

The Potential of Deep Learning for Gas Sensor Evaluation and Calibration

DISSERTATION

zur Erlangung des Grades
des Doktors der Ingenieurwissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von

Yannick Robin

Saarbrücken

2024

Tag des Kolloquiums: 05. Juni 2024

Dekan: Prof. Dr.-Ing. Michael Vielhaber

Berichterstatter: Prof. Dr. Andreas Schütze

Prof. Dr. Santiago Marco

Vorsitz: Prof. Dr. Romanus Dyczij-Edlinger

Akad. Mitarbeiter: Dr.-Ing. Amine Othmane

Danksagung

Diese Arbeit wäre natürlich nicht ohne die Unterstützung von vielen Personen möglich gewesen. Im Folgenden möchte ich mich bei einigen dieser Personen bedanken.

Allen voran möchte ich mich bei meinem Doktorvater Prof. Dr. Andreas Schütze für die Möglichkeit bedanken, diese Arbeit anfertigen zu können. In vielen Gesprächen hat er mir das Thema rund um die Gassensoren nähergebracht und gezeigt, an welchen Stellen ich mit meiner Forschung ansetzen musste. Im gleichen Atemzug möchte ich mich bei Prof. Dr. Santiago Marco dafür bedanken, dass er als Gutachter für diese Arbeit fungiert und mir bei verschiedenen Konferenzen wichtige Gedankenanstöße gegeben hat.

Natürlich dürfen nicht all die wunderbaren Kollegen am Lehrstuhl für Messtechnik vergessen werden. Ohne den Austausch auf Arbeitsebene wäre es mir nicht möglich gewesen, ein solches Verständnis für die Themen Gassensorik und maschinelles Lernen aufzubauen. Sie haben alle maßgeblich zu dem Erfolg dieser Arbeit beigetragen. Dennoch möchte ich besonders Johannes, Dennis, Christian und Tobias hervorheben, ohne die ich niemals die Daten bekommen hätte, die ich für meine Forschung benötigt habe. Ebenfalls möchte ich mich bei Payman bedanken, der mir während meiner Arbeit am Lehrstuhl bei Fragen rund um Deep Learning zur Seite gestanden und große Teile dieser Arbeit inhaltlich kontrolliert hat. Zuletzt möchte ich Tizian für seine starken Neven danken.

Neben der Arbeitswelt haben mich auch meine Freunde und Familie stets bei meinem Ziel des Dokortitels unterstützt. Hier möchte ich mich vor allem bei meinen Eltern Birgit und Noel, meiner Schwester Jana, meiner Oma Irmgard und meiner Freundin Kathrin bedanken. Ihr habt mich immer unterstützt und den ganzen Werdegang erst möglich gemacht. Vielen Dank!

Zuletzt möchte ich mich auch bei meinen Freunden dafür bedanken, dass ihr mich die anstrengende Arbeitswelt auch mal vergessen lassen habt. Besonders möchte ich mich hier bei Christina bedanken, die ebenfalls große Teile dieser Arbeit kontrolliert hat.

Abstract

Metal oxide semiconductor gas sensors are promising candidates for selectively measuring harmful pollutants indoors. However, they suffer from their lack of selectivity, sensor-to-sensor variance, and drift over time.

Advanced calibration and operation modes are required to overcome some of these sensor drawbacks. During calibration, the sensor is exposed to many gas mixtures to build robust, data-driven models. Based on the sensor response, these models deduce the target gas concentration present. Special operation modes like temperature-cycled operation are used to gain additional information from the transient behavior of the sensor. However, calibration can be costly, time-consuming, and complicated, even without complex operation modes.

Within this thesis, a new data-driven model for the evaluation and calibration of metal oxide semiconductor gas sensors is introduced. The newly developed model, TCOCNN, is a multi-layer convolutional neural network. Together with methods from the field of deep learning, like transfer learning, it is possible to tackle long calibration times and sensor-to-sensor variation. It was shown that it is possible to reduce the calibration time by up to 99.3 % and significantly reduce the influence of sensor-to-sensor variance. In some aspects, the TCOCNN surpasses state-of-the-art methods and provides insights into the model's inner workings, the temperature cycle, and the sensor itself.

Zusammenfassung

Metalloxid-Halbleiter-Gassensoren sind ein vielversprechender Kandidat für die Messung einzelner schädlicher Gase in der Innenraumluft. Allerdings leiden die Sensoren unter der starken Varianz zwischen den Sensoren, dem Drift über die Zeit, und der fehlenden Selektivität.

Um den Sensor für die selektive Messung von schädlichen Gasen nutzbar zu machen, muss der Sensor kalibriert und in komplexen Betriebsmodi (bspw. Temperaturzyklus) verwendet werden. Während der Kalibrierung arbeitet der Sensor im Temperaturzyklus und wird verschiedenen Gasgemischen ausgesetzt. Die daraus resultierenden Daten werden dazu genutzt, ein Modell zu trainieren, das die Konzentration der Zielgase vorhersagen kann. Diese aufwendige Kalibrierung ist bereits ohne Temperaturzyklus teuer, komplex und zeitintensiv.

Deshalb wird innerhalb dieser Arbeit ein neues datengetriebenes Modell vorgestellt. Die neue Methode basiert auf einem mehrschichtigen convolutional neural network und wird als TCOCNN bezeichnet. Es konnte gezeigt werden, dass das TCOCNN in manchen Aspekten signifikant bessere Ergebnisse als die klassischen Methoden erzielt. Des Weiteren konnten fortgeschrittene Methoden des Deep Learning (bspw. Transfer Learning) dazu genutzt werden, die Schlüsselfragen rund um Metalloxid-Halbleiter-Gassensoren zu lösen. Beispielsweise konnte die Kalibrierzeit um bis zu 99,3 % reduziert werden, während trotzdem gezielte Einblicke in den Temperaturzyklus, den Sensor und die Funktionsweise des Modells möglich sind.

Appended Papers

- Paper 1** Y. Robin, J. Amann, T. Baur, P. Goodarzi, C. Schultealbert, T. Schneider, and A. Schütze: High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning, *MDPI open access journal: Atmosphere* (2021)
- Paper 2** Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, and C. Bur: Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning, *MDPI open access journal: Atmosphere* (2022)
- Paper 3** Y. Robin, J. Amann, T. Schneider, A. Schütze, and C. Bur: Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors, *MDPI open access journal: Atmosphere* (2023)

Published articles have been reprinted with the permission of the copyright holders.

Author's contribution to appended papers

Paper 1 I conceptualized the paper with respect to the application of deep learning for gas sensor calibration. Furthermore, I developed the methodology and software. Moreover, I validated the results, performed the formal analysis, made the visuals, and wrote the original draft and the final version of the paper.

Paper 2 I conceptualized the paper with respect to the application of transfer learning from deep learning for calibration transfer in the field of gas sensors. Furthermore, I developed the methodology and software. Moreover, I validated the results, performed the formal analysis, made the visuals, and wrote the original draft and the final version of the paper.

Paper 3 I conceptualized the paper with respect to the comparison of different calibration transfer methods. Furthermore, I developed the methodology and software. Moreover, I validated the results, performed the formal analysis, made the visuals, and wrote the original draft and the final version of the paper.

Other appended papers

- Paper A** Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, and C. Bur: Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays, *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)* (2023)
- Paper B** P. Goodarzi, Y. Robin, A. Schütze, and T. Schneider: Deep convolutional neural networks for cyclic sensor data, Eprint *arXiv* 2308.06987 (2023)

Published articles have been reprinted with the permission of the copyright holders.

Author's contribution to other appended papers

Paper A I conceptualized the paper with respect to the application of explainable AI for gas sensor calibration. Furthermore, I developed the methodology and software. Moreover, I validated the results, performed the formal analysis, made the visuals, and wrote the original draft and the final version of the paper.

Paper B I conceptualized the paper with respect to the application of deep learning for condition monitoring and cyclic sensor data within the use-case of hydraulic systems. Furthermore, I developed the methodology and software. Moreover, I validated the results, performed the formal analysis, and reviewed the original draft of the paper.

Other Related Publications

- Data set i** Y. Robin, J. Amann, C. Bur, and A. Schütze: Transfer Learning Dataset for Metal Oxide Semiconductor Gas Sensors, *Zenodo* (2022)
- Paper i** Y. Robin, P. Goodarzi, T. Baur, C. Schultealbert, A. Schütze, and T. Schneider: Machine Learning Based Calibration Time Reduction for Gas Sensors in Temperature Cycled Operation, *IEEE I2MTC 2021 International Instrumentation and Measurement Technology Conference (digital)* (2021)
- Paper ii** T. Dorst, Y. Robin, T. Schneider, and A. Schütze: Automated ML Toolbox for Cyclic Sensor Data, *MSMM 2021 Joint Virtual Workshop of ENBIS and MATHMET Mathematical and Statistical Methods for Metrology* (2021)
- Paper iii** T. Baur, C. Schultealbert, Y. Robin, P. Goodarzi, T. Schneider, and A. Schütze: Accurate Quantification of Formaldehyde at ppb Level for Indoor Air Quality Monitoring, *IMCS 2021, International Meeting on Chemical Sensors, Digital Conference* (2021)
- Paper iv** T. Dorst, Y. Robin, S. Eichstädt, A. Schütze, and T. Schneider: Influence of synchronization within a sensor network on machine learning results, *Journal of Sensors and Sensor Systems* (2021)
- Paper v** Y. Robin, J. Amann, P. Goodarzi, T. Baur, C. Schultealbert, T. Schneider, and A. Schütze: Überwachung der Luftqualität in Innenräumen mittels komplexer Sensorsysteme und Deep Learning Ansätzen, *15. Dresdner Sensor-Symposium* (2021)

-
- Paper vi** S. Pültz, Y. Robin, A. Schütze, T. Schneider, Y. Koch, E. Kirchner, D. Quirnheim Pais, and L. Rauber: Automated Condition Monitoring for Helical Gears based on measuring Instantaneous Angular Speed with Magnetoresistive Sensors, *Sensoren und Messsysteme 2022* (2022)
- Paper vii** Y. Robin, J. Amann, P. Goodarzi, A. Schütze, and C. Bur: Transfer Learning to Significantly Reduce the Calibration Time of MOS Gas Sensors, *ISOEN 2022 - International Symposium on Olfaction and Electronic Nose* (2022)
- Paper viii** Y. Robin, J. Morsch, T. Schneider, A. Schütze, and C. Bur: Insight in Dynamically Operated Gas Sensor Arrays with Shapley Values for Data Segments, *MNE EUROSENSORS 2022* (2022)
- Paper ix** C. Schnur, Y. Robin, P. Goodarzi, T. Dorst, A. Schütze, and T. Schneider: Development of a bearing test-bed for acquiring data for robust and transferable machine learning, *IEEE I2MTC 2023, International Instrumentation and Measurement Technology Conference* (2023)

Contents

Abbreviations	XV
1 Introduction	1
1.1 Motivation	2
1.2 Organization	3
2 Theoretical Background	5
2.1 Gas Sensors	5
2.2 Gas Mixing Apparatus	10
2.3 Design of Experiment	13
2.4 Machine Learning	15
2.4.1 General Machine Learning	16
2.4.2 Classic Machine Learning	19
2.4.3 Neural Network Basics / Deep Learning	22
2.5 Data-Driven Sensor Calibration	40
2.5.1 Challenges for Sensor Calibration	42
2.5.2 State-Of-The-Art	45
3 Results and Discussion	53
3.1 Results and Discussion: Introduction	53
3.2 Paper 1 – High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning	57
3.2.1 Synopsis	58
3.3 Paper 2 – Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning	89
3.3.1 Synopsis	90
3.4 Paper 3 – Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors	117
3.4.1 Synopsis	118

3.5	Paper A – Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays	143
3.5.1	Synopsis	144
3.6	Paper B – Deep convolutional neural networks for cyclic sensor data . .	153
3.6.1	Synopsis	154
4	Conclusion	161
5	Outlook	163
	References	165
A	Appendix: Gaussian Process	194
	List of Figures	197
	List of Tables	200

Abbreviations

ADC	Analog Digital Converter
Adam	Adaptive Moment Estimation
AI	Artificial Intelligence
ALA	Adaptive Linear Approximation
CNN	Convolutional Neural Network
DL	Deep Learning
DoE	Design of Experiment
DS	Direct Standardization
FESC	Feature Extraction Selection Classification
FESR	Feature Extraction Selection Regression
GMA	Gas Mixing Apparatus
GC-MS	Gas Chromatography-Mass Spectrometry
IAQ	Indoor Air Quality
LDA	Linear Discriminant Analysis
LMT	Lab for Measurement Technology
MFC	Mass Flow Controller
ML	Machine Learning
MLP	Multi-Layer Perceptron
MOS	Metal Oxide Semiconductor
PCA	Principal Component Analysis
PDS	Piecewise Direct Standardization
PLSR	Partial Least Squares Regression
RH	Relative Humidity

RMSE	Root-Mean-Squared Error
SVM	Support Vector Machine
SVOC	Semi Volatile Organic Compound
TC	Temperature-Cycle
TCO	Temperature-Cycled Operation
UGM	Unique Gas Mixture
VOC	Volatile Organic Compound
VVOC	Very Volatile Organic Compound
WHO	World Health Organization
XAI	Explainable Artificial Intelligence

1 Introduction

This thesis focuses on gas sensing with Metal Oxide Semiconductor (MOS) gas sensors, primarily on how it can be done, the problems, and how to solve some of them. The main gas sensing application analyzed is Indoor Air Quality (IAQ) monitoring. The importance of appropriate IAQ monitoring is demonstrated in [1]. There, it was revealed that in 2019, over 6 million people died due to air pollution [1]. In the same context, the World Health Organization (WHO) has set a goal to reduce air pollution by 2030 substantially to prevent related deaths [2]. Specifically, IAQ monitoring is essential as humans spend up to 90 % of their time indoors, which makes them particularly vulnerable to indoor air pollution [3–5].

Currently, IAQ monitoring aims to measure harmful gases within the relevant concentration ranges. The gases of interest for IAQ monitoring are usually carbon monoxide, ozone, radon, and Volatile Organic Compounds (VOCs) (today, most of the time total VOC concentration) [6–9]. This information can be used to control ventilation or warn the room’s occupants to leave. For this thesis, the main objective is to accurately and selectively measure single VOCs [10], including Very Volatile Organic Compounds (VVOCs) [11], and Semi Volatile Organic Compounds (SVOCs) [12]. However, selectively measuring important VOCs is complex since indoor air consists of hundreds of different gases that can interfere with the measurement. Furthermore, not all VOCs are relevant for IAQ monitoring. Some are less harmful, like ethanol or isopropanol, while others, like benzene or formaldehyde, can already cause serious health issues at low concentrations [13]. Currently, a popular method to rate IAQ is to calculate the total VOC concentration with the help of the CO_2 equivalent. Pettenkofer [7] had already discovered the relationship between bad IAQ and the exhaled amount of CO_2 by humans. However, since humans are not the sole source for the presence of VOCs, the CO_2 equivalent can be very inaccurate. For example, cleaning products, furniture, and cooking emit VOCs but do not emit CO_2 [14]. Furthermore, it is impossible with this method to detect single harmful VOCs at low concentrations to ensure human safety, and alternative methods must be developed. One approach would be to use the

gold standard, e.g., Gas Chromatography-Mass Spectrometry (GC-MS). While these instruments are very reliable, they only allow for relatively slow measurements, require expert knowledge (complex calibration), and are expensive. One promising alternative might be MOS gas sensors.

1.1 Motivation

As stated previously, IAQ is paramount for human health. Promising candidates for measuring single harmful VOCs are MOS gas sensors to ensure good IAQ [15]. MOS gas sensors were chosen for this thesis because they are low cost, easy to use, and sensitive to various gases [16–18]. Nevertheless, they are rarely deployed to detect single harmful VOCs for multiple reasons.

First, MOS gas sensors are not very selective, meaning the detection of single harmful gases cannot be easily achieved and is still subject to recent studies [19]. Likewise, sensor response and target gas dependencies are too complex to be explained by an analytic physical-chemical model [18]. Therefore, data-driven approaches are necessary, leading to numerous calibration samples, which can be costly in terms of time and money. Furthermore, the sensor-to-sensor variance requires every sensor to be independently calibrated, making it even more expensive for the manufacturer [20]. Other major drawbacks of MOS gas sensors are the stability over time and sensor poisoning [21, 22]. Both drawbacks change the properties of the sensor so that frequent recalibration is necessary to continuously measure the target gases, which is unsuitable for industrial and especially consumer applications.

In the past, multiple attempts have been made to tackle those drawbacks and make MOS gas sensors viable for a wide range of gas sensing applications. One state-of-the-art solution to improve the lack of selectivity is Temperature-Cycled Operation (TCO), which has already been thoroughly studied in [18, 23] and is also used throughout this thesis. In TCO, the sensor is heated to different temperature levels. The transient responses of the sensor can then be used to derive the concentration of different gases present. Similarly, other methods like calibration transfer and randomized calibration are used to reduce the calibration time and to reduce the effect of sensor-to-sensor variance on data-driven models [24, 25]. However, none of these methods have been implemented in large-scale MOS gas sensor deployment for accurate IAQ monitoring.

One of the reasons might be that calibration is still too costly in terms of time and money and, therefore, not feasible for commercial applications.

This thesis aims to improve existing methods to finally make MOS gas sensors suitable for the selective measurement of VOCs. The main goal is to reduce calibration times and sensor-to-sensor variance further. Therefore, some of the already existing methods are combined with new techniques from the field of Deep Learning (DL), like convolutional neural networks, transfer learning, and Explainable Artificial Intelligence (XAI). To put the new results in perspective, they are compared with state-of-the-art methods like classic Machine Learning (ML), e.g., Feature Extraction Selection Regression (FESR), and classic calibration transfer methods, e.g., Direct Standardization (DS).

The challenges of unknown interfering gases, sensor drift, and poisoning are only roughly introduced and should be further analyzed in future publications. Furthermore, all described methods should be tested for a wider range of applications like breath analysis, health monitoring, outdoor air monitoring (air pollution monitoring), and industrial emission monitoring to validate their effectiveness [5, 26, 27].

1.2 Organization

The thesis is structured as follows: In the Theoretical Background chapter, the foundations are presented to understand the core topics of this thesis. This chapter introduces the MOS gas sensor together with the TCO. Furthermore, the Gas Mixing Apparatus (GMA) is introduced to understand how calibration samples are collected. The design of the experiment is also explained to complement the data recording process. The following sub-chapter introduces ML, covering various data-driven approaches ranging from classical ML to Deep Learning (DL) and their advantages as well as disadvantages. In the last part of the Theoretical Background chapter, the drawbacks of MOS gas sensors are introduced together with state-of-the-art solutions. In the Results and Discussion chapter, the three core papers that tackle the major drawbacks of MOS gas sensors are presented in detail and placed in the context of other state-of-the-art publications. Additionally, two conference papers are discussed to give a broader view of neural networks in general and how they can be applied to cyclic sensor data. The results are summarized in the Conclusion chapter and put into the larger context for MOS gas sensors and their relevance for manufacturers. Possible further extensions are listed in the Outlook.

2 Theoretical Background

2.1 Gas Sensors

It is essential to mention that the presented sensor concept is not exhaustive and covers only some essentials of gas sensors. In this regard, electrochemical sensors, infrared gas sensors, and others are not discussed [28]. The foundation of this thesis are Metal Oxide Semiconductor (MOS) gas sensors. Tetsuro Seiyama proposed the first zinc-oxide semiconductor gas sensor in 1962 [29]. It was demonstrated that by heating the sensor to high temperatures of around 400 °C, the resistance of the metal oxide semiconductor changes rapidly depending on the gas present. Analyzing the response made detecting individual gases at different concentration levels possible. Since then, multiple materials (e.g., TiO_2 [30], WO_3 [31], SnO_2 [30, 32]), structures (e.g., nanoparticles [33], nanoflakes [34], nanowires [35]) and operation modes (static operation, dynamic exposure [36], and Temperature-Cycle (TC) [37, 38]) have been developed. Thorough reviews on this matter can be found in [39, 40]. However, the general sensor concept remains the same and can be described as follows [41]. Typically, the MOS gas sensor system consists of a micro hotplate with the sensing element on top. Figure 2.1 (a) shows that the sensing elements consist of two engaging metal wires. The two wires are not connected; instead, the space in between is filled with a metal oxide semiconductor material (most commonly SnO_2 [40]). The general conductance of the metal oxide semiconductor layer (e.g., SnO_2) is caused by missing oxygen atoms, which results in freely moving electrodes in the lattice [42]. Although the metal oxide semiconductor material generally allows a current to flow, the resistance changes with respect to the free charge carriers, which depends on the surrounding gases. Two processes must be considered to understand the influence of the surrounding gas on the amount of freely moving electrons. Firstly, physisorption, this effect attracts gases to the sensor's surface with the help of Van der Waals forces [43], and secondly, chemisorption [44], which allows the adsorption of gases and, thereby, the transfer of electrons. One example is oxygen; its high electronegativity binds freely moving electrons near the surface, hence increasing

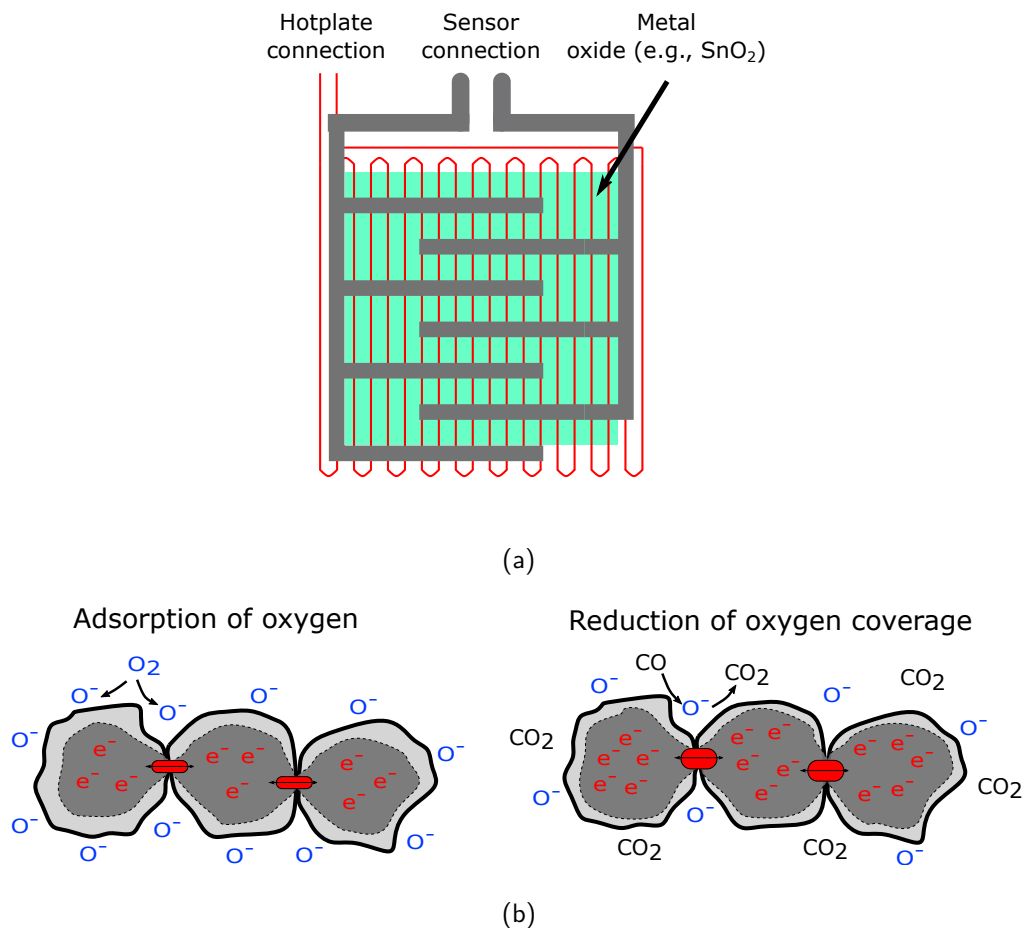


Figure 2.1: a) Basic MOS sensor model (adapted from [42]). b) Model for grain boundary effect during adsorption and reduction of oxygen. The change in resistance is indicated by the width of the conduction channel between grains illustrated in red (adapted from [42, 46]).

the resistance and thereby changing the sensor's electrical properties. Reducing gases can then recombine with the chemisorbed oxygen (O^-) on the surface, releasing the electrons and allowing them to move again freely through the lattice [40], causing a decrease in resistance. Of course, other gases can also bind to the gas sensor's surface, increasing or decreasing the amount of freely moving electrons (e.g., O_3 [45]). Which gas is more likely to bind to the surface or recombine with the gases at the surface depends on the temperature [19].¹ Therefore, the micro-hotplate can be used to control the temperature and thereby manipulate the sensor's response.

Although the presence or absence of oxygen is the dominant effect that determines how many freely moving electrons are available, the grain boundaries define the general

¹The material and the corresponding morphology also influence the selectivity of the sensor.

resistance. This effect is illustrated in Figure 2.1 (b). Here, it can be seen that the described effect of oxygen binding electrons at the surface has the most significant impact at grain boundaries of the MOS substrate. This is because the depletion of electrons there has the most significant influence on the resistance. Therefore, the sensor's resistance differs over multiple magnitudes if covered entirely with oxygen molecules or if no oxygen is adsorbed at the surface. Since the general dependencies between surrounding gases and sensor response are known, it is possible, under well-defined conditions, to calculate the exact sensor response to a specific known gas or to measure the surrounding gas based on the sensor response [47, 48]. Nevertheless, this is only feasible if the sensor is operated under specific laboratory conditions (e.g., exposed to single gases, known humidity; cf. Figure 2.2). Therefore, calculating is unsuitable for real-world applications since it is impossible to determine every parameter during operation (surface-state) [49, 50]. The most popular approach to still predict the specific gas concentration is to build a data-driven model by calibrating the sensor [24, 27, 51]. With the help of multiple training samples, the model learns the relation between the sensor response and the applied gas mixtures.

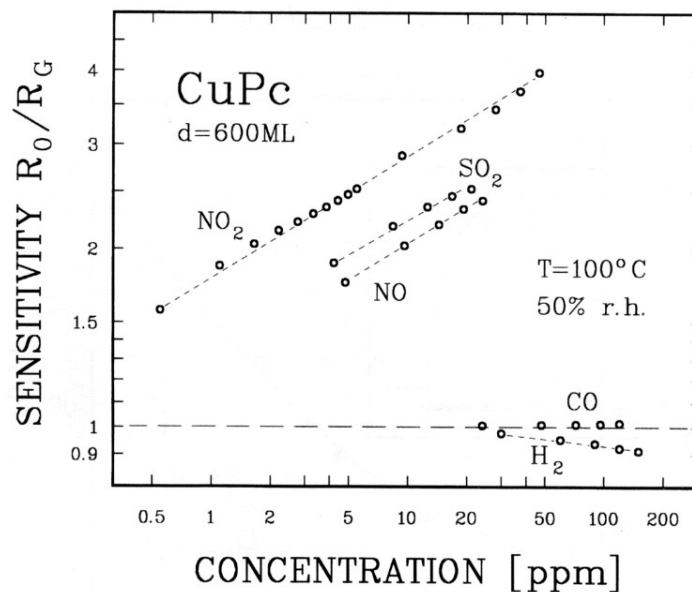


Figure 2.2: Double logarithmic dependency between sensor response and gas concentration. Reprinted with permission of [52], © 1995 Shaker.

The sensor effects captured with the training samples vary according to the operation modes of the sensor. The most common approach is to operate the sensor in pure air under a stable temperature, expose the sensor to the target and interfering gases,

and then clean it with pure air (transient sensor response). The contamination and cleaning process shows gas mixture-specific response patterns that can be learned [16, 36, 53]. However, this can be challenging for real-world use cases as complex switching mechanisms between target air and purified air are required, where purifying itself can be difficult [54]. Another approach is to use a sensor array (multiple different sensors) under stable temperature and constant gas mixture [55]. This method relies on a multisensor array and analyses the static response of the system to the gas mixture. Every mixture has a specific fingerprint that can be learned [56]. The method used throughout this thesis is temperature modulation. In this case, the gas sensor is covered with a stable gas mixture, and with the help of micro hotplates, the temperature of the sensor is modulated [15, 37, 38, 57, 58]. Today, because of the micro hotplate and the micro-structured sensor elements, it is possible to heat the sensor within milliseconds to high temperatures. Similarly, low temperatures are achieved because the system quickly cools down as soon as the micro hotplate is turned off [59]. With the help of the different temperature steps and the transient behavior of the sensor, it is possible to have a virtual gas sensor that makes it possible to identify different gases selectively [60]. Multiple temperature patterns have been proposed for this method, each with benefits and drawbacks. Examples are given as ramping the temperature up and down [61], quickly changing between two temperatures [62], or complex heating patterns [63, Paper 2]. An example for a TCO is shown in Figure 2.3.

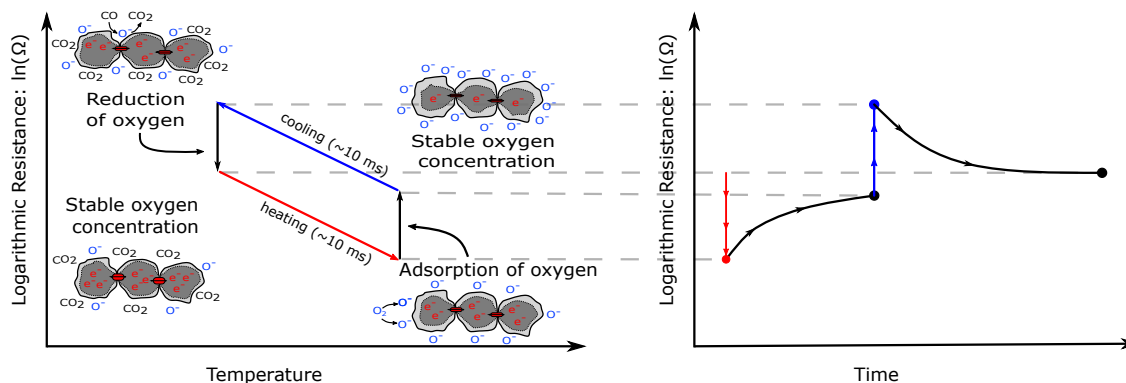


Figure 2.3: Working principle of the temperature-cycled operation regarding oxygen coverage. During heating, the general resistance of the sensor decreases. Afterward, the oxygen molecules adsorb on the sensor, increasing the resistance. Then, the sensor is cooled down, which generally increases the resistance. Additionally, reducing gases recombine with O^- during the low-temperature phases, lowering the overall resistance (adapted from [18, 41, 46]).

The TC consists of two different temperature steps, each with a specific duration. The information about the surrounding gas can be extracted by analyzing the transient behavior. The general principle is that during high-temperature phases, the sensor's O^- coverage at the surface is renewed, and during low-temperature phases, reducing gases can recombine with the O^- . The different temperature levels during a TC allow different gases to combine with the O^- to change the sensor response. Based on the gas-specific transient behavior, deducing the target gas concentration is possible. The fundamentals of TCO, like the rate constant of the dynamic processes, are covered within the Sauerwald-Baur model [18, 19, 23].

In order to gather the sensor responses, it is necessary to measure the sensor's resistance. The most basic approach to read out the sensor resistance is to connect the sensor to one high-precision resistor and a voltage source, as shown in Figure 2.4.

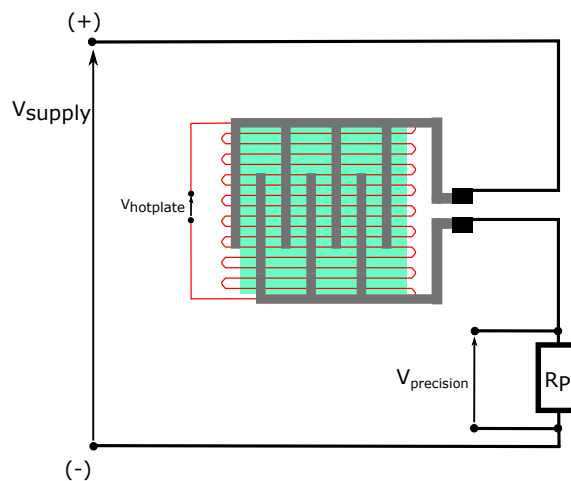


Figure 2.4: Example configuration for a readout circuit (adapted from [64]).

The physical quantity measured is the voltage at the precision resistor (R_p). Since the primary circuit works as a voltage divider, it is possible to calculate the resistance of the gas sensor. For the data in this thesis, the voltage across the sensor is fixed, and the current that flows through the sensor is measured. This method can be improved by using a logarithmic amplifier before the Analog Digital Converter (ADC) [65].² Using the logarithmic amplifier has the benefit that less information on the transient behavior is lost because the change in resistance over several orders of magnitudes is captured with higher precision. This effect of the exponential change in resistance is described in the Sauerwald-Baur model [18, 23].

²Some digital sensors have the required electronics already integrated.

The MOS gas sensors used throughout this thesis are the SGP40 and SGP30 sensors from Sensirion (Sensirion AG, Stäfa, Switzerland). These sensors consist of four sub-sensors/pixels with a micro hotplate. For model building, all pixels are used simultaneously. Although only MOS gas sensors are covered within this thesis, the subsequently introduced methods for sensor calibration can be applied to a wide range of sensors.

2.2 Gas Mixing Apparatus

As mentioned above, a data-driven model must be built to predict specific gas concentrations with the help of a MOS gas sensor.³ To collect the data for sensor calibration, the sensor must be exposed to many known gas mixtures. Either the sensor is calibrated in laboratory conditions with artificially generated gas mixtures [66, 67] or in the field with reference instruments [68, 69]. Both methods come with benefits and drawbacks. However, this thesis focuses on calibrating sensors under laboratory conditions. Although calibrating under laboratory conditions sounds more manageable, it also comes with challenges. This is because of multiple reasons: the applied gas mixture can consist only of a selection of gases that represent the use-case (hundreds of different gases in the real world), the gases within the mixtures can be difficult to handle (e.g., the VVOC formaldehyde [13]), the independent gas concentrations can differ based on the allowed thresholds in several orders of magnitudes (e.g., formaldehyde 0 - 100 ppb, carbon monoxide 200 - 5000 ppb [5, 70]), not all gases are available in bottles, mixing gases with an acceptable uncertainty is complex, and creating reproducible gas mixtures with a relatively small uncertainty can be difficult. Thus, an advanced apparatus that automatically mixes the desired gas mixtures and automatically applies the well-known gas mixtures to the sensor or sensor systems is necessary.

Over the last few years, different methods to build such an apparatus have been developed. Those approaches can mainly be divided into closed loop [67, 71] or continuous flow [66, 72, 73]. Based on ISO 6145 [74], multiple iterations have been made at the Lab for Measurement Technology (LMT) to build the most optimal continuous flow Gas Mixing Apparatus (GMA) (cf. Figure 2.5) for the use cases covered, e.g., IAQ, outdoor air quality, exhaust gas streams, or breath analysis [75–78]. Since the beginning, the system has been designed with a continuous total flow realized with a carrier gas in the form of zero air (sometimes nitrogen). Zero air is generated with the Ultra

³Building the data-driven model and collecting the calibration samples can be understood as calibrating the sensor.

Zero Air generator GT 30000 plus-EU (Schmidlin Labor + Service GmbH, Dettingen, Germany). The surrounding air is sucked into the system, purified from all VOCs, H_2 , CO , and the sum of all hydrocarbons is reduced to a maximum of 0.1 ppm [79]. The flow of zero air through the system and sensor chamber is controlled with an Mass Flow Controller (MFC) (1000 ml min^{-1}).⁴ The GMA50A module of MKS Instruments Germany was chosen because of its fast switching times and high precision [76–78]. Apart from the carrier gas, the system consists of multiple gas lines that provide the sensor chamber with the humidified air, target, and background gases [77, 78]. Figure 2.5 shows the different types of gas lines. The most simple line used is the normal line. There, the gas from a gas bottle is directly forwarded through an MFC to the gas mixing chamber ($0 - 20 \text{ ml min}^{-1}$). A more complex version of the normal line consists of a two-stage dilution. The target gas and zero air are mixed in the first stage within a mixing pipe. This first mixture is controlled by two MFCs, one for the target gas in air and the other for the zero air. The MFC for the gas bottle is usually smaller in order to achieve a dilution as high as possible ($10 - 20 \text{ ml min}^{-1}$ vs. 500 ml min^{-1}). After the first mixing stage, not the entire flow is forwarded to the sensor chamber; instead, only a fraction is forwarded with the help of a third MFC of the same size as the one at the gas bottle. This multi-stage dilution ensures that the impurities of the different gas bottles have a minimal impact. For example, a grade five rating gas bottle still has 10 ppm of impurities, which can significantly affect the calibration. With the help of predilution, this can be reduced to concentrations smaller than 1 ppb [78]. Furthermore, this two-stage approach allows reaching concentrations as small as 1:12505000 of the original gas bottle concentration, as reported by [77]. However, the additional mixing sections make these more complex dilution lines harder to build. A third gas line can be designed by using a permeation oven instead of a gas bottle. This is usually done if the gas is unavailable in a gas bottle but within a permeation tube. Nevertheless, the principle remains the same: the target gas and zero air are mixed, and only a portion is forwarded through an additional MFC. The last unique line is then used to humidify the system’s gas flow. Humidification is essential since humidity significantly impacts MOS gas sensors. So far, all lines are designed with very low residual humidity.⁵ In order to vary the Relative Humidity (RH), the system contains one humidification line that allows to humidify the total flow. This line is regulated with the help of zero air and a single MFC. The dry air that passes through the MFC is humidified with the help of a

⁴The MFC is a module that can control the throughput of gas [76].

⁵Much smaller than 1 %.

water bubbler and forwarded to the sensor chamber. The different lines are combined in the gas mixing chamber, and the final gas mixture is generated with the specified gas concentration at the desired humidity at room temperature. The most advanced GMA was built recently and consists of up to 14 lines to generate the most complex mixtures. A total of 14 lines is very well suited for indoor air applications, where ten lines can be used for VOCs and at least two lines for background gases that can influence the prediction of the target VOC. Ten VOCs are suitable since it is possible to categorize most VOCs into ten groups, as described later within the Design of Experiment section. Figure 2.5 illustrates the complete setup with all components.

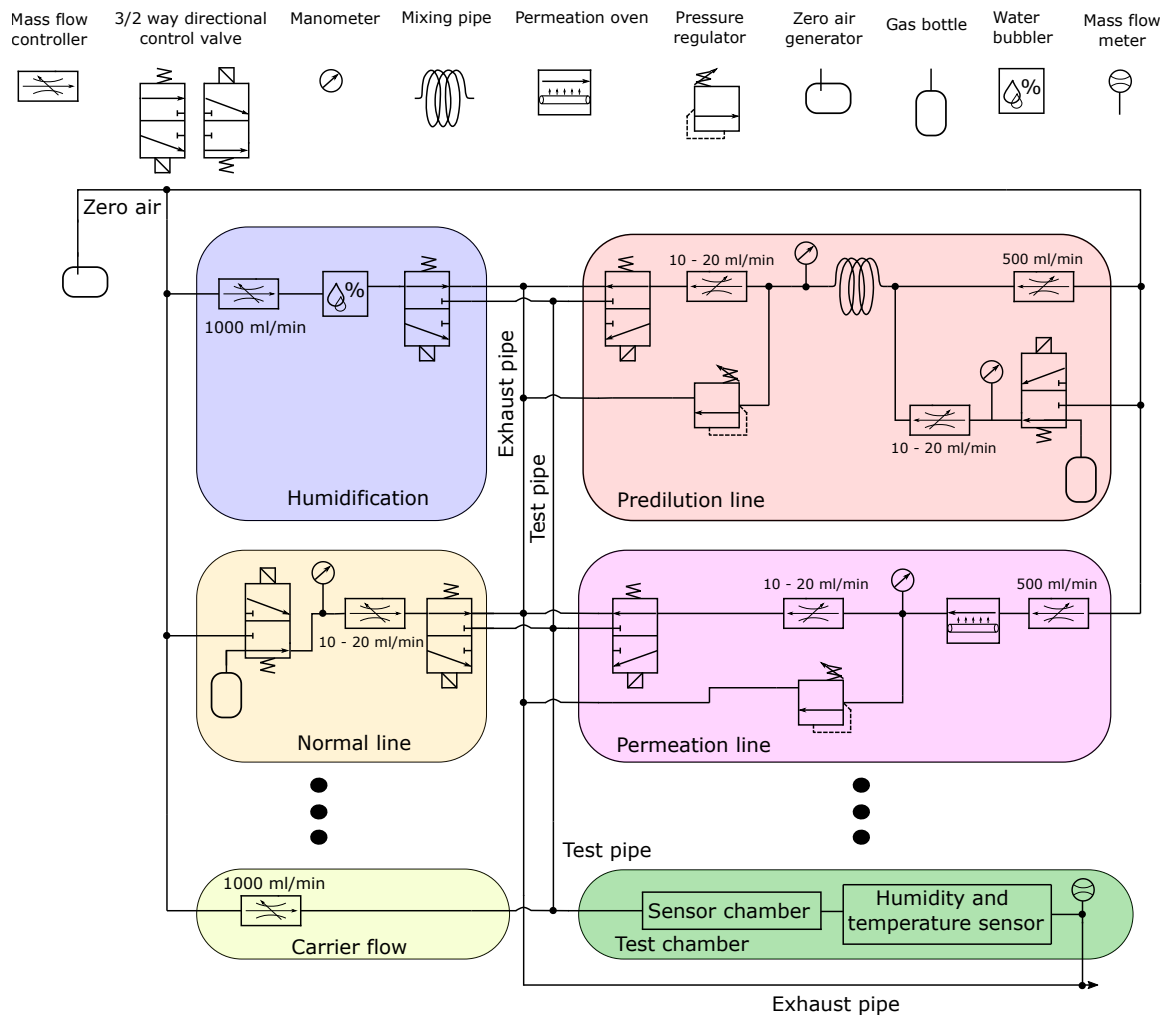


Figure 2.5: Schematic of the most complex GMA at LMT today (adapted from [77]).

This description should only briefly overview the topic of GMA design. A more detailed overview can be found in the following publications, which also dive into dead volumes, timing within the GMA mixing sections, and efficiency regarding the gas consumption [75–78].

2.3 Design of Experiment

After introducing the MOS gas sensors and the GMA to record the calibration data, the next step is to record meaningful calibration samples in order to be able to build a stable and robust data-driven model that is capable of performing the target task. As described earlier, the main application in this work is IAQ monitoring. For IAQ, VOCs are one of the most critical substances to detect. However, detecting harmful VOCs requires careful gas sensor calibration. During calibration, multiple environmental influences must be taken into account to build a reliable model. For example, various gases that can influence the prediction in the real world and the different humidity levels must be considered. The process of accounting for known influences for calibration is called Design of Experiment (DoE) and is currently investigated by many researchers [24, 80–83]. However, because this thesis mainly focuses on evaluating gas sensor calibration, the topic of DoE is only briefly introduced using an IAQ example. The first important question for the DoE for gas sensors is: In which environment does the sensor operate later? This question is essential because the selection of interfering gases, background gases, and the RH would vary significantly based on the use case (e.g., breath analysis vs. IAQ). Therefore, the DoE for IAQ monitoring is always done with realistic background concentrations of hydrogen and carbon monoxide and a realistic RH [15, 84]. Furthermore, indoor air consists of hundreds, if not thousands, of VOCs, making it impossible to use all of them within the calibration. Therefore, the next step in the DoE is to select the most critical VOCs. The selection of the most important VOCs can be done with the help of the substance groups theory [15], which states that all VOCs can be categorized into ten subgroups. For smaller datasets, the dominant substance of each category or, if possible, multiple main contributors can be used. The eight most important VOC groups for IAQ monitoring and their main contributors are listed in Table 2.1.

Besides the different groups and substances per group, Table 2.1 also indicates the different concentration ranges typically encountered in indoor air. For a more sophisticated DoE, the maximum allowed indoor concentration over a more extended

Table 2.1: The eight most important chemical classes for VOCs extracted from analytical studies [85, 86] together with representatives, P90, and P95 quantiles for reference measurements. Reprinted with permission of Ref. [15]. T. Baur, 2023.

Chemical Class (representative)	P90 in $\mu\text{g}/\text{m}^3$ (ppb)	P95 in $\mu\text{g}/\text{m}^3$ (ppb)
Alcohols (ethanol)	320 (~ 170)	520 (~ 790)
Aldehydes (formaldehyde)	340 (~ 270)	480 (~ 390)
Alkanes (n-hexane, n-heptane)	180 (~ 50)	350 (~ 90)
Aromatics (toluene)	190 (~ 50)	370 (~ 90)
Esters (ethyl acetate)	140 (~ 30)	280 (~ 70)
Ketones (acetone)	250 (~ 100)	420 (~ 170)
Terpenes (limonene, α -pinene)	170 (~ 30)	330 (~ 60)
Organic acid (acetic acid)	150 (~ 60)	240 (~ 100)

period is given by [5, 70]. With this information, it is possible to define the different gas mixtures and the corresponding concentration ranges that can be used to calibrate the sensor for IAQ monitoring. The next task is to define the Unique Gas Mixture (UGM) in order to calibrate the sensor. Therefore, the different gas mixtures that are applied to the sensor must be specified. Since the data-driven model should be as robust and accurate as possible, it is necessary to have data samples covering the complete range for all gases. The first experiments done in this field used sequential calibration [24, 68]. Figure 2.6 shows an example of this scheme. In sequential calibration, all gases are increased step-wise gas per gas. The drawback of this approach is that it requires many samples⁶ to cover all concentration ranges for all gases, and it introduces systematics that the model can learn (overfitting [24]). That implies that the model might learn the dependency between time and gas concentration instead of the actual gas concentration and corresponding sensor response. One method to prevent this is to apply different gases in ascending and descending order. Nevertheless, this doubles the calibration time and does not prevent overfitting. Therefore, the only suitable way is to use randomized calibration [24]. In this case, the gas concentrations are not predefined and separately selected; instead, random concentrations are randomly picked from a predefined distribution for every UGM. This ensures that no systematic is introduced that the data-driven model might learn. However, multiple methods can help improve the randomization, which enables building an even more robust and stable model.

⁶In the case of 4 concentrations per gas, $4^{\text{numbergases}}$ UGMs would be necessary.

Examples are random subsampling, Latin hypercube sampling [87, 88], and orthogonal sampling [89]. For most datasets in this work, Latin hypercube sampling is used. For Latin hypercube sampling, the first step is to define the probability distribution of every gas on hand. For this work, most of the time, equal distribution over a given range is assumed.

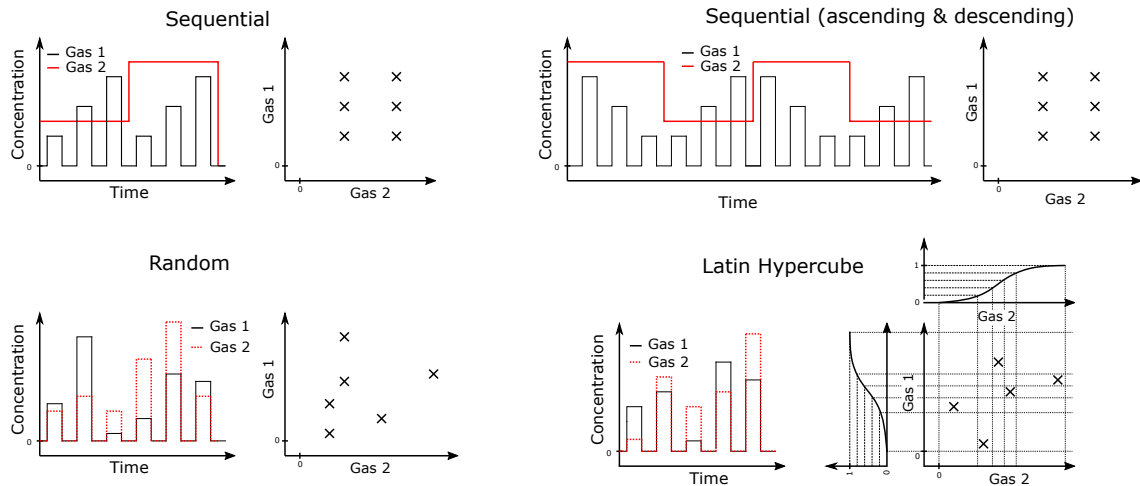


Figure 2.6: Example for a possible design of experiments concerning the selection of gas concentrations. Sequential subsampling, random subsampling, and Latin hypercube subsampling for calibration are shown.

The next step is to select how many UGMs (sample points) should be used. Afterward, the range for all gases is split into sections of similar probability (equidistant for equal distribution). Then, each sample is randomly picked for each gas from the concentration range. After selecting this first UGM no other UGM can have one concentration from the same sub-range of the specific gases. Figure 2.6 illustrates this approach in more detail. The benefit of Latin hypercube sampling is minimizing the correlation between different gases. A robust model can be built where predicting one gas based on two or more others is impossible.

2.4 Machine Learning

After the data is recorded, the next step is to build a data-driven model that predicts the target gas concentration based on the raw sensor output. However, before the specific data-driven models for gas sensor calibration are introduced, the general concepts of

analytical model building, Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) are introduced in the following.

2.4.1 General Machine Learning

In the first step, the terms analytical model building, AI, ML, and DL need to be structured. Analytical model building usually describes the workflow of a scientist. In this approach, a human analyses the problem, and afterward, an analytical model in the form of a closed-form mathematical solution is derived [90]. This formula describes the situation and allows the problem to be solved. Examples would be any model in the realm of physics (e.g., CO_2 IR gas sensors) [91].

Since analytic model building based on human calculation can be tedious or even impossible for some tasks, a new branch called AI evolved. AI uses the computer and its computing power to solve the problem [92]. Nowadays, AI is closely related to algorithms that can learn from data. However, in the domain of AI, not every model needs to be capable of learning. Instead, they can work based on a simple algorithm written by a human. Therefore, AI contains a sub-set called ML, representing the approaches capable of learning based on collected data. Instead of a human making observations, computer-based algorithms can make them based on previously seen data. ML can again be split into classic ML (e.g., decision trees, support vector machines, k-Nearest Neighbors, simple artificial neural networks) and DL. DL only differs from classical ML in terms of the model used to learn. While ML covers all learning algorithms (also DL algorithms), DL explicitly describes the methods that contain neural networks with more than one hidden layer [93].

After introducing the general idea of AI and ML, it is essential to understand that different problems require different learning approaches to solve the task. For ML, these methods can be divided into four domains: supervised learning, semi-supervised learning, unsupervised learning, and reinforcement learning [94–97]. The approaches differ in the availability of the data and the target/label to solve the task at hand. The data describes the current observation (e.g., pictures, numeric tables, or sensor readings). Based on the data, further information about the current observation is derived (e.g., what animal can be seen in the picture). The target describes the information that the model should extract from the data during inference (unavailable to the model when used in practice).⁷

⁷Inference refers to the model being used to classify a new observation.

For supervised learning, the data and the corresponding targets are available during the training. This allows a model to learn the dependencies between the data and the target. The model can then be used for evaluating new but similar observations. This approach is used in various fields, from computer vision to gas sensing tasks [98]. For semi-supervised learning [99], only some data with the desired target is available during training. One typical example is novelty detection for condition monitoring tasks [100]. In this case, only the data and the information that the machine was in working condition are available. The task is to build a model that analyzes new observations and decides if the machine is still working or deviates from its original state [101]. For unsupervised learning, only the data is available, and the task is to find patterns within the data. This approach includes methods like clustering [102, 103]. Reinforcement learning is slightly different from the just described methods [97]. For this approach, the data is not gathered from an external process. Instead, the learning algorithm is trained online in a virtual environment or via human feedback. Therefore, the data and the corresponding target (reward and punishment) are dynamically generated through operation and are used to train the model. This allows the algorithm, e.g., to learn to play games [104].

Within this work, exclusively supervised ML methods are used. Within supervised learning, the task can be either classification or regression. Most of the time, the task can be defined as finding a projection function $O_w(x)$ that predicts the correct output for an input instance x [105]. The task of finding the optimal $O_w(x)$ is often transformed into the minimization of a loss function. The loss function thereby represents the model's performance on the desired task. An example of a regression loss is given in Equation 2.1, with $|D|$ representing the number of observations in training set D , O the model output with parameter w (can be any model), x the independent observations ($x_i \in D$), and the corresponding target \hat{y} .

$$L(D) = \frac{1}{|D|} * \sum_{j=1}^{|D|} (O_w(x_j) - \hat{y}_j)^2 \quad (2.1)$$

A prediction is made with the help of the found projection function O_w . This function performs the projection of the test instance x_t to the output (classification or regression). A simple example of linear regression without bias is given in Equation 2.2.

$$O_w(x_t) = w * x_t \quad (2.2)$$

Before specific ML models are introduced, the most crucial part is to understand how to validate and test the model [106, 107] and how to rate their performance. First, the methods to rate the performance of a ML model need to be introduced. For classification, one common approach is to calculate the classification error. This error quantifies the amount of wrongly classified instances. However, this metric is not always optimal. In the case of medical studies, for example, it is sometimes more critical to avoid predicting false negatives [108]. Therefore, multiple metrics were developed, like F1-score, recall, sensitivity, and specificity. For regression problems, the most popular metric is the Root-Mean-Squared Error (RMSE). Yet, sometimes the RMSE is misleading, and other metrics need to be assessed, like R-squared. Therefore, it is always important to choose the most meaningful metric for the use case.

The next step is to perform a meaningful validation by splitting the data correctly. The data should at least be split into training and testing while focusing on solving the target task.⁸ The model learns from the training data, and the model's performance is tested using the remaining test data. However, most ML models require optimization before they reach optimal performance. Therefore, the training data is usually again split into training and validation [106]. As before, the training data is used to train the model, and the validation data is used to rate the performance for different model optimizations. The performance on the validation set can subsequently be used to find optimal model parameters that possibly perform best on the test data. If only a section of the original train data is left out for validation, the approach is called leave-out validation [109]. Cross-validation is performed for more robust models. For cross-validation [110], the evaluation (same model optimization parameter) is done multiple times, and in each step, different subsets of the original training data are left out. Throughout all iterations, every data point is precisely once in the validation set, and the mean performance across all validation sub-sets defines the performance of the specific model. This approach usually provides a more reliable estimation of the model's performance on the test data based on the validation error. Many more validation methods can be used to validate the model further, as described in [106]. Specifically crucial is appropriate validation and testing for detecting overfitting [106]. Overfitting means the model has learned unrelated systematic (e.g., noise) and has yet to learn the general underlying dependencies to perform well in real-world scenarios [111]. This can happen for multiple reasons, such as sub-optimal training data or insufficient data that

⁸One Example: If the stability over time of a model should be tested, the test data should focus on this task.

does not incorporate all aspects of the real world. Therefore, having an appropriate DoE and finding the optimal training, validation, and test split for the specific use case is always important.⁹

2.4.2 Classic Machine Learning

Classic supervised ML is defined in this work as all ML methods except neural networks, including DL [112]. Examples include support vector machines [113], Gaussian processes [114], linear regression [115], or decision trees [116, 117]. Those algorithms are used to learn the dependencies between input data and target. Classic ML performs exceptionally well for tabular data [118] (e.g., medical datasets)¹⁰ but can also be used together with raw sensor readings, images, or text. However, classical ML algorithms, solely used for regression and classification, tend to fail in this case because the information within the data is not easily accessible (e.g., due to the curse of dimensionality [119]).¹¹ Therefore, this work introduces classical ML not only as the regression and classification algorithm but as a stack/pipeline of algorithms used on the data [120, 121]. The stack consists of any pre-transformation of the raw data, a corresponding feature extraction, followed by feature selection, and a final classification or regression algorithm. This more sophisticated approach is called from here on Feature Extraction Selection Classification (FESC) or Feature Extraction Selection Regression (FESR).

The pre-transformation strongly depends on the data and is necessary to unify the input data and make the data suitable for the ML task. Text, for example, is tokenized, or pictures are scaled in size and value. During feature extraction, the pre-transformed input is converted into meaningful features. After extraction, each feature describes a specific property of the raw data (e.g., a feature from the frequency domain, a feature from the time domain, or other extracted features). This process highlights properties that may not have been obvious in the raw data and simultaneously suppresses unwanted noise. The feature extraction process is especially complex since it depends on the use case. Sometimes, it is necessary to design specific features for a use case based on domain-specific knowledge [122]. In industrial applications, the data is often transformed with the help of Fourier transformation to extract specific frequency bands that are

⁹For example, group-based validation is essential if many samples are very similar within the training data to check for interpolation capabilities.

¹⁰Tabular data is specified as strongly heterogeneous data, for example, medical datasets (blood pressure, height, and patient weight).

¹¹Curse of dimensionality means that the model cannot learn the intended dependencies because of too many input features/dimensions.

characteristic of the analyzed machine. Similarly, methods like Principal Component Analysis (PCA) can create new features for tabular data. But even with specifically designed features, it is still possible to run into the curse of dimensionality [119], which means there are too many features, and the classification or regression still fails. Therefore, feature selection is performed to eliminate unnecessary or redundant features. For this task, there are three general fields: filter methods (calculates a metric independent of the classification/regression algorithm, e.g., Relief or Pearson correlation), wrapper methods (searches through subspace of features with the help of an additional classification/regression algorithm, e.g., recursive feature elimination), and embedded methods (feature selection is part of the classification/regression algorithm, e.g., decision trees) [123, 124]. The last part of the FESC/FESR can then still make use of all the algorithms that classic ML provides (e.g., Support Vector Machine (SVM), decision trees, Gaussian processes). This can be done because after the feature extraction and selection, the data is basically transformed into tabular data with a reduced feature set, and the algorithm will perform reasonably well.

Two examples of a framework that performs feature extraction, selection, and classification/regression are the FESC/FESR toolbox [125] and the DAV³E toolbox (Data Analysis and Verification/Visualization/Validation Environment) [49] both developed at the LMT. Both toolboxes provide similar functions for feature extraction, selection, and classification/regression. However, the FESC/FESR toolbox is tailored explicitly for an automated approach. Therefore, different combinations of algorithms (extraction, selection, and classification/regression) are tested in a predefined validation scenario to find the optimal combination for the use case automatically. The feature extraction algorithms are selected to be complementary, which means that features are either extracted from the time domain, from the time/frequency domain, or from the frequency domain. The feature selection is generally used in wrapper fashion, and the inner functions are complementary (filter or wrapper). As explained above, the final learning algorithm (classification or regression) can be any algorithm that suits the use case. In the FESC/FESR toolbox, 15 algorithm combinations are tested for regression and classification (five extractors, three selectors, one classification/regression). The five feature extraction methods are introduced in more detail in the following. The statistical moment's extractor divides the signal into N equidistant segments and calculates the mean, variance, skewness, and kurtosis for every segment [120]. For the best Daubechies wavelet extractor, the most critical time-frequency coefficients are returned based on the highest mean coefficients observed during training [126, 127]. The best Fourier

coefficients extractor similarly returns the top 10 % frequency bands with the highest mean energy based on the training [128]. The principle component analysis extractor returns the first 500 PCA components. The Adaptive Linear Approximation (ALA) extractor splits the signal into linear segments based on the reconstruction error of the training data and returns the mean and slope for these segments [120, 129, 130]. Feature selection is twofold. First, one of three algorithms is used to specify the ranking used in the second stage to find the optimal subset (wrapper method). The first ranking method tests for the Pearson correlation and reduces the features by default to 500 essential features. The ranking within these 500 features is based on the correlation coefficient. The Pearson selector can either be used as the stand-alone ranking method or as a pre-selector for the other two computationally expensive ranking methods to limit the number of features from the start. The recursive feature elimination algorithm [131] repeatedly removes one feature from the 500 pre-selected features based on the relevance (either importance within the support vector machine with linear kernel or the factor within a least squares regression) to reevaluate the ranking of the feature set. Thereby, getting a different ranking compared to the Pearson selection of the features is possible. The last ranking method is Relieff [132]. This method considers the locality of the different classes based on the different features. Features that separate the classes well will get a higher ranking. The actual selection happens in wrapper fashion by iteratively testing the top 200 features depending on the ranking of one of the three algorithms and testing the subsets from 1 - 200 with the help of the chosen classification or regression algorithm. The best subset based on a 10-fold cross-validation is selected for the final model building. For classification, only one algorithm is currently available in the toolbox; this algorithm consists of two stages. First, the remaining features are transformed with the help of Linear Discriminant Analysis (LDA) [133, 134]. Afterward, the Mahalanobis distance between the new sample and the mean of each class is used for classification [133]. Mahalanobis was chosen because it also considers the scattering of the training data.¹²

Complementary to classification, linear regression is a suitable algorithm for the regression task. This method directly attempts to find a w and b for Equation 2.3, which minimizes Equation 2.4 where \hat{y} represents the actual target.

$$O_w(X) = Y(X) = w * X + b \tag{2.3}$$

¹²This stack of 15 possible combinations showed promising results for industrial datasets [120].

$$L(D) = \frac{1}{|D|} * \sum_{j=1}^{|D|} (w * x_j + b - \hat{y}_j)^2 \quad (2.4)$$

However, linear regression can fail if X contains correlated variables. Therefore, Partial Least Squares Regression (PLSR) [135, 136] is used as it works better under such conditions because it creates new orthogonal features.¹³ The data X and target Y are decomposed into $X = TP^T + E$ and $Y = UC^T + F$ to find a projection C from T to Y , where T contains new orthogonal features ($Y = TC^T + G$ with G).¹⁴ This is similar to the principal component regression, where X is transformed with the help of principal component analysis. However, in this case, the transformation takes X and Y into account in order to find T and U with maximized covariance. PLSR is an iterative approach. In each step n (n represents the number of components), the scores t_n of X are estimated with $t_n = X_n * w_n$, where w_n is the weight vector $w_n = X_n^T * y$ ("covariance" feature to y). With t_n estimated, it is possible to calculate the loadings of x (p_n). Afterward, the coefficients are calculated with the help of linear regression ($c_n = y^T * t_n$), where c_n is only a scalar that represents the contribution of t_n for the final prediction. After each step, the matrix X is deflated $X_{n+1} = X_n - t_n * p_n^T$, and the process is repeated until X is empty or the number of components is reached. The prediction is made with $Y = X * B + G$, where B can be constructed with $B = W * (P^T * W)^{-1} * c$ (project c back to original space), and G is a constant value ($c_0 - p_0^T * B$). The matrices W , P , T , and the vector c are constructed by combining all intermediate results in a large matrix or vector [135]. For PLSR, the algorithm used is the SIMPLS algorithm [137].

A detailed introduction of the FESC/FESR toolbox and the algorithms can be found in [125, 133], and a short description of DAV³E (Data Analysis and Verification/Visualization/Validation Environment) is given in [49]. Compared to the FESC/FESR toolbox, DAV³E is specifically tailored for gas sensor applications and includes a graphical user interface.

2.4.3 Neural Network Basics / Deep Learning

Before it is discussed how ML can help to calibrate gas sensors, a few general neural network basics and DL methods need to be introduced. As described above, neural networks are a part of ML. They are ML models that can learn to solve a task with the

¹³Capital letters represent matrices, and lowercase letters are vectors.

¹⁴ T & U : scores, P & C : loading's, E & F & G : residuals.

help of backpropagation, gradient descent, and the data together with the corresponding targets. In general, neural networks are heuristic learning algorithms that build a computation path that transforms the input into an output (in this case, regression or classification). The basis of the neural networks in the form of the perceptron was developed by Frank Rosenblatt [138] in 1958. Since then, a massive development followed, leading to DL as we know it today [139]. The following section describes the essential parts of a neural network.

2.4.3.1 Neurons and Neural Networks

The general structure of a neuron/perceptron can be found in Figure 2.7. A neuron was inspired by biology, respectively the brain, and consists of n inputs, weights, a bias, an activation function, and one output [138]. During processing, each input is scaled with the corresponding weight. Afterward, all scaled inputs are summed up, and the bias is added. The weights and the bias represent the learnable parameters and define the dependencies between the input and output of the neuron. After multiplication and summation, the result is processed with a so-called activation function, cf. Equation 2.5 [140], with x_i being the input.

$$O_w(X) = f_{activation}(net) = f_{activation}\left(\sum_{i=1}^n w_i * x_i + b\right) \quad (2.5)$$

The activation function can have multiple forms; the most popular are sigmoid, hyperbolic tangent, and rectified linear unit (ReLU). Those activation functions need to be nonlinear, differentiable, and fast to calculate. An exception is the ReLU function, which is not differentiable. However, this is only at one point, which can be sufficiently approximated. An activation function is needed because it introduces nonlinearity and thereby creates various complex features and nonlinear segmentations. With different activation functions, different forms of features can be created.

Today, the most popular activation function is the ReLU activation. The ReLU function is especially useful as it has the special properties of generating linear dependencies in the positive range while still being nonlinear (e.g., linearly separable classes). Furthermore, the ReLU activation does not suffer from the vanishing gradient, like sigmoid activation (zero gradients for large negative and positive sums), which prevents the neuron from learning [141]. Nevertheless, ReLU is not the optimal activation function

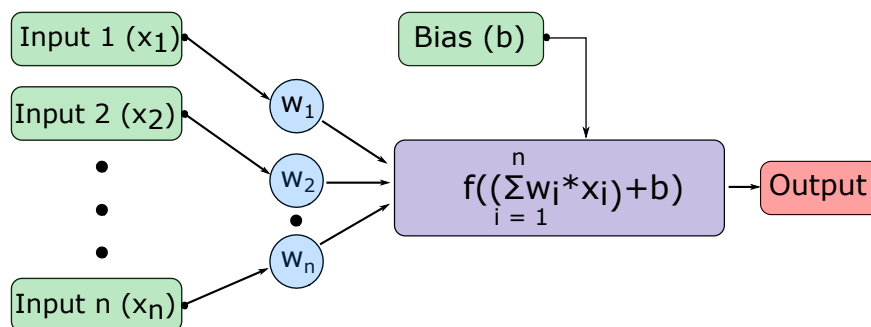


Figure 2.7: Example of one neuron.

since it suffers from drawbacks like dying neurons [142]. Therefore, activation functions are still heavily researched [143].

A neural network consequently consists of multiple neurons that resemble a network (cf. Figure 2.8). The neurons are usually structured in layers, and the processing chain proceeds from left to right. The neurons on the left are called the input layer, while the neurons on the right resemble the output layer. The layers of neurons in the middle represent the so-called hidden layers. If multiple hidden layers are present, the neural network is called deep neural network/Multi-Layer Perceptron (MLP) [144]. Input data primarily determines the left part of the neural network. For example, different layers are needed depending on the input. If the input is a picture, a Convolutional Neural Network (CNN) might be the best choice, while for an ensemble of features, a fully connected layer might work best. The output layer will change according to the task on hand. For classification problems, the last layer is typically a fully connected layer with as many neurons as classes trained. The activation function of this layer is then a sigmoid/softmax function that resembles the probability of a particular class. For regression problems, the last layer is usually a fully connected layer with only one neuron, and the activation is typically the identity. However, regression tasks can also have multiple outputs if multiple regression tasks need to be performed within the same network simultaneously. The hidden layers in the middle can typically have various forms and shapes.

In a simple regression network, as represented in Figure 2.8, the output of every neuron can be calculated with Equation 2.6, with j being the layer number, k the neuron in the current layer, and i the input number. The overall function then depends on the weights and biases of the whole network. As for every ML algorithm, the goal is

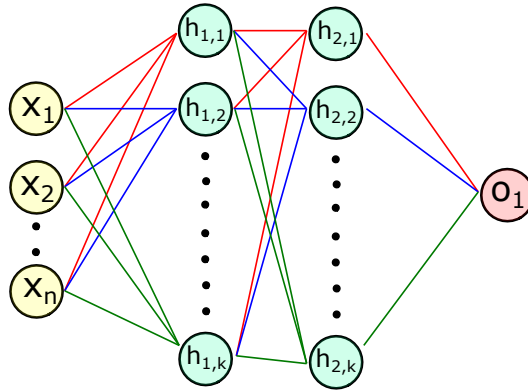


Figure 2.8: Example of one neural network consisting of two fully connected hidden layers.

to adapt the computation path to optimize the regression or classification task. As introduced above, this is done by minimizing the loss function. During training, the neurons' weights and biases are optimized to reduce the loss. The metric or loss to determine the performance for regression is usually the root mean squared error, and for classification, the cross-entropy loss; however, many more loss functions can be used for specific use-cases [145].

$$O_{j,k}(x) = f_{activation}(net_j) = f_{activation}\left(\sum_{i=1}^n w_{i,j,k} * O_{j-1,i}\right) + b_{j,k} \quad (2.6)$$

The following chapter dives more into detail to better understand how a neural network works and how they are optimized.

2.4.3.2 Backpropagation and Gradient Descend

Backpropagation and gradient descent are fundamental components for tuning the trainable parameters (weights and biases) in the neural network [146, 147] to minimize the loss function. During backpropagation [140, 148–150], the gradient for each neuron involving its input, output, and error/loss is calculated (chain rule [151]). With the help of gradient descent, the weights are adjusted to find a minimum (most of the time local minimum) of the loss function L (e.g., mean squared error or cross-entropy loss). Equation 2.7 demonstrates how the loss of a single sample is calculated. O_j represents the output of the specific neuron j (currently investigated). The sum iterates over all outputs the network possesses. Equation 2.8 shows how the gradient for a specific weight is calculated based on one observation with the help of the chain rule. The parameters

are the same as in Equation 2.6, and O_i is the output of the neuron from the layer to the left.

$$L(x) = \frac{1}{2} * \sum_{j=1}^n (O_j(x) - \hat{y}_j)^2 \quad (2.7)$$

$$\frac{\partial L}{\partial w_{i,j}} = \frac{\partial L}{\partial O_j} * \frac{\partial O_j}{\partial net_j} * \frac{\partial net_j}{\partial w_{i,j}} = \frac{\partial L}{\partial O_j} * f'_{activation}(net_j) * O_i \quad (2.8)$$

As shown in Equation 2.9 δ is now specified as the loss of the neuron with respect to the input of the neuron.

$$\delta = \frac{\partial L}{\partial O_j} * f'_{activation}(net_j) \quad (2.9)$$

In order to calculate the gradient for a specific weight, two cases have to be distinguished: either the neuron is an output neuron or a neuron in the hidden layer. With the help of Equations 2.10, 2.11, it is possible to calculate every gradient in a network from the end to the beginning (backpropagation).¹⁵ A more detailed derivation can be found in [140].

$$\delta_{output} = (O_j - \hat{y}_j) * f'_{activation}(net_j) \quad (2.10)$$

$$\delta_{hidden} = \sum_{k=1}^n (\delta_{k,previous} * w_{k,previous}) * f'_{activation}(net_j) \quad (2.11)$$

After the gradients are accessible, the gradient descent algorithm tries to find the optimal weights and biases regarding the objective function (loss). The process of finding the optimal weights to minimize the objective function is illustrated in Figure 2.9, and the basic algorithm can be found in Equation 2.12 [147].¹⁶

When training from scratch, the different weights and biases are randomly initialized (e.g., Glorot initializer [153]). During training, the data is repeatedly fed through the network. In each epoch/step, the objective function based on the training data is traversed to find the optimum (e.g., gradient descent) [154]. Thus, the weights (w) are adjusted slightly in each step until a suitable solution is found. The amount the weights are allowed to change in each iteration is determined by the learning rate α (usually

¹⁵Previous: layer to the right, k: neuron in the layer.

¹⁶ W_{ij} represents the weight between neuron i from the layer to the left and the neuron j from the current layer.

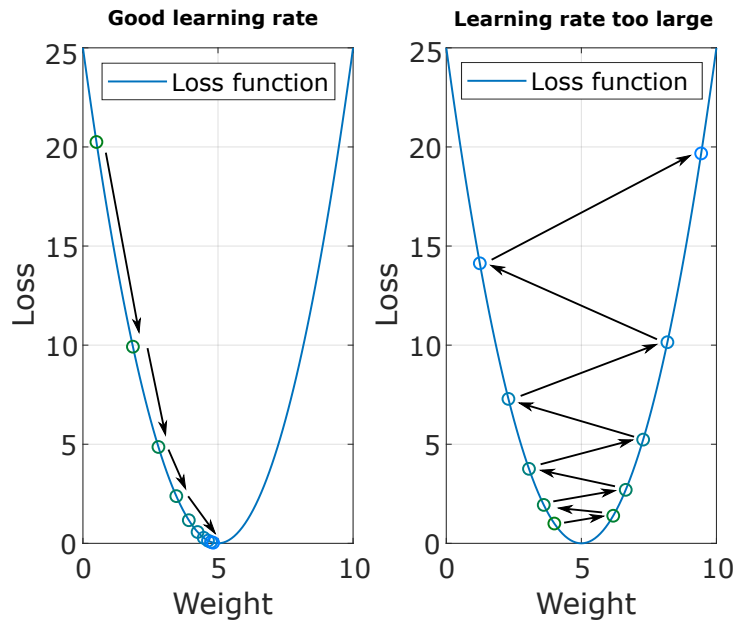


Figure 2.9: Illustration of gradient descent to find the optimum (adapted from [152]).

between 0 and 1) and the gradient. If the learning rate is too large, it is possible that the weights are adjusted too much in each iteration, and the network will not find a good solution. If the learning rate is too small, finding the minimum takes infinite time (cf. Figure 2.9).

$$W_{ij}^{new} = W_{ij}^{old} - \alpha * \frac{\partial L}{\partial W_{ij}} = W_{ij}^{old} - \alpha * \delta_j * O_i \quad (2.12)$$

The gradient descent algorithm is one of the earliest methods combined with neural networks to optimize the weights [138, 147]. Since then, the algorithm has changed. Although gradients are still calculated or estimated, many improvements have been made to search through the loss function in order to find the minimum faster and more precisely. While for gradient descent, the gradient of the complete dataset is calculated before weights are updated, the gradient is nowadays approximated with the help of methods like stochastic gradient descent [155, 156]. Thereby, a single random instance is used to approximate the gradient of the whole dataset, and the weights are updated with each sample. That usually speeds up the training process as progress is made with each instance. However, this method leads to quite noisy descents to the minimum. Therefore, stochastic mini-batch gradient descent was introduced [156, 157]. This approach estimates the gradient with a set of samples (mini-batch, e.g., 64), allowing faster convergence. Nevertheless, this approach tends to have a lot of

unnecessary orthogonal movement to the original gradient, and therefore, new methods like RMSprop [158, 159] and momentum [160] were developed—RMSprop updates Equation 2.12 with Equation 2.13. RMSprop minimizes the orthogonal movement to the actual gradient by normalizing the gradients (usually $\beta = 0.9$ and $\epsilon = 10^{-8}$) [158].

$$W_{ij}^{new} = W_{ij}^{old} - \alpha * \frac{\partial L}{\partial W_{ij}} * \frac{1}{\sqrt{RMS_{dW_{ij}}^{new} + \epsilon}} \quad (2.13)$$

$$RMS_{dW_{ij}}^{new} = RMS_{dW_{ij}}^{old} * \beta + (1 - \beta) * \frac{\partial L}{\partial W_{ij}}^2 \quad (2.14)$$

Similarly, momentum remembers the previous gradients (cf. Equations 2.15, 2.16) and finds a mean gradient closer to the actual gradient, thereby speeding up the training process ($\beta = 0.999$) [160]. Both methods profit from the benefit of calculating the gradient based on a subset of input samples and speed up the convergence with the help of memory terms and first and second-order approximation of the gradient. The publications [158, 160] showed that all those methods provide several drawbacks and benefits. However, the new methods provide faster and better results than basic gradient descent.

$$W_{ij}^{new} = W_{ij}^{old} - \alpha * M_{dW_{ij}}^{New} \quad (2.15)$$

$$M_{dW_{ij}}^{New} = M_{dW_{ij}}^{Old} * \beta + (1 - \beta) * \frac{\partial L}{\partial W_{ij}} \quad (2.16)$$

Today's state-of-the-art approach for optimizing a neural network is Adaptive Moment Estimation (Adam). Compared to the other methods, Adam combines the benefits of mini-batches, statistic gradient descent, RMSprop, and momentum in one approach and extends those by a bias correction term. This additional term prevents the two other terms from converging to zero. Because Adam can combine the different benefits of all of the mentioned methods, it is the most widely used technique [147, 161]. Equation 2.19 summarizes how the different methods are combined. A more detailed review of all the new developments around the gradient descent algorithm can be found in [147, 162].

$$W_{ij}^{new} = W_{ij}^{old} - \alpha * \frac{M_{dW_{ij}}^{New}}{\sqrt{RMS_{dW_{ij}}^{new} + \epsilon}} \quad (2.17)$$

$$M_{dW_{ij}}^{New} = (M_{dW_{ij}}^{Old} * \beta_1 + (1 - \beta_1) * \frac{\partial L}{\partial W_{ij}}) * \frac{1}{1 - \beta_1^{numIteration}} \quad (2.18)$$

$$RMS_{dW_{ij}}^{new} = (RMS_{dW_{ij}}^{old} * \beta_2 + (1 - \beta_2) * \frac{\partial L}{\partial W_{ij}}) * \frac{1}{1 - \beta_2^{numIteration}} \quad (2.19)$$

If a minimum is found, it is essential to note that in every data-driven modeling, the optimized loss function and the underlying data are only a sparse representation of reality. Therefore, an optimum of any model does not mean that a generally applicable model was found. Furthermore, it is always essential to test different methods as there is no algorithm that fits all tasks best [163, 164].

2.4.3.3 Special Hidden Layer

After introducing the general concepts of neural networks (neurons, backpropagation, gradient descent), the next step is to build the network in detail. In order to achieve the best possible performance, the neural network has to be designed with care. The most crucial part, thereby, is to select suitable neural network layers. Distinct hidden layers and their tunable parameters can be used between network input and output to tailor the model for a specific problem [98].

The most widely known layer is the fully connected layer (derived from the perceptron [138]). This layer consists of multiple neurons, and each neuron takes in the output of every neuron from the previous layer. Afterward, the inputs are scaled with the corresponding weights, summed up, and modified with the activation function, as explained above. Therefore, the number of weights is defined by the number of inputs and number of neurons in the fully connected layer (plus one for each neuron for the bias). The output size of this layer is then defined by the number of neurons in this layer. The tunable parameters that can be used to optimize this layer for a specific task are usually the number of neurons and the activation function. This layer is often used as a regression or classification layer at the end of convolutional neural networks [98, 165] or to process tabular data in medicine [166].

The convolutional layer [167, 168] (cf. Figure 2.10) is mostly used for computer vision [169, 170] or speech recognition tasks [171]. This layer's unique property is that it can extract local features by sliding a kernel over the input. When sliding over the data, the kernel is convoluted in each step with the data to extract meaningful features [98]. The extracted features are based on the kernel weights learned during training (e.g., edges or slopes within the input image). Figure 2.10 gives an example of such a convolution. The input data for a convolutional layer usually has three dimensions: x and y resemble the usual size of the data frame (e.g., picture), and the third dimension represents the

number of channels (e.g., color channels in pictures). The tunable parameters of such a layer are the kernel size, striding, padding, and the number of filters. The first parameter to discuss is the kernel size. This parameter defines the window size regarding width and height. The kernel is moved over the data based on the second parameter, the striding size. This parameter defines how the kernel is moved from the upper left corner over the whole frame. Usually, the striding and kernel sizes are chosen to cover the essential features best. However, it is possible that the image is not properly/evenly processed with a specific kernel and striding size. In this case, it is necessary to perform padding at the edges to cover the image evenly. For padding, different methods are viable: extend the data by 0 around the edges or copy the pixels at the border. A more complete summary of padding can be found in [172]. The last parameter that can be tuned for convolutional layers is the number of filters. Multiple filters lead to multiple independent kernels to extract different features. More than one filter is reasonable as each filter usually extracts independent features, generating many additional robust features. The number of weights in a convolutional layer can be calculated with Equation 2.20. The parameter x_{kernel} represents kernel size in x direction, y_{kernel} defines the kernel in y , and $numFilters_{Out}$ stands for the number of filters present in the convolutional layer.

$$numW = x_{kernel} \times y_{kernel} \times numChannels_{In} \times numFilters_{Out} \quad (2.20)$$

The output of a convolutional layer with an Input of $20 \times 20 \times 3$, 100 filters, and a striding and kernel size of 2×2 then has an output size of $10 \times 10 \times 100$ and contains $2 \times 2 \times 3 \times 100$ weights and 100 biases.

Another layer is the batch normalization layer, usually used to introduce an additional regularization effect into the neural network. The layer is placed between two hidden layers, standardizing each intermediate output independently during training and inference (subtract mean, then divide by standard deviation). During training, the standardization is done with the help of the mini-batch and during inference with the previously learned mean and standard deviation based on the training data. There are two standard methods for learning the two moments' (mean and standard deviation) used during inference. One way is to calculate them during the whole training process with the help of momentum (similar to the momentum optimizer above) [173]. Alternatively, the parameters can be calculated in the last epoch in one step [174]. It is important to mention that Keras, the most commonly used library for DL, uses the first method [173], while Matlab, which is used for this work, uses the second [174]. This can lead

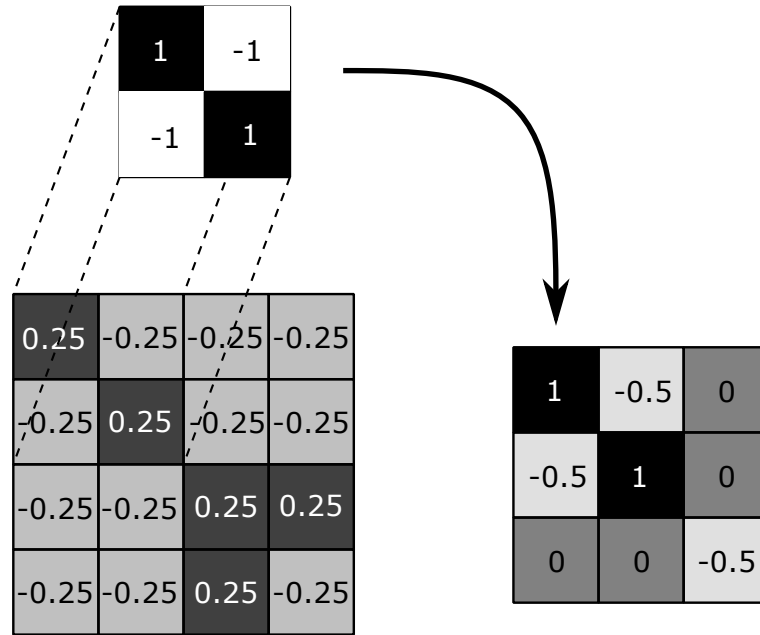


Figure 2.10: Example for a convolutional layer with an input image with size 4×4 , a kernel size of 2×2 , a striding size of 1×1 , and without padding.

to significant differences; however, Matlab performs slightly better. In any case, the batch normalization layer leads to minimized variance in the output of the neurons across the network, which eases finding suitable features for subsequent layers [175, 176]. During training, this is especially helpful in reducing the covariate shift. Covariate shift describes the problem that the input of subsequent layers can be vastly different in distribution during each training epoch, which slows down the training process due to large weight changes. Furthermore, the problem of vanishing and exploding gradients is reduced by always remaining in the sensitive range of many activation functions [176–179]. A complete explanation of batch normalization that considers the scaling parameter can be found in [176–179].

Likewise, the Dropout layer [180] is also used for regularisation. Instead of passing all outputs to the next layer, this layer sets the value of a predefined number of outputs of a specific layer to zero during training (in every iteration, random outputs). This is done to force the model to generalize, to not rely on single features, and to adapt to the general problem [175, 180].

Although the activation function was already introduced, some libraries (e.g., Matlab and Keras) allow users to add layers to the network without an activation function. The activations are later placed separately and can, therefore, be listed as separate layers.

The fully connected layer, convolutional layer, batch-normalization layer, dropout layer, and activation layer are the layers used within this work. Nevertheless, a complete list with more details can be found in [181, 182] (e.g., self-attention layer, LSTM - layer, GRU - layer). After introducing the primary layers, one important property that all layers with trainable parameters share is the possibility of applying regularization to the weights. This can be any regularization function [175, 183]. These regularizations only differ in the way they penalize overcomplicated models. Throughout this work, a constant L2 regularization is used. L2 regularization keeps the weights of features within the network small by punishing large weights to avoid getting overcomplicated models [183]. The alternative L1 regularization can reduce the number of features. L2 can be interpreted similarly to the C parameter of a support vector machine [113]. With a large C , all the emphasis is assigned to the error, which leads to overcomplicated models (overfitting). At the same time, a small C tries to minimize the two-norm of the separating hyperplane, which leads to simpler models.

2.4.3.4 Hyperparameters

When faced with an ML problem, the first step after analyzing the data is finding a suitable model. The model can be from the classic ML domain like linear regression [184], and decision trees [116], or from the DL domain like convolutional neural networks [185]. In any case, the specific hyperparameters of the model should be optimized with the help of the training and validation data. If one chooses neural networks, the first step is to find a suitable architecture. It is recommended to search for similar tasks already solved with neural networks. After the network is specified, the next step is to optimize the architecture and corresponding hyperparameters to solve the desired task, which is commonly referred to as neural architecture search [186–188]. Neural architecture search can include investigating different types of layers, how they are interconnected, and optimizing the different hyperparameters of the layers. As covered in this work, architecture optimization only includes finding the optimum number of layers and their optimal hyperparameters. The hyperparameters optimized are usually the tunable parameters specified for the different layers and a few training parameters like learning rate, regularization, and mini-batch size. Throughout this work, the number of

neurons in a fully connected layer, the number of filters together with the striding and kernel size in a convolutional layer, the dropout rate, and the initial learning rate are optimized. For the optimization task, the first step is to find promising ranges for each parameter (e.g., tune the number of neurons between 100 - 2000). These ranges can be determined by analyzing networks used for similar tasks or by first testing which range provides reasonable results on the validation data. After suitable ranges are defined for all parameters, searching for a suitable network configuration is possible. One method would be to test every combination (grid-search), which can be extremely computationally expensive. Therefore, automated hyperparameter optimization should be used. Examples include random search, gradient-based optimization, or Bayesian optimization [188]. Previous studies showed that Bayesian optimization can provide faster and better results than the other methods [187–189]. This study introduces Bayesian optimization specifically as a higher order ML model that searches through the predefined ranges of parameters to find the set of parameters that minimizes the loss of the network. Unlike an exhaustive grid search, Bayesian optimization selects the most promising set of hyperparameters for subsequent evaluations instead of searching iteratively through every combination. This ensures that it is possible to find the optimum as fast as possible, which is crucial for neural networks, where evaluating one model can already take several minutes, hours, or even days [Paper 1]. The following explains how Bayesian optimization identifies the most promising candidates for the subsequent evaluation.

The Bayesian optimization [190–192] first takes a few uniformly distributed guesses for the hyperparameters and evaluates the neural network with these parameters (e.g., Matlab makes four random guesses [193]). A Gaussian process regression model is built with the help of these four data points and all subsequent evaluations. This regression model works similarly to kernel regression. However, the main difference is that this algorithm returns the expected loss value of the network for the selected hyperparameter and the standard deviation of this prediction. Because the uncertainty is assumed to be Gaussian, the response of the Gaussian process can be expressed as shown in Equation 2.21.¹⁷ How the Gaussian process regression works in detail is covered in Appendix A and in the following publications [194–196].

$$h(x) \sim \mathcal{N}(\mu(x), \sigma(x)) \tag{2.21}$$

¹⁷ \mathcal{N} : normal distribution; μ : mean; σ : standard deviation.

After the first few random evaluations, the Gaussian regression model can estimate the loss of the neural network together with the corresponding uncertainty for all possible hyperparameter combinations. Thereby, it is possible to quickly evaluate the Gaussian regression model for many different sets of hyperparameters and identify the next most promising set of hyperparameters. Convenient methods are here the lower confidence bound ($x_{bestNext} = \operatorname{argmin}_x(\mu(x) - 1.95 * \sigma(x))$), or the improvement probability (how likely is the next point to be better), or the expected improvement (expected improvement of point x: sum over all possible values smaller than the current best times probability) [193]. The more complex improvement estimators can also consider the specific model's training time or consider a trade-off between exploitation and exploration.¹⁸ A more in-depth explanation can be found in [193], and an example is given in Figure 2.11.

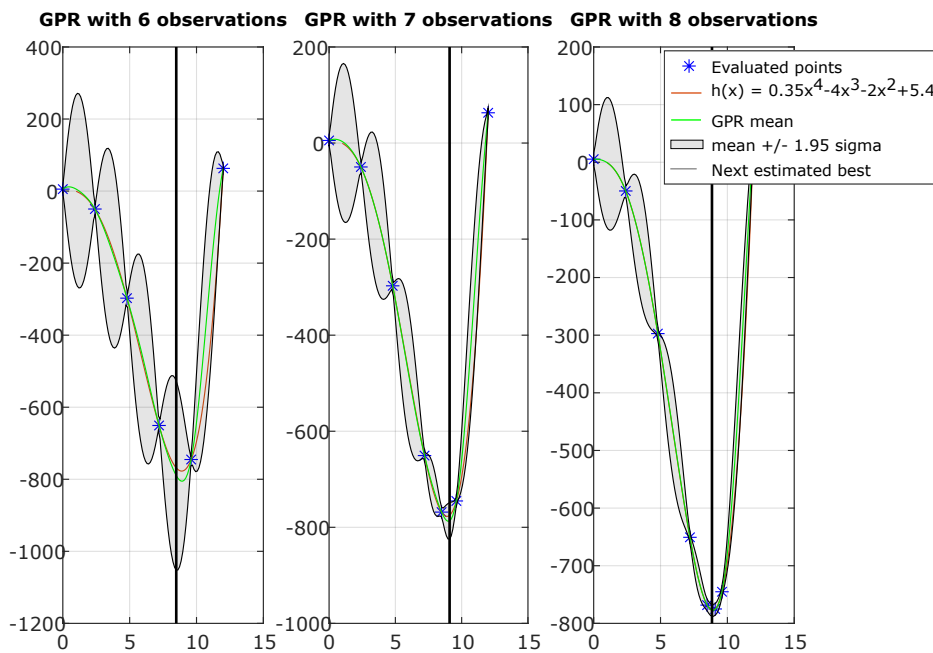


Figure 2.11: Example of Bayesian optimization with a Gaussian Process Regression (GPR) for multiple steps with lower confidence bound (adapted from [197]).

¹⁸Exploitation focuses on the area where it is more likely to find a minimum, while exploration also investigates unlikely/new areas.

2.4.3.5 Advanced Methods

It is widely known that DL works quite well for computer vision and large language models [169–171]. However, deep neural networks are frequently outperformed when the neural network is trained from scratch (random initialization). Especially, decision trees can easily outperform neural networks for tabular datasets [118, 198]. However, if transfer learning is applied, neural networks can again compete with the other techniques regardless of the form of the data [199]. Furthermore, neural networks suffer from the image of being impossible to understand because they are mostly seen as black box models. Therefore, it is crucial to understand the model’s inner workings to build faith in its capabilities. Consequently, the following paragraphs introduce transfer learning and Explainable Artificial Intelligence (XAI) for DL models.

Transfer Learning In general, ML aims to find the optimum of the objective function. However, as no infinite amount of data is available, and the user designs the optimization function, which only approximates the real world, it is impossible to find the global optimum. Nevertheless, the more data available, the more general the model can become. If only a small dataset is available, this can be problematic. In this case, good performance may be achieved on the training data while the model fails on the test data (e.g., overfitting [175]). Still, there are methods to find a suitable model quickly with only a small dataset. One method that has highly improved the prediction quality of DL models for computer vision and is frequently used to adapt large language models is called transfer learning [200–202]. For transfer learning, it is not necessary to train a new model from scratch every time; instead, an existing model trained on a similar task can be reused by slightly adjusting it (cf. Figure 2.12).

During the standard training process of a neural network, the weights are adjusted to resemble the optimal projection from input to output. In the case of training from scratch, the different layers’ weights are randomly initialized (e.g., Glorot initializer [153]). For transfer learning, the weights of a previously trained network are used as a starting point. These old weights usually come from a network trained on a similar task but on a different dataset. The idea is that the underlying function optimized in the first model is similar to that optimized for the new model (cf. Figure 2.13).

Therefore, it is unnecessary to search the whole space again; instead, restrict the search to a more focused and efficient region. This is beneficial if only a small training set is available and the data is only a sparse representation of the underlying data

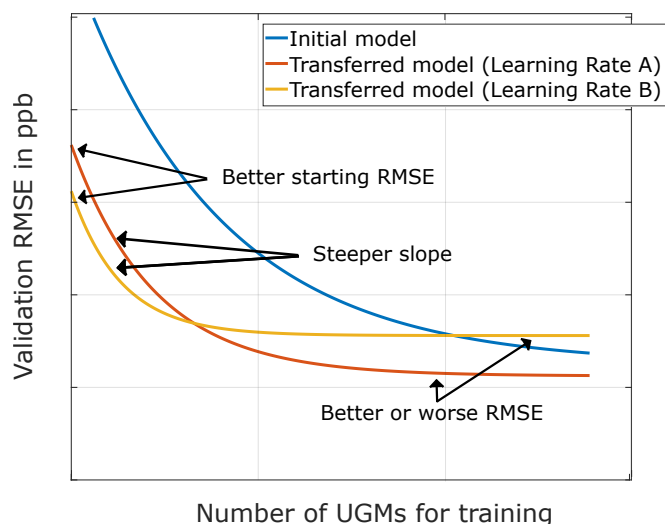


Figure 2.12: Transfer Learning: The effect of transfer learning for different hyperparameters and number of training samples. Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.

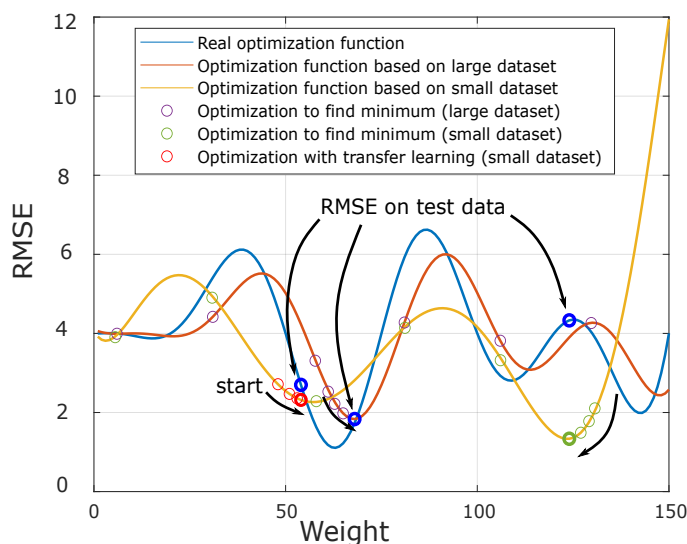


Figure 2.13: Transfer Learning: The effect of transfer learning for different dataset sizes. The optimization function, if infinite data is available, is illustrated together with the slightly different optimization functions that result from incomplete datasets. A comparison of the optimization is shown between a large dataset (from scratch), a small dataset (from scratch), and a small dataset with transfer learning. For transfer learning, the learning rate is reduced, and only limited data is available.

distribution. The restriction is usually done by limiting the learning rate (fine-tuning) or freezing some weights. An example is shown in Figure 2.13. For larger learning rates,

the subsequent point reached through backpropagation is far from the starting point set by the weights of the old model. Small learning rates show that if a suitable starting point is chosen, it is possible to achieve good results by only adapting slightly to the new training samples.

Of course, multiple methods from the field of transfer learning have their origin outside of neural networks and can, therefore, be applied to classical ML and DL (e.g., Direct Standardization (DS) [73] or instance weighing [203]). However, they do not perform as well as transfer learning from the field of DL.

Explainable AI The problem with overcomplicated neural networks is that they tend to be hard to understand. This is unacceptable for some areas that concern human safety or health (e.g., self-driving cars or medical diagnosis [204–206]). Therefore, XAI methods have been developed to understand the dependencies between input and output.

For the classical ML methods (e.g., FESR), the features most correlated with the target or interfering disturbances are easy to extract and understand. This is much more difficult for neural networks because the feature extraction happens internally and can not be easily accessed. Thus, new methods have to be developed to get a more sophisticated understanding of the internally extracted features. These methods can generally be divided into black box and white box explainer [207]. Black box explainers do not interfere with the model’s inner workings but determine the relevance score based on the input and output of a ML model. These methods have the benefit that they are not restricted to neural networks but can be used with every model. White box models, on the other hand, take the model’s inner workings into account to determine in which part of the input most of the information is hidden. A few examples are Local Interpretable Model-Agnostic (LIME) [208], occlusion map (occlusion sensitivity) [167], Class Activation Maps (CAM) [209], Gradient Attribution Map (gradient map) [210], and their derivatives, to only name a few. In the following, the methods used within this thesis are explained in more detail. One frequently used method in computer vision is called occlusion map/occlusion sensitivity (cf. Figure 2.14) [167]. This black box method slides a cover over the image, and the image is reevaluated at each position. The difference between the prediction with occlusion and the original is calculated, determining the importance score (IS) (cf. Equation 2.22, 2.23).

$$IS = P_{original} - P_{occluded}(classification) \quad (2.22)$$

$$IS = \left| \frac{RV_{original} - RV_{occluded}}{\text{mean}(RV_{training})} \right| (regression) \quad (2.23)$$

The cover usually has a size of $a \times b$, where a represents the number of vertical pixels, and b represents the number of pixels in the horizontal direction of the cover. The user usually determines these values specifically for the use case. Furthermore, the elements within the cover to occlude the image can have multiple values. Either they replace the original values in the occluded area with a constant value (e.g., grey in picture classification), or they replace the original value with the expected value for that pixel (generated from training data) [167, 211, Paper A]. The cover is moved over the picture based on the striding size (similar to convolutional layers). The striding size determines the distance the cover is moved over the picture in one step. In the example of a striding of 5×10 , the cover would first be moved in the x direction (10 pixels per step) until it hits the border, then moved 5 pixels down and back to the left of the image. After that, the cover is again moved in the x direction. With every pixel covered, the process is finished. Similar to the cover size, the striding depends on the use case.¹⁹ This process results in an occlusion map smaller than the original picture with or without padding. Therefore, interpolation is needed to bring the occlusion map back to the original form, which then resembles an importance score of every pixel.

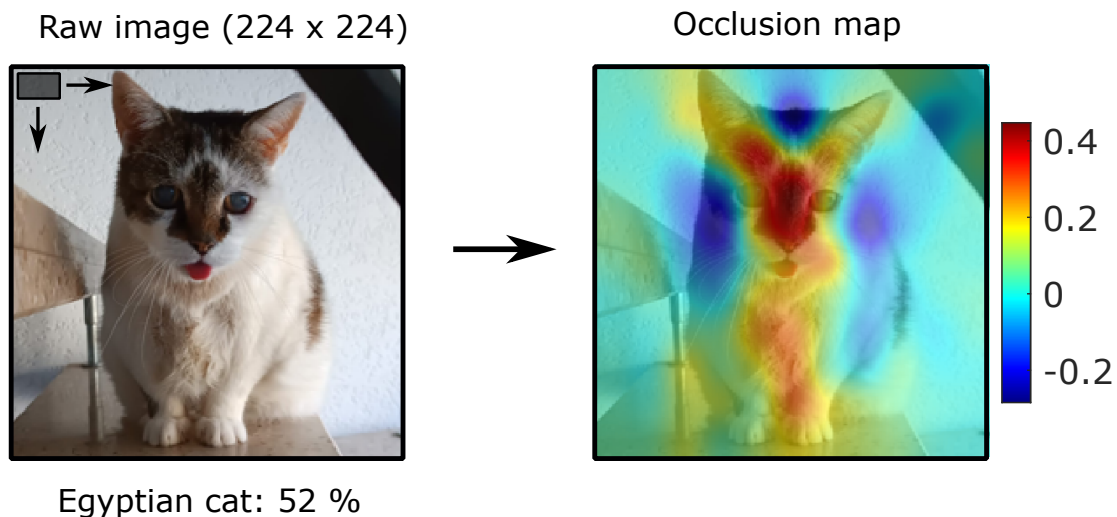


Figure 2.14: Example of an occlusion map for computer vision. The task was to identify the object in the picture and to identify the most important sections for that prediction (adapted from [211, 212]).

¹⁹If the cover and striding size do not fit perfectly, it is possible to perform padding.

Another promising method is gradient map [210, 213]. This method works as a white box explainer. The gradient through the network regarding a specific input instance concerning the class or regression output is calculated. This can be written as shown in Equation 2.24 [210, 213, 214]. Thereby, IS represents the importance score, RO , the output layer, I , the input, and S , the specific input instance. The higher the gradient of the particular pixel, the more critical this pixel is for the prediction. However, this method can lead to noisy importance scores, and therefore, the resulting gradient map is smoothed with a moving average filter. As for the occlusion map, the filter size is specific to the use case.

$$IS = \left. \frac{\partial RO}{\partial I} \right|_S \quad (2.24)$$

An extension of this algorithm is called Grad-CAM [215] (Gradient-weighted Class Activation Mapping). In this method, the gradients are not calculated back to the inputs but to the last convolutional layer. That means an importance score for the output of the last convolutional layer is calculated. Since this layer has positional encoding and resembles a specific feature of the input (every filter resembles a feature), it is possible to map the importance score of these feature maps back to the input. Therefore, the feature maps are multiplied by the importance score and added to represent a smaller image with an importance score. Then, the weighted feature map (usually much smaller than the original picture) is rescaled to the original size, which results in an importance score for every pixel from the original image, as shown in Figure 2.15.

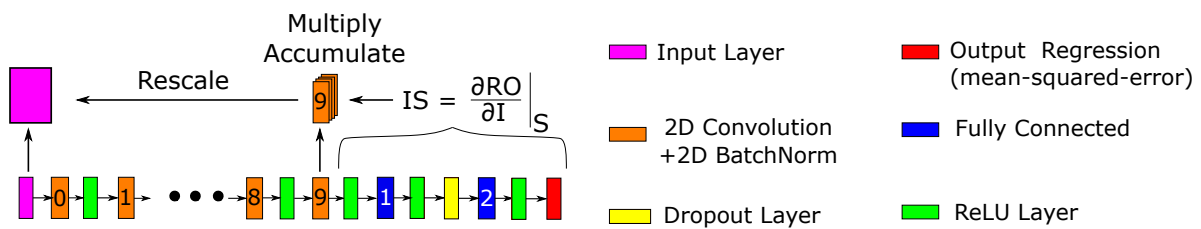


Figure 2.15: Overview of Grad-CAM that illustrates the calculation of the importance scores (adapted from [215]).

So far, the introduced methods only give insight into the importance score of a single instance. These methods are called local explainers. They work exceptionally well for pictures where it is known which features the neural network should look at, as a human can confirm it. However, this is impossible for sensor calibration because it is unknown

in which part of the signal the information is embedded. Therefore, these methods need to be extended to make global interpretations. Thus, the following method from local to global is used [216, 217]. The mean scores across the independent instances are calculated, as the region of importance is expected to stay consistent despite the regression output. Therefore, it should be possible to identify the most crucial part for sensor calibration. However, a verification scheme is necessary since this task is much more challenging to verify by a human, compared to pictures where it is known which features are important [Paper A].

2.5 Data-Driven Sensor Calibration

Every sensor needs proper calibration before it can be accurately used. However, calibration is especially important for MOS gas sensors because the relationship between sensor response and gas concentration is much more complex than for other sensors like temperature sensors [24]. This is because of two points. First, the MOS gas sensor is based on a chemical process prone to many disturbances. Second, the MOS gas sensor does not directly measure a specific gas concentration; instead, the sensor's resistance is measured under different conditions, and thereby, an indirect measurement is used to estimate the gas concentrations [18, 23]. The following calibration approach varies depending on the form of the data on hand and the ML task. For example, if only the static sensor responses are available, boosted decision trees might be the best option [55]. If the dynamic response of the sensor can be used, a DL model might be favorable [Paper 1]. However, the principles are the same for any gas sensor. First, an observation that is later used for model building needs to be defined. In the specific use case of the MOS gas sensor operated in TCO, an observation consists of all samples recorded from all sub-sensors from one sensor during a complete TC. An example of one observation is presented in Figure 2.16.

In this instance, the TC consists of twelve high and low-temperature steps, and the sensor response is sampled at 10 Hz (duration 144 s). The high-temperature steps are always set to 400 °C (duration of 5 s), and the interlaced low-temperature steps are increased from 100 to 375 °C in 25 °C steps (duration of 7 s). This specific shape is used because the sensor is "reset" at high-temperature phases (covered with oxygen), and different gases react with the sensor at different low-temperature phases. An exception is sub-sensor 3, which is only modulated between 300 °C and 200 °C. In this case,

one observation consists of a 4 (sub-sensors) \times 1440 (samples) Matrix, which can be displayed as an image (cf. Figure 2.17).

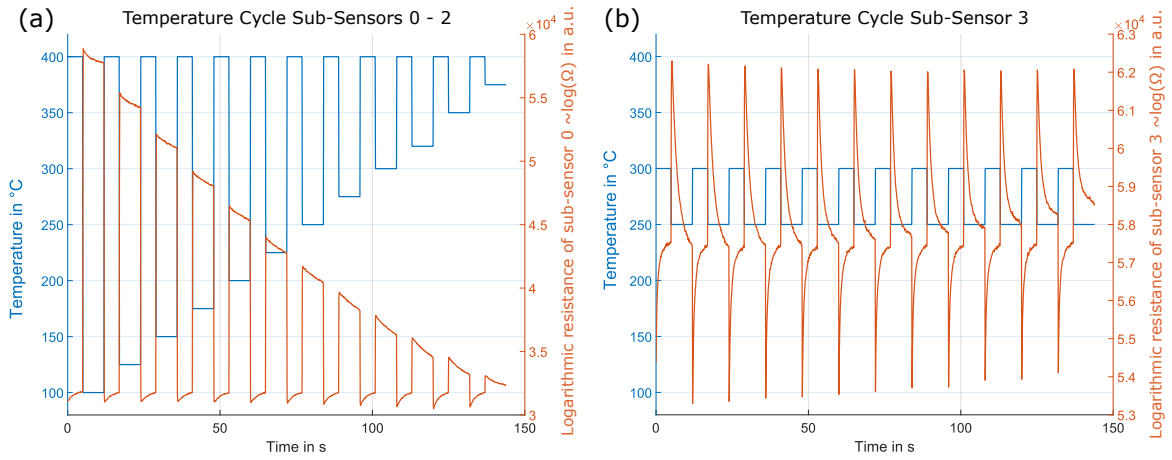


Figure 2.16: Example of the sensor signals from an SGP40 (Sensirion AG, Stäfa, Switzerland [218]) with four sensor pixels sampled at 10 Hz. Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.

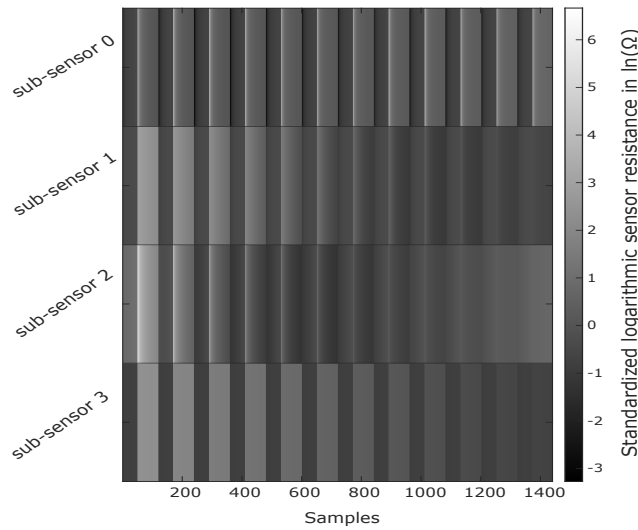


Figure 2.17: Example of the sensor signals displayed as an image from an SGP40 (Sensirion AG, Stäfa, Switzerland [218]) with four sensor pixels sampled at 10 Hz (adapted from Paper A).

The following calibration can be understood as building the data-driven model that learns the dependencies between raw sensor signal and target gas concentration based on multiple observations during training (cf. Figure 2.18).

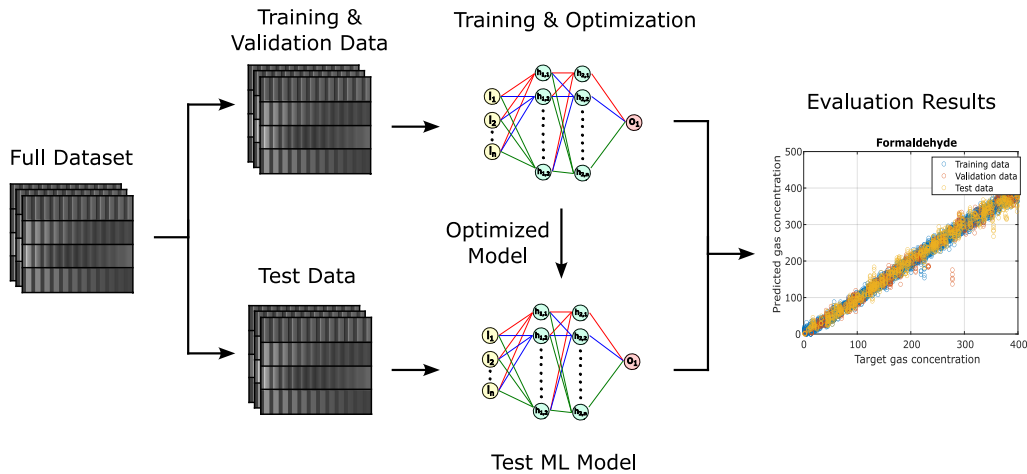


Figure 2.18: Evaluation pipeline for ML algorithms.

The model that learns the dependencies between input and output can be from the FESR or DL domain. In any case, the model is later used in real-world operations to estimate single gas concentrations. The following subsections discuss the challenges faced with MOS gas sensor calibration and the state-of-the-art approaches to solve these challenges in more detail.

2.5.1 Challenges for Sensor Calibration

2.5.1.1 Calibration Time

One of the main reasons MOS gas sensors are not yet widely used to predict single harmful VOCs for accurate IAQ monitoring is that an extensive calibration is necessary. It was shown in [15] that a calibration time of multiple weeks is needed to accurately measure harmful VOCs at ppb level. Because of the lack of selectivity [19, 219] and stability [220], even more, training samples are required for building a stable model under a wide variety of conditions [24, 221]. Furthermore, each sensor needs to be calibrated individually because of the manufacturing tolerances of the sensor itself or the temperature shifts in the micro hotplate (sensor to sensor variance) [222]. Although the manufacturers try to reduce this scattering, every sensor still needs multiple weeks of calibration. This extensive calibration is not suitable for commercial applications. First, the long calibration times would require massive GMAs that can calibrate hundreds of sensors simultaneously; otherwise, the amount of calibrated sensors would need to be

massively reduced. Second, the amount of gas essential to calibrate multiple sensors for several weeks is costly for the manufacturers. Furthermore, it is not fully understood how many different gas mixtures a model needs to have seen and which mixtures are necessary for the specific use cases before the model is accurately calibrated [223, Paper 2].

2.5.1.2 Drift

Another massive problem of MOS gas sensors is that they are based on a physical-chemical process. This fact in itself is not a problem. However, these processes tend to be not fully reversible. Therefore, MOS gas sensors tend to drift over time [224]. Drift can be described by the sensor changing its inner properties over time, thereby making the calibration partially obsolete. The extent of this problem can be emphasized by the number of research articles published on this topic [20, 21, 25, 73, 225]. The drift can be identified when the sensor is operated after multiple weeks in the same condition as calibrated before. The measured resistance will be different from the beginning. Figure 2.19 illustrates the quasi-static signal of the sensor operated at 400 °C.

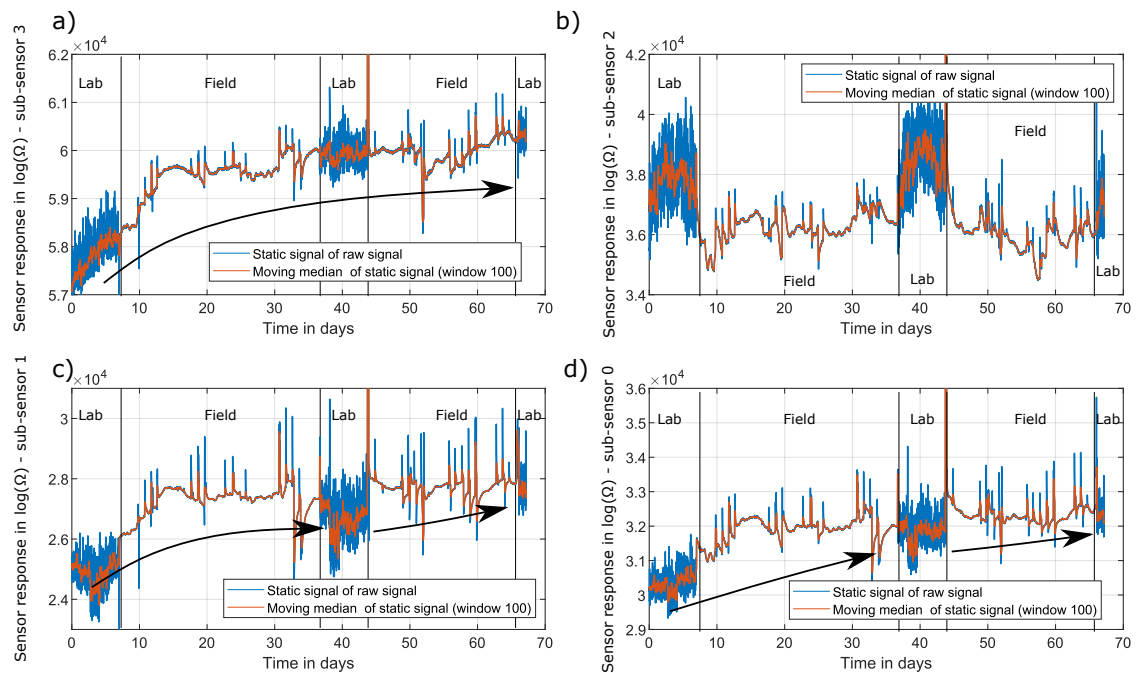


Figure 2.19: Example of drift for the four sub-sensors of an SGP40 over 70 days. Different sub-sensors show a different severity of drift over time (@ 400 °C). Data used for visualization from [226].

It can be seen that the resistance changes over time, especially when analyzing the early operation of sub-sensors 0, 1, and 3. For these sub-sensors, it is possible to see a steep increase in resistance. The first drastic change is associated with burn-in, not the normal drift over time. However, it shows the challenges faced when building a data-driven model that needs to be stable over multiple months. While the problem of the burn-in effect can usually be bypassed by operating the sensor for a more extended period in normal atmospheric conditions before calibration [227], this is not applicable for drift. Although it can be seen that the increase in resistance over time is minor, after some time, it still can be observed and will definitely influence the data-driven model. Likewise, it can be assumed that this drift over time also affects the dynamic response, making it harder to find a robust model.

The severity of the difference between a prediction of a linear ML model (PLSR) right after the training and after several weeks can be seen in Figure 2.20. The prediction after 30 days shows a significant offset and a different slope (the sensor's sensitivity might be altered). More complex models can even amplify this effect since they tend to show nonlinear effects as well [Paper 1]. Currently, the state-of-the-art approach to tackle this problem is to use the sensor only as long it is within the specified calibration time or recalibrate the model after several weeks, which is unsuitable for commercial applications for the same reasons as long calibration times.

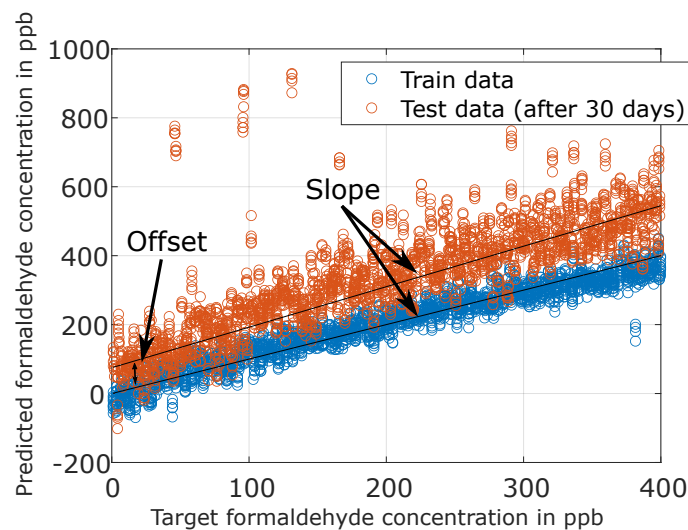


Figure 2.20: Example of how drift over time can influence the prediction accuracy; offset and slope can be altered (adapted from [Paper 1, 15]).

2.5.1.3 Poisoning

Poisoning can be defined as the sensor surface being altered so that oxygen or other oxidizing gases cannot adsorb at the surface [220, 228]. This implies that the sensor can no longer be used since the sensor is no longer sensitive to the target gas, or the sensor response is altered, so a recalibration is necessary. It was shown in [228] that poisoning with siloxane can help to make a MOS gas sensor more selective to hydrogen but also eliminates the sensitivity of the sensor towards multiple VOCs. The major drawback of poisoning of MOS gas sensors is that it can always happen, and it can happen very quickly. After poisoning, the sensor response is irreversibly changed, and the calibration is inoperable. Therefore, every calibration model should be able to detect poisoning and signal that something unusual has happened. The difficulty is distinguishing between novel gas mixtures and sensor poisoning. There are already publications addressing sensor poisoning and how it can be dealt with [220, 229, 230]. One example shows that it is possible to detect sensor poisoning for MOS gas sensors operated in TCO [231, 232]. This thesis only covers sensor poisoning in the Outlook section regarding future work and how it may be approached but does not provide a novel solution.

2.5.2 State-Of-The-Art

The following subsections introduce state-of-the-art methods to calibrate gas sensors and solve the issues mentioned above. These methods were previously developed and described in various publications [16, 25, 27, 53, 68, 69, 121], where they have proven to be suitable.

2.5.2.1 Calibration

One established approach to build a data-driven regression model for calibrating a MOS gas sensor is called the FESR approach. This approach is widely used under different names and describes the general feature extraction, selection, and regression process. The raw sensor data is transformed into features, and the most important features are further used together with any regression algorithm to predict the gas concentration [15, 24, 233–235]. They mainly differ in the feature extraction methods (e.g., linear segments, derivative, time of dynamic processes, min/max values [53]) and the final regression method (e.g., linear regression, PLSR, XGBoost) [121]. A different approach uses a regression algorithm directly on the raw sensor signal without feature extraction. This

is especially successful when analyzing tabular data (e.g., electrochemical sensor arrays) [55, 236]. Another suitable calibration strategy for gas sensors is to perform classification instead of regression by either classifying specific concentrations or differentiating between harmless and harmful concentrations. For this task, it is again possible to apply feature extraction, selection, and classification or use classification directly on the raw data of a sensor array. For classification, the most popular approaches are k-nearest neighbors (k-NN), support vector machines (SVMs), or simple artificial neural networks, as demonstrated in [236]. A recent review from 2022 by Sagar et al. [121] summarized the most recent developments regarding gas sensor calibration. In this thesis, the FESR toolbox is partially used to generate a baseline accuracy to rate all results based on state-of-the-art methods. The features are extracted with the help of the methods introduced above (Fourier transformation, adaptive linear approximation, wavelet transformation, principle component analysis, or statistical moments (with mean and slope)). Afterward, the features are pre-selected with the help of Pearson correlation and further reduced in an iterative process with the help of recursive feature elimination least squares regression. The final regression is performed with PLSR with varying components [15]. However, the detailed model-building approaches are introduced in the corresponding papers.

2.5.2.2 Calibration Transfer

As previously mentioned, one of the major drawbacks of MOS gas sensors is that due to manufacturing tolerances (e.g., differences in a micro hotplate, different doping), it is not possible to reuse the calibration model of one sensor [25, 73, 237–239]. This consequently leads to prolonged and independent calibrations, which are costly in terms of money and time. Therefore, multiple approaches for calibration transfer have been developed to reuse the same model across sensors to reduce the required calibration time. The task is usually defined by transferring the calibration model between master and slave sensors (calibration transfer). Thus, a large dataset of the master sensor and a smaller dataset from the slave is available. Within the large calibration dataset from the master sensor, the master sensor system is exposed to many different calibration gas mixtures. The smaller transfer dataset from the slave sensor contains only the sensor responses from a subset of the calibration mixtures. The goal is to achieve the best possible accuracy for the slave sensor on additional, independent test data [25, 73].

The following introduces the general principles of the most popular methods for calibration transfer. One approach is called signal standardization and aims to match the signal of multiple independent sensor systems to get the same response across devices. In this case, the task described above is solved by building an initial model with the available data from the master sensor. Afterward, a projection is created between master and slave responses that modify the slave responses so that after projection, the slave response resembles the master responses (cf. Figure 2.21) [73, 239].

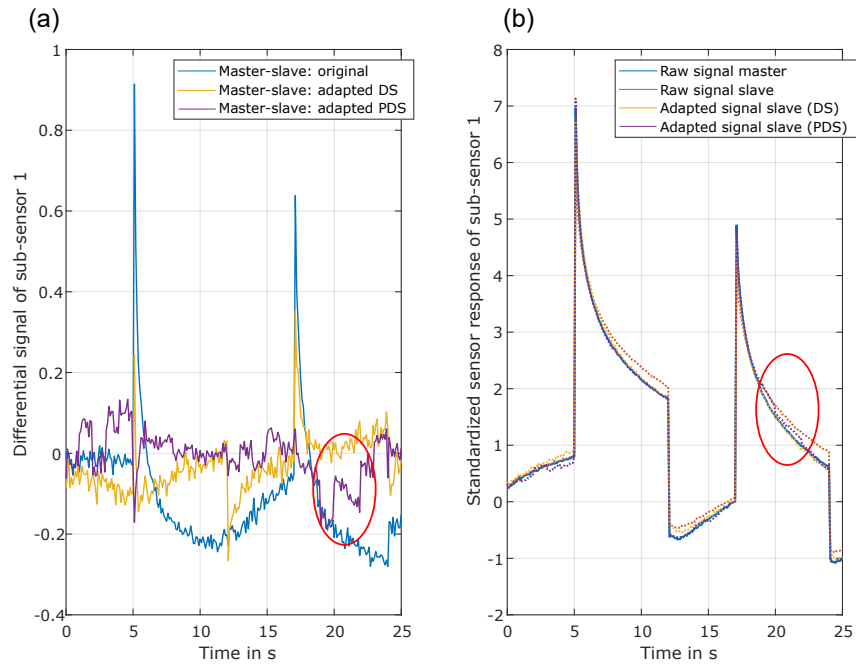


Figure 2.21: a) Differential signal between original and adapted signal. b) Sensor response of the master sensor, the initial sensor response from the slave sensor, and the adapted signal from the slave sensor (Direct Standardization (DS) and Piecewise Direct Standardization (PDS)). Only a section (0 s - 25 s) of one Temperature-Cycle (TC) is shown for better visibility, and only the signal of sub-sensor 1 is shown. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.

In the gas sensing community, signal standardization is the most popular method for calibration transfer. Within signal standardization, many different methods were developed, e.g., Direct Standardization (DS) [240], Piecewise Direct Standardization (PDS) [241], windowed piecewise direct standardization [242], standardization error-based model improvement [242], and many more. All of these algorithms build another ML model that is able to learn the projection and predict the expected master response based on the slave sensor response. The most significant differences are the ML techniques

used to standardize the data (e.g., multi-linear regression, PLS2, neural networks) and the number of samples used within an observation (e.g., Direct Standardization (DS) vs. Piecewise Direct Standardization (PDS)) in order to predict the corresponding samples in the master space [25, 235, 243, 244]. Similar to this approach, methods like orthogonal signal correction [245] and general least squares weighting are used [73] to project the master and slave response in a new sub-space to suppress inter-sensor variance in the signal. Another more high-level approach to solve the task of calibration transfer is to build the initial model with the calibration data from the master sensor and then apply the data from the new sensor to this model. This results in having the target and the new response from the initial model. With this, it is again possible to build a data-driven model that uses the newly predicted target as input and the original target as the output target. In this way, it is possible to learn the dependencies between the two devices (Equation 2.25) [246].

$$y^{new} = c * y^{predicted} + b \quad (2.25)$$

A fourth approach for calibration transfer is global modeling. In this case, the calibration data from multiple master sensors is used to build the initial model. Global modeling can help since there is a broader variety in the data, and thereby, more general models can be found that can be used directly with new slave sensor systems [55, 233]. Finally, there is model expansion. For this method, the initial model is built with the data of the master sensors and with the data of the slave. Compared to global modeling, an essential addition is that additional samples are weighted to ensure the model focuses explicitly on the new samples from the slave sensor [247]. Different methods have been developed for this task. Weighting can happen by duplicating the slave responses to have them multiple times within the training dataset. Another approach is to adapt the loss function to punish an error more severely for the transfer samples (Tikhonov regularization [248] or joint-Y partial least squares regression [249]). A review from 2012 from Marco et al. [235] and 2018 from Rudnitskaya [25] showed the principle of all these approaches.

However, since most of the methods have only been tested and described in their respective publication, as stated by Rudnitskaya [25], only those most widely used are further introduced and later used for comparison. For chemical sensing, signal standardization methods are the most widespread approaches. By Fernandes [237] and Fonollosa [73, 239], it was shown that DS and PDS could outperform orthogonal signal

correction and generalized least squares regression. Therefore, the two methods from the field of signal standardization are introduced in more detail. Compared to other implementations, the focus lies on the implementation based on multi-linear regression [240]. The multi-linear multi-variable problem is illustrated in Equation 2.26 [73, 240].

$$X_{slave}^{new} = C * X_{slave}^{old} \quad (2.26)$$

Parameter C is thereby learned, based on Equations 2.27, 2.28.

$$X_{master}^{old} = C * X_{slave}^{old} \quad (2.27)$$

$$C = X_{master}^{old} * (X_{slave}^{old})^+ \quad (2.28)$$

The only difference to linear regression is that C is not a vector but a matrix due to the target having more than one dimension. An extension to this approach does not use the entire slave sensor response at once to predict a specific master response but instead only a part of the raw signal (PDS [241]). The theory is that with a smaller section to transform and fewer dependent variables, the inversion of the response matrix should be more reliable (Equation 2.29 [241]).

$$C_{S \times o; z} = (X_{Master; S \times o}^{old})_z * (X_{Slave; S \times o}^{old})_z^+ \quad (2.29)$$

In this case, C is calculated on smaller sub-sets of dimension $\mathbb{R}^{o \times S}$. S stands for the number of samples used within one observation, while o stands for the size of the transfer set. Moreover, z represents the number of sections the original signal is divided into. The complete C is then constructed by placing the independent C s on the diagonal of a matrix, as shown in Equation 2.30.

$$C = \begin{bmatrix} C_{S \times o; 1} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & 0 \\ 0 & \cdots & 0 & C_{S \times o; z} \end{bmatrix} \quad (2.30)$$

Although the above-mentioned methods can significantly reduce the calibration time, they still do not provide the needed decrease in calibration time for accurate quantification of single VOCs. Currently, around 20 UGMs are required to reach acceptable accuracy,

but more than five gas tests are not suitable for commercial application (no GMA needed). This thesis, therefore, aims to develop a method that is capable of surpassing the existing methods for calibration time reduction to make MOS gas sensor viable for accurate IAQ monitoring.

2.5.2.3 Drift Compensation

Another major drawback of MOS gas sensors or chemical sensors, in general, is that they are prone to irreversible effects. That means the electrical property of the sensor may change over time. This is usually called sensor drift and can be observed for near-infrared spectroscopy [238] and other chemical sensors [25]. A paper from 2022 addresses the cause of sensor baseline drift from the chemical perspective and suggests a few possible solutions on the sensor side [224]. However, this work focuses on the methods to reduce the impact of drift from the data science perspective. In the case of drift, data is available from the sensor's starting state to build the initial model. Additionally, test data is available from the same sensor after multiple weeks. For gas sensors, it is often the case that the initial model can no longer be used as the prediction will be inaccurate [25]. The review papers of Rudnitskaya [25] and Sagar [121] show several methods that have been proven helpful for drift counteraction. One approach assumes that the drift can always be found in the same direction for multiple sensors. These methods, therefore, try to find the direction of drift in the raw signal and then try to remove this effect. One example could be component correction [22]. They use PCA with reference samples from the starting state and reference samples from later in time (similar to orthogonal signal correction [245]). The first component is now believed to contain the variation caused by drift. Therefore, the first component can then be subtracted from the raw signal to be still able to use the initial model.²⁰ Similarly, it is also possible to identify specific features resistant to drift [53]. The ML model is then only trained on the new features, which results in drift-resistant models. Another successful method that works similarly, is slope and bias correction. This method is closely related to the time series domain. In this case, the time information when a sample was recorded is available (previous samples are known with the target). Therefore, the raw signal can be corrected if it is known how drift manifests in the sensor, or it can be used to correct the actual predictions to reduce the drift. Multiple methods have been developed. Examples are autoregressive moving averages, Kalman

²⁰A few more advanced methods have been developed, e.g., correlated information removing based interference suppression (CIRIS) [250].

filters [251], or methods based on recurrent neural networks combined with ensemble methods [252]. The last subset of methods for drift compensation is global model building. As for calibration transfer, the approach uses multiple sensors (different ages) for model building, and the goal is to find a model that can already compensate for drift as drift-resistant features are extracted. Similar to calibration time reduction, the improvement still seems insufficient for broad MOS gas sensor deployment. This might be the case because they usually require frequent reference samples, the models can not be transferred between use cases, and they also suffer from sensor-to-sensor variation. In this thesis, drift is only briefly covered. Therefore, the baseline approach to rate the performance is global model building, as this is reasonably easy to implement. Moreover, global modeling is most suited for the datasets dedicated to drift because calibration samples are only available for a certain amount of time before the sensor is again used in the field. Hence, evaluating other methods with the dataset is not easily possible.²¹

2.5.2.4 Neural Networks for Gas Sensor Calibration

Another standard approach for calibrating gas sensors is to use neural networks. Neural networks are used because they are popular and can outperform classic ML algorithms [236, 253]. The way neural networks are applied for gas sensor calibration varies extensively. For dynamically operated gas sensors, convolutional neural networks [Paper 1, 223, 254, 63, 255], or recurrent neural networks [256] are frequently used. Also, feature extraction and selection can be used to create tabular data and subsequently use fully connected neural networks or special tabular data neural networks [236, 252, 257] for regression or classification. Besides the application for sensor calibration, neural networks are also used directly for drift compensation by generating domain-independent features [258, 259]. Furthermore, long short-term memory neural networks in combination with ensemble learning [252] are used to reduce drift over time. They have also been shown to work slightly better in sensor-to-sensor generalization [253] and are already used for simple calibration transfer methods (e.g., different target beverages) [260]. Finally, neural networks in combination with XAI have been used to gain information about the importance of pre-calculated features [261]. Compared to the mentioned publications, this thesis shows the use of convolutional neural networks to calibrate a MOS gas sensor for a regression task and apply transfer learning to perform sensor-to-sensor recalibration and use XAI method on the raw signal to validate the results all at once. Furthermore,

²¹Some of those approaches can also be used to deal with sensor poisoning. However, they are also not yet used in larger commercial applications.

the exceptionally complex IAQ datasets validate the significance of the results and show that the introduced methods can be used to tackle the most important tasks for MOS gas sensing. However, directly comparing all already existing neural network solutions with the newly proposed methods is problematic as these approaches are often incomparable. Some publications exclusively try to achieve the best possible performance with the help of neural networks and rarely try to shorten the calibration time, reduce sensor-to-sensor variance, or use raw data from TCO in the use case of IAQ monitoring. This problem is further amplified by different datasets for different use cases with many different gases and sensor operation modes. Therefore, the approach developed within this thesis is only compared to the state-of-the-art approaches, and the comparison with other DL methods should be part of future research.

3 Results and Discussion

3.1 Results and Discussion: Introduction

Within the following publications, the TCOCNN is developed to investigate the benefits of deep neural networks and advanced methods from the field of DL for gas sensor calibration and evaluation in the realm of IAQ monitoring. The TCOCNN is a custom convolutional neural network tailored explicitly for calibrating MOS gas sensors operated in TCO. For model building, DL was chosen as advanced methods like transfer learning can significantly leverage the prediction quality. For the architecture, a convolutional neural network was selected since the gas sensor response can be interpreted similarly to an image, and those networks have proven well-suited to extract specific features for computer vision. The TCOCNN consists of multiple convolutional layers followed by two fully connected layers, as illustrated in Figure 3.1. The input of the TCOCNN is an image (cf. Figure 2.16). The picture has the dimensions of $n \times m$, where n represents the number of sub-sensors or sensors within a sensor array, and m defines the number of samples in one observation. An example of this can be seen in Figure 2.16. In this example, one SGP40 with four sub-sensors in TCO (144 seconds @ 10 Hz) was used, which resulted in a 4×1440 input matrix. For the output, a single output neuron is usually used to predict a specific target gas concentration.²²

Besides the general structure of the network, the TCOCNN has many parameters that were static during every evaluation in the following papers to make them more manageable. Static parameters were the solver (Adam), the L2 regularization (0.0001), the learning rate schedule (drop learning rate by 0.9 every second epoch), the number of epochs (75), and the mini-batch size (50 observations). Furthermore, the shape of the convolutional layers after the first two layers was fixed²³, the ReLU and Batch

²²Predicting multiple gases is possible with more than one neuron in the output layer.

²³Odd convolutional layer: kernel 2×1 ; striding 2×1 ; even convolutional layer: 1×1 ; striding 1×1 ; number of filters doubles every second convolutional layer.

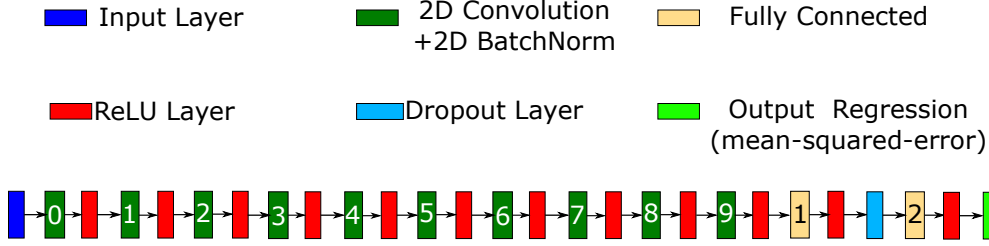


Figure 3.1: Neural network architecture of the TCOCNN (adapted from [262]). An example configuration with ten convolutional layers (later optimized). Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.

normalization was the default of Matlab, no normalization within the input layer was utilized, and only one output neuron was used.

However, many other hyperparameters were tuned throughout the papers with the help of Bayesian optimization to reach optimal performance for predicting the target gas (cf. Table 3.1). One of the parameters picked for optimization was the initial learning rate of the Adam solver. This parameter was varied between 10^{-5} and 10^{-3} to adapt to changing complexity (e.g., number of neurons). Another parameter that was tuned was the number of convolutional layers. This parameter was varied between 6 and 12, but only an even number of layers could be chosen.

Table 3.1: One example for possible hyperparameter ranges for the TCOCNN (adapted from Paper 1).

Initial Learning Rate (Log Scale)	Number of Filters (First Two Layers)	Kernel Size (First Two Layers)	Stride Size (First Layer)	Dropout	Number of Neurons (FC)
$\times 10^{-5}$ - $\times 10^{-3}$	10 - 100	15 - 100	5 - 35	30 - 50 %	500 - 2500

Additional parameters tuned with Bayesian optimization were the number of filters, the kernel, and the striding size of the first two convolutional layers. The number of filters was chosen between 10 and 100, the kernel size was varied between 15 and 100, and the striding size was varied between 5 and 35. This was done because the features changed slightly depending on the target, and multiple different features were needed. Another critical parameter that varied was the dropout rate. Although, in theory, the batch normalization layer replaces the need for a dropout layer, we found that it still positively affects the overall performance. The last parameter adjusted was the number of neurons in the second to last fully connected layer since this parameter significantly impacts accuracy.

Within the following publications, the TCOCNN is tested with various IAQ datasets. Those datasets consist of a selection of VOCs, carbon monoxide, hydrogen, and a wide variety of humidity as typical interference. The specific datasets are introduced within the respective paper, and a rough overview is given in Table 3.2.

Table 3.2: Rough overview of the complexity of the datasets used in the respective paper [Paper 1, Paper 2, Paper 3].

	Paper 1	Paper 2	Paper 3
Sensor	SGP30	SGP40	SGP40
TC duration	120 s	144 s	144 s
Humidity	25 - 70 % RH	25 - 80 % RH	25 - 75 % RH
Carbon monoxide	150 - 2000 ppb	100 - 2000 ppb	200 - 2000 ppb
Hydrogen	400 - 4000 ppb	400 - 2000 ppb	400 - 2000 ppb
Acetone	14 - 1000 ppb	3 - 500 ppb	0 - 1000 ppb
Ethanol	4 - 1000 ppb	1 - 500 ppb	0 - 1000 ppb
Formaldehyde	1 - 400 ppb	1 - 300 ppb	0 - 600 ppb
Toluene	4 - 1000 ppb	1 - 250 ppb	0 - 2000 ppb
Acetic acid	-	1 - 500 ppb	0 - 1000 ppb
Ethyl acetate	-	1 - 500 ppb	0 - 1000 ppb
Isopropanol	-	1 - 500 ppb	0 - 1000 ppb
Xylene	-	2 - 500 ppb	0 - 1000 ppb
Acetaldehyde	-	-	0 - 1000 ppb
Limonene	-	-	0 - 300 ppb
n-hexane	-	-	0 - 1000 ppb

Regarding the computational cost, it has to be stated that during training and inference, the TCOCNN is much more costly. In a simplified evaluation stack based on the FESR approach (raw signal same as in Figure 2.16), the computational cost during inferring can be summarized as follows. For feature extraction, the sensor signal is split into 144 segments, and the mean and slope are calculated for each segment. This basically results in no parameters to be stored and a total of roughly 30000 multiply-accumulate operations during inference. For the final model building, a PLSR with 20 components is chosen, which leads to 1153 parameters to be stored and 1152 additional multiply-accumulate operations during inference. This means a total of 1153 parameters need to be stored, and approximately 31000 multiply-accumulate operations need to be performed. Compared to this approach, an average TCOCNN has 10 million parameters and requires 100 million multiply-accumulate operations during inference. It still might be possible to calculate a single output of a TCOCNN in 2 minutes, but this would

require a microcontroller that can perform 100 million multiply-accumulate operations in two minutes and store more than 50 megabytes of variables.²⁴ Although the TCOCNN is more costly in terms of computational cost, it is still expected to help tackle some of the main challenges regarding MOS gas sensors.

The goal of this thesis is to develop a viable calibration scheme for IAQ monitoring by tackling the main challenges for MOS gas sensors. The primary target is drastically reducing the calibration time and sensor-to-sensor variance to make it feasible to quickly calibrate MOS gas sensors to predict single harmful VOCs with the TCOCNN for IAQ applications (laboratory and field). The remaining drawbacks of drift over time, sensor poisoning, and unknown interfering gases are only briefly discussed and should be analyzed in future research.

The following main questions are tackled in the subsequent publications:

- Is the TCOCNN capable of predicting the concentrations of different VOCs?
- Can the TCOCNN outperform the FESR approach in normal sensor calibration?
- Does the TCOCNN still show acceptable results in real-world environments?
- Is it possible to significantly reduce the calibration cost for IAQ monitoring with calibration transfer based on DL methods (transfer learning)?
- What are all the hyperparameters for transfer learning that significantly influence the calibration transfer?
- What is the effect of global model building for initial model building on transfer learning?
- How does calibration transfer based on transfer learning perform compared to state-of-the-art methods?
- Can XAI from DL help to better understand the sensor or the used TC?
- Is it possible to transfer all the insights about the TCOCNN to other domains like condition monitoring?

²⁴Estimates are based on the mathematical equations for the corresponding model [125, Paper 1].

3.2 Paper 1 – High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning

Y. Robin, J. Amann, T. Baur, P. Goodarzi, C. Schultealbert, T. Schneider, A. Schütze

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

Atmosphere 2021, 12(11), 1487;

The original paper can be found in the online version at <https://www.mdpi.com/1351366> or DOI: <https://doi.org/10.3390/atmos12111487>

© 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). Reprinted, with permission, from Y. Robin, J. Amann, T. Baur, P. Goodarzi, C. Schultealbert, T. Schneider, A. Schütze; *High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning; Atmosphere 2021.*

3.2.1 Synopsis

The first paper introduces the TCOCNN as a specifically tailored CNN for gas sensor calibration and evaluation. In an initial test, the performance of the TCOCNN regarding general accuracy, field test capabilities, and stability over time was tested and compared to state-of-the-art FESR methods. The dataset to test the newly developed model was first published in [15] and contained data from the laboratory and field. The laboratory samples were recorded with a custom Gas Mixing Apparatus (GMA) under constant flow that allows mixing two background gases, four VOCs, and varying the relative humidity. The two background gases were carbon monoxide and hydrogen, while acetone, toluene, formaldehyde, and ethanol were used as VOCs. The dataset consisted of three calibration phases interlaced with two field tests. Multiple UGMs were recorded during each calibration phase for sensor calibration. Every UGM was constructed with the help of Latin hypercube sampling, and the uniformly distributed gas concentration ranges given in Table 3.3.

Table 3.3: Concentration ranges for all gases within gas mixtures during the calibration phases. Reprinted with permission of Ref. [15, Paper 1]. Y. Robin and T. Baur, 2023.

Substance	Min.	Max.	Extended
Carbon monoxide	150 ppb	2000 ppb	-
Hydrogen	400 ppb	2000 ppb	4000 ppb
Humidity	25 % RH	70 % RH	-
Acetone	14 ppb	300 ppb	1000 ppb
Toluene	4 ppb	300 ppb	1000 ppb
Formaldehyde	1 ppb	400 ppb	-
Ethanol	4 ppb	300 ppb	1000 ppb
VOC _{sum}	300 ppb	1200 ppb	-

During the first two calibration phases, 100 UGMs were recorded, with every gas in the normal range and an additional 100 UGMs per gas in the extended range. The third calibration phase consisted of 200 additional UGMs where some gases were exchanged, and the remaining gases were within the normal range. During each of the 1200 UGMs, ten TCs were recorded, and five TCs per UGM were available in the final dataset.²⁵ For the field tests, the sensors were operated in an office over multiple weeks. Field tests are necessary to test the calibration models in real-world environments. During operation in the field, release tests were performed to validate the models and to test their capability

²⁵The recording of one UGM took 20 minutes in this case.

to detect relative changes. For the release tests, controlled amounts of acetone, toluene, ethanol, isopropanol, and hydrogen were released. The SGP30 gas sensor used within this dataset is similar to the one shown in the theoretical background section. As before, the four gas-sensitive layers of the sensor were operated in TCO. The TC used for this dataset consisted of ten high and low-temperature steps. During high-temperature phases, the sensor was always heated to 400 °C, and the low-temperature phases ranged from 100 - 375 °C in 25 °C steps, where the temperatures 225 °C and 250 °C were left out. The first goal was to find the optimal hyperparameters for the TCOCNN. The parameter optimization was done with the entire first calibration and the first 100 UGMs from the second calibration (total of 600 UGMs). The training, validation, and test split was based on complete UGMs in 70/10/20 fashion. The model was optimized with Bayesian optimization using training and validation data. A suitable model with a reasonable RMSE was found for every gas in the dataset. After hyperparameter optimization, the influence of the number of observations per UGM and independent UGMs on model building were analyzed. It was shown that the model's accuracy increases with the number of observations. However, having additional unique UGMs is more beneficial than observations per UGM. With the second test, the performance of the TCOCNN regarding drift was analyzed. First, only the initial calibration was used for model building (500 UGMs). The second scenario then utilized the first 600 UGMs for training. In both cases, the remaining UGMs from the second calibration were used for testing. It was shown that the observed drift can be compensated. However, the TCOCNN needs multiple UGMs from the second calibration to compensate for drift (training with 600 UGMs), which is similar to the results of the FESR approach. For a more sophisticated statement on drift compensation with neural networks, a new dataset containing samples from multiple sensors after extended periods is necessary.

After the general performance of the TCOCNN was analyzed, the following evaluation focused on comparing the FESR approach as presented in [15] and the TCOCNN. The FESR approach in [15] consisted of dividing the raw signal into 120 equidistant segments, calculating the mean and slope of each segment, performing feature selection with the help of recursive feature elimination least squares regression, and using a PLSR with 20 components for regression. By comparing the performance on the 70/10/20 split, it was revealed that the TCOCNN outperforms the FESR approach regarding all tested gases by a significant margin. The most promising improvement was observed for formaldehyde. There, the RMSE was more than halved. The selected hyperparameter ranges for the Bayesian optimization can explain the considerable improvement. These ranges were

selected based on previous studies [221] focusing on formaldehyde measurements, which resulted in optimal ranges for this gas. Similar improvements for the other gases might be possible with a more extended Bayesian optimization. Another reason for the superior performance of the TCOCNN might be the internally generated complex features that capture more specific characteristics of the sensor response.

This publication's final evaluation compared the FESR and TCOCNN regarding their capability to predict the target gases in real-world environments (cf. Figure 3.2). Since the reference instruments were not calibrated, the aim of this study was to predict realistic relative concentration changes and natural background concentrations with minimum noise.

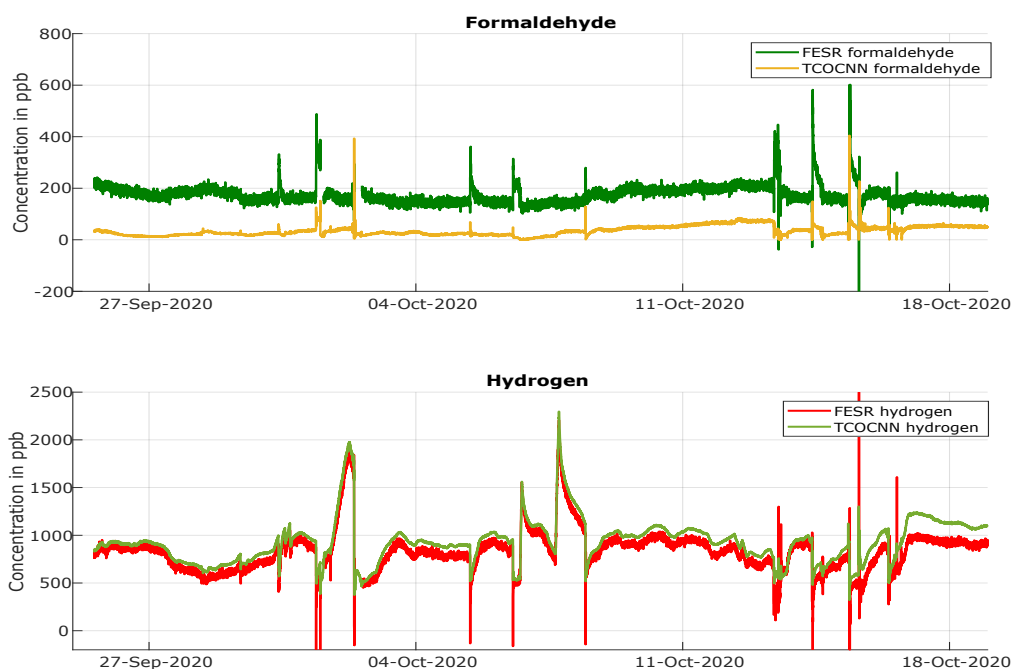


Figure 3.2: Comparison of the results obtained during field tests with the FESR and TCOCNN models for formaldehyde and hydrogen. Reprinted with the permission of Ref. Paper 1. Y. Robin, 2023.

Within the publication, it was possible to demonstrate that the TCOCNN can predict more stable and realistic gas concentrations. This was especially prominent when analyzing formaldehyde and hydrogen. The TCOCNN showed much less noise and a more realistic baseline for formaldehyde and hydrogen, with minimum values

during ventilation of 30 - 40 ppb for formaldehyde and 500 ppb for hydrogen (natural background [263]).

Regarding the release tests (cf. Figure 3.3), it was possible to show that the TCOCNN could reach similar precision in predicting relative changes in the gas concentration compared to the FESR approach. Similarly, the TCOCNN models did not show a significant effect during the release of interfering gases, comparable to the FESR approach. This indicates the robustness of the TCOCNN and FESR to the release of other VOCs. The results were compared to the reference instruments X-pid 9500 and TD-GC-MS to validate the results further. It was possible to show that the instruments used exhibited similar relative changes in gas concentration, and the closest match was achieved between the TCOCNN and the gold standard TD-GC-MS. However, those instruments were not calibrated; therefore, making a conclusion about absolute accuracy is impossible.

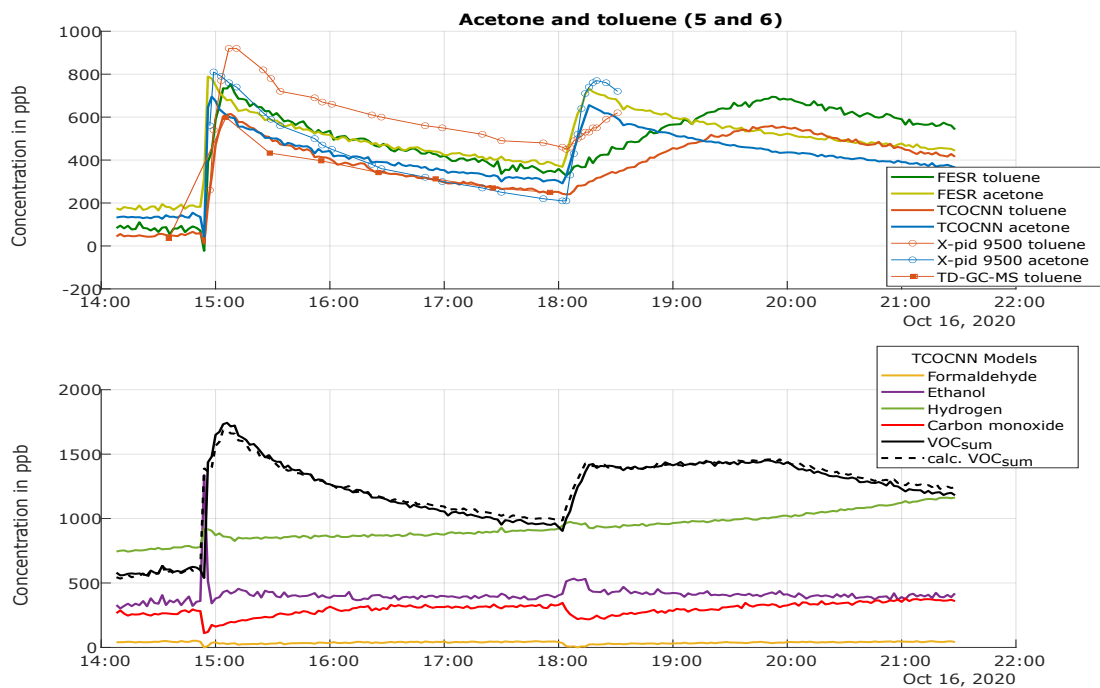


Figure 3.3: Prediction of gas concentrations during release tests 5 and 6 (acetone and toluene) showing the various models trained compared to the analytical measurements (adapted from [15]). Reprinted with permission of Ref. Paper 1. Y. Robin, 2023.

Consequently, it can only be stated that the TCOCNN performs similarly to the FESR approach regarding the capability to detect relative concentration changes close to the expected amount of released gas. As a last experiment, not only trained gases were released, but also gases from the same chemical group (unknown interfering gases). This was done to test if it is possible to train with one representative of a chemical group to predict the sum concentration of the others. Consequently, this would reduce the need for training with highly toxic gases. However, this was only partly successful. While this theory was successfully tested for toluene and xylene, it did not work for ethanol and isopropanol. Therefore, this theory must be further tested to apply to a broader range of gases.

For IAQ monitoring, the results indicate that the newly developed TCOCNN is a promising alternative to the FESR approach to predict single VOCs accurately in laboratory conditions and the field. However, many open questions must be addressed before the model can be widely used in commercial applications. The most critical topic that must be solved is the long calibration time. The calibration took several weeks for this experiment, which is unsuitable for broad application. Therefore, the following papers discuss a possible solution to reduce the calibration time significantly.

The main takeaways of this publication are:

- The TCOCNN can successfully predict gas concentrations in the laboratory and field.
- It is more important to train a ML model on a manifold of UGMs than having multiple observations per UGM.
- With samples that contain drift, it is possible to learn to compensate for drift.
- The TCOCNN can outperform classic ML solutions regarding sensor calibration and evaluation for some aspects, but by the cost of higher computational effort both during training and inference.
- The TCOCNN provides better prediction quality regarding real-world data.
 - Less Noise.
 - More realistic baseline.

- The TCOCNN can predict relative changes in the target gas concentration in real-world environments.
 - Similar performance compared to FESR.
 - Predicts similar relative changes in concentration compared to reference instruments (X-pid 9500 and TD-GC-MS).

Still, open questions/tasks are:

- Calibration still takes multiple weeks.
- Sensor-to-sensor variance was not discussed.



Article

High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning

Yannick Robin ^{*} , Johannes Amann, Tobias Baur , Payman Goodarzi , Caroline Schultealbert, Tizian Schneider and Andreas Schütze

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany; j.amann@lmt.uni-saarland.de (J.A.); t.baur@lmt.uni-saarland.de (T.B.); p.goodarzi@lmt.uni-saarland.de (P.G.); c.schultealbert@lmt.uni-saarland.de (C.S.); t.schneider@lmt.uni-saarland.de (T.S.); schuetze@lmt.uni-saarland.de (A.S.)

* Correspondence: y.robin@lmt.uni-saarland.de

Abstract: With air quality being one target in the sustainable development goals set by the United Nations, accurate monitoring also of indoor air quality is more important than ever. Chemiresistive gas sensors are an inexpensive and promising solution for the monitoring of volatile organic compounds, which are of high concern indoors. To fully exploit the potential of these sensors, advanced operating modes, calibration, and data evaluation methods are required. This contribution outlines a systematic approach based on dynamic operation (temperature-cycled operation), randomized calibration (Latin hypercube sampling), and the use of advances in deep neural networks originally developed for natural language processing and computer vision, applying this approach to volatile organic compound measurements for indoor air quality monitoring. This paper discusses the pros and cons of deep neural networks for volatile organic compound monitoring in a laboratory environment by comparing the quantification accuracy of state-of-the-art data evaluation methods with a 10-layer deep convolutional neural network (TCOCNN). The overall performance of both methods was compared for complex gas mixtures with several volatile organic compounds, as well as interfering gases and changing ambient humidity in a comprehensive lab evaluation. Furthermore, both were tested under realistic conditions in the field with additional release tests of volatile organic compounds. The results obtained during field testing were compared with analytical measurements, namely the gold standard gas chromatography mass spectrometry analysis based on Tenax sampling, as well as two mobile systems, a gas chromatograph with photo-ionization detection for volatile organic compound monitoring and a gas chromatograph with a reducing compound photometer for the monitoring of hydrogen. The results showed that the TCOCNN outperforms state-of-the-art data evaluation methods, for example for critical pollutants such as formaldehyde, achieving an uncertainty of around 11 ppb even in complex mixtures, and offers a more robust volatile organic compound quantification in a laboratory environment, as well as in real ambient air for most targets.



Citation: Robin, Y.; Amann, J.; Baur, T.; Goodarzi, P.; Schultealbert, C.; Schneider, T.; Schütze, A. High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning. *Atmosphere* **2021**, *12*, 1487. <https://doi.org/10.3390/atmos12111487>

Academic Editor: Stéphane Le Calvé

Received: 15 October 2021

Accepted: 5 November 2021

Published: 10 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Keywords: volatile organic compounds (VOCs); indoor air quality (IAQ); deep neural networks; neural network architecture search; temperature-cycled operation (TCO)



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With indoor air quality (IAQ) being one of the most common and unavoidable threats to human health and also one of the most difficult to determine accurately, it is more important than ever to be able to make accurate measurements of IAQ [1]. Especially dangerous are volatile organic compounds (VOCs), which can lead to serious health problems. For example, extensive exposure to formaldehyde can cause cancer [2]. Even the United Nations agree in their goals for sustainable development that pollution is a goal of the greatest importance and that the number of deaths and illnesses from hazardous chemicals and air, water, and soil pollution and contamination should be substantially

reduced by 2030 [3]. Several reasons are making an accurate measurement of IAQ difficult. First of all, indoor air contains hundreds or even thousands of compounds, some of them benign, others toxic, even at very low concentrations, making accurate quantification of each of them impossible, at least for routine and continuous measurements [4]. Second, analytical measurement systems that are capable of providing measurements of the most relevant VOCs are very expensive, are too slow for real-time application, and require expert knowledge to operate and calibrate [5]. Third, due to the difficulty of providing comprehensive measurements, too little is known about the cause and effect of various gases and especially of their combined effect [6]. Currently, CO₂ is the primary indicator used for IAQ estimation as there is a direct relation between VOC concentration in room air and the CO₂ concentration if the VOC levels are caused by human presence, as already described by Pettenkofer in 1858 [7]. However, dangerous VOCs are also released from building materials, furniture, and activities such as cooking and cleaning, which do not release CO₂ [8–10]. For this study, VOCs represent a diverse spectrum from very volatile (VVOC) to semivolatile (SVOC) organic compounds [11]. In this study, we concentrated on VOCs with a high-to-medium vapor pressure including the carcinogens formaldehyde and benzene, which are considered as two of the most toxic substances in indoor air with guideline threshold values in the low ppb range according to the WHO [11]. Therefore, comprehensive VOC monitoring is required to provide a universal indicator for IAQ, e.g., as a basis for demand-controlled ventilation to reduce the overall burden on people [12].

We recently reported a new approach for IAQ monitoring based on low-cost metal oxide semiconductor (MOS) gas sensors (chemiresistor) combined with temperature-cycled operation (TCO) and pattern recognition to interpret the resulting complex response patterns [11,13]. In these studies, we used linear machine-learning (ML) models based on feature extraction followed by feature selection and finally regression (FESR model) to predict the concentration of various VOCs and other relevant gases individually, as well as the sum concentration of all VOCs [13]. As deep learning has proven to be very successful for the interpretation of complex patterns [14], this study provides a first test of deep-learning-based methods utilizing advanced ML techniques such as convolutional neural networks (CNNs) [15] in combination with neural architecture search (NAS) [16] for improved IAQ monitoring.

Previous studies have also successfully addressed the combination of gas sensors and deep learning [17–23]. Most of these studies have addressed higher concentrations in the ppm range [19–22] and were based on multisensor arrays [17,18,21]. Only some also used dynamic operation, but with a simple operating mode for the gas sensor with two temperatures only [19,20,22]. In some studies, the evaluation target was limited to the classification of different gases [19,23]. For a more complete overview, the reader is referred to a recent review paper on smart gas sensing technologies [24].

Therefore, the goal of this study is to show that this new deep-learning model for gas sensors should be capable of making accurate and reliable predictions for the concentration of multiple VOCs in indoor air, again based on the raw data obtained from a low-cost MOS sensor system using TCO to improve their selectivity, sensitivity, and stability [25]. Furthermore, we wanted to confirm that these models can outperform the predictions of the benchmark [13] (established linear data-driven models) at the ppb level in the laboratory environment and field tests. The benchmark was based on classic statistical approaches such as linear segmentation, principal component analysis, and a partial least-squares regression (PLSR). Finally, we compared the predictions of the deep-learning model with state-of-the-art analytical measurement systems, which are the gold standard for IAQ monitoring. Ideally, the novel approach should be considerably less costly, but able to provide high-quality data with high temporal resolution while requiring less expert knowledge, thus being easier to use.

The dataset used throughout this study was published by Baur et al. [13], and the results of the corresponding publication were used as a reference. The dataset was based on an SGP30 sensor (Sensirion AG, Stäfa, Switzerland) with four gas-sensitive layers [26],

operated using TCO for improved selectivity, sensitivity, and stability. The sensor was lab-calibrated using complex random gas mixtures [27] and then tested during operation in a typical office environment with as little human presence as possible over several weeks. Several release tests of VOCs and hydrogen were performed to validate the sensor response and to compare the performance of the model predictions of the MOS sensor system to analytical instruments [13].

2. Materials and Methods

2.1. Dataset

In order to evaluate the capabilities of the newly developed deep-learning approaches for accurate quantification of different gas concentrations in indoor air, the dataset published in Baur et al. 2021 [13] was used. This dataset utilizes advanced calibration and operation techniques together with the low-cost sensor system of the SGP30 to generate a comprehensive dataset for monitoring complex mixtures that are typical of indoor air situations. In addition to various VOCs (acetone, ethanol, formaldehyde, toluene, with formaldehyde being highly toxic, while acetone, ethanol, and toluene represent VOCs with comparatively low hazard potential), relevant inorganic gases, i.e., hydrogen and carbon monoxide, as well as relative humidity (RH), were also included in the calibration scheme, as these have a strong influence on MOS sensors (see Figure 1b). Thus, the sensors needed to be calibrated, and a machine learning model needed to be developed to discriminate interfering gases and various VOCs and to provide quantitative data on the various gas concentrations, as well as the total VOC concentration to allow comprehensive IAQ monitoring. Note that we used VOC_{sum} to describe the total VOC concentration to distinguish this from the TVOC value obtained by analytical measurements, where only VOCs with medium volatility are considered. Gas sensors, on the other hand, also detect VOCs with high volatility, so-called very volatile organic compounds (VVOCs), such as acetone, ethanol, and formaldehyde, which are not considered in the analytical TVOC value [11]. The dataset was based on random gas mixtures [27] generated in an automatic gas-mixing system [28]. With the help of this dataset, complex data-driven models for different gases can be built and evaluated in laboratory environments, as well as in real indoor air scenarios.

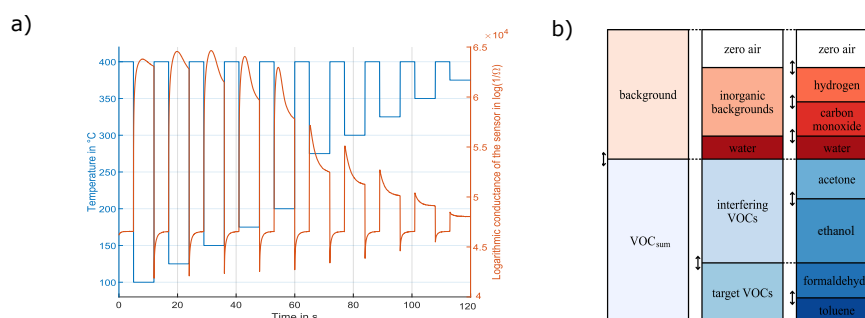


Figure 1. (a) Temperature-cycled operation ranging from 100–375 °C, together with one example of the logarithmic conductance of one sensor element (adapted from [13]) and (b) the gas composition for calibration containing background gases, as well as the target volatile organic compounds (VOCs) (adapted from [27]).

Regarding the sensor setup, the dataset utilizes an SGP30 MOS gas sensor and TCO, as illustrated in Figure 1. The sensor's output represents the resistance of the four different gas-sensitive layers over time sampled at 20 Hz. Thus, a single cycle consists of 2400 raw data samples for each of the four gas-sensing layers' resistance during TCO. This complex operation mode achieves a wide detection spectrum of the sensor system in terms of the concentration range and the gases that can be detected [29]. For improved data evaluation,

the sensor resistance patterns were converted to the logarithmic conductance of the sensor according to the Sauerwald–Baur model [30,31].

The dataset itself consists of multiple recordings of the SGP30 sensor. Those recordings can be divided into calibration phases performed in the lab with typically hundreds of well-known unique gas mixtures interlaced with field tests during which the actual gas composition and the concentrations are not known. With the help of the calibration phases, it is possible to build data-driven models for individual gases from a single sensor element. During calibration, the sensor was exposed to various gas compositions that always contained the six different gases plus relative humidity (RH), as illustrated in Figure 1, to reflect a simplified indoor environment. The concentration ranges of the various gases are given in Table 1. Furthermore, extended ranges for the VOCs as stated in Table 1 were used, to train the model also for gas compositions outside of the normally expected range, which might occur during specific exposure situations in real life, and were simulated by release tests performed in the field study (Table 2).

Table 1. Concentration ranges for all gases within gas mixtures during the calibration phases [13].

Substance	Min.	Max.	Extended
Carbon monoxide	150 ppb	2000 ppb	-
Hydrogen	400 ppb	2000 ppb	4000 ppb
Humidity	25% RH	70% RH	-
Acetone	14 ppb	300 ppb	1000 ppb
Toluene	4 ppb	300 ppb	1000 ppb
Formaldehyde	1 ppb	400 ppb	-
Ethanol	4 ppb	300 ppb	1000 ppb
VOC _{sum}	300 ppb	1200 ppb	-

Table 2. A subset of all release tests performed in [13]. Specifically listed are the release tests, which were further analyzed within this study.

Release	Time	Substance (Type of Release)	Released Amount of Substance (Approx. Increase in Room Conc.)
5	16 October, 14:50	Acetone (evaporation) Toluene (evaporation)	~600 ppb ~600 ppb
6	16 October, 18:00	Acetone (evaporation) Toluene (evaporation)	~600 ppb ~600 ppb
7	2 November, 16:50	Toluene (evaporation)	~600 ppb
9	4 November, 16:22	Acetone (evaporation)	~600 ppb
13	10 November, 14:30	Isopropyl alcohol (evaporation)	~600 ppb
14	11 November, 15:49	m/p-Xylene (evaporation)	~600 ppb
15	12 November, 15:08	Toluene (evaporation) m/p-Xylene (evaporation)	~600 ppb ~600 ppb
16	13 November, 14:30	Acetone (evaporation) Toluene (evaporation) Ethanol (evaporation)	~600 ppb ~600 ppb ~664 ppb
17	16 November, 17:06	Hydrogen (MFC, gas cylinder)	2000 ppb

During the study, three calibration phases and two field test phases were completed (see Figure 2). The initial calibration phase and the first recalibration consisted of 100 unique gas mixtures (UGM) with the typical gas concentration ranges plus 100 additional UGM for

each of the extended concentration ranges for acetone, ethanol, toluene, and hydrogen. This resulted in a total of 500 unique gas mixtures for each of these two calibration phases. The two field test periods were performed between the calibration phases. The recalibrations were necessary to test the stability of the models, i.e., that these were still capable of reliable predictions after several weeks and that they could also suppress or compensate the drift caused by the limited stability of the gas-sensing layers. During calibration, each unique gas mixture was offered in the custom-built gas mixing apparatus for 20 min, i.e., for ten temperature cycles, as described above. Because of the nonideal synchronization between the gas-mixing system and the electronics running the temperature cycle and the delay in the gas exchange within the system, only 5 out of each 10 temperature cycles were later used for evaluation, where the gas concentration was constant. These cycles are called core samples and ensured that all measurements used for model building were recorded under stable gas compositions.

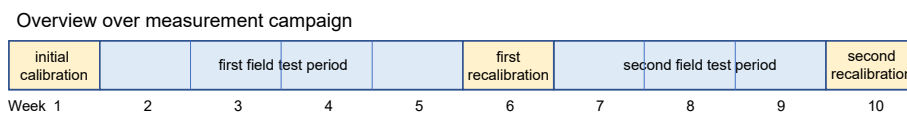


Figure 2. An illustration of the complete experiment over ten weeks, including calibration phases and field tests.

After the initial calibration, the first field test was performed in a partly controlled environment over a period of four weeks. Partly controlled means that the room was ventilated regularly and that there was no human presence in the room unless required for the release tests described below. During the field tests, the indoor air concentrations for the six trained gases were evaluated and compared with the expected values, as well as with analytical reference instrumentation for VOCs and hydrogen. To validate the quantitative prediction of the model for various gases, release tests were performed for acetone, ethanol, toluene, and hydrogen, as listed in Table 2, after thorough ventilation. Note that we did not release formaldehyde due to its high toxicity during the field tests. These tests were performed to allow an evaluation of whether the ML models can correctly detect the released compound and accurately monitor the concentration during release. The released amount was always chosen so that the concentration in the room should reach 600 ppb when the released substance was evenly distributed in the room while neglecting any losses through ventilation or adsorption on surfaces. During some of these tests, analytical instruments were used to monitor the release in parallel with the MOS sensor system. For online monitoring, a portable gas chromatograph with photo-ionization detection (GC-PID: X-pid 9500, Dräger Safety AG & Co KGaA, Lübeck, Germany) was used for VOCs and a gas chromatograph with a reducing compound photometer (GC-RCP: Peak Performer 1, Peak Laboratories LLC, Mountain View, CA, USA) for hydrogen. In addition, samples were collected on Tenax tubes (Markes International Ltd, Llantrisant, Wales, UK) for VOC monitoring in indoor air and later analyzed using thermo-desorption gas chromatography mass spectrometry (TD-GC-MS, Thermo Fisher Scientific Inc., Waltham, MA, USA). Further experimental details were given in Baur et al. [13].

2.2. Model Building

Two machine-learning approaches were used for model building. The first method, which we used as a benchmark here based on [13], utilizes feature extraction (FE) in the form of linear segmentation, standardization, feature selection (FS) based on recursive feature elimination (RFE), together with least-squares regression, a gas-mixture-based cross-validation, and an optimization scheme to find the optimum model regarding the number of selected features and the components used for the partial least-squares regression (PLSR) [13]. This approach is called Feature Extraction Selection Regression (FESR). Linear segmentation means in this case that the four different logarithmic conductance patterns obtained from the gas-sensitive layers are divided into 120 equidistant segments each and the mean and slope are calculated for all segments, resulting in a total of 960 features per

temperature cycle. The data from all unique gas mixtures offered during calibration were then split into 80% for training and 20% for testing. After this step, the 300 most important features according to the recursive feature elimination (RFE) least-squares regression (LSR) ranking were selected for further use. To find a suitable number of features and PLSR components, a gas mixture-based 10-fold cross-validation was performed on the 80% training data. Here, all core samples from 10% of the unique gas mixtures were excluded from the training for validation to find a suitable compromise for the hyperparameters, to achieve a low root-mean-squared error (RSME) with a low number of features and PLSR components. This ML model was developed using the open-source MATLAB toolbox DAV³E [32], and the approach was described in more detail in [13].

The second model-building approach then utilizes the TCOCNN architecture (see Figure 3), a 10-layer deep convolutional neural network (CNN) [15]. A similar network was first introduced in [33] and successfully utilized to predict the formaldehyde concentration for the laboratory calibration measurements. For this contribution, the structure of this network was adapted to predict not only one gas concentration at a time, but the concentrations of all gases offered during calibration, i.e., acetone, ethanol, formaldehyde, toluene, the total concentration of all VOCs (VOC_{sum}), and also the inorganic gases carbon monoxide and hydrogen. The CNN structure was derived from the original ResNet model from [33] to reduce the overall complexity.

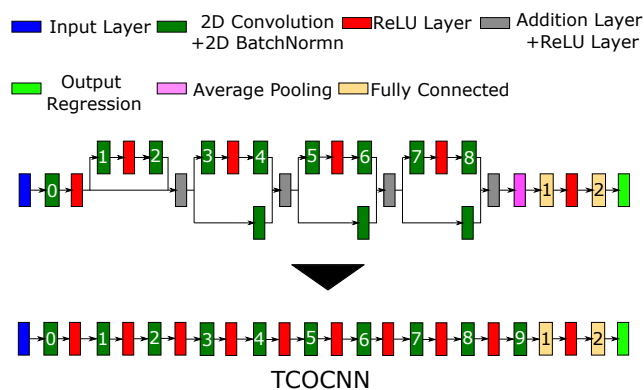


Figure 3. Original network structure from [33] together with the newly derived general architecture of the TCOCNN.

To build a gas-specific model, the general architecture as illustrated in Figure 3 was used. For each gas, the data were randomly split according to the gas mixtures into 70% training data, 10% validation data, and 20% for testing. After the data split, a neural architecture search (NAS) was performed on the training and validation data. This approach searches through a predefined search space of the parameters of the neural network (Table 3) to find the optimal hyperparameters for each gas concentration to be predicted by minimizing the RMSE for the validation data [16].

The NAS varies the parameters listed in Table 3 using a Bayesian optimization search with the remaining parameters, as specified in Table 4. In total, 30 different combinations of the parameter were tested, and the model with the smallest validation RMSE was considered the best model. The Bayesian optimization strategy was chosen to speed up the NAS. Since the training process of one TCOCNN on a GPU already requires up to 20 min, an extensive search through the complete search space would have not been feasible. Therefore, Bayesian optimization was performed to find an acceptable solution in a reasonable time. The optimization that was performed in this specific case was based on the Gaussian process method [34], and the optimized cost function was the validation RMSE.

Table 3. Parameter ranges for every neural architecture search (NAS).

Initial Learning Rate (Log Scale)	Number of Filters (First Two Layers)	Kernel Size (First Two Layers)	Stride Size (First Layer)	Dropout	Number of Neurons (FC)
1×10^{-4} – 9×10^{-3}	60–240	40–80	15–45	30–50%	1000–2500

Table 4. Parameters, which are kept constant during the evaluation.

Parameter	Value
L2-regularization	0.0001
Stride size, even layer	1×2
Kernel size, other layer	1×2
Learn rate drop rate	0.9
Mini-batch size	50
Stride size, odd layer	1×1
Epochs	75
Learn rate drop period	2

The parameters chosen for this optimization were the initial learning rate, the number of filters in the first two layers, the kernel size of the first two layers, the stride size of the first layer, the dropout rate, and the number of neurons in the last fully connected layer [15]. The initial learning rate had to be a part of the optimization as this parameter should be adjusted according to the network complexity. The hyperparameters of the first convolutional layers have proven to have a large influence on the prediction quality and were therefore an important part of the optimization. Additionally, the dropout rate and the number of neurons in the fully connected layers are also parameters worth considering. The ranges for the different target gases were based on the best parameters found in [33]. For carbon monoxide and ethanol, the NAS had to train 60 different TCOCNNs to reach sufficient results as the model building seemed to be more difficult for these gases. Furthermore, the NAS for ethanol had to be restricted to a range from 15 to 35 for the stride size of the first layer to find a suitable result faster.

2.3. Data Evaluation

As a carcinogenic gas, formaldehyde is of great importance for indoor air quality. Thus, as the first step, we evaluated the suitability of the model for predicting the formaldehyde concentration in the ppb range in a complex mixture of other gases [33]. Here, a model for formaldehyde was trained on the initial calibration dataset with a gas-mixture-based data split of 70% training, 10% validation, and 20% for testing.

To determine the required complexity of the calibration (note that one hundred unique gas mixtures offered for 20 min each resulted in a total calibration time of 33 h; the extended calibration with higher VOC concentrations, therefore, required almost 7 d in total), the same model was built with fewer core samples and/or fewer unique gas mixtures to reproduce the results achieved in [33] and also to show the influence of more core samples compared to more unique gas mixtures (UGM).

In the next step, the effect of sensor drift, which is often observed for chemical and especially MOS gas sensors [35,36], was examined. Here, three different models were compared for the prediction of the formaldehyde concentration. For the first model, only the initial calibration including extended concentrations was used for model building and the data of the second part of the first recalibration (with extended concentrations) were used for testing. This model should show significant sensitivity to sensor drift and various effects on the prediction quality such as offset or linearity errors and increased uncertainty. The second model then uses only three instead of all four gas-sensitive layers from the sensor system. The excluded layer is the one most prone to sensor drift as observed in

previous studies [37], and the model trained on these data was used to investigate the cause of the different drift effects. The last model then includes all gas-sensitive layers for training, but extends the calibration data to include the 100 unique gas mixtures of the first recalibration in the standard concentration range. By including parts of the first recalibration in the training, it was expected that the model could suppress drift effects, as these were included in the training data [38]. Again, the second part of the first lab recalibration was used to test the prediction of these models.

After validating the TCOCNN model in general, the results achieved for the laboratory tests with the deep-learning approach were compared with the FESR model published previously [13]. Therefore, TCOCNN models for all seven targets were trained with the help of the NAS on the initial calibration and the first part of the first recalibration to reduce the drift effects. The data split was performed as explained before (70% training, 10% validation, and 20% testing). For comparison, the RMSEs on the test data of the different models are compared for the different gases. This step allows comparing the prediction quality and capability of the different data-driven models.

After demonstrating the quality of the prediction of the TCOCNN approach for the lab data, the deep network was also applied to data from real indoor air environments during field tests. Again, the models were trained using the lab calibration data with the complete initial calibration and the first part of the first recalibration to predict all trained targets during the field test. First, the overall prediction of the TCOCNN for the field test data was compared to the FESR model. This should indicate if the predictions of the FESR model and the TCOCNN are consistent. Furthermore, the standard deviations of both predictions were calculated to estimate the uncertainty of a prediction based on one temperature cycle assuming that the gas concentration was changing only slowly during the field test. To determine the standard deviation, a period with minimal signal changes was selected, here between 4 October 12:00, and 5 October 0:00, and the model predictions during this period were smoothed with the help of a sliding window with a length of 1 h. The standard deviation between the original model output and the smoothed data was calculated as an estimate of the noise level of the different models.

Next, the predictions obtained during release tests were compared to investigate the quantitative performance of the two different models. Furthermore, the prediction qualities of the TCOCNN models were analyzed regarding their cross-influence. In addition, the TCOCNN output was compared with the results obtained from the analytical instruments to further evaluate the capabilities compared to state-of-the-art systems.

Finally, the models were tested regarding their capability to detect gases not contained in the calibration, but belonging to the same chemical class as one trained gas (Table 5). Here, release tests performed with *m/p*-xylene (an aromatic) and isopropyl alcohol (an alcohol) were considered to determine the ability of the trained models to extrapolate to similar chemical compounds. This would show if the systematic approach with the MOS sensor, dynamic operation, and ML modeling could quantify individual gas components or provide an estimate of the total concentration of a certain chemical class.

Table 5. Chemical classes investigated in this publication [13].

Chemical Class (Representative)	P90 in $\mu\text{g}/\text{m}^3$ (ppb)	P95 in $\mu\text{g}/\text{m}^3$ (ppb)
Alcohols (ethanol)	320 (~170)	520 (~280)
Aldehydes (formaldehyde)	340 (~270)	480 (~390)
Aromatics (toluene)	190 (~50)	370 (~90)
Ketones (acetone)	250 (~100)	420 (~170)

3. Results

3.1. Calibration Results

Figure 4 shows that, in general, fewer data samples significantly increased the RMSE and also the uncertainty or rather variation of the RMSE. The specific RMSE mean and variance values illustrated in Figure 4 were based on the RMSEs achieved on the same training, validation, and testing data in 10 different runs using the TCOCNN.

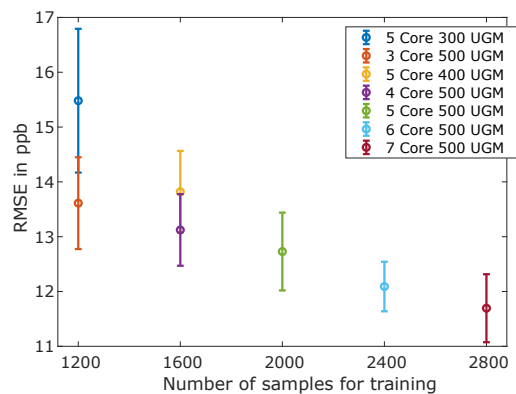


Figure 4. Obtained root-mean-squared error (RMSE) values for formaldehyde vs. the number of core samples and unique gas mixtures.

Obviously, fewer data samples degraded the prediction quality. Furthermore, Figure 4 illustrates with the comparison of five core samples for three-hundred UGM and three core samples for five-hundred UGM, i.e., both with one-thousand five-hundred cycles in total, that the number of unique gas mixtures is more important to achieve a high prediction quality than the number of core samples. Thus, a calibration should be biased towards containing more UGM with a shorter duration, resulting in fewer usable core samples, in accordance with the results of Robin et al. [33]. Note, however, that reducing the number of core samples did not decrease the overall duration of the calibration linearly, as temperature cycles recorded during a change of the gas composition cannot be evaluated. Moreover, Figure 4 illustrates that the difference between four core samples for five-hundred UGM and five core samples for four-hundred UGM, i.e., a total of two-thousand cycles each, resulted in only a minimal difference of the RMSE. Thus, it can be assumed that more than 500 UGM would not lead to a significant further reduction of the RMSE. This is also shown by the RMSE stagnating for more than five core samples, in agreement with previous results [33]. The best RMSE for formaldehyde for this dataset was achieved for five-hundred unique gas mixtures and seven core samples with an RMSE of 11.7 ppb. Nevertheless, because of the synchronization errors between the gas-mixing apparatus and the recording system, all further measurements were based on five core samples only, as these were always recorded under stable conditions.

For Figure 5a,b, the models were trained on the initial calibration data and the prediction was performed for the second part of the first recalibration (with extended gas concentrations). Figure 5c shows the results of a TCOCNN model that was trained on data including the initial calibration and the first part of the first recalibration with the prediction again performed on the second part of the first recalibration. Figure 5a shows that without any drift compensation, the prediction performance degraded severely over a period of six weeks (first field test period in a normal office environment) with a strong bias towards lower predicted concentrations and much higher uncertainty or variance of the prediction. Note that the TCOCNN did not predict negative concentration values; instead, many low concentrations were predicted as 0 ppb. Figure 5b illustrates that the one gas-sensitive layer that was excluded accounted for most of the drift, i.e., the major

part of the bias and the higher scatter of the predictions. Thus, excluding this layer already significantly improved the prediction quality, as indicated by the reduced variance, smaller linearity error, and reduced offset. The remaining linearity error can be attributed at least in part to a change of the formaldehyde test gas bottle between the initial calibration and the first recalibration. The test gas bottle concentrations had an uncertainty of 20%, which was most probably the cause for the remaining systematic linearity error. However, the gas-sensitive layer left out for the quantification of formaldehyde is in fact important for the detection and quantification of other VOCs, such as toluene, and also to reduce the cross-sensitivity to these gases. Thus, all available information should be used for building a comprehensive data-based model, and a different approach for reducing the drift is necessary. As previously reported, including data that already contain drift in the calibration, a so-called extended calibration [38], can improve the performance considerably, so this approach was also tested here. Figure 5c shows that the prediction based on extending the calibration dataset to include also data from the first recalibration after four weeks of field operation significantly improved the prediction quality. This model showed only a slight increase in the variance and a small offset and linearity error. Again, the linearity error could also be due to the change of the test gas bottles between the initial calibration and first recalibration, which would result in a systematic error.

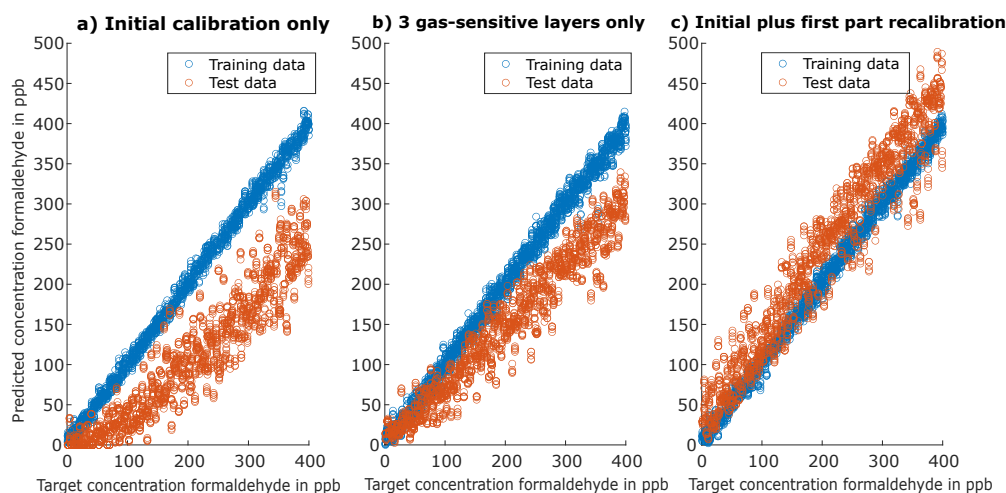


Figure 5. Evaluation of the target gas formaldehyde over several weeks. Test data always consist of the data of the second part of the first recalibration. (a) Results based on training with initial calibration only. (b) Results based on training with initial calibration only, but without one gas-sensitive layer that shows large drift over time. (c) Results based on training with the initial calibration and the first part of first recalibration.

This following section of the results demonstrates the performance achieved with the TCOCNN approach for the prediction of all gases present in the calibration dataset. The results obtained for formaldehyde on the number of core samples and unique gas mixtures, as well as the proven approach for compensating sensor drift were transferred to build the models for the other target gases. Thus, the presented results were always based on training with five core samples and extended calibration containing data from the initial calibration and the first part of the first recalibration. As before, a gas-mixture-based validation was performed. Accordingly, the data were split into 70% training, 10% validation, and 20% testing. In Table 6, the hyperparameters selected with the help of the training data, the validation data, and the NAS are listed. This shows that for most gases, the stride sizes of the first layer need to be larger than in the following layers.

Table 6. Optimized hyperparameters found during neural architecture search (NAS).

Substance	Initial Learning Rate (Log Scale)	Number of Filters (First Two Layers)	Kernel Size (First Two Layers)	Stride Size (First Layer)	Dropout	Number of Neurons (FC)
Acetone	1×10^{-4}	142	48	34	31.65%	1084
Toluene	2×10^{-4}	240	69	17	37.06%	2462
Formaldehyde	3×10^{-4}	183	55	18	49.75%	1188
Ethanol	2×10^{-4}	228	73	34	49.55%	2310
VOC _{sum}	1×10^{-4}	77	78	44	30.89%	1373
CO	6×10^{-4}	151	52	30	34.96%	2468
Hydrogen	1×10^{-4}	77	41	19	49.07%	1088

Figure 6 illustrates the RMSE results of the 70/10/20 data split on the initial calibration and the first part of the first recalibration for the FESR model (blue) and the mean value plus the standard deviation of the TCOCNN (orange) with the hyperparameters listed in Table 6. This shows that the TCOCNN achieved at least the same and often a significantly lower RMSE compared to the FESR model for all gases. The most significant improvement was achieved for formaldehyde, where the mean RMSE of the TCOCNN was less than half of the RMSE of the FESR model (15.4 ppb for TCOCNN vs. 31.3 ppb for FESR). This significant improvement for formaldehyde was probably due to the fact that the underlying model of the TCOCNN was originally optimized for formaldehyde quantification, i.e., more hyperparameters were optimized for formaldehyde than for the other gases. Thus, extending the NAS to include also the parameters that were kept constant in this study might result in similar improvements also for the other gases. Nevertheless, the results clearly showed that the TCOCNN models outperformed the FESR models regardless of the gas on which they were trained. Moreover, the variations of the RMSE caused by the different initializations of the TCOCNN were relatively small; thus, a stable model was achieved even if the network was trained only once. The variance of the RMSE is not given for the FESR method as the PLSR is deterministic, i.e., always produces the same result with the hyperparameters specified during the 10-fold cross-validation.

3.2. General Field Test Results

After showing that the TCOCNN can successfully predict the concentration of various gases in complex laboratory environments, this part focuses on quantifying the trained gases in a real indoor air environment. First, the general prediction quality of the TCOCNN for the various gases was compared to the predictions of the FESR model. Figure 7 illustrates the prediction of indoor air between September 26 and October 18 for formaldehyde and hydrogen. These two gases were chosen as they showed relevant aspects; the results for the other gases were similar. First, a constant offset was observed between both models, with the FESR model for formaldehyde indicating significantly higher concentrations than the TCOCNN model (average offset 140 ppb), while for hydrogen, the prediction of the TCOCNN was slightly higher with an average offset of 98 ppb. These differences were probably caused by the presence of additional gases in the room, which were not part of the calibration and were therefore not (fully) compensated by the data-based models and/or by the gas concentrations of the trained gases in the indoor environment outside of the trained ranges. Without reference measurements, it is not possible to determine which value is correct; in fact, both could be similarly off with one prediction being too high, the other too low. Nevertheless, at least for hydrogen, the baseline of the TCOCNN model seemed to be more realistic, as after ventilation events, the TCOCNN model indicated concentrations around 500 ppb, corresponding to the natural background level [39]. For formaldehyde as well, the lower average concentrations indicated by the TCOCNN model seemed more realistic, as the FESR model indicated concentrations well above the

WHO recommended limit value of 80 ppb [40]. This will be investigated in the future with formaldehyde reference measurements as described in the relevant standards [40]. More importantly, however, both models were in agreement concerning the relative changes of the gas concentrations, which would be required to indicate changes in the indoor air quality, e.g., for demand-controlled ventilation.

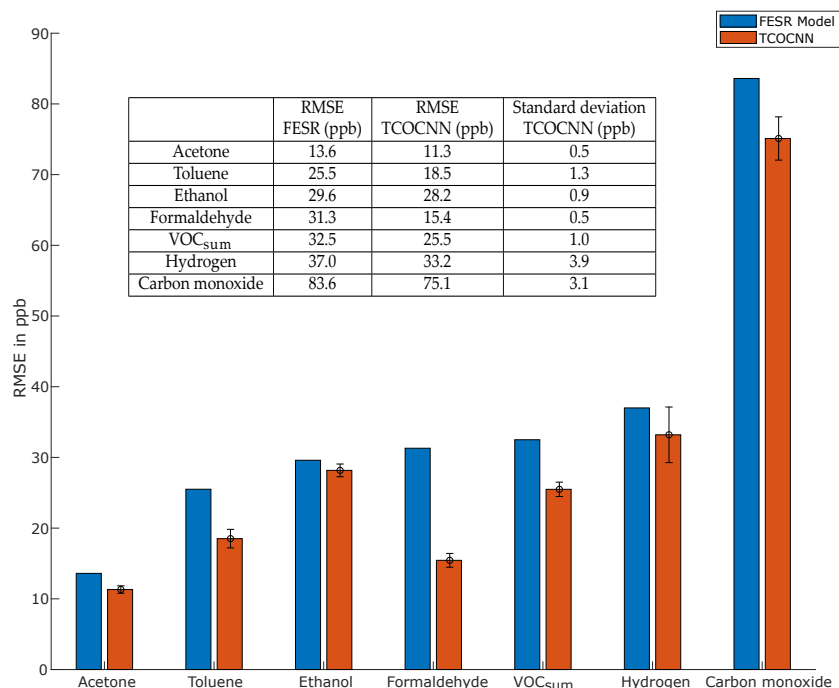


Figure 6. Comparison of the RMSE values obtained with FESR (adapted from [13]) and the TCOCNN.

In addition, the FESR models showed much higher noise or short-term fluctuations compared to the TCOCNN models. This was specifically dominant for hydrogen and formaldehyde. To quantify this effect further, the standard deviation was calculated for all gases for both models. For this, a fairly stable time period without release events (4 October 12:00, to 5 October 00:00) was chosen, and the standard deviation between the model predictions and their hourly average as the estimated mean signal was calculated, resulting in the values given in Table 7. The ratio of the noise levels of the FESR model vs. the TCOCNN model was between 1.4 (for ethanol) and 5.2 (for hydrogen) to 6.2 (for formaldehyde), which is also evident from Figure 7. Furthermore, some predictions of the FESR model were below zero, which was not the case for the TCOCNN. These short events were caused by ventilating the room in which the experiments were performed, which probably resulted in very low gas concentrations below the calibrated range. Thus, it was not surprising that the models were not able to quantify the gases correctly in these conditions. Taking all observations into account, we concluded that for all gases, the overall quality of the TCOCNN model was more suitable for monitoring real indoor air as no false-negative values were obtained and the noise in a room where the gas composition changes only slowly is much lower. The absolute error of both models can only be determined with calibrated reference measurements, which were not available for this study, but which will be performed in the near future.

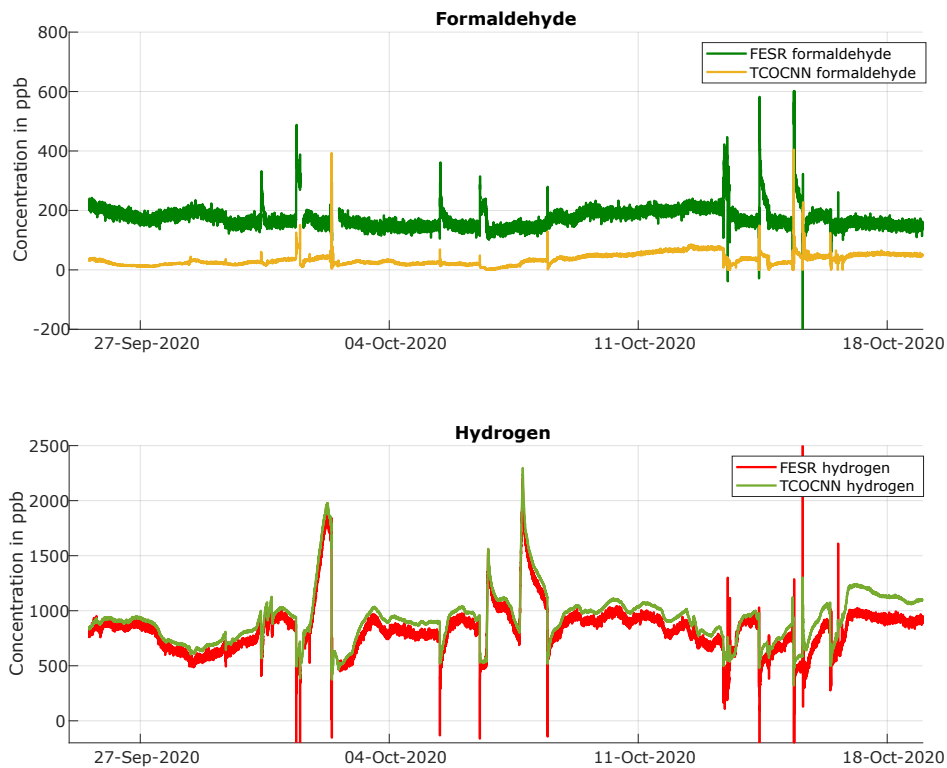


Figure 7. Comparison of the results obtained during field tests with the FESR and TCOCNN models for formaldehyde and hydrogen.

Table 7. Standard deviation during the field test of the TCOCNN and FESR.

	Standard Deviation FESR (ppb)	Standard Deviation TCOCNN (ppb)
Toluene	9.7	2.0
Formaldehyde	9.9	1.6
Carbon monoxide	33.6	5.5
VOC _{sum}	15.8	4.5
Acetone	5.7	2.6
Ethanol	15.0	10.7
Hydrogen	19.0	3.7

3.3. Results of the Release Tests for the Trained Gases

After the general observations of the similarities and differences between both models, this section focuses on the release tests performed during the field test period. Since formaldehyde as a carcinogenic gas and carbon monoxide as a toxic gas could not be actively released, the following results concentrated on release tests of acetone, ethanol, toluene, and hydrogen.

First, the releases of individual VOCs were analyzed in detail. Figure 8a illustrates the release of toluene on November 2 (Test 7). Both the FESR and the TCOCNN models

predicted similar average concentrations over time, but with higher noise for the FESR model, as observed before. The FESR model indicated an increase of approximately 600 ppb, which is in accordance with the amount released, cf. (Table 2), while the TCOCNN indicated a slightly smaller increase of 500 ppb. In addition to the MOS gas sensors, the release was also monitored with the portable GC-PID (X-pid 9500), which indicated a similar rapid increase and slow decrease of the toluene concentration, but a higher absolute value with an increase of approximately 700 ppb. Note, however, that the limit of quantification (LOQ) for the X-pid 9500 is 1 ppm for toluene according to the manufacturer (500 ppb for acetone).

For acetone, Figure 8b shows an offset between the absolute concentrations predicted by the TCOCNN and FESR models: the baseline concentration of the TCOCNN model was approximately 72 ppb, while the FESR model indicated a baseline concentration of approximately 120 ppb. This difference of 48 ppb was also observed during the release of acetone with both models indicating the same increase of approximately 450 ppb and showing the same decline vs. time. The X-pid 9500 indicated a baseline value similar to the FESR model, but again predicted a larger increase of approximately 700 ppb with the same shape over time. The expected increase caused by the amount of acetone released was 600 ppb, but the actual concentration at the site of the sensors could be higher or lower depending on the distribution in the room and also secondary effects, such as adsorption on surfaces. Nevertheless, both data-driven models were clearly capable of detecting acetone with a high temporal resolution.

Figure 8c illustrates the release of hydrogen from a test gas bottle with an expected maximum concentration increase of approximately 2 ppm. The graphs show the corresponding values indicated by the TCOCNN and FESR models, as well as the GC-RCP reference instrument. As already observed in Figure 7, the baseline concentration indicated by the TCOCNN model was slightly higher compared to the FESR model, and the noise level of the TCOCNN was much smaller compared to the FESR model. The increase of the hydrogen concentration indicated by both models was similar (around 1500 ppb) and realistic compared to the amount of released gas, especially considering the relatively slow release over several hours, where some gas exchange and therefore loss of hydrogen is unavoidable. The GC-RCP (limit of detection 10 ppb) indicated a similar increase of the hydrogen concentration, but with an even lower baseline compared to both sensor models. We suspect that the RCP was underestimating the hydrogen concentration slightly [13], which was especially evident during ventilation events where the GC-RCP indicated concentrations well below the natural background concentration of 500 ppb [41]. Additionally, the RCP also showed a larger noise level compared to the TCOCNN model. Regarding the other models, only ethanol and carbon monoxide showed a cross-influence, which was relatively small compared to the released amount of hydrogen (see Figure A1).

Finally, Figure 8d illustrates the values of the respective TCOCNN models during a simultaneous release of acetone, ethanol, and toluene. For toluene, the peak increase was approximately 400 ppb, which is slightly lower than during the release test of toluene only, similarly for acetone with an increase of approximately 360 ppb. Nevertheless, all three models detected the release of the various compounds with a high temporal resolution, which is especially evident when observing the different shapes of the release peaks: acetone with the lowest boiling point showed the sharpest peak, while toluene with a comparatively high boiling point showed a much broader and rounded release peak. To further elucidate the simultaneous evaluation of the various data-based models, Figure A1 shows the behavior of all other models during those release tests, and the following figure shows as an example all calculated model outputs during a specific release test including the VOC_{sum} model, indicating the sum concentration of all VOCs.

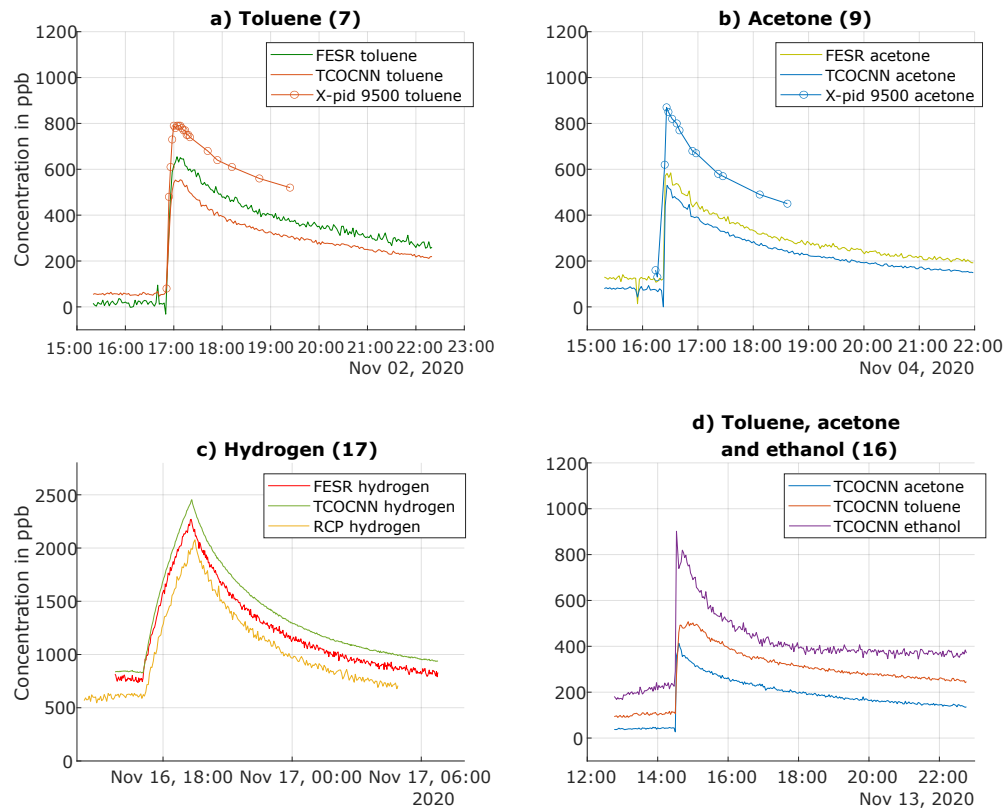


Figure 8. Prediction of gas concentrations during release tests for various trained gases using different evaluation models and a comparison with the results of the analytical instruments (adapted from [13]).

Figure 9 illustrates the TCOCNN model output during two simultaneous releases of acetone and toluene. The increase of the acetone and toluene concentrations indicated by the models was similar to the previous test shown above (Tests 7 and 9). For toluene and acetone, the signal increased by approximately 550 ppb, which is close to the expected value of 600 ppb. For both gases, the X-pid 9500 again indicated a significantly higher concentration increase, but with a similar shape over time. Parallel TD-GC-MS (LOQ approximately 50 ppb for toluene) analysis with samples taken over 30 min intervals showed absolute concentrations, as well as an increase of the toluene concentration of approximately 550 ppb, which is very similar to the values obtained from the TCOCNN model; acetone was not evaluated with the TD-GC-MS method in this study as the sampling protocol would have to be adjusted for accurate quantification of this VVOC. Again, the FESR model showed an offset compared to the TCOCNN model with slightly higher baselines and also somewhat larger concentration increases for both gases; the absolute values of the FESR model were between the results obtained with the X-pid 9500 and the TD-GC-MS. Note, however, that the GC-MS was not calibrated before these measurements. Finally, both the FESR and TCOCNN models, as well as the X-pid 9500 indicated that the toluene increase during the second simultaneous release was much slower (no Tenax samples were collected during this second release). This slower increase was probably caused by the significantly lower temperature during the second release (at night), resulting in much slower evaporation of toluene. Again, all three methods—X-pid 9500 and both

MOS data-based models—showed the same shape over time, indicating their potential for monitoring IAQ in real-time.

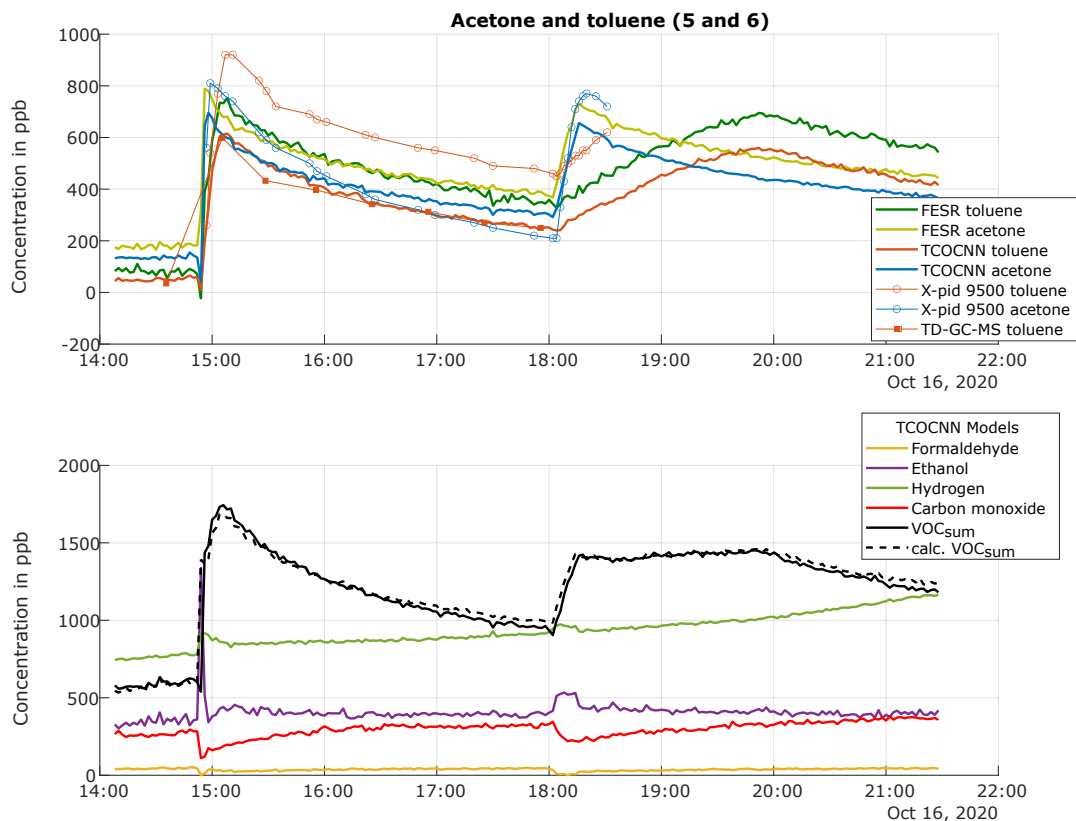


Figure 9. Prediction of gas concentrations during Release Tests 5 and 6 (acetone and toluene) showing the various models trained compared to the analytical measurements (adapted from [13]).

The outputs of the other TCOCNN models are shown in the lower part of Figure 9. Note that the trained VOC_{sum} model (solid black line) actually showed a nearly identical behavior compared to adding the concentrations indicated by all four separate VOC models for acetone, ethanol, formaldehyde, and toluene with an increase of approximately 1100 ppb during the first combined release. The VOC_{sum} model also showed the different evaporation speeds during the second release with a first fast increase caused by the release of acetone followed by an almost constant concentration due to the offsetting effects of increasing toluene and decreasing acetone concentrations.

For the other two VOCs, ethanol and formaldehyde, which were not released and were therefore expected to have a constant concentration, the TCOCNN models showed very little cross-sensitivity: a small drop was observed for formaldehyde and a short sharp increase for ethanol at the first release, but both models recovered their previous baseline quickly. These short-term effects were probably caused by the person performing the release test entering the room. However, a significant cross-sensitivity was observed for carbon monoxide, where the model output dropped by approximately 170 ppb during both releases and then recovered only slowly. A similar, but opposite effect was observed for the hydrogen model, which showed a minor increase during both release tests. Furthermore, a general baseline drift of the indicated hydrogen concentration was observed with a different

behavior over time compared to the released VOCs. This effect can be attributed to VOC decomposition, leading to an increase of hydrogen; in fact, the model predictions for hydrogen over time were in good agreement with the GC-RCP reference instrument [41].

The behavior regarding the cross-influence of different gases was observed for all release tests. Independent of the specific VOC released, the other VOC models showed only minor effects, while carbon monoxide showed a significant cross-sensitivity, which was probably caused by the comparatively low sensitivity of the SGP30 to carbon monoxide [37]. Hydrogen actually showed large variations during the field tests, which were not correlated with the release tests, indicating other sources inside the room with a diurnal pattern. The VOC_{sum} signal always accurately indicated the combined concentration of the various VOCs. The release of hydrogen did not result in a significant response of any other model, illustrating the high selectivity achieved for hydrogen, as previously reported for the FESR model [41].

3.4. Results of Release Tests for Gases Not Trained

To further elucidate the selectivity of the various models, release tests were performed with gases not included in the calibration, but from the same chemical classes, i.e., m/p-xylene as a second aromatic compound and isopropyl alcohol as a second alcohol. Again, we compared the performance of the TCOCNN model with the FESR model. Figure 10 illustrates the predictions of both models calibrated for toluene during the release of m/p-xylene (Test Number 14). Both the FESR model and the TCOCNN model indicated a similar evolution over time of the toluene concentration. The indicated increase for the FESR was approximately 450 ppb, so again, close to the theoretically expected increase of 600 ppb, while the increase with the TCOCNN was slightly smaller with 350 ppb. The X-pid 9500, on the other hand, showed a large offset with a baseline value of almost 500 ppb and an increase similar to the FESR model. These results showed that both data-driven models were capable of quantifying aromatics, i.e., chemicals from the same chemical class as the calibrated toluene, in agreement with previous results for VOC identification [42]. The other VOC models were not influenced by the release of m/p-xylene, indicating good selectivity (see Figure A2 in Appendix A). The VOC_{sum} model also responded to the release of m/p-xylene, again similar to toluene. Finally, a significant cross-sensitivity of the carbon monoxide model was also observed during the release of m/p-xylene, similar to the release of toluene.

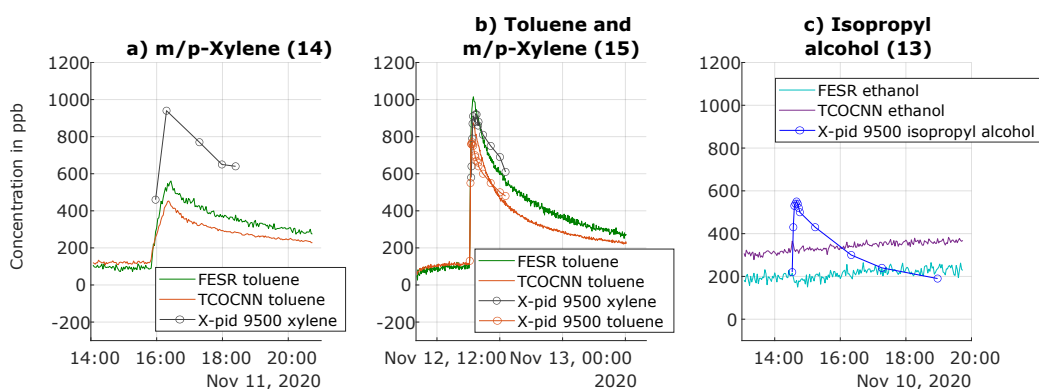


Figure 10. Prediction of gas concentrations for release tests of gases not contained in the calibration. (a) Release Test Number 14 of m/p-xylene. (b) Release Test Number 15 of toluene and m/p-xylene. (c) Release Test Number 13 of isopropyl alcohol (adapted from [13]).

The second release test in Figure 10 illustrates that the combined release of toluene and m/p-xylene could be detected by both data-based models, as well as the X-pid 9500, which can also discriminate between both aromatics, while the data-driven models could

not discriminate the two gases; the indicated increase was close to the sum of the previous releases. Again, a similar behavior was observed for the other TCOCNN models of the other target gases (see Figure A2 in Appendix A). Note that discrimination between both aromatic compounds might be possible with the data-based models if both gases are included separately in the calibration.

On the other hand, both the TCOCNN, as well as the FESR model did not indicate the release of isopropyl alcohol as ethanol, as shown in Figure 10c. Only a small upwards drift for ethanol is visible. We cannot provide a conclusive answer, but the upward drift already started before the release of isopropyl alcohol, so this was probably due to some unrelated background event. Interestingly, the FESR model showed a slight decrease when isopropyl alcohol was released, but otherwise, the same trend as the TCOCNN model. Regarding the other TCOCNN models, no model showed a significant reaction to this gas (see Figure A2 in Appendix A) not included in the calibration. Only the X-pid 9500 was capable of detecting and quantifying isopropyl alcohol as well. This result showed that the models were not always capable of indicating gases of the same chemical class, in this case alcohols. While this might be seen as a drawback, because a more complex calibration would be required to obtain a valid VOC_{sum} model, when considering formaldehyde, a high selectivity also to other aldehydes such as acetaldehyde is preferable to discriminate the health impact of the various gases.

4. Discussion

This contribution discussed the application of a deep-learning model for evaluating the complex data patterns recorded with a metal oxide gas multisensor (SGP30) in temperature-cycled operation (TCO) to independently determine multiple gas concentrations from a single sensor element. The novel deep-learning approach named the TCOCNN was based on a 10-layer CNN design. In this work, the performance of the TCOCNN was studied both concerning the optimization using different data configurations by varying the number of temperature cycles per gas mixture (core samples) and the number of unique gas mixtures (UGM). As expected, the number of UGM is more important to achieve a low RMSE of the prediction than the number of (basically redundant) samples. Note that, when taking the number of independent variables into account (six gases plus RH), the number of unique gas exposures for reliably achieving a stable ML model is actually fairly low. A full factorial calibration with seven parameters at four levels each would result in a total of more than 16,000 tests. Second, we studied the potential to suppress drift, which is often observed for MOS gas sensors, by extending the calibration data to also include data from an additional one-hundred gas exposures obtained in the second calibration run after four weeks of operation in the field. This extended calibration succeeded in greatly reducing the offset, linearity, and noise error observed when only the original calibration data were used to build the model. With this approach, a stable prediction of the formaldehyde concentration with an uncertainty of approximately 42 ppb was achieved over a period of at least 6 wk. Third, the TCOCNN approach was successfully tested not only for formaldehyde, but also for predicting at the ppb level the concentrations of the other gases included in the calibration plus an additional model for the sum of all VOCs, VOC_{sum} . Here, a neural architecture search with Bayesian optimization was performed to select suitable hyperparameters for the TCOCNN models. The results showed that stable models were reproducibly obtained with this approach, achieving a performance at least as good as the previous linear models in terms of the RMSE for the calibration data. The improvement was most significant for formaldehyde, where the RMSE was more than halved. The different initializations of the TCOCNN only resulted in negligible variation of the RMSE between 1 ppb and 8 ppb, indicating the excellent reproducibility of the model-building approach. Applying these models to data from the field tests showed that the TCOCNN models had lower noise for real field data compared to the previous FESR model and did not predict negative gas concentrations even when operated outside the calibrated gas concentration range. We did observe significant offsets between both ML models,

which were probably caused by unknown gases not contained in the calibration, but present during the field tests. This will require further analysis with calibrated reference instruments to determine which model provides higher absolute accuracy. However, variations of the gas concentrations can be accurately monitored with high temporal resolution, as demonstrated with various release tests during the field test period. In fact, the indicated concentration increases of the released gas closely matched the expected theoretical values and often significantly outperformed the mobile GC-PID and GC-RCP instruments used for comparison both when releasing hydrogen or single VOCs and when simultaneously releasing VOC mixtures. The best absolute agreement was observed for the TCOCNN model and the gold standard TD-GC-MS for toluene monitoring. Minimal cross-sensitivity was observed for the six gases tested in this study with only the carbon monoxide model showing a slightly higher cross-sensitivity to VOCs, probably due to the low overall sensitivity of the SGP30 to CO. Finally, release tests were performed with gases not contained in the calibration. Here, two different results were observed for m/p-xylene and isopropyl alcohol. While the TCOCNN for toluene was also able to detect and quantify m/p-xylene with reasonable accuracy, neither the ethanol model nor any other reacted to isopropyl alcohol. This shows that in some cases (toluene and m/p-xylene), the sensor actually detects a certain chemical class, here aromatics, while in others, the gases (ethanol and isopropyl alcohol), although belonging to the same chemical group, here alcohols, induce unique sensor response patterns allowing discrimination and quantification of the individual components. This aspect will require further examination, as both effects are beneficial in some ways and undesirable in others. Quantifying all gases from the same chemical group after the calibration of only a single representative would greatly reduce the complexity of the sensor calibration. On the other hand, being able to quantify individual gases even against others from the same chemical class is important for the accurate determination of relevant indoor pollutants such as formaldehyde (vs. acetaldehyde and others) or benzene (vs. toluene and xylene). The presented systematic approach could provide the basis for the development of high-performance application-specific VOC sensor systems taking target and interfering gases into account.

Regarding the computation time for hyperparameter tuning and model training, it should be noted that the TCOCNN model training requires much more time. While the FESR method requires up to 24 h for a full evaluation including hyperparameter optimization, the TCOCNN including NAS requires several days. Therefore, a reduction of the computational complexity of the TCOCNN is desirable for future investigations.

5. Conclusions and Outlook

All-in-all, the novel TCOCNN model presented here outperformed state-of-the-art ML models such as the FESR approach both in laboratory measurements and field tests, achieving higher accuracy and lower noise with the same temporal resolution, especially in real application environments. Furthermore, the TCOCNN model achieved similar quantification performance as the tested analytical systems, which however were more robust in the case of unknown gases.

On the other hand, the TCOCNN approach is still not fully investigated, as it is not yet clear on which features the model is basing its decision and how the hyperparameters influence the model performance for various target gases. Furthermore, the absolute accuracy has to be determined with calibrated reference instruments, which were not available for this study. Similarly, the selectivity and quantification performance for gases from the same chemical group as one of the trained gases needs to be studied further to make full use of this effect to reduce the calibration complexity while still achieving the required level of selectivity. Finally, we are planning to investigate methods such as transfer learning to reduce or even eliminate the required recalibration for drift compensation.

Author Contributions: Conceptualization, Y.R., T.B. and A.S.; methodology, Y.R., P.G. and J.A.; software, Y.R. and P.G.; validation, Y.R., T.B., P.G. and J.A.; formal analysis, Y.R.; investigation, Y.R.; resources, A.S.; data curation, J.A.; writing—original draft preparation, Y.R.; writing—review and editing, Y.R., T.B., C.S., J.A., T.S. and A.S.; visualization, Y.R.; supervision, Y.R., T.B., T.S. and A.S.; project administration, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: Part of this research was performed within the project “SE-ProEng” funded by the European Regional Development Fund (ERDF). We acknowledge support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) and Saarland University within the funding program Open Access Publishing.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The underlying data are available on Zenodo and were published with our first publication on the field tests: DOI:10.5281/zenodo.4593853. Title: Measuring Hydrogen in Indoor Air with a Selective Metal Oxide Semiconductor Sensor: Dataset. Authors: Johannes Amann, Tobias Baur, Caroline Schultealbert, <https://zenodo.org/record/4593853> (accessed on 17 May 2021).

Acknowledgments: We thank Rainer Lammertz Pure Gas Products for providing the Peak Performer 1 reference instrument and Dräger Safety AG & Co KGaA for providing the X-pid 9500 for this study.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional neural network
FE	Feature extraction
FESR	Feature extraction selection regression
FS	Feature selection
GC-PID	Gas chromatograph with photo-ionization detection
GC-RCP	Gas chromatograph with reducing compound photometer
IAQ	Indoor air quality
LOQ	Limit of quantification
MFC	Mass flow controller
ML	Machine learning
MOS	Metal oxide semiconductor
NAS	Neural architecture search
PLSR	Partial least squares regression
PM	Particulate matter
RH	Relative humidity
RFE	Recursive feature elimination
RMSE	Root-mean-squared error
SVOC	Semivolatile organic compounds
TCO	Temperature-cycled operation
TD-GC-MS	Thermo-desorption gas chromatography mass spectrometry
UGM	Unique gas mixtures
VOC	Volatile organic compounds
VVOC	Very volatile organic compounds

Appendix A

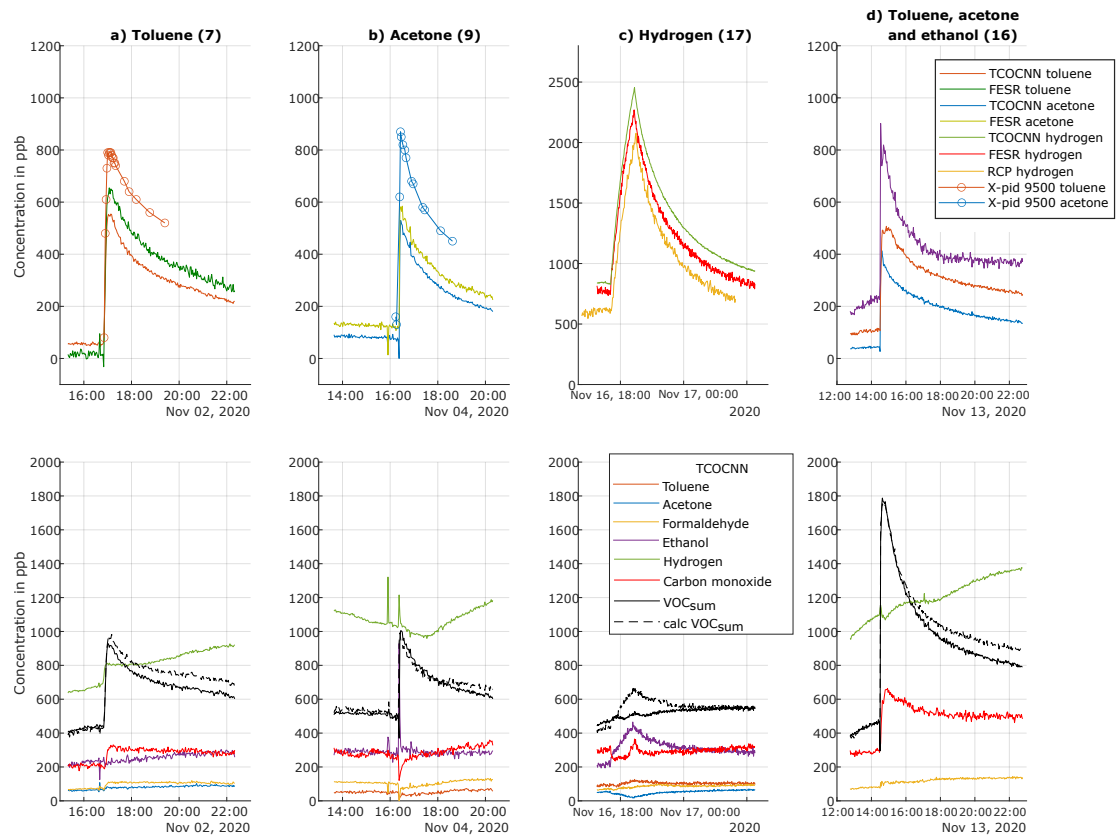


Figure A1. Prediction of gas concentrations during the release tests for various trained gases using different evaluation models and comparison with the results of analytical instruments together with all predictions of the TCOCNN for the other gases. (a) Release Test Number 7 of toluene. (b) Release Test Number 9 of acetone. (c) Release Test Number 17 of hydrogen. (d) Release Test Number 16 of toluene, acetone and ethanol (adapted from [13]).

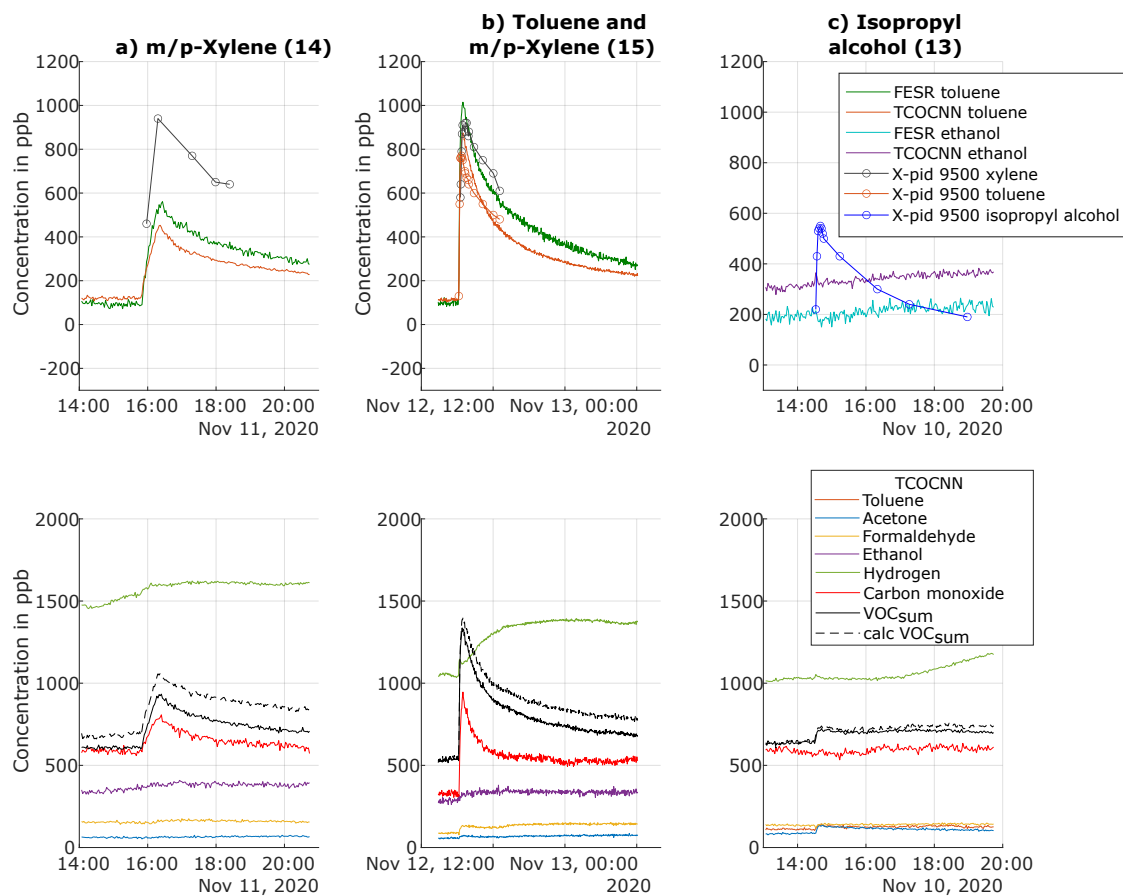


Figure A2. Prediction of gas concentrations for the release tests of gases not contained in the calibration together with all predictions of the TCOCNN for the other gases. (a) Release Test Number 14 of m/p-xylene. (b) Release Test Number 15 of toluene and m/p-xylene. (c) Release Test Number 13 of isopropyl alcohol (adapted from [13]).

References

- Asikainen, A.; Carrer, P.; Kephelopoulou, S.; De Oliveira Fernandes, E.; Wargocki, P.; Hänninen, O. Reducing burden of disease from residential indoor air exposures in Europe (HEALTHVENT project). *Environ. Health* **2016**, *15*, 61–72. [CrossRef] [PubMed]
- Hauptmann, M.; Lubin, J.H.; Stewart, P.A.; Hayes, R.B.; Blair, A. Mortality from Solid Cancers among Workers in Formaldehyde Industries. *Am. J. Epidemiol.* **2004**, *159*, 1117–1130. [CrossRef] [PubMed]
- United Nations, Department of Economic and Social Affairs, Sustainable Development. Ensure Healthy Lives and Promote Well-Being for All at All Ages. Available online: <https://sdgs.un.org/goals/goal3> (accessed on 15 October 2021).
- Valero, E. *Advanced Nanomaterials for Inexpensive Gas Microsensors: Synthesis, Integration and Applications*; Elsevier: Amsterdam, The Netherlands, 2020.
- Molhave, L.; Nielsen, G.D. Interpretation and Limitations of the Concept “Total Volatile Organic Compounds” (TVOC) as an Indicator of Human Responses to Exposures of Volatile Organic Compounds (VOC) in indoor air. *Indoor Air* **1992**, *2*, 65–77. [CrossRef]
- Salthammer, T. Very volatile organic compounds: An understudied class of indoor air pollutants. *Indoor Air* **2014**, *26*, 25–38. [CrossRef]
- Pettenkofer, M. *Über den Luftwechsel in Wohngebäuden*; Literarisch-Artistische Anstalt der J.G. Cotta’schen Buchhandlung: München, Germany, 1858.
- Yeoman, A.M.; Shaw, M.; Carslaw, N.; Murrells, T.; Passant, N.; Lewis, A.C. Simplified speciation and atmospheric volatile organic compound emission rates from non-aerosol personal care products. *Indoor Air* **2020**, *30*, 459–472. doi: 10.1111/ina.12652. [CrossRef]

9. Coggon, M.M.; McDonald, B.C.; Vlasenko, A.; Veres, P.R.; Bernard, F.; Koss, A.R.; Yuan, B.; Gilman, J.B.; Peischl, J.; Aikin, K.C.; et al. Diurnal Variability and Emission Pattern of Decamethylcyclpentasiloxane (D5) from the Application of Personal Care Products in Two North American Cities. *Environ. Sci. Technol.* **2018**, *52*, 5610–5618. [[CrossRef](#)]
10. Mølhave, L. Indoor air pollution due to organic gases and vapours of solvents in building materials. *Environ. Int.* **1982**, *8*, 117–127. [[CrossRef](#)]
11. Schütze, A.; Baur, T.; Leidinger, M.; Reimringer, W.; Jung, R.; Conrad, T.; Sauerwald, T. Highly Sensitive and Selective VOC Sensor Systems Based on Semiconductor Gas Sensors: How to? *Environments* **2017**, *4*, 20. [[CrossRef](#)]
12. Haddad, S.; Synnefa, A.; Marcos, M.Á.P.; Paolini, R.; Delrue, S.; Prasad, D.; Santamouris, M. On the potential of demand-controlled ventilation system to enhance indoor air quality and thermal condition in Australian school classrooms. *Energy Build.* **2021**, *238*, 110838. [[CrossRef](#)]
13. Baur, T.; Amann, J.; Schultealbert, C.; Schütze, A. Field Study of Metal Oxide Semiconductor Gas Sensors in Temperature Cycled Operation for Selective VOC Monitoring in Indoor Air. *Atmosphere* **2021**, *12*, 647. [[CrossRef](#)]
14. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe Nevada, CA, USA, 3 December–6 December 2012; Curran Associates Inc.: Red Hook, NY, USA, 2012; Volume 1, pp. 1097–1105.
15. Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, L.; Wang, G.; et al. Recent Advances in Convolutional Neural Networks. *arXiv* **2017**, arXiv:1512.07108v6.
16. White, C.; Neiswanger, W.; Savani, Y. BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search. *arXiv* **2020**, arXiv:1910.11858v3.
17. Vito, S.D.; Massera, E.; Piga, M.; Martinotto, L.; Francia, G.D. On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario. *Sens. Actuators B Chem.* **2008**, *129*, 750–757. [[CrossRef](#)]
18. Szczurek, A.; Szcwówka, P.; Licznarski, B. Application of sensor array and neural networks for quantification of organic solvent vapours in air. *Sens. Actuators B Chem.* **1999**, *58*, 427–432. [[CrossRef](#)]
19. Han, L.; Yu, C.; Xiao, K.; Zhao, X. A New Method of Mixed Gas Identification Based on a Convolutional Neural Network for Time Series Classification. *Sensors* **2019**, *19*, 1960. [[CrossRef](#)]
20. Wang, S.; Hu, Y.; Burgues, J.; Marco, S.; Liu, S.C. Prediction of Gas Concentration Using Gated Recurrent Neural Networks. In Proceedings of the 2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Genova, Italy, 31 August–2 September 2020. [[CrossRef](#)]
21. Chen, Z.; Zheng, Y.; Chen, K.; Li, H.; Jian, J. Concentration Estimator of Mixed VOC Gases Using Sensor Array With Neural Networks and Decision Tree Learning. *IEEE Sens. J.* **2017**, *17*, 1884–1892. [[CrossRef](#)]
22. Xu, Y.; Meng, R.; Zhao, X. Research on a Gas Concentration Prediction Algorithm Based on Stacking. *Sensors* **2021**, *21*, 1597. [[CrossRef](#)]
23. Benrekia, F.; Attari, M.; Bouhedda, M. Gas Sensors Characterization and Multilayer Perceptron (MLP) Hardware Implementation for Gas Identification Using a Field Programmable Gate Array (FPGA). *Sensors* **2013**, *13*, 2967–2985. [[CrossRef](#)]
24. Feng, S.; Farha, F.; Li, Q.; Wan, Y.; Xu, Y.; Zhang, T.; Ning, H. Review on Smart Gas Sensing Technology. *Sensors* **2019**, *19*, 3760. [[CrossRef](#)]
25. Bastuck, M. Improving the Performance of Gas Sensor Systems with Advanced Data Evaluation, Operation, and Calibration Methods. Ph.D. Thesis, Department Systems Engineering, Shaker Verlag, Saarland University, Düren, Germany, 2019.
26. Ruffner, D.; Hoehne, F.; Bühler, J. New Digital Metal-Oxide (MOx) Sensor Platform. *Sensors* **2018**, *18*, 1052. [[CrossRef](#)]
27. Baur, T.; Bastuck, M.; Schultealbert, C.; Sauerwald, T.; Schütze, A. Random gas mixtures for efficient gas sensor calibration. *J. Sens. Sens. Syst.* **2020**, *9*, 411–424. [[CrossRef](#)]
28. Helwig, N.; Schüler, M.; Bur, C.; Schütze, A.; Sauerwald, T. Gas mixing apparatus for automated gas sensor characterization. *Meas. Sci. Technol.* **2014**, *25*, 055903. [[CrossRef](#)]
29. Schultealbert, C.; Baur, T.; Schütze, A.; Sauerwald, T. Facile Quantification and Identification Techniques for Reducing Gases over a Wide Concentration Range Using a MOS Sensor in Temperature-Cycled Operation. *Sensors* **2018**, *18*, 744. [[CrossRef](#)] [[PubMed](#)]
30. Baur, T.; Schütze, A.; Sauerwald, T. Optimierung des temperaturzyklischen Betriebs von Halbleitersensoren (Optimization of temperature-cycled operation of semiconductor gas sensors). *TM-Tech. Mess.* **2015**, *82*, 187–195. [[CrossRef](#)]
31. Schultealbert, C.; Baur, T.; Schütze, A.; Böttcher, S.; Sauerwald, T. A novel approach towards calibrated measurement of trace gases using metal oxide semiconductor sensors. *Sens. Actuators B Chem.* **2017**, *239*, 390–396. [[CrossRef](#)]
32. Bastuck, M.; Baur, T.; Schütze, A. DAV³E a MATLAB toolbox for multivariate sensor data evaluation. *J. Sens. Sens. Syst.* **2018**, *7*, 489–506. [[CrossRef](#)]
33. Robin, Y.; Goodarzi, P.; Baur, T.; Schultealbert, C.; Schütze, A.; Schneider, T. Machine Learning based calibration time reduction for Gas Sensors in Temperature Cycled Operation. In Proceedings of the 2021 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Glasgow, UK, 17–20 May 2021; pp. 1–6. [[CrossRef](#)]
34. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian Optimization of Machine Learning Algorithms. *arXiv* **2012**, arXiv:1206.2944v2.
35. Vergara, A.; Vembu, S.; Ayhan, T.; Ryan, M.A.; Homer, M.L.; Huerta, R. Chemical gas sensor drift compensation using classifier ensembles. *Sens. Actuators B Chem.* **2012**, *166–167*, 320–329. [[CrossRef](#)]

36. Artursson, T.; Eklöv, T.; Lundström, I.; Martensson, P.; Sjöström, M.; Holmberg, M. Drift correction for gas sensors using multivariate methods. *J. Chemom.* **2000**, *14*, 711–723. [[CrossRef](#)]
37. Amann, J.F. Möglichkeiten und Grenzen des Einsatzes von Halbleitersensoren im temperaturzyklischen Betrieb für die Messung der Innenraumluftqualität—Kalibrierung, Feldtest, Validierung. Master's Thesis, Universität des Saarlandes, Saarbrücken, Germany, 2021.
38. Bur, C.; Engel, M.; Horras, S.; Schütze, A. Drift compensation of virtual multisensor systems based on extended calibration. In Proceedings of the IMCS2014—The 15th International Meeting on Chemical Sensors (Poster Presentation), Buenos Aires, Argentina, 16–19 March 2014.
39. Schleyer, E.B.R.; Wallasch, M. *Das Luftmessnetz des Umweltbundesamtes*; Umweltbundesamt: Dessau-Roßlau, Germany, 2013.
40. WHO. WHO Regional Office for Europe Centers of Disease Control, WHO Guidelines for Indoor Air Quality: Selected Pollutants; World Health Organization: Copenhagen, Denmark, 2010; Volume 9, ISBN 978-92-890-0213-4.
41. Schultealbert, C.; Amann, J.; Baur, T.; Schütze, A. Measuring Hydrogen in Indoor Air with a Selective Metal Oxide Semiconductor Sensor. *Atmosphere* **2021**, *12*, 366. [[CrossRef](#)]
42. Schütze, A.; Gramm, A.; Ruhl, T. Identification of Organic Solvents by a Virtual Multisensor System With Hierarchical Classification. *IEEE Sens. J.* **2004**, *4*, 857–863. [[CrossRef](#)]

3.3 Paper 2 – Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning

Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, C. Bur

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

Atmosphere 2022, 13(10), 1614;

The original paper can be found in the online version at <https://www.mdpi.com/1864658> or DOI: <https://doi.org/10.3390/atmos13101614>

© 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). Reprinted, with permission, from Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, C. Bur; Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning; Atmosphere 2022.

3.3.1 Synopsis

After introducing the TCOCNN and demonstrating that the custom neural network can accurately quantify target VOCs in laboratory conditions as well as field tests, the next step is to tackle one of the essential issues with MOS gas sensor calibration. This paper shows that it is possible to significantly reduce the calibration time with the help of calibration transfer based on transfer learning from DL. A new dataset was required to analyze the capabilities of transfer learning for calibration transfer in an IAQ scenario. Therefore, a new dataset containing multiple UGMs from various sensors in a more complex indoor air scenario with more gases was recorded. The dataset was again recorded with the help of a custom GMA and consisted of 906 UGMs constructed with the help of Latin hypercube sampling. The corresponding ranges for the seven VOCs (acetic acid, acetone, ethanol, ethyl acetate, formaldehyde, toluene, and xylene), the two background gases (carbon monoxide, hydrogen), and the relative humidity can be found in Table 3.4. The 906 UGMs were split into three segments to allow for a better approximation of lower gas concentrations. Each segment used different concentration ranges for all gases and the relative humidity to build the dataset. Three SGP40 sensors (sensor A - C) were used to record the dataset across all 906 UGMs.²⁶ Sensors A and B were from the same batch, while sensor C was from a new batch. The operation mode for all sensors was once again TCO. Compared to the previous paper, the cycle had a duration of 144 s with twelve high and low-temperature phases. The high temperature was again set to 400 °C, and the low temperatures were increased from 100 to 375 °C in 25 °C steps; however, every temperature step was recorded this time. Sub-sensor 3 was different, as it was only modulated between 200 °C and 300 °C.

After introducing the dataset, explaining transfer learning for calibration transfer is essential. As presented in the Theoretical Background section, transfer learning is a technique to reuse a working model on a new but similar dataset. An example of transfer learning from computer vision would be to reuse an object detection network to detect new objects in pictures with the help of only a few additional training images. This publication reused a model built with one sensor for a second sensor by retraining the model with as few transfer samples/UGMs as possible. Compared to regular training, transfer learning takes the weights of the previously trained model as a starting point for retraining. Furthermore, a lower learning rate is applied (fine-tuning), or some layer weights are frozen. This should guarantee optimal adaption to the new sensor

²⁶It took 24 minutes to record one UGM.

Table 3.4: Concentration ranges for all target gases and the number of unique gas mixtures (UGMs) for each range within the dataset. Relative Humidity (RH) was varied between 25 % and 80 %. Reprinted with permission of Ref. Paper 2. Y. Robin, 2023.

	Segment 1 (1 - 200)	Segment 2 (201 - 500)	Segment 3 (501 - 906)
Carbon monoxide	100 - 2000 ppb	100 - 2000 ppb	100 - 2000 ppb
Hydrogen	400 - 2000 ppb	400 - 2000 ppb	400 - 2000 ppb
Acetic acid	1 - 50 ppb	1 - 150 ppb	1 - 500 ppb
Acetone	3 - 50 ppb	3 - 150 ppb	3 - 500 ppb
Ethanol	1 - 50 ppb	1 - 150 ppb	1 - 500 ppb
Ethyl acetate	1 - 50 ppb	1 - 150 ppb	1 - 500 ppb
Formaldehyde	1 - 50 ppb	1 - 150 ppb	1 - 300 ppb
Toluene	1 - 75 ppb	1 - 75 ppb	1 - 250 ppb
Xylene	2 - 150 ppb	2 - 150 ppb	2 - 500 ppb

without overfitting to the much smaller transfer dataset. After the general concepts were discussed within the paper, the first step was to split the data into training, validation, and testing. Similarly to Paper 1, the data was divided based on the UGMs in 70/10/20 fashion to test the TCOCNN. This split was the same across sensors and evaluations to guarantee fair comparison. After the data split, the TCOCNN was optimized for the independent gases with the help of Bayesian optimization, using sensor A’s training and validation data. By using the optimized hyperparameters, individual calibration models were built for every gas and sensor. This was done to test the general performance of the TCOCNN on the new dataset. It was shown that the TCOCNN achieved reasonable results for all sensors with the hyperparameters found with sensor A. However, sensor C showed slightly worse performance, which can be attributed to sensor C being from a new batch.

The next step was to analyze which fine-tuning method works best for this use case. Therefore, an initial model was built with sensor A, and the transfer samples were randomly selected from the pool of training samples from sensors B and C. For the fine-tuning itself, three different learning rates were tested together with freezing the convolutional part of the neural network. It was demonstrated that selecting the learning rate reached halfway during the regular learning processes ($LR * 0.9^{15}$)²⁷ is optimal for small transfer sets (cf. Figure 3.4). For most gases, 50 UGMs were sufficient to reach

²⁷LR represents the initial learning rate which depends on the different target gases.

acceptable performance compared to training a model from scratch with all 700 training UGMs (double the RMSE). This suggests that it is possible to reduce the calibration time by up to 93 % and still find a suitable model. However, this depends on the target gas and the sensor used for transfer. For example, it was demonstrated that slightly worse performance was achieved when sensor C from the new batch was used. This indicates that the model built with one sensor for initial model building cannot be generalized across batches.

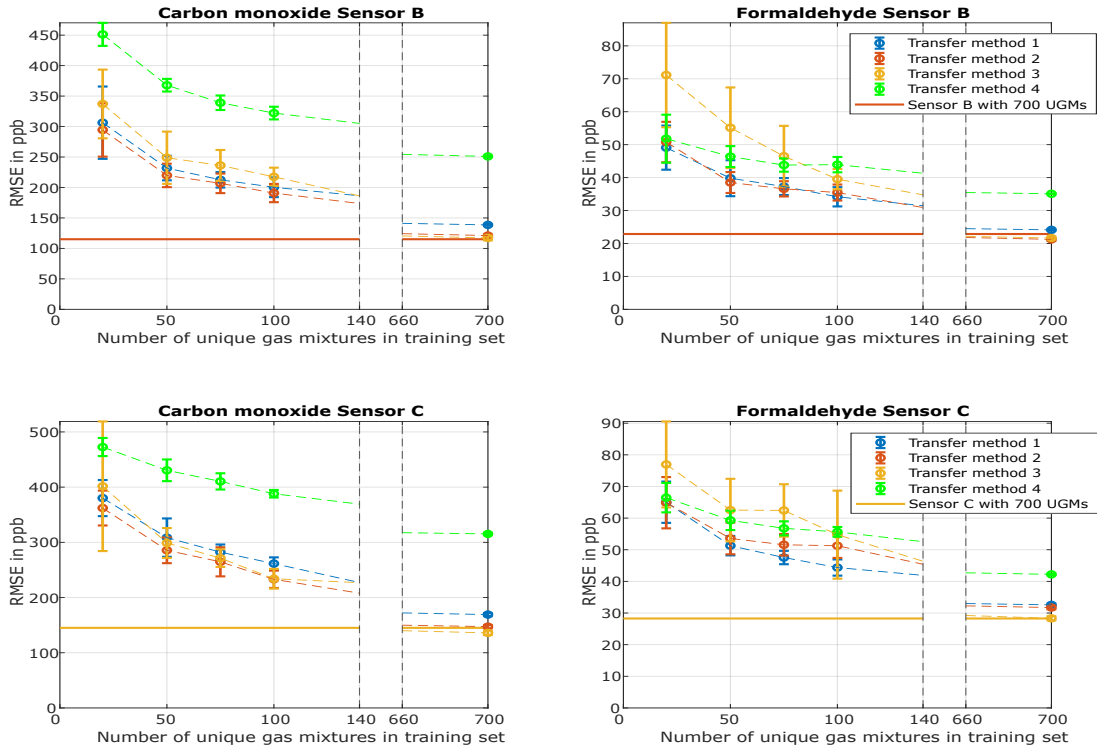


Figure 3.4: Comparison of different transfer methods for carbon monoxide and formaldehyde. Transfer method 1: learning rate set to the value reached at the end of original training. Transfer method 2: learning rate set to the value reached halfway through original training. Transfer method 3: learning rate set to the original value. Transfer method 4: implicit feature extraction fixed, only fully connected layer can be trained. Reprinted with permission of Ref. Paper 2. Y. Robin, 2023.

For further validation, the results achieved with the learning rate set to $LR * 0.9^{15}$ were compared with building the models from scratch. There, it was demonstrated that transfer learning leads to considerably better results than the model built from scratch. This is especially true for small transfer sets where much smaller RMSE values were achieved across all gases. Apart from the learning rate, DL-based transfer learning has many

more parameters that can lead to even better calibration transfer [Paper 2]. Therefore, the impact of the selected UGMs on the quality of the transferred model was analyzed. This was done with the help of performing multiple evaluations with a static transfer set (same UGMs) and a dynamic transfer set (different UGMs from the training set). A much higher variance was observed for the dynamic set, which showed it is possible to achieve far better results with "good" 20 UGMs compared to 50 "bad" UGMs. While not tested in this work, the Kennard-Stone algorithm [25] might be a suitable solution to find the optimal subset of UGMs. During the evaluation, it was discovered that the results could be drastically improved with the help of normalizing the input by subtracting the mean and dividing the result by the standard deviation for every observation and each sub-sensor independently. In these experiments, the evaluations were repeated with the normalization, and the results showed an overall improved performance while also confirming the previous results. The last evaluation compared the results to global modeling based on the FESR approach. The comparison showed that the performance of global FESR and transferred TCOCNN are very similar for small datasets. However, for slightly larger datasets, it could be seen that the TCOCNN outperforms global model building. This can be attributed to the specifically tailored ML model for the new sensor.

In conclusion, it can be stated that DL-based transfer learning for calibration transfer can significantly reduce the needed calibration time for IAQ applications (93 %). Although many parameters are tunable for transfer learning, encouraging results are already achieved with straightforward methods. For future work, it is necessary to compare this newly developed method with state-of-the-art approaches (signal standardization) to estimate the efficiency. Furthermore, optimizing this approach for industrial use is essential, where only 1 - 5 gas tests are used for sensor calibration (no GMA needed). Global model building for initial model building is mainly analyzed in the following paper as this is a promising candidate for finding more general initial models.

The main takeaways of this publication are:

- The same hyperparameter can be used across different sensors.
- Normalization improves the prediction quality significantly.
- Transfer learning can be used for calibration transfer.
- Different transfer strategies perform well for different tasks.

- Best performance for a small dataset with $LR * 0.9^{15}$.
- Best performance for a large dataset with $LR * 0.9^0$.
- Transfer learning can be used to reduce the calibration time significantly.
 - Reduction by up to 93 %.
 - Performance depends on the target gas.
- The selection of transfer UGMs is highly relevant.
- Transfer learning can generate tailored models that perform better than global ones.
- Transfer learning comes with additional hyperparameters.
 - Transfer strategies (fine-tuning, freezing).
 - Selected transfer samples.

Open questions/tasks are:

- Can transfer learning outperform state-of-the-art calibration transfer methods?
- What is the effect on the performance of calibration transfer if multiple sensors are used for initial model building?

Article

Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning

Yannick Robin ^{*}, Johannes Amann , Payman Goodarzi , Tizian Schneider , Andreas Schütze and Christian Bur 

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

* Correspondence: y.robin@lmt.uni-saarland.de

Abstract: In this study, methods from the field of deep learning are used to calibrate a metal oxide semiconductor (MOS) gas sensor in a complex environment in order to be able to predict a specific gas concentration. Specifically, we want to tackle the problem of long calibration times and the problem of transferring calibrations between sensors, which is a severe challenge for the widespread use of MOS gas sensor systems. Therefore, this contribution aims to significantly diminish those problems by applying transfer learning from the field of deep learning. Within the field of deep learning, transfer learning has become more and more popular. Nowadays, building a model (calibrating a sensor) based on pre-trained models instead of training from scratch is a standard routine. This allows the model to train with inherent information and reach a suitable solution much faster or more accurately. For predicting the gas concentration with a MOS gas sensor operated dynamically using temperature cycling, the calibration time can be significantly reduced for all nine target gases at the ppb level (seven volatile organic compounds plus carbon monoxide and hydrogen). It was possible to reduce the calibration time by up to 93% and still obtain root-mean-squared error (RMSE) values only double the best achieved RMSEs. In order to obtain the best possible transferability, different transfer methods and the influence of different transfer data sets for training were investigated. Finally, transfer learning based on neural networks is compared to a global calibration model based on feature extraction, selection, and regression to place the results in the context of already existing work.

Keywords: air quality; MOS gas sensors; deep learning; calibration time reduction; transfer learning



Citation: Robin, Y.; Amann, J.; Goodarzi, P.; Schneider T.; Schütze A.; Bur, C. Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning. *Atmosphere* **2022**, *13*, 1614. <https://doi.org/10.3390/atmos13101614>

Academic Editor: László Bencs

Received: 25 August 2022

Accepted: 28 September 2022

Published: 2 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Global Burden of Diseases, Injuries, and Risk Factors Study (GBD) 2019 has shown that air pollution caused the premature death of around 6.7 million people in 2019 [1]. In order to lower the number of premature deaths, it is essential to analyze outdoor and indoor air regarding their toxic or harmful components. Since air generally consists of many different components, it is almost impossible to determine every substance within. Therefore, this contribution focuses on quantifying volatile organic compounds (VOCs), which are of great importance because there is a large variety of substances, some safe, like ethanol, and others highly toxic even at low concentrations like formaldehyde [2]. Today, only analytical measurement systems can quantify specific components (e.g., VOCs) with reasonable accuracy at the ppb level. However, these measurement systems are costly, require expert knowledge to operate, and are often not capable of real-time measurements. This prevents the widespread use of analytical measurements for reducing the risk associated with exposure to dangerous VOCs, especially in indoor air. One promising solution for a low-cost and easy-to-operate measuring system is provided by metal oxide semiconductor (MOS) gas sensors. Previous studies showed that such systems, together with complex operating modes and deep learning, can quantify single VOCs in complex environments at the ppb level [3]. The system, in this case, consists of one or multiple MOS gas sensors operated dynamically to obtain complex signal patterns. In particular,

temperature cycled operation (TCO) has been demonstrated to greatly increase sensitivity, selectivity, and stability [4]. With the help of machine learning, a data-driven calibration model is built that analyses the complex sensor response patterns and can thereby predict gas concentrations of individual gases even in complex mixtures. However, due to even minute production tolerances of the gas sensing layer or the μ -hotplate for setting the temperature, it is necessary to calibrate every system individually, which requires several days to reach sufficient accuracy for complex gas mixtures. This greatly increases the cost and limits the widespread use of these advanced gas sensor systems. There are already transfer methods to significantly reduce the calibration time, such as direct standardization, orthogonal signal correction, global modeling, and many more, as shown in [5–11]. These methods calibrate a global model based on a few master sensors and use this for all additional sensors. In some cases, they also map the signals of additional sensors to the signals of the master sensors with the help of a few transfer samples. Thereby, the sensor signal of the new sensors is altered slightly to match the signal pattern of the master sensors. This allows for using the same global data-driven model with data from many different sensors. It was shown that those methods, together with a few transfer samples, can, for example, account for temperature differences caused by the hotplate or compensate for sensor drift. Nevertheless, those methods require that the sensor to be calibrated is operated under the exact same conditions, i.e., gas compositions and concentration ranges, during calibration as the master model. Thus, it is not possible to transfer models between different use cases. One promising solution for this problem might be transfer learning from the field of deep learning [12–14]. This method is often used for image classification tasks [15]. In this case, classification models are not built from scratch every time; instead, already existing models are adjusted to classify different objects in a picture.

This study adopts and applies transfer learning to data of a commercially available MOS gas sensor (SGP40, Sensirion AG, Stäfa, Switzerland). A deep convolutional neural network is trained on one master MOS gas sensor to build an initial model, which is then used as the starting point for other sensors to be calibrated. The gases analysed are the seven target VOCs acetic acid, acetone, ethanol, ethyl acetate, formaldehyde, toluene, and xylene, and the two background gases carbon monoxide and hydrogen, as well as the relative humidity. It was previously shown by [16,17] that it is possible to use such methods to reduce the calibration time of sensors significantly. To the best of our knowledge, not many scientific publications address deep transfer learning for gas sensor calibration; therefore, more research is required. Compared to the previous studies, we have tackled more complex situations, i.e., more than ten independent gas components with a focus on VOCs at low ppb concentrations are analyzed as a regression problem. The influence of different hyperparameters on transferability are investigated. For example, the influence of different transfer learning methods and the gases used for the transfer between sensors are analyzed. Furthermore, the results are compared with already existing work in the form of a conventional global calibration model based on feature extraction, selection, and regression.

2. Materials and Methods

2.1. Dataset

The dataset used throughout this study to evaluate the benefits of transfer learning in the field of deep learning is generated with our recently developed novel gas mixing apparatus (GMA), which allows up to 16 independent gases to be arbitrarily mixed over a wide concentration range. Details about this GMA are described in [18] and further details can be found in [19,20]. The GMA is used to provide well-known complex gas mixtures to three commercially available MOS sensors, i.e., SPG40 (Sensirion AG, Stäfa, Switzerland), with two sensors from one batch and the third from a different one. Every SGP40 holds four different gas-sensitive layers and is operated using temperature cycled operation (TCO) as shown in Figure 1. For this study, the temperature of sub-sensors 0–2 from all SGP40 is switched between high-temperature phases at 400 °C for 5 s, followed by

low-temperature phases varying in 25-degree steps from 100 °C to 375 °C and lasting 7 s each. The temperature of sub-sensor 3 is only changed between 300 °C and 250 °C due to the lower temperature stability of this gas-sensitive layer. Each temperature cycle lasted 144 s with the logarithmic sensor resistance sampled at 10 Hz. Temperature cycling of the independent hotplates and read-out of the four gas-sensitive layers is achieved with the integrated electronic using custom software based on a protocol provided by Sensirion under a non-disclosure agreement. Further details are described in [19]. The data used for building the data-driven models always consists of one full temperature cycle of all gas-sensitive layers with the raw signal modified based on the Sauerwald-Baur model [21,22], i.e., a total of 5760 data points. For future reference, one complete temperature cycle is treated as an observation.

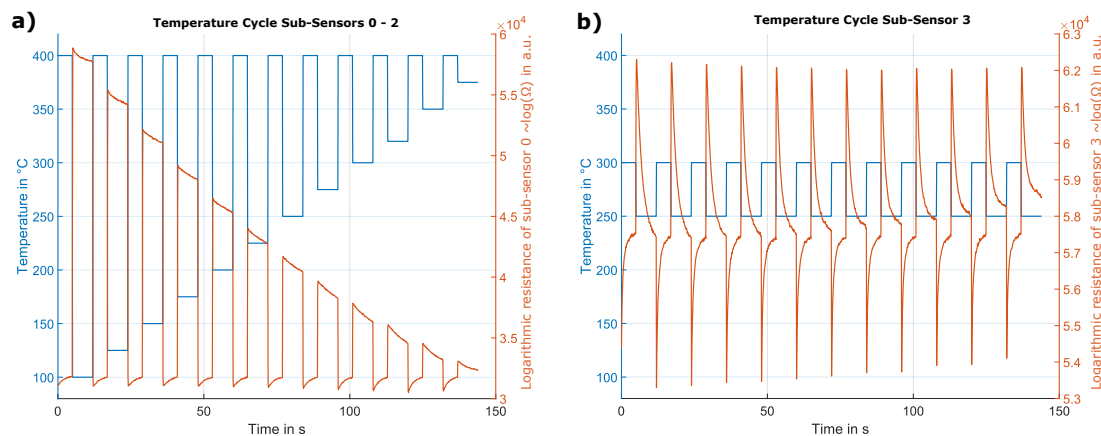


Figure 1. Temperature cycle together with an example sensor response sampled at 10 Hz. High-temperature phase with a duration of 5 s and a low-temperature phase of 7 s with an overall duration of 144 s. (a) temperature cycle for sub-sensor 0–2 and a typical sensor response of sub-sensor 0; (b) temperature cycle for sub-sensor 3 and a typical sensor response.

In order to calibrate the sensor for a complex environment, the gas mixtures applied by the GMA represent realistic gas mixtures as found in indoor environments consisting of eight different VOCs plus two inorganic background gases, CO and hydrogen. Furthermore, the relative humidity @20 °C is varied between 25–80%, resulting in a total of 11 independent variables. The eight VOCs chosen for calibration are acetic acid, acetone, ethanol, ethyl acetate, formaldehyde, isopropanol, toluene, and xylene, representing various VOC classes. They contain benign and hazardous VOCs (cf. Figure 2). During the experiment, multiple unique gas mixtures (UGMs) were randomly defined based on predefined concentration distributions. Latin hypercube sampling [23,24] was applied to minimize correlations between the different gases. All gas sensors were simultaneously exposed to these UGMs for several minutes to record multiple temperature cycles (TCs) or observations for each mixture. For Latin hypercube sampling, the experiment is divided into three different sections. For each of these segments, the concentration of every gas is randomly picked from a uniform distribution, cf. Table 1. The specific gases and concentration ranges used for this study are based on [19,24,25].

In total, 906 UGMs were set, exposing all three SGP40 sensors simultaneously for approx. 10 TCs (1440 s), yielding an overall calibration duration of more than 15 days. Because of synchronization problems between GMA and the sensors' data acquisition, only three to four TCs or observations per UGM are used for further evaluation. The target for the data-driven models is an accurate prediction of the gas concentration for each gas, VOC, or inorganic background gas individually. As an additional target, the sum concentration

of all VOCs within the mixtures is also predicted as this TVOC_{sens} value might be a suitable indicator for indoor air quality. Because of minimal sensor response to isopropanol, it is excluded as an independent target and contributor to the TVOC_{sens} target.

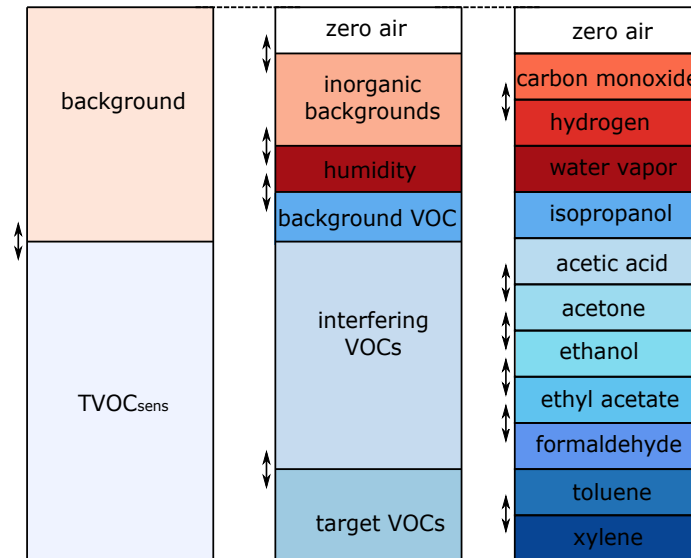


Figure 2. Overview of all substances within gas mixtures (adapted from [3,24]).

Table 1. Concentration ranges for all target gases together with the number of unique gas mixtures (UGMs) for each range within the dataset.

	Segment 1 (1–200)	Segment 2 (201–500)	Segment 3 (501–906)
Carbon monoxide	100–2000 ppb	100–2000 ppb	100–2000 ppb
Hydrogen	400–2000 ppb	400–2000 ppb	400–2000 ppb
Acetic acid	1–50 ppb	1–150 ppb	1–500 ppb
Acetone	3–50 ppb	3–150 ppb	3–500 ppb
Ethanol	1–50 ppb	1–150 ppb	1–500 ppb
Ethyl acetate	1–50 ppb	1–150 ppb	1–500 ppb
Formaldehyde	1–50 ppb	1–150 ppb	1–300 ppb
Toluene	1–75 ppb	1–75 ppb	1–250 ppb
Xylene	2–150 ppb	2–150 ppb	2–500 ppb

This dataset consists of the data of three SGP40 sensors. Throughout this study, sensor A will be used to build the initial models from which the calibration models for sensor B (same batch) and sensor C (different batch) are derived. Sensors from different batches have been chosen to investigate the effect of transferring a data-driven model across batches, as one would expect larger differences in sensor parameters. Sensors A and B are from the same batch, and sensor C is from a different one.

2.2. Model Building and Methods

A ten-layer deep convolutional neural network (TCOCNN [3]) is used to derive the data-driven model (cf. Figure 3). The input for this neural network is one observation represented in a 4x1440 array. The first dimension describes the different gas-sensitive

layers per sensor, and the second dimension covers the time domain (144 s sampled at 10 Hz). In previous studies, it is shown that this TCOCNN achieves similar or slightly better results than classical approaches based on feature extraction, selection, and regression (FESR) [3] and can be used for transfer learning [16]. Other contributions also showed that neural networks can be used to calibrate gas sensors and achieve similar results to established methods [26]. The focus of this contribution will be on the TCOCNN from [3] as a starting point for transfer learning. This will also include the normalization of the input matrix. Currently, this is done by calculating the mean and standard deviation of the complete 4×1440 matrix of one observation and then normalizing this frame. This standard approach for pictures is not the best for sensor-based data, where normalization would ideally be done for each sub-sensor.

The hyperparameters of the neural network are optimized independently for each analyzed gas, similar to the optimization done in [3]. Hyperparameter tuning of the TCOCNN is performed with Bayesian optimization and neural architecture search (NAS) [27,28]. The optimized parameters are the initial learning rate of the optimizer, the number of filters, and kernel size, and striding size of the convolutional layers, the dropout rate, and the number of neurons of the fully connected layer. The optimization for all sensors is based on sensor A and is performed on 500 randomly chosen UGMs (450 for training and 50 for validation). The optimization goal was to minimize the root-mean-squared error (RMSE) on the validation data as described in [3]. In our typical workflow, the final model would be trained for each individual sensor from scratch after hyperparameter tuning with 700 UGMs and tested on the remaining 200 UGMs.

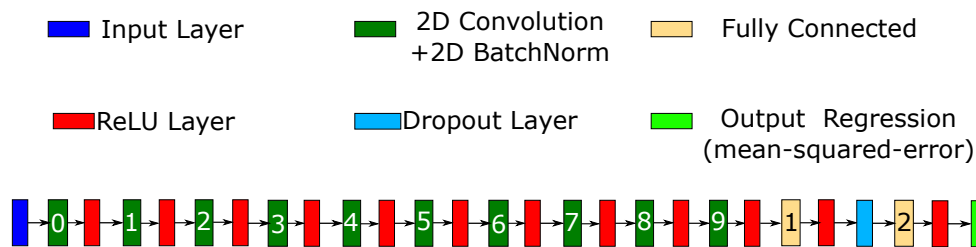


Figure 3. Architecture of the TCOCNN neural network (adapted from [29]).

2.2.1. Transfer Learning

In order to improve the training process of the TCOCNN, primarily by reducing the number of UGMs required for calibration, transfer learning is now introduced [12–15]. Transfer learning in the scope of deep learning is generally used to adapt the information of an already existing model to a new task or a new domain. For neural networks, this information is stored within the weights and biases, as they are used to transform the input to form the output. Therefore, they contain information on the dependencies between input and output. Thus, those weights and biases can be used for a similar task and must only be adjusted with additional training. The exact definition of transfer learning and how it is used with and without deep learning can be seen in multiple surveys [30–32]. In our case, we want to maintain the information within the weights and biases about interpreting different signal patterns for specific gas concentrations and transfer this information to other sensors [16]. This is done by not initializing the model with random weights and biases. Instead, the weights and biases from a previously calibrated network (different sensor or sensors) are used as a starting point (initial/global model). This ensures that the neural network starts not without any information about the task at hand. This means that, before the training/calibration starts for the new sensor, the neural network contains information on how to transform the input. Nevertheless, a further adjustment of the weights and biases in the form of additional training is necessary because of production differences between every sensor (new domain).

The benefits of transfer learning can be seen in Figure 4. This scheme was initially proposed for classification by [12] and is here adapted to a regression problem for gas sensor calibration. It shows that, with proper transfer learning together with suitable data, it is possible to improve the starting accuracy, increase the learning slope, and perhaps even achieve better results than the initial model. However, it is also illustrated that the amount of data, here the number of UGMs, and the transfer learning method are essential to achieve the desired improvement. An improvement means either reaching an acceptable accuracy, here a required measurement uncertainty expressed as RMSE, with fewer data, or achieving higher accuracy with a similar amount of data. This study will analyze both possible improvements for our use case.

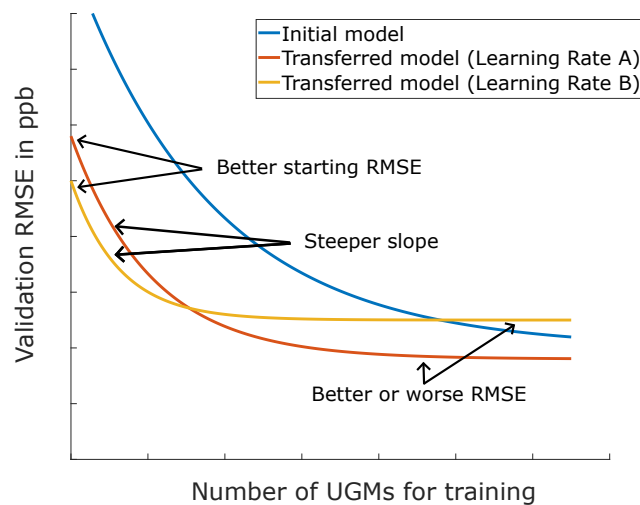


Figure 4. Expected effect of transfer learning over the number of training samples and transfer methods.

In order to transfer the gas sensor calibration from one sensor (sensor A) to one of the others (sensor B or C), multiple methods are available, which were used for other data-driven problems, like computer vision or speech recognition [15,33]. Throughout this contribution, four different methods for transfer learning will be tested. Three of them belong to the field of fine-tuning [34], where not only certain parts of the model are retrained, but the whole model is adjusted to the new domain (e.g., sensor), which means that every weight and bias is readjusted. Those three fine-tuning methods vary in the rate the network parameters are allowed to change, which the initial learning rate can control. The lower the initial learning rate, the less the parameter within the network will change. Method one sets the initial learning rate to the learning rate the TCOCNN has reached at the end of the training of the initial model, which is low because, during TCOCNN training, the learning rate is decreased after every two epochs by 0.9. The second method sets the initial learning rate to the value reached halfway through the initial training process, while the third set the learning rate to the value at the beginning of the initial training. On the other hand, the fourth method keeps all parameters of the convolutional layers fixed and only adapts the parameters in the last two fully connected layers to the new sensors [34]. Thus, this method basically keeps the implicit feature extraction [35] as derived for the model of sensor A and only retrains the regression part of the network. This allows for gaining insights if an adaptation of the implicit feature extraction is necessary for new sensors.

2.2.2. Global Model

After introducing transfer learning, it is also important to compare this method with already existing approaches like global model-building [7]. Here, a global model-building approach means one model trained once on all available data samples, i.e., covering different gas sensors. Based on this definition, the transfer learning approach based on neural networks, as introduced above, is not a global calibration method, as individual models for different sensors are obtained. For practical application, a comparison between transfer learning and global modeling is important as a valid global model would completely eliminate the necessity for individual gas sensor calibration. In this case, we chose not to compare this with neural networks but instead used established methods based on feature extraction, selection, and regression (FESR). Usually, we use this approach only on single sensors, as shown in [19], but this approach is extended to multiple sensors in this study. As for the neural network, the data is split into 700 UGMs for model building and 200 UGMs for testing data. For feature extraction, adaptive linear approximation (ALA) is used. This extraction method divides the raw signal into sections and calculates the mean and slope of these sections as features. The segmentation is optimized based on the reconstruction error on the full training set that is achieved with the calculated mean and slope values. Details are given in [36,37]. The feature selection is performed with 10-fold cross-validation (based on all observations) of the training data. In this case, the training set is split randomly into ten parts, and the RMSE is calculated for the different training sets (different numbers of features) across all folds. First, the features are sorted based on their Pearson correlation to the target gas to reduce the number of features to a manageable level, here 200. Then, the feature set is gradually increased from 1–200 features. The resulting sets are tested with the help of a partial least squares regression (PLSR) with a maximum of 100 components [38,39], and the set with the smallest RMSE across all ten folds is used for later evaluations (e.g., [37]). The final regression is then built with the help of a PLSR with a maximum of 100 components and the data available from multiple sensors.

2.3. Data Evaluation

In order to evaluate the capabilities of transfer learning, this section summarises the strategies followed to test transfer learning. Before any evaluations are performed, the data are split into training (700 UGMs) and testing (200 UGMs) data. The 200 test UGMs are the same throughout all evaluations and also for different sensors to make the comparison fair.

As stated in the modeling section, the TCOCNN is first optimized for the different gases. In addition, 500 random UGMs are picked from the 700 available training UGMs from sensor A. The Bayesian optimization is then performed with 450 UGMs for training and 50 for validation. The model parameters that achieve the lowest validation RMSE are then chosen for the rest of this contribution for the specific gas. A detailed explanation of this process can be found in [3], albeit for a different sensor model.

In the next step, the RMSE that these models can achieve is determined, i.e., the RMSE when training the model with the selected hyperparameters with all 700 training UGMs and testing on the 200 independent test UGMs. This training process is repeated ten times for every gas and for all sensors to determine the stability of the achieved models. The mean RMSE values of the training are used as reference RMSE values for the subsequent transfer learning.

In the first part, different methods for transfer learning are studied and compared. Thus, an initial model is built based on the 700 training UGMs for sensor A. Since this model building was already performed in the previous step for calculating the baseline ten times, the model picked for transfer is the last model built during the baseline calculation. With this model and a subset of the UGMs (“transfer samples”), the performance of the four different transfer learning methods is tested to analyze which method(s) work best for specific gases. The main question is how many transfer samples are required to reach an acceptable accuracy. This is analyzed by varying the number of transfer samples from 20 to

700 UGMs, i.e., up to the full data set of the original training. Furthermore, the methods are tested for sensors B and C to analyze their performance for sensors from the same and from a different batch. The metric to compare the different methods and the number of training samples is the RMSE reached on the test data set, i.e., the 200 testing UGMs, which are always the same regardless of which scenario is tested. All pieces of training are performed multiple times to gain insight into the stability of the tested approach. The paper presents the results for only a few relevant gases, and the remaining results are shown in Appendix A.

The second part then compares the best model achieved by transfer learning and a model built from scratch. Therefore, the accuracy in terms of RMSE of the models with transfer learning and no transfer learning are compared across all sensors. This gives an overview of the benefits transfer learning can provide. As for the first test scenario, the number of samples used for training varies from 20 to 700. Furthermore, the influence of the specific samples used for training the models is compared by randomly picking 20–500 training samples from the 700 available samples and training each specific case 10 times to estimate the variation caused by different training sets (random subsampling). In order to compare this variation with the variation during normal training with a fixed set, a static training set is also trained ten times. This allows for estimating how much variation is caused by the neural network itself and how much by the random subsampling.

In order to confirm the achieved results from the previous sections, the experiments are repeated with an optimized standardization of the raw data and the different transfer strategies. For this experiment, the raw signal of each of the four sub-sensors is normalized independently for each temperature cycle as the raw signals of the sub-sensors differ significantly because of the gas-sensitive materials and the different temperature ranges. This experiment shows that the general method is always applicable when used with a different normalization for the neural network.

The last part of the results compares the transfer learning approach with a simplified global calibration approach based on established methods like feature extraction, selection, and regression. Similar to the neural network approach, the data of all sensors are based on the same split of the data set into training (700 UGMs) and test data (200 UGMs) as for the evaluation above. As mentioned above, the feature extraction is based on the adaptive linear approximation. The feature selection is performed by finding the optimal subset of Pearson sorted features based on the validation set (10-fold cross-validation). In order to test this method as a global calibration scheme, all training UGMs of one sensor, together with additional samples from another sensor, are used to build the model. Here, the same evaluations as for the neural networks are performed: Models are trained with 700 UGMs from sensor A together with 20–700 additional samples from sensor B and tested on the 200 test UGMs from sensor B. Finally, the evaluation with 20–500 randomly selected UGMs is repeated ten times to evaluate the influence of different training sets also for the global approach.

The previously described evaluation steps are visualized in Figure 5, to make the process easier to understand.

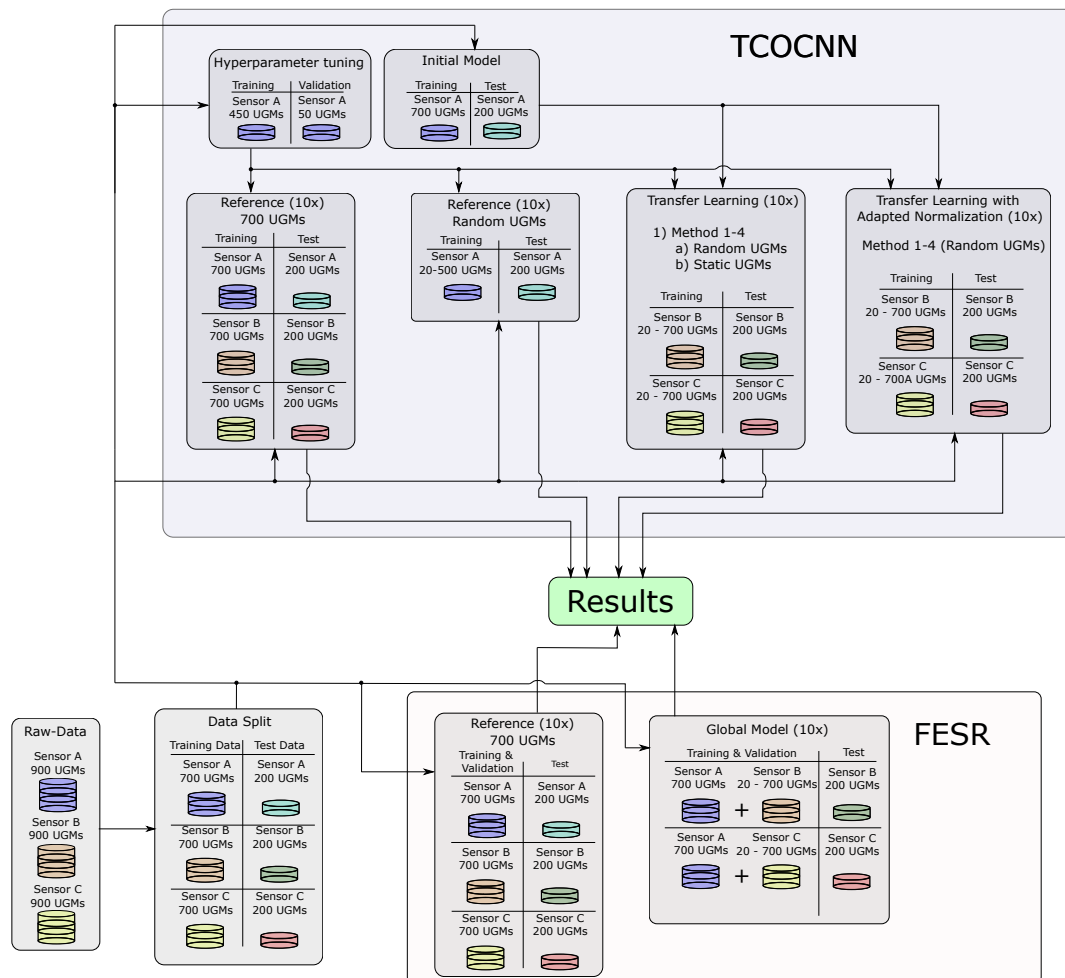


Figure 5. Schematic of all performed evaluations to gather the necessary results.

3. Results

Before the transfer methods can be compared, the reference values must be obtained to rate the transfer methods (cf. Figure 6). The mean RMSE values for sensors A and B are very similar for the different gases. The largest difference regarding the achieved RMSE divided by the mean concentration of the specific gas is observed for ethanol, where the relative RMSE for sensor A is 3.2% smaller than sensor B. Thus, the hyperparameter tuning performed to find an optimized architecture for a specific gas based on sensor A can also be used as architecture for other sensors from the same batch. For some gases, sensor B outperforms sensor A, although the network was not optimized for this specific sensor. On the other hand, the performance of sensor C (different batch), while comparable for some gases with RMSE values close to the sensors A and B, is significantly reduced for acetone and hydrogen compared to the other sensors. This effect cannot be attributed to sensor drift as sensor C was used over a similar period as the other two. The most probable explanation for the observed difference could therefore be a slight variation of one or more gas-sensitive layers between the batches.

As already used above, another important measure to rate the quality of the TCOCNN is the RMSE divided by the mean concentration. Here, the best result is achieved for hydrogen with a deviation of only 4%, the average deviation for the other gases is between 10 and 30%, which is still a reasonable result that can be useful for IAQ applications. One exception is toluene. For this gas, the deviation is 45%, which is still useful but not as promising as the other gases. Nevertheless, the RMSE of this gas can be significantly reduced with the adapted normalization (21.7 ppb), and a reasonable RMSE divided by the mean concentration of 28.6% can be achieved. Thereby, it can be stated that the TCOCNN can be a useful tool for calibrating gas sensors.

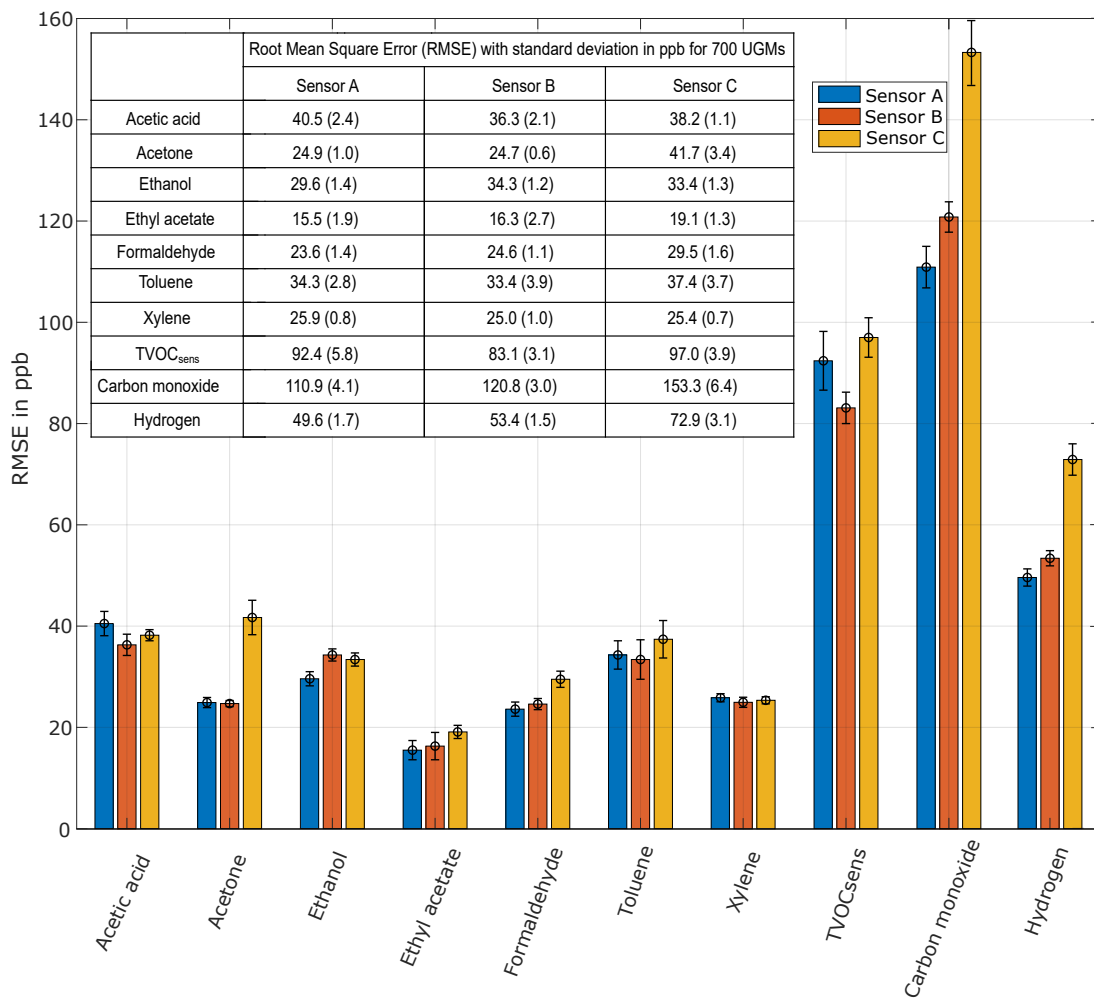


Figure 6. Mean root-mean-squared error (RMSE) and standard deviation for all gases trained with 700 UGMs for later reference.

Figure 7 shows the results of the first test scenario where different transfer learning methods are compared. The results are shown exemplarily for carbon monoxide and formaldehyde, two gases that are of great importance for air quality assessments. Carbon monoxide was chosen because of its toxic properties and also because it is a ubiquitous background or interfering gas when monitoring VOCs. Formaldehyde was chosen because

of its carcinogenic properties even at low concentrations [2]. Within Figure 7, the dashed lines indicate the accuracy achieved with different transfer methods over the number of training UGMs. The solid lines indicate the best RMSE achieved without transfer learning using 700 UGMs over ten trials. For sensor B and carbon monoxide, transfer method 2 (learning rate between min and max) performs best across all different transfer sets. Method 1 (initial learning rate at minimum) performs similarly well for small transfer sets, while method 3 (initial learning rate at maximum) performs slightly better for very large transfer sets. Similar behavior can be observed for formaldehyde, where methods 1 and 2 are very similar for small transfer sets, while methods 2 and 3 yield almost identical results for larger transfer sets. For carbon monoxide and formaldehyde, it can be seen that transfer method 4 (static feature extraction) performs worst, at least for larger transfer sets. That allows the assumption that, although the different sensors achieve similar performance for the full dataset when trained from scratch (Table 1), they rely on slightly different features. Therefore, the implicit feature extraction, as well as the regression within the fully connected layers, needs to be transferred between models. Method 4 achieves competitive results only for formaldehyde and very small transfer sets with 20 UGMs. Furthermore, it can be observed that method 3, especially for small transfer sets (e.g., 20 UGMs), shows a large variation of the RMSE values. Thus, the method lacks robustness. This behavior is not observed for the other methods, which allows the interpretation that learned dependencies could be maintained more easily when reducing the initial learning rate for the parameters of the TCOCNN. This behavior for method 3 is counterproductive to the general goal of this contribution. Since we want to reduce the calibration time, achieving good models for small transfer sets is the highest priority. Thus, method 2 seems to be the most promising transfer method for further evaluation. Nevertheless, one should note that method 3 outperforms all other methods for all gases when used with the full 700 UGM transfer sets. For this scenario, method three, on average, achieves results even better than the absolute best RMSE obtained for the different gases without transfer learning. This shows that pre-trained networks can improve the accuracy of different sensors even when not reducing the calibration time, apparently by using the additional information contained in measurements of other sensors. Here, results of only the most relevant gases are presented. Results of the remaining gases are provided in Appendix A.

To compare the benefits of transfer learning with results when models are trained from scratch, Figure 8 shows the comparison of transfer method 2 for sensors B and C together with a model trained from scratch on sensor A, again for the two relevant target gases carbon monoxide and formaldehyde. The solid lines within the figures indicate the best possible RMSE achieved with 700 UGMs over ten tries and are used as a reference for all further evaluation. The dashed blue curve illustrates the achieved accuracy for sensor A, i.e., when no transfer learning is applied, while the dashed orange and yellow lines show the results achieved with transfer learning. In order to capture the variance caused by the different UGMs used in the training set, the standard deviation is marked for all sensors and training sets. The blue line, i.e., sensor A without prior information, always starts at much higher RMSE values than the yellow and orange lines for sensors B and C with transfer learning and also shows a larger standard deviation. This implies that the quality of the results is significantly improved when applying transfer learning, as the models reproducibly achieve better accuracy. Furthermore, for both gases, the benefit of transfer learning is most significant when applied to sensors from the same batch. This is demonstrated by the orange line (sensor B), which is significantly lower than the yellow line (sensor C) and exhibits a smaller variance similar to results reported previously [16]. This is plausible as the difference in terms of both the various gas-sensitive layers and the μ -hotplate should be much smaller between sensors A and B compared to sensor C. Note, however, that the performance of sensor C is not as good as for sensors A and B also when trained from scratch. Overall, all training curves converge towards the best possible model achieved with 700 UGMs without transfer learning, with sensor B achieving a slightly lower RMSE for formaldehyde with additional transfer learning compared to training the

model from scratch. Note that this demonstrates that more than 700 UGMs would not improve the accuracy significantly, indicating that 700 UGMs are sufficient even for the complex gas mixtures (10 gases plus humidity) studied here.

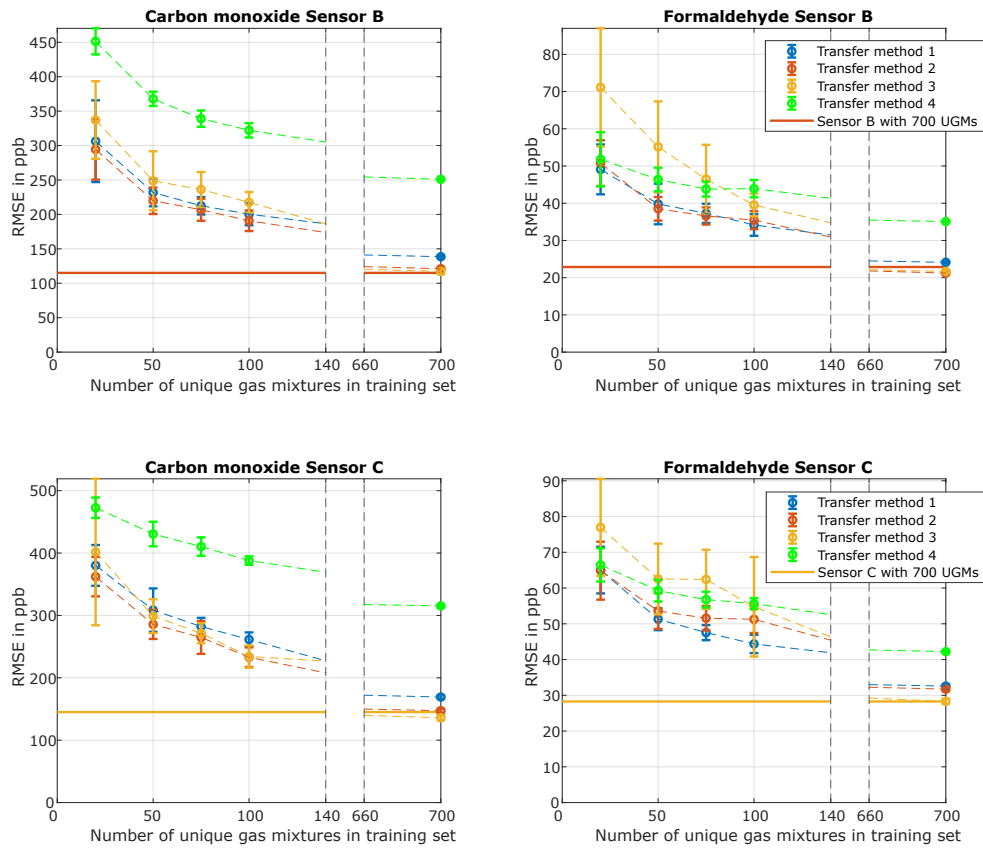


Figure 7. Comparison of different transfer methods for carbon monoxide and formaldehyde. Transfer method 1: learning rate set to the value reached at the end of original training. Transfer method 2: learning rate set to the value reached halfway through original training. Transfer method 3: learning rate is set to the original value. Transfer method 4: implicit feature extraction fixed only fully connected layer can be trained.

In conclusion, it can be stated that transfer learning can significantly reduce the required calibration time by up to 93%, i.e., 50 UGMs instead of 700, while increasing the RMSE only by approx. a factor of 2 even for sensors from a different batch. In practical applications, the possible calibration time reduction will depend on the required performance, i.e., the acceptable RMSE for a specific use case.

After the general findings are discussed above, Table 2 provides quantitative values for carbon monoxide and formaldehyde. Considering the ambient threshold limit value for CO of 10 mg/m³ [40], corresponding to approx. 8.5 ppm, the accuracy achieved with small transfer sets is excellent and significantly lower than a model generated from scratch without transfer learning, as shown for sensor A. Similarly, for formaldehyde, the WHO limit of 80 ppb [41] could be monitored with sensors trained by transfer learning on small calibration data sets.

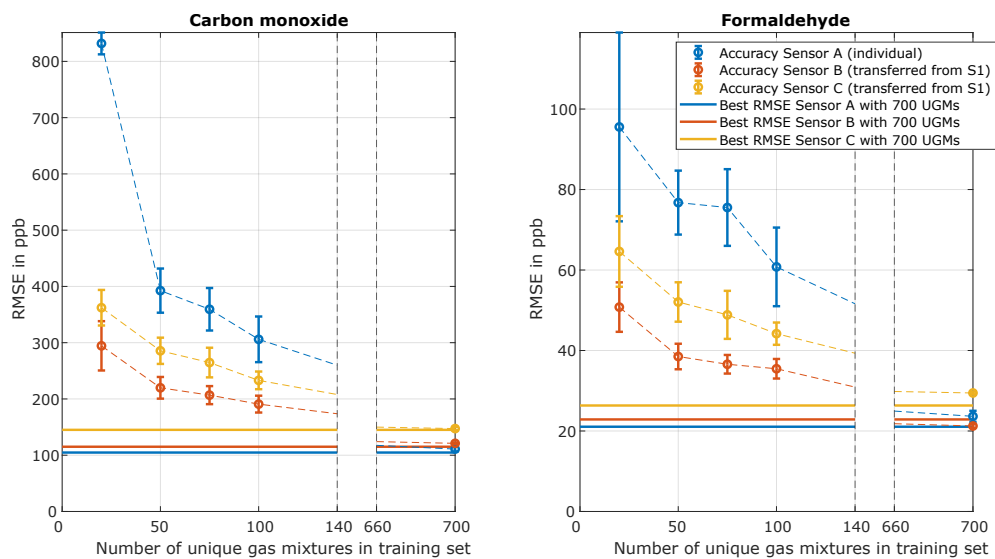


Figure 8. Comparison of carbon monoxide and formaldehyde with and without transfer learning over the different number of training samples and sensors.

Table 2. Detailed RMSE values for carbon monoxide and formaldehyde for Figure 8 with a static training set or training set with random subsampling.

	Carbon Monoxide in ppb			Formaldehyde in ppb		
	Sensor A	Sensor B (Transfer)	Sensor C (Transfer)	Sensor A	Sensor B (Transfer)	Sensor C (Transfer)
700 UGMs mean RMSE (std)	110.9 (4.1)	121.1 (1.6)	147.3 (3.0)	23.6 (1.4)	21.2 (0.4)	29.4 (0.3)
Initial model before transfer		532.3	782.9		55.1	73.0
75 UGMs mean RMSE (std) static training set	332.6 (8.4)	194.3 (1.9)	256.3 (2.3)	67.0 (2.5)	36.1 (0.5)	44.4 (0.4)
75 UGMs mean RMSE (std) random subsampling	359.4 (37.8)	206.8 (16.0)	264.7 (26.3)	75.5 (9.5)	36.6 (2.3)	48.9 (6.0)
50 UGMs mean RMSE (std) static training set	374.8 (7.3)	206.0 (1.5)	238.1 (2.8)	76.1 (0.9)	36.8 (0.4)	46.7 (0.2)
50 UGMs mean RMSE (std) random subsampling	392.5 (39.2)	219.8 (19.3)	285.6 (23.3)	76.8 (8.0)	38.5 (3.2)	52.1 (4.9)
20 UGMs mean RMSE (std) static training set	856.4 (8.0)	268.3 (3.9)	349.1 (2.8)	80.3 (1.4)	53.3 (1.5)	72.4 (2.4)
20 UGMs mean RMSE (std) random subsampling	831.9 (19.3)	294.4 (43.8)	362.1 (31.6)	95.6 (23.5)	50.8 (6.1)	64.6 (8.8)

Furthermore, Table 2 and Figure 9 also demonstrate the effect of different sets of transfer samples for transfer learning. The static sets, where the training was repeated ten times with the same data sample, always show a slightly smaller mean RMSE with a much smaller variation than the training set with random subsampling. In the example shown, it seems the randomly chosen static training sets for formaldehyde and carbon monoxide were a lucky choice as they resulted in a lower mean RMSE. More significant is the standard deviation, which is significantly larger for the training sets with random subsampling. It seems clear that the changing UGMs cause additional variance within the training sets.

A closer look at Figure 9 shows that a poor choice for 75 transfer samples results in a higher RMSE for CO (207 ppb plus 16 ppb) than a good choice for 50 UGMs (220 ppb minus 19 ppb). This indicates that the choice of samples for transfer learning is important to obtain the best possible results, i.e., even shorter calibration times or better performance with the same calibration time. When analyzing the distribution of the different training sets, a good training set is usually signified by samples spanning the full concentration range of the target gas. However, the distributions of other gases are also relevant for the achieved performance. Therefore, the design of experiment (DoE) for optimal transfer learning-based calibration is a topic for future research.

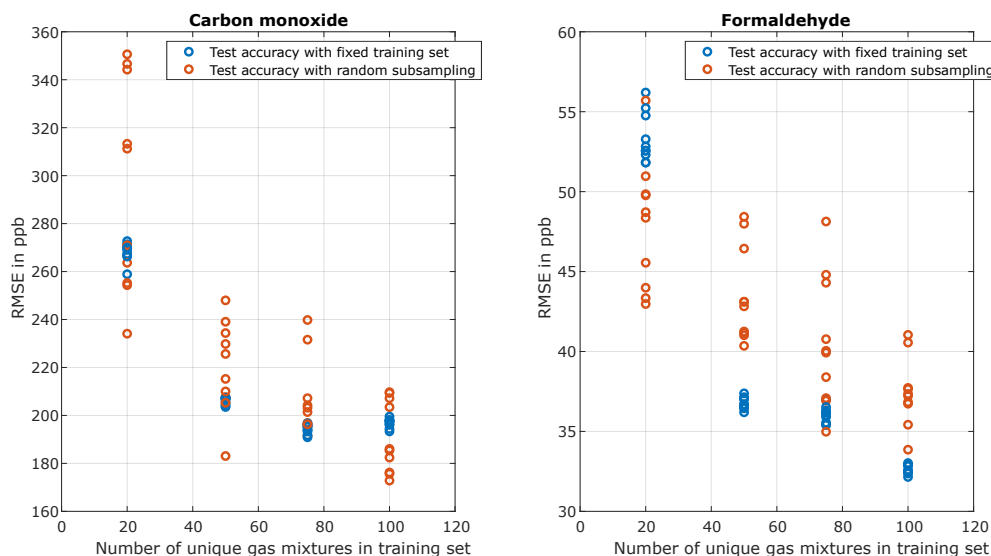


Figure 9. Comparison of multiple training instances for carbon monoxide and formaldehyde. Training and testing were performed 10 times for a static training set with always the same UGMs for training and testing (blue) and training sets with random subsampling where always different UGMs for training and the same for testing were used (orange).

The analysis is repeated for sensor B using the adapted normalization strategy with the results shown in Figure 10 for formaldehyde and xylene. Formaldehyde was chosen to have one example throughout this study, and xylene was selected because the difference between both normalization strategies was one of the largest. The figure shows the former and the adapted normalization results in the upper and lower part, respectively. The best RMSE for formaldehyde achieved with the adapted normalization is slightly better, and the improvement for xylene is significant with a reduction of the RMSE of 25%. Thus, normalization substantially impacts model building, and a suitable normalization can significantly improve the ML model accuracy. At the same time, the general shape of the improvement in accuracy with an increasing number of transfer samples is very similar for both gases, independent of the applied normalization. The obtained models can yield slightly higher or even lower RMSE values for both scenarios depending on the transfer method. The ranking of the different transfer methods is the same, and in each case, transfer method two achieves the best overall results. The only significant difference is the smaller mean RMSE achieved with the adapted normalization, which can be attributed to the improved preprocessing allowing for a better model accuracy. This result shows that the findings for the transfer learning method might apply not only to this specific example but might also be a suitable approach for transfer learning for gas sensors in general.

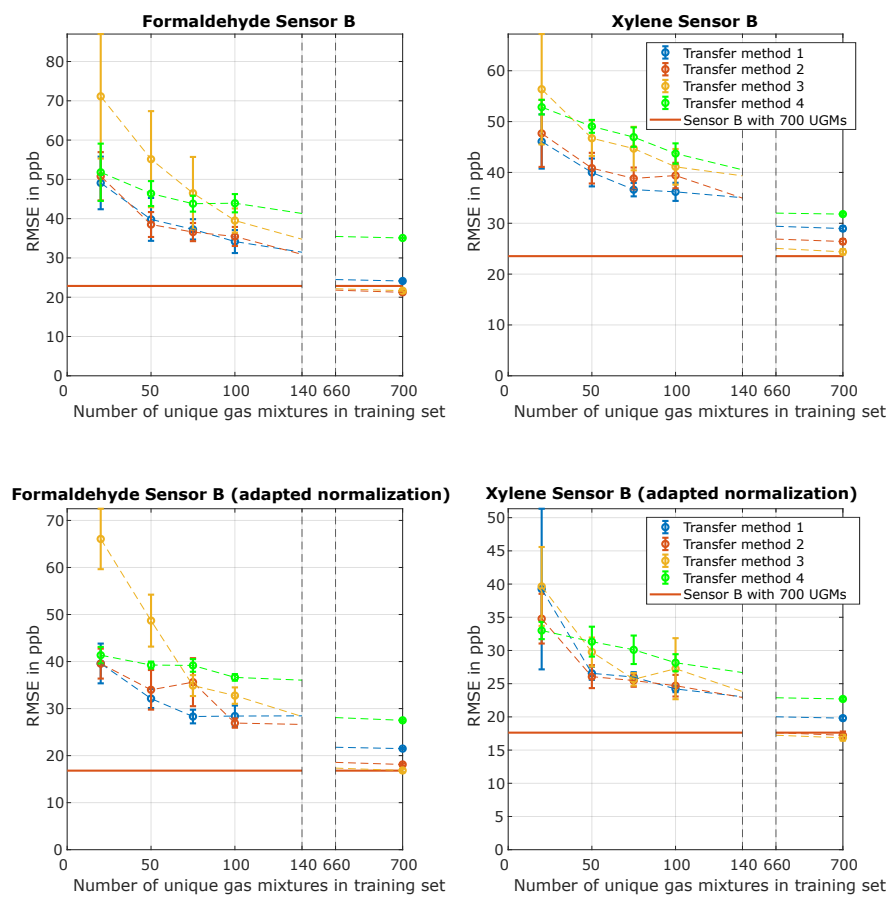


Figure 10. Comparison of different normalization methods for transfer learning regarding formaldehyde and xylene. The upper figures show the achieved accuracy over the different number of transfer samples with the former normalization, and the lower part shows the accuracy with the adapted normalization.

Again, as for the former normalization, the models obtained with transfer learning show a better performance than without both in terms of overall accuracy and training stability as indicated by the standard deviation (cf. Table 3).

Table 3. RMSE values together with the standard deviation for the adapted normalization with and without transfer learning for formaldehyde and xylene.

Mean RMSE (std) in ppb	20 UGMs	50 UGMs	75 UGMs	100 UGMs	150 UGMs	500 UGMs
Without transfer learning (formaldehyde)	87.6 (10.4)	78.5 (3.3)	72.0 (9.3)	61.0 (6.7)	40.3 (5.2)	21.0 (1.0)
Transfer method 2 (formaldehyde)	39.6 (3.2)	34.0 (4.2)	35.6 (5.1)	26.9 (1.0)	26.6 (2.0)	19.9 (0.6)
Without transfer learning (xylene)	65.4 (5.6)	50.9 (9.1)	38.1 (2.0)	32.4 (1.9)	26.5 (1.4)	20.0 (0.5)
Transfer method 2 (xylene)	34.8 (3.7)	26.1 (1.7)	25.5 (0.9)	24.7 (1.6)	22.9 (1.0)	18.8 (0.7)

Finally, we compare results achieved with transfer learning (here with the former, sub-optimal normalization) and a simplified global model. Figure 11 shows the RMSE

achieved for formaldehyde for the two models when trained with the same number of UGMs recorded with two different sensors (sensor A, sensor B) and tested on the same test data. First, the solid lines indicate the best possible RMSE achieved when training with only 700 UGMs from sensor A and testing with the test data (200 UGMs) also from sensor A. In this scenario, the TCOCNN (21.1 ppb) performs slightly better than the established FESR methods (25.9 ppb), but this also depends on the target gas, and in general, the performance achieved with both methods is comparable as demonstrated previously [3]). The dashed lines indicate the accuracies achieved with the additional samples from sensor B and tested with data from sensor B. Comparing the general trend and the standard deviations, both approaches are quite similar regarding the number of UGMs. Thus, additional samples from the new sensor can improve the prediction quality for the global model and the neural network. The selection of UGMs (DoE) is most important for achieving a low RMSE. Furthermore, in the stages between 20 and 100 UGMs, the differences between both approaches are very small. They can most probably be attributed to the slightly better initial accuracy of the transfer learning model. The largest difference is observed if both models are trained with 700 additional UGMs. In this case, the transfer model converges towards the best possible RMSE while the global model settles at a higher RMSE. This can be attributed to the generalization of the global model, which builds the best possible model for both sensors simultaneously, thus not achieving the lowest RMSE for a sensor-specific model. For transfer learning, on the other hand, the new model is specifically adapted to the new sensor B and will therefore show a higher RMSE for the original sensor A (84.4 ppb). A larger dataset with more sensors is required to compare the pros and cons of individual transfer learning and global modeling more in-depth. In addition, methods for signal correction like direct standardization or orthogonal signal correction should be analyzed in this context [5,6].

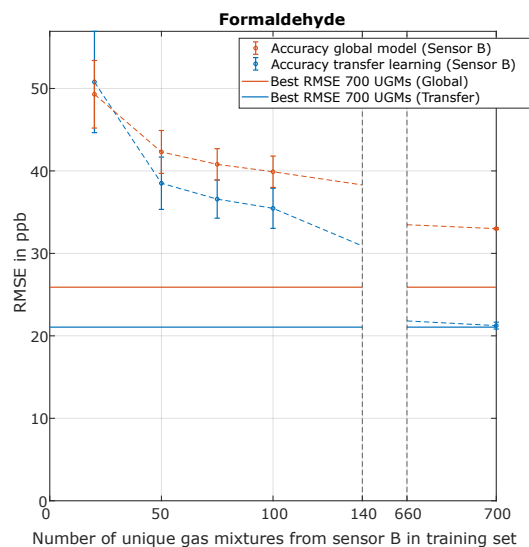


Figure 11. Comparison between transfer learning and global modeling over the different number of training samples (additional samples from sensor B).

4. Discussion

This paper demonstrates that a significant reduction of the calibration time for accurate quantification of multiple gases at a ppb level within complex environments is possible with transfer learning. Various methods of transfer learning can be applied to reduce the training time. This study achieved the best results with transfer method 2 (complete adaptation of all CNN parameters with initial learning rate for transfer learning set to the value reached

halfway through the original model training). This allowed a reduction of 93% (50 instead of 700 unique gas mixtures) while increasing the RMSE by less than a factor of 2. However, the optimal method cannot be generalized as this will vary according to the specific sensor model, the target gas, and the use case, i.e., concentration range and interfering gases. Furthermore, we could show that transferring data-driven models from one sensor to another always results in better accuracy than building the models from scratch, both when analyzing long or short calibration times. Again, the number of transfer samples required for building an acceptable data-driven model will vary, i.e., between sensors from different batches, and also strongly depends on the required accuracy. Furthermore, the influence of varying transfer samples was analyzed, which demonstrated that transfer learning could only develop its full potential if the transfer samples were selected with care. However, the selection of suitable samples is not straightforward and not yet fully understood. However, it would seem reasonable to follow the same rules as for the design of experiments in ML model building in general. Finally, transfer learning was compared with global modeling. For the small dataset with only two sensors studied here, we could show that both model-building approaches (transfer learning and global modeling) yield reasonably similar results, especially when using only a few additional samples from a second sensor. When using the entire dataset, the differences between transfer learning and global modeling become evident, i.e., transfer learning achieves a higher accuracy for a sensor-specific model, while global modeling yields greater generalization for different sensors. For a more in-depth analysis of the benefits and drawbacks of both methods, a larger dataset with multiple different sensors is required.

5. Conclusions

In conclusion, transfer learning can be used to significantly reduce the calibration time of MOS gas sensors by up to 93% while maintaining an RMSE that is only double the best possible. It was shown that different transfer methods provide additional benefits and drawbacks and that the selection of UGMs is most important. Nevertheless, a suggestion of which method or UGMs to use could not be made because this highly depends on the use case. Finally, we compared deep transfer learning with the global modeling approach. We found that deep transfer learning performs similarly to global modeling, although it might be essential to analyze this comparison in more detail with a larger dataset (multiple sensors).

For further research, it might be interesting to not only use a single sensor to build the initial model but to use a global calibration model based on multiple sensors and combine this approach with transfer learning. Furthermore, a wider variety of transfer methods should be tested and compared with deep transfer learning, as proposed in this work. Likewise, transfer learning should include methods from deep learning and extend to complementary approaches such as direct standardization or other signal-related transfer methods. This would also allow for combining different datasets with only one or few (target) gases in common but different backgrounds. This could help analyze the transferability between backgrounds and improve the understanding of which gases interfere with each other on the sensor.

Author Contributions: Conceptualization, Y.R., C.B. and A.S.; methodology, Y.R., P.G. and J.A.; software, Y.R. and P.G.; validation, Y.R. and J.A.; formal analysis, Y.R.; investigation, Y.R.; resources, A.S.; data curation, J.A.; writing—original draft preparation, Y.R.; writing—review and editing, Y.R., J.A., C.B., T.S. and A.S.; visualization, Y.R.; supervision, Y.R., C.B., T.S. and A.S.; project administration, A.S. and C.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partly performed within the project “VOC4IAQ” funded by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) through the program Industrial Collective Research (AiF-iGF) under the grant number 22084N/1. The authors thank the European Regional Development Fund (ERDF) for supporting their research within the project number 14.2.1.4-2019/1. We acknowledge support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) and Saarland University within the ‘Open Access Publication Funding’ programme.

Data Availability Statement: The underlying data for this work can be accessed at <https://doi.org/10.5281/zenodo.6821340>, (accessed on 17 May 2022). Robin, Yannick, Amann, Johannes, Bur, Christian, and Schütze, Andreas. (2022). Transfer Learning Dataset for Metal Oxide Semiconductor Gas Sensors (1.0).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- CNN Convolutional neural network
- DoE Design of experiment
- FE Feature extraction
- FESR Feature extraction selection regression
- FS Feature selection
- IAQ Indoor air quality
- ML Machine learning
- MOS Metal oxide semiconductor
- NAS Neural architecture search
- PLSR Partial least squares regression
- PM Particulate matter
- RH Relative humidity
- RMSE Root-mean-squared error
- TC Temperature-cycle
- TCO Temperature-cycled operation
- UGM Unique gas mixture
- VOC Volatile organic compounds

Appendix A

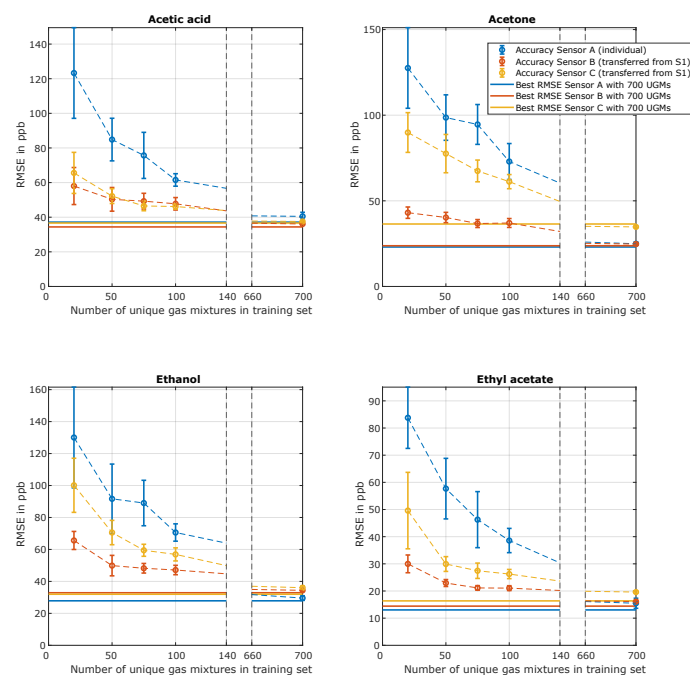


Figure A1. Comparison of acetic acid, acetone, ethanol, and ethyl acetate with and without transfer learning over the different number of training samples and sensors.

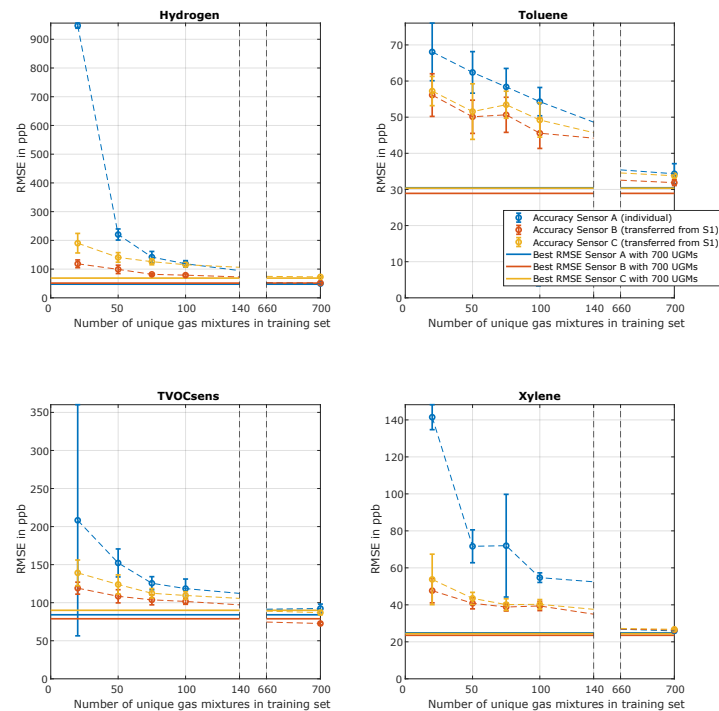


Figure A2. Comparison of hydrogen, toluene, TVOC_{sens}, and xylene with and without transfer learning over the different number of training samples and sensors.

References

- GBD 2019 Risk Factors Collaborators. Global burden of 87 risk factors in 204 countries and territories, 1990–2019: A systematic analysis for the Global Burden of Disease Study 2019. *Lancet* **2020**, *396*, 1223–1249. [[CrossRef](#)]
- Hauptmann, M.; Lubin, J.H.; Stewart, P.A.; Hayes, R.B.; Blair, A. Mortality from Solid Cancers among Workers in Formaldehyde Industries. *Am. J. Epidemiol.* **2004**, *159*, 1117–1130. doi: [[CrossRef](#)] [[PubMed](#)]
- Robin, Y.; Amann, J.; Baur, T.; Goodarzi, P.; Schultealbert, C.; Schneider, T.; Schütze, A. High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning. *Atmosphere* **2021**, *12*, 1487. [[CrossRef](#)]
- Schütze, A.; Sauerwald, T. Dynamic operation of semiconductor sensors. In *Semiconductor Gas Sensors*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 385–412. [[CrossRef](#)]
- Fonollosa, J.; Fernández, L.; Gutiérrez-Gálvez, A.; Huerta, R.; Marco, S. Calibration transfer and drift counteraction in chemical sensor arrays using Direct Standardization. *Sens. Actuators B Chem.* **2016**, *236*, 1044–1053. [[CrossRef](#)]
- Fernandez, L.; Guney, S.; Gutierrez-Galvez, A.; Marco, S. Calibration transfer in temperature modulated gas sensor arrays. *Sens. Actuators B Chem.* **2016**, *231*, 276–284. [[CrossRef](#)]
- Miquel-Ibarz, A.; Burgués, J.; Marco, S. Global calibration models for temperature-modulated metal oxide gas sensors: A strategy to reduce calibration costs. *Sens. Actuators B Chem.* **2022**, *350*, 130769. [[CrossRef](#)]
- Fonollosa, J.; Neftci, E.; Huerta, R.; Marco, S. Evaluation of calibration transfer strategies between Metal Oxide gas sensor arrays. *Procedia Eng.* **2015**, *120*, 261–264. [[CrossRef](#)]
- Laref, R.; Losson, E.; Sava, A.; Siadat, M. Calibration Transfer to Address the Long Term Drift of Gas Sensors for in Field NO₂ Monitoring. In Proceedings of the 2021 International Conference on Control, Automation and Diagnosis (ICCAD), Grenoble, France, 3–5 November 2021. [[CrossRef](#)]
- Jaeschke, C.; Padilla, M.; Glöckler, J.; Polaka, I.; Leja, M.; Veliks, V.; Mitrovics, J.; Leja, M.; Mizaikoff, B. Modular Breath Analyzer (MBA): Introduction of a Breath Analyzer Platform Based on an Innovative and Unique, Modular eNose Concept for Breath Diagnostics and Utilization of Calibration Transfer Methods in Breath Analysis Studies. *Molecules* **2021**, *26*, 3776. [[CrossRef](#)] [[PubMed](#)]
- Vito, S.D.; D'Elia, G.; Francia, G.D. Global calibration models match ad-hoc calibrations field performances in low cost particulate matter sensors. In Proceedings of the 2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May–1 June 2022. [[CrossRef](#)]

12. Torrey, L.; Shavlik, J. Transfer Learning. In *Handbook of Research on Machine Learning Applications*; IGI Global: Hershey, PA, USA, 2009.
13. Bozinovski, S. Reminder of the First Paper on Transfer Learning in Neural Networks, 1976. *Informatica* **2020**, *44*, 291–302. [[CrossRef](#)]
14. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *arXiv* **2020**, arXiv:1911.02685.
15. Plested, J.; Gedeon, T. Deep transfer learning for image classification: A survey. *arXiv* **2022**, arXiv:2205.09904.
16. Robin, Y.; Amann, J.; Goodarzi, P.; Schütze, A.; Bur, C. Transfer Learning to Significantly Reduce the Calibration Time of MOS Gas Sensors. In Proceedings of the 2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May–1 June 2022. [[CrossRef](#)]
17. Yadav, K.; Arora, V.; Jha, S.K.; Kumar, M.; Tripathi, S.N. Few-shot calibration of low-cost air pollution (PM_{2.5}) sensors using meta-learning. *arXiv* **2021**, arXiv:2108.00640.
18. Arendes, D.; Lensch, H.; Amann, J.; Schütze, A.; Baur, T. P13.1—Modular design of a gas mixing apparatus for complex trace gas mixtures. In Proceedings of the Poster at Dresdner Sensor-Symposium, Online, 6–8 December 2021; AMA Service GmbH: Wunstorf, Germany, 2021; [[CrossRef](#)]
19. Baur, T.; Amann, J.; Schultealbert, C.; Schütze, A. Field Study of Metal Oxide Semiconductor Gas Sensors in Temperature Cycled Operation for Selective VOC Monitoring in Indoor Air. *Atmosphere* **2021**, *12*, 647. [[CrossRef](#)]
20. Helwig, N.; Schüler, M.; Bur, C.; Schütze, A.; Sauerwald, T. Gas mixing apparatus for automated gas sensor characterization. *Meas. Sci. Technol.* **2014**, *25*, 055903. [[CrossRef](#)]
21. Baur, T.; Schütze, A.; Sauerwald, T. Optimierung des temperaturzyklischen Betriebs von Halbleitersensoren (Optimization of temperature cycled operation of semiconductor gas sensors). *tm-Tech. Mess.* **2015**, *82*, 187–195. [[CrossRef](#)]
22. Schultealbert, C.; Baur, T.; Schütze, A.; Böttcher, S.; Sauerwald, T. A novel approach towards calibrated measurement of trace gases using metal oxide semiconductor sensors. *Sens. Actuators Chem.* **2017**, *239*, 390–396. [[CrossRef](#)]
23. Loh, W.L. On Latin hypercube sampling. *Ann. Stat.* **1996**, *24*, 2058–2080. [[CrossRef](#)]
24. Baur, T.; Bastuck, M.; Schultealbert, C.; Sauerwald, T.; Schütze, A. Random gas mixtures for efficient gas sensor calibration. *J. Sens. Sens. Syst.* **2020**, *9*, 411–424. [[CrossRef](#)]
25. Hofmann, H.; Plieninger, P. Bereitstellung einer Datenbank zum Vorkommen von flüchtigen organischen Verbindungen in der Raumluft. In *WaBoLu Hefte*; Umweltbundesamt: Dessau-Roßlau, Germany, 2008. Available online: <https://www.umweltbundesamt.de/sites/default/files/medien/publikation/long/3637.pdf> (accessed on 17 May 2022).
26. Kobald, A.; Weimar, U.; Barsan, N. Regression Model for the Prediction of Pollutant Gas Concentrations with Temperature Modulated Gas Sensors. In Proceedings of the 2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May–1 June 2022. [[CrossRef](#)]
27. White, C.; Neiswanger, W.; Savani, Y. BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search. *arXiv* **2020**, arXiv:1910.11858v3.
28. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian Optimization of Machine Learning Algorithms. *arXiv* **2012**, arXiv:1206.2944v2.
29. Robin, Y.; Amann, J.; Goodarzi, P.; Baur, T.; Schultealbert, C.; Schneider, T.; Schütze, A. Überwachung der Luftqualität in Innenräumen mittels komplexer Sensorsysteme und Deep Learning Ansätzen. In Proceedings of the Vorträge at Dresdner Sensor-Symposium, Online, 6–8 December 2021, AMA Service GmbH: Germany, 2021. [[CrossRef](#)]
30. Xu, W.; He, J.; Shu, Y. Transfer Learning and Deep Domain Adaptation. In *Advances and Applications in Deep Learning*; IntechOpen: London, UK, 2020. [[CrossRef](#)]
31. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [[CrossRef](#)]
32. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A Survey on Deep Transfer Learning. *arXiv* **2018**, arXiv:1808.01974.
33. Wang, S.H.; Xie, S.; Chen, X.; Guttery, D.S.; Tang, C.; Sun, J.; Zhang, Y.D. Alcoholism Identification Based on an AlexNet Transfer Learning Model. *Front. Psychiatry* **2019**, *10*. [[CrossRef](#)] [[PubMed](#)]
34. Li, Z.; Hoiem, D. Learning without Forgetting. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
35. Yamashita, R.; Nishio, M.; Do, R.K.G.; Togashi, K. Convolutional neural networks: An overview and application in radiology. *Insights Imaging* **2018**, *9*, 611–629. [[CrossRef](#)] [[PubMed](#)]
36. Olszewski, R.T.R. Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data. Ph.D. Thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, 2001. [[CrossRef](#)]
37. Schneider, T.; Helwig, N.; Schütze, A. Automatic feature extraction and selection for condition monitoring and related datasets. In Proceedings of the 2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Houston, TX, USA, 14–17 May 2018. [[CrossRef](#)]
38. Kowalski, B.R. *Chemometrics Mathematics and Statistics in Chemistry*; Springer: Berlin/Heidelberg, Germany, 2013.
39. de Jong, S. SIMPLS: An alternative approach to partial least squares regression. *Chemom. Intell. Lab. Syst.* **1993**, *18*, 251–263. [[CrossRef](#)]

40. Umweltbundesamt. Information on Carbon Monoxide (CO). 2017. Available online: https://www.umweltbundesamt.de/sites/default/files/medien/370/dokumente/infoblatt_kohlenmonoxid_eng_0.pdf (accessed on 24 August 2022).
41. WHO Regional Office for Europe. *WHO Guidelines for Indoor air Quality: Selected Pollutants*; World Health Organization: Copenhagen, Denmark, 2010. [[CrossRef](#)]

3.4 Paper 3 – Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors

Y. Robin, J. Amann, T. Schneider, A. Schütze, C. Bur

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

Atmosphere 2023, 14(7), 1123;

The original paper can be found in the online version at <https://www.mdpi.com/2377952> or DOI: <https://doi.org/10.3390/atmos14071123>

© 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). Reprinted, with permission, from Y. Robin, J. Amann, T. Schneider, A. Schütze, C. Bur; Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors; Atmosphere 2023.

3.4.1 Synopsis

The previous papers demonstrated that the TCOCNN performs superior to the FESR approach and that transfer learning can significantly reduce the calibration time. Within this publication, an in-depth comparison of calibration transfer methods was performed, and global model building for initial model building for calibration transfer was analyzed. Thereby, the capabilities of the TCOCNN and the FESR approach to generalize across multiple sensors were studied. Likewise, their general ability to apply to various sensors simultaneously and their compatibility with calibration transfer methods were tested. Once again, a new dataset was designed to be able to perform a comprehensive study. The dataset was recorded with the most recent GMA developed at LMT. This time, eleven different VOCs were mixed, namely acetaldehyde, acetic acid, acetone, ethanol, ethyl acetate, formaldehyde, isopropanol, limonene, n-hexane, toluene, and xylene. Acetone was selected as the target gas because it is relevant for IAQ monitoring and showed the best overall results. For a realistic background, hydrogen and carbon monoxide were chosen. Similarly, the relative humidity varied between 25 % and 75 %. The corresponding ranges and distribution of the 930 UGMs are illustrated in Figure 3.5.

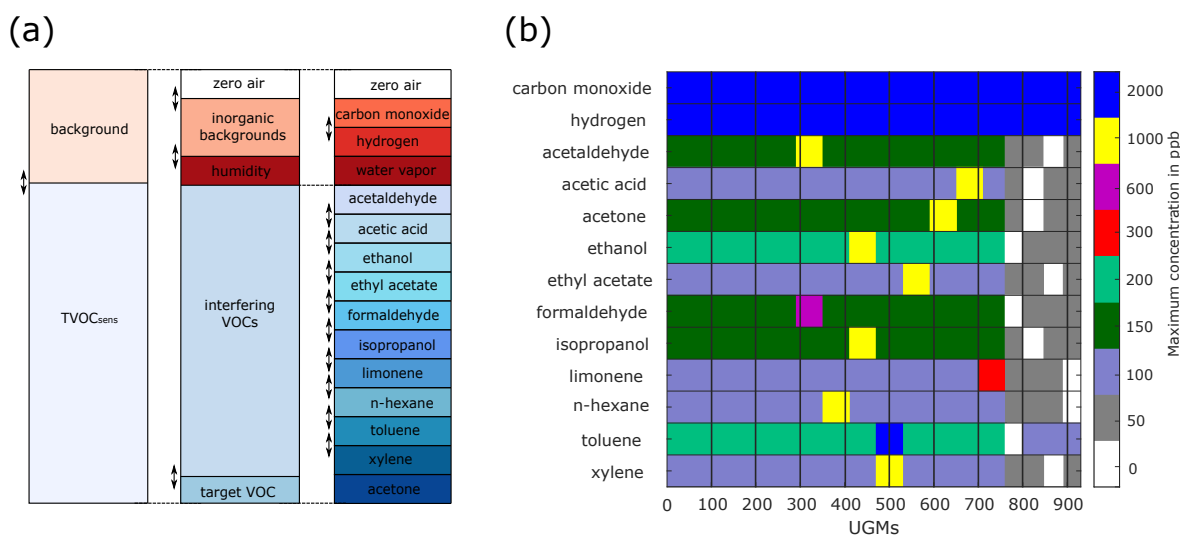


Figure 3.5: Overview of the gases included in the randomized calibration. Each UGM contains all of the shown gases. (a) The composition of the different UGMs (adapted from [24, Paper 2]). (b) All the maximum concentrations during recording. The lowest concentration for all VOCs during the measurement is 0 ppb, for carbon monoxide 200 ppb, and for hydrogen 400 ppb. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.

To be able to test for calibration transfer, a total of seven sensors (SGP40) were used to record the 930 different UGMs. Those sensors were again operated in TCO with twelve high-temperature steps (400 °C) interlaced with twelve low-temperature steps (100 - 375 °C). The same exception as before was applied to sub-sensor 3, which was only modulated between 200 °C and 300 °C. As in the previous publications, the first step was identifying the optimal hyperparameter regarding the target gas with the help of Bayesian optimization for the TCOCNN. Afterward, the algorithm stack for the reference FESR method was defined. The stack consisted of the Adaptive Linear Approximation (ALA) feature extractor [129, 130], followed by the Pearson selection to identify the optimal number of features.²⁸ A PLSR with a maximum of 100 components was chosen for the final regression algorithm. For this procedure, ALA was used as it showed excellent performance in [234]. For training, validating, and testing, a 70/10/20 split was used, which was constant throughout all evaluations.

When trained with one sensor and tested with the corresponding test data, the TCOCNN reached a baseline RMSE of around 15 ppb while the FESR approach reached an RMSE of ~ 20 ppb. When the setup for both methods was completed, it was first analyzed which method generalizes best. Consequently, models with each method were built with 1 to 6 sensors and tested with the combined test data. There, it was demonstrated that the FESR method struggles to find a model that performs well for multiple sensors simultaneously. This differs from the TCOCNN, which found better models with every sensor added (RMSE 12 ppb). This indicates that the TCOCNN can find more general features independent of the sensor. The next step was to train a model on one sensor and test it with another to see if the model could be used for multiple sensors. Therefore, models were built with only one of the sensors, 1 to 6, and tested with sensor 7. There, it was revealed that the similarity between sensors strongly influences the accuracy. Furthermore, the selection of the model also had a significant influence. This can be attributed to the different features selected with the FESR and TCOCNN. Consequently, this indicates that training a model on one sensor and applying it to another does generally not work. The next task was to analyze the effect of global modeling on generalization. Therefore, models were built with 1 to 6 sensors and tested on sensor 7. Compared to the previous evaluation, multiple sensors were used simultaneously to construct the model. It was shown that the FESR approach and the TCOCNN can be used to build a global initial model that can generalize across sensors. Similarly to the previous evaluation, the TCOCNN performed best in finding

²⁸Wrapper method from the FESR toolbox was used to identify the optimal subset.

more general models. For acetone, it was possible to reduce the RMSE from 70 ppb (one sensor) to 30 ppb with the help of the TCOCNN and all six sensors. Although the RMSE was doubled compared to training from scratch with the complete training data, the results indicate promising performance. To improve the results even further, the effect of calibration transfer methods was analyzed in detail. In the following evaluation, multiple calibration transfer methods were tested to evaluate their performance. The initial models were built with 1 to 6 sensors. The four calibration transfer methods, Direct Standardization (DS), Piecewise Direct Standardization (PDS), transfer learning, and global model building with transfer samples, were tested with 5, 25, 125, and 600 transfer UGMs (cf. Figure 3.6).

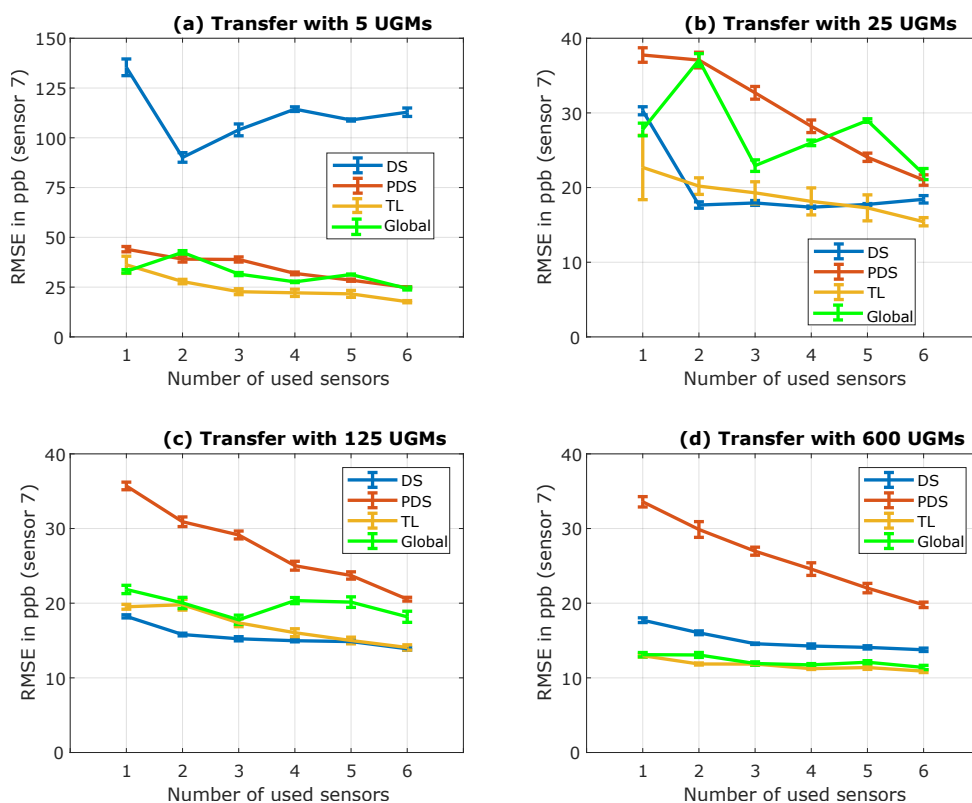


Figure 3.6: Comparison of Direct Standardization (DS), Piecewise Direct Standardization (PDS), transfer learning for DL (TL), and global model building concerning the TCOCNN. Different numbers of UGMs for transfer learning are used in the different sub-plots. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.

Multiple conclusions were drawn from this experiment. It was obvious that, for the TCOCNN, using more sensors for initial model building significantly improved the generalization capabilities. The best method to achieve low RMSE values for very small datasets was to take six sensors for initial model building and use transfer learning from DL. This combination achieved an RMSE with only five transfer UGMs of 17.7 ppb (calibration time reduction of 99.3 %).²⁹ Similarly, global model building and PDS reached decent results of around 25 ppb. The slightly better performance of transfer learning can be attributed to the model being specially trained for the new sensor. DS needed 25 transfer UGMs to achieve reasonable results but then achieved similar results to transfer learning. The poor performance of DS can be attributed to the complex calculation of the correction matrix C . When using the largest dataset of 600 transfer UGMs, transfer learning and global modeling became ever closer, and DS and PDS started to saturate at slightly higher RMSE values. The saturation of those two methods can be attributed to the specific projection that is calculated (considering all sensors). At the same time, global model building and transfer learning can focus more on the target sensor. Besides the results achieved for the TCOCNN, similar outcomes were observed by applying the methods to the FESR approach. However, because of the worse baseline from the first two evaluations, the results were never quantitatively comparable to the TCOCNN. Furthermore, PDS did not work well for the FESR approach.

The initial evaluations showed that the TCOCNN outperforms the FESR approach regarding its capability to generalize across sensors. This was especially prominent when building a model with six sensors and testing it on the remaining. There, the TCOCNN showed the capability to find universally applicable features. In the final comparison of all calibration transfer methods, it was shown that it is possible to further reduce the long calibration time and sensor-to-sensor variance compared to Paper 2. The most significant improvement for calibration transfer was achieved with the help of global model building, which further leverages the performance of DL-based transfer learning. Moreover, it was demonstrated that this method outperforms state-of-the-art approaches like signal standardization. The improvement was especially prominent for small transfer datasets, which are most important for industry (calibration time reduction of 99.3 %). For IAQ monitoring, this indicates that it is possible to calibrate a MOS gas sensor within two hours for a wide range of applications.³⁰ Two hours or five UGMs is suitable

²⁹A 15 ppb baseline is achieved by training the TCOCNN with 700 UGMs.

³⁰One UGM had a duration of 24 minutes. Usually, ten observations are recorded per UGM, and only stable observations are later used.

for large-scale calibration since no GMA is needed (gas test), and productivity can be maintained compared to today's standards. To better understand how this significant improvement was possible, the following paper introduces XAI methods, which allows for a detailed insight into the model's inner workings, the sensor itself, and the used TC.

The main takeaways of this publication are:





- TCOCNN is better for finding a model applicable to multiple sensors simultaneously.
- Calibration transfer between sensors without global model building or calibration transfer is rarely possible.
- Global model building without transfer samples helps the model generalize.
 - With multiple sensors in training, decent results can be achieved.
 - TCOCNN outperforms the FESR approach.
- All calibration transfer methods show promising results.
- Best result for five transfer UGMs is achieved with six sensors for initial model building and transfer learning from DL.
 - Most important for industry and IAQ monitoring.
 - RMSE: 17.7 ppb.
 - r-squared: > 0.99 .
 - Calibration time reduction of 99.3 % (8 days down to 2 hours).
 - Suitable for industrial application.
 - Next best methods are global modeling with transfer samples and Piecewise Direct Standardization (PDS) with an RMSE: ~ 25 ppb.
- Similar results can be achieved with the FESR approach.

Open questions/tasks are:

- Is it possible to explain the inner workings of the TCOCNN to understand why it outperforms the FESR approach?
- Can the TCOCNN be applied to different fields (e.g., condition monitoring)?

Article

Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors

 Yannick Robin ^{*}, Johannes Amann , Tizian Schneider , Andreas Schütze  and Christian Bur 

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany; j.amann@lmt.uni-saarland.de (J.A.); t.schneider@lmt.uni-saarland.de (T.S.); schuetze@lmt.uni-saarland.de (A.S.); c.bur@lmt.uni-saarland.de (C.B.)

* Correspondence: y.robin@lmt.uni-saarland.de

Abstract: Although metal oxide semiconductors are a promising candidate for accurate indoor air quality assessments, multiple drawbacks of the gas sensors prevent their widespread use. Examples include poor selectivity, instability over time, and sensor poisoning. Complex calibration methods and advanced operation modes can solve some of those drawbacks. However, this leads to long calibration times, which are unsuitable for mass production. In recent years, multiple attempts to solve calibration transfer have been made with the help of direct standardization, orthogonal signal correction, and many more methods. Besides those, a new promising approach is transfer learning from deep learning. This article will compare different calibration transfer methods, including direct standardization, piecewise direct standardization, transfer learning for deep learning models, and global model building. The machine learning methods to calibrate the initial models for calibration transfer are feature extraction, selection, and regression (established methods) and a custom convolutional neural network TCOCNN. It is shown that transfer learning can outperform the other calibration transfer methods regarding the root mean squared error, especially if the initial model is built with multiple sensors. It was possible to reduce the number of calibration samples by up to 99.3% (from 10 days to approximately 2 h) and still achieve an RMSE for acetone of around 18 ppb (15 ppb with extended individual calibration) if six different sensors were used for building the initial model. Furthermore, it was shown that the other calibration transfer methods (direct standardization and piecewise direct standardization) also work reasonably well for both machine learning approaches, primarily when multiple sensors are used for the initial model.

Keywords: indoor air quality; metal oxide semiconductor; volatile organic compounds; calibration transfer; deep learning; direct standardization



Citation: Robin, Y.; Amann, J.; Schneider, T.; Schütze, A.; Bur, C. Comparison of Transfer Learning and Established Calibration Transfer Methods for Metal Oxide Semiconductor Gas Sensors. *Atmosphere* **2023**, *14*, 1123. <https://doi.org/10.3390/atmos14071123>

Academic Editor: Alexandra Monteiro

Received: 13 June 2023
Revised: 27 June 2023
Accepted: 5 July 2023
Published: 7 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As early as 2005, people spend up to 90% of their time indoors [1,2]. Since then, multiple studies have shown that indoor air quality is paramount for human health [2–4]. Within indoor air, volatile organic compounds (VOCs) can be harmful components that can cause severe health issues [3–5]. Contamination of only a few parts per billion (ppb) over an extended period with the most dangerous VOCs like formaldehyde or benzene can already have serious consequences [3,4]. However, since not every VOC is harmful (e.g., ethanol or isopropanol), the WHO sets the maximum allowed concentration and maximum exposure for every VOC separately. The difficulty with measuring VOCs in indoor air is that hundreds of different VOCs and many background gases (ppm range) are present and interfere with the measurement [4,6]. Therefore, selectively detecting single harmful VOCs at the relevant concentration levels (e.g., formaldehyde < 80 ppb [5]) in front of complex gas mixtures with a high temporal resolution is essential for advanced indoor air quality monitoring. Today the most common approach for indoor air quality assessments is to estimate indoor air quality based on the CO₂ concentration [7]. However, this does not

allow for the detection of single harmful VOCs since most of the time, a mixture of VOCs is emitted, and not all VOC sources emit CO₂ [3,8]. The current state-of-the-art systems capable of solving the task of being selective to multiple single harmful VOCs are GC-MS or PTR-MS systems. Unfortunately, those systems cannot provide the needed resolution in time (except PTR-MS), they require expert knowledge to operate, they require accurate calibration, and they are expensive. One popular alternative is gas sensors based on metal oxide semiconductor (MOS) material. They are inexpensive, easy to use, highly sensitive to various gases, and provide the needed resolution in time. However, they come with issues that prevent them from being even more widely used in different fields (e.g., breath analysis [9,10], outdoor air quality monitoring [11,12] or indoor air quality monitoring [13]). Those problems are that they need to be more selective to be able to detect specific gases [14]; they drift over time [15], making frequent recalibrations necessary (time and effort); and they suffer from large manufacturing tolerances [16], which has a significant effect on the sensor response. Some of those issues have already been addressed. The following publications have covered the problem regarding selectivity [17,18]. Moreover, in [19–21], drift over time was analyzed, and in [20,22], the calibration of those sensors when considering manufacturing tolerances was studied.

Compared to those studies, this work analyzes multiple methods that claim to reduce the needed calibration time. As a first approach, the initial calibration models trained on single sensors are tested regarding their ability to generalize to new sensors [23,24]. The methods used are either from classic machine learning like feature extraction, selection, and regression or advanced methods from deep learning. Afterward, calibration transfer methods are tested to improve those results with as few transfer samples/observations as possible (e.g., direct standardization and piecewise direct standardization [21,25]). Direct standardization and piecewise direct standardization are used to match the signal of different sensors in order to use the same model for various sensors. Thus, it is possible to eliminate the need for extensive calibration for new sensors. Direct standardization and piecewise direct standardization in their most basic forms were selected because they are easy to apply and can be used with any model since the input is adjusted. Furthermore, those methods showed superior performance over other transfer methods like orthogonal signal correction or Generalized Least Squares Weighting [25] if MOS gas sensors operated in temperature cycled operation were used. More advanced versions of those methods, like direct standardization based on SVM [22], are not used since the first comparison should be with the most basic method to achieve an appropriate reference. For future experiments, the comparison can be extended to more sophisticated techniques. As a different transfer method, baseline correction was specifically not used because the TCOCNN produces highly nonlinear models that might not be suitable for this approach. Similarly, adaptive modeling, as shown in [26], is not used because it is not suited for performing random cross-validation (no drift over time). However, in order to still take a wider variety of approaches into account, global models are built that take the calibration sensor and the new sensor into account. A more thorough review of the broad field of calibration transfer can be found in [27].

As the new method for calibration transfer, transfer learning is used to transfer an initial model to a new sensor [28,29]. Transfer learning was chosen as it showed excellent results in computer vision for a long time [30–33] and recently showed promising results for calibration transfer for dynamically operated gas sensors [28,29]. Furthermore, this method helps overcome the problem of extensive recalibration of sensors used in different conditions. Specifically, the benefit is that the new calibration can still rely on large datasets recorded previously but also be relatively specific to the new environment because of the retraining, which is more challenging to achieve in global modeling.

Afterward, the results are compared to analyze the benefit of the different calibration transfer methods.

Different methods and global modeling for initial model building are analyzed and compared to those in different articles. Furthermore, the calibration transfer between various sensors is studied. The gas chosen for this work is acetone, which is not as harmful as formaldehyde or benzene but provides the most detailed insight into the desired effects, as the initial models showed the most promising accuracy.

2. Materials and Methods

2.1. Dataset

The dataset used throughout this study was recorded with a custom gas mixing apparatus (GMA) [34–36]. The GMA allows us to offer precisely known gas mixtures to multiple sensors simultaneously. The latest version of the GMA can generate gas mixtures consisting of up to 14 different gases while also varying the relative humidity [37]. A specific gas mixture of predefined gas concentrations and relative humidity is called a unique gas mixture (UGM). Within this work, a unique gas mixture consists of zero air, two background gases (carbon monoxide, hydrogen), relative humidity, and eleven different VOCs, as illustrated in Figure 1. Since many different UGMs are required to build a regression model for a gas sensor, multiple UGMs are necessary. This dataset consists of 930 UGMs, randomly generated with the help of Latin hypercube sampling [38,39]. Latin hypercube sampling implies that each gas concentration and the relative humidity is sampled from a predefined distribution (in this case, uniformly distributed) such that the correlation between the independent targets is minimized. This prevents the model from predicting one target based on two or more others. This method has been proven functional in previous studies [39]. However, this process is extended with extended and reduced concentration ranges at low (0–50 ppb) and very high (e.g., 1000 ppb) concentrations. All concentration ranges can be found in Figure 1b. The range for the relative humidity spanned from 25% to 75%. A new Latin hypercube sampling was performed every time a specific range was adjusted. Moreover, because only one observation per UGM is not statistically significant, ten observations per UGM are recorded. However, the GMA has a time constant and the new UGMs could not be applied immediately, so five observations had to be discarded. Nevertheless, this resulted in 4650 observations for the calibration dataset.

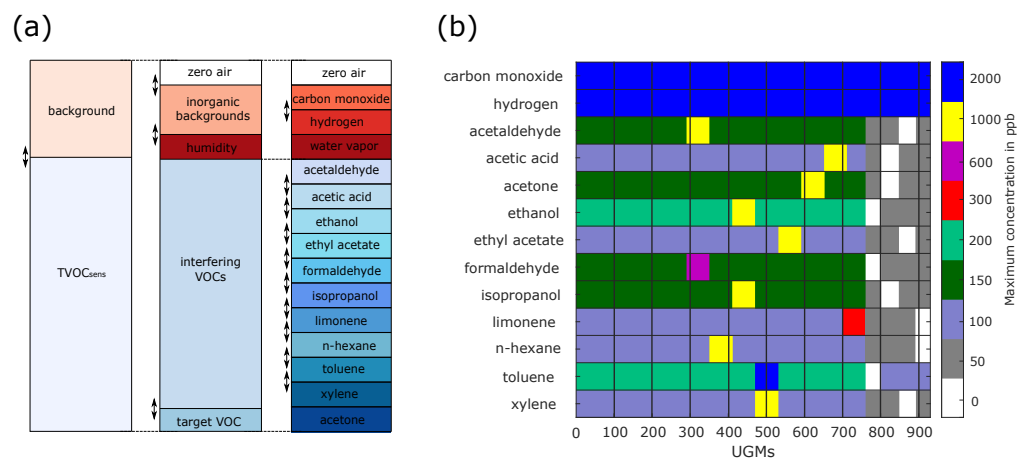


Figure 1. Overview of the gases included in the randomized calibration. Each UGM contains all of the shown gases. (a) The composition of the different UGMs (adapted from [29,39]). (b) All the maximum concentrations during recording. The lowest concentration for all VOCs during the measurement is 0 ppb; for carbon monoxide, 200 ppb, and for hydrogen, 400 ppb.

After discussing the UGMs applied to the different gas sensors, the next important part of the dataset is the sensor used and how the sensor is operated. The sensors used within this dataset are SGP40 sensors from Sensirion (Sensirion AG, Stäfa, Switzerland). Those sensors have four different gas-sensitive layers on four individual micro-hotplates. A non-disclosure agreement made it possible to operate the sensors in temperature cycled operation (TCO) [40]. Temperature cycled operation means that with the help of the micro-hotplates of the sensor, the independent gas-sensitive layers can be heated in specific temperature patterns during operation. One temperature cycle for sub-sensors 0–2 (gas-sensitive layer) consists of 24 phases. First, the sub-sensor is heated to 400 °C for 5 s, followed by a low-temperature phase at 100 °C for 7 s. This pattern is repeated twelve times in one full temperature cycle with increasing low-temperature phases (an increase of 25 °C per step). This leads to twelve high- and low-temperature steps, as illustrated in Figure 2. The temperature cycled operation for sub-sensor 3 is slightly different; here, the temperature cycle repeats the same high and low-temperature levels. The high temperature is always set to 300 °C, and the low temperature to 250 °C (cf. Figure 2). As described earlier, a temperature cycled operation was used to increase the selectivity of the different sensors. Therefore, the whole temperature cycle takes 144 s, resulting in 1440 samples (sample rate set to 10 Hz). The sensor response during one temperature cycled operation results in a matrix of 4×1440 and represents one observation. In total, the response of seven SGP40 sensors (S1–S7) for all UGMs is available for this study.

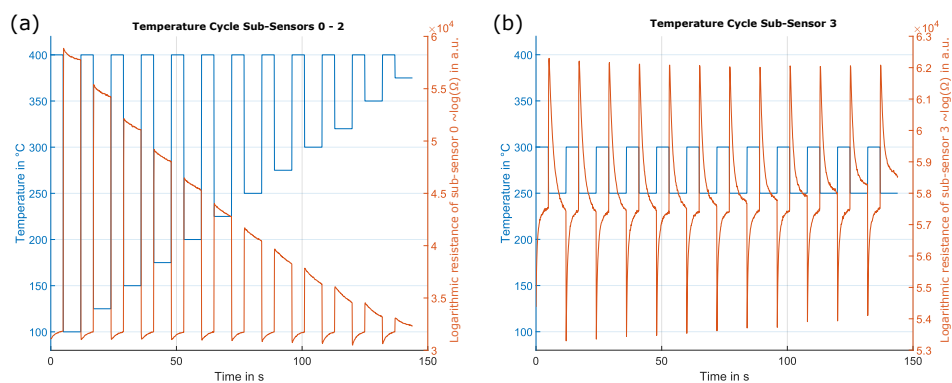


Figure 2. Sensor response of one SGP40 operated in temperature cycled operation. (a) Temperature cycle for sub-sensors 0–2 in blue with the corresponding sensor response of sub-sensor 0 in red. (b) Temperature cycle for sub-sensor 3 in blue with the corresponding sensor response of sub-sensor 3 in red (Reprinted with permission from Ref. [29], 2022, Y. Robin).

2.2. Model Building

In the first step, the calibration dataset is divided into 70% training, 10% validation, and 20% testing. A crucial point regarding the data split is that the splits are based on the UGMs rather than observations. In order to make the fairest comparison possible, this split is static across all different model-building methods and sensors throughout this study, which means that for every evaluation, the same UGMs are in either training, validation, or test set.

After the data split, two different methods for model-building are introduced. One model-building approach is feature extraction, selection, and regression (FESR), which was intensively studied earlier [13,40,41]. The other method, TCOCNN, was developed recently in [29,42] and has already proven to challenge the classic methods.

2.2.1. Feature Extraction, Selection, and Regression

The first machine learning approach introduced is feature extraction, selection, and regression (FESR). This method first extracts sub-sensor-wise features from the raw signal, selects the most important ones based on a metric, and then builds a regression model to predict the target gas concentration. The algorithm can learn the dependencies between raw input and target gas concentration during training. If multiple SGP40 sensors are used for training, the input size of the model does not change. Instead, the model only gets more observations to learn.

This study uses the adaptive linear approximation as a feature extraction method [43]. Although the algorithm can identify the optimal number of splits, this time, the algorithm is forced to make exactly 49 splits for each sub-sensor independently, which ensures that every temperature step can be accurately reconstructed. The position of the optimal 49 splits is determined by the reconstruction error, as described in [44], cf. Figure 3. The mean and slopes are calculated on each resulting segment. Since there are four sub-sensors and 50 segments each, this results in 400 features per observation.

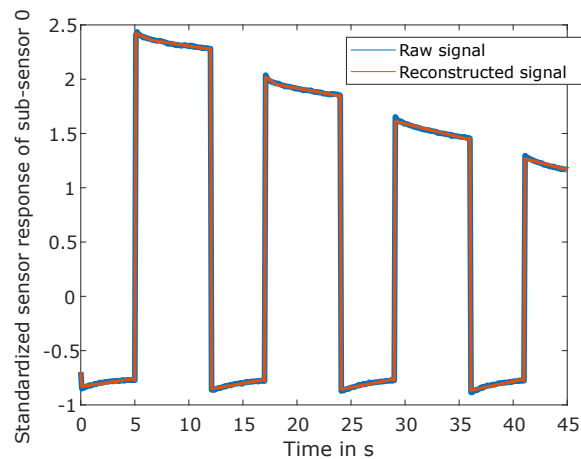


Figure 3. Raw signal in blue together with the reconstructed signal in red from features extracted from adaptive linear approximation for sub-sensor 0. Only a section (0 s–45 s) of one temperature cycle is illustrated for better visibility, and only the signal of sub-sensor 0 is shown.

Afterward, features are selected based on their Pearson correlation to the target gas to reduce the number of features to the most essential 200. After that, a partial least squares regression (PLSR) [45] with a maximal number of 100 components was trained on 1–200 Pearson-selected features in a 10-fold cross-validation based on training and validation data to identify the best feature set. Finally, another PLSR was trained with the best feature set on training and validation data to build the final model. This combination of methods achieves reasonable results, as reported earlier [46].

2.2.2. Deep Learning: TCOCNN

The TCOCNN is a convolutional neural network [42,47] specifically tailored for MOS gas sensors operated in temperature cycled operation. Figure 4 gives an example of the network. The TCOCNN takes as an input a 4×1440 matrix. Four represents the number of sub-sensors per gas sensor, and 1440 is the number of sample points in the temperature cycles.

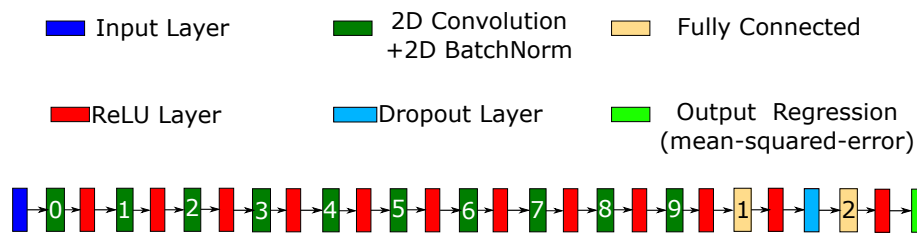


Figure 4. Neural network architecture of the TCOCNN (adapted from [48]). One example configuration with ten convolutional layers (later optimized) (Reprinted with permission from Ref. [29]. 2022, Y. Robin).

The network consists of multiple hyperparameters that can be tuned with the help of the training data, the validation data, and a neural architecture search. The hyperparameters adjusted within this study are the kernel width (10–100) of the first two convolutional layers, the striding size (10–100) of the first two convolutional layers, the number of filters in the first layer (80–150), the depth of the neural network (4–10; including the last two fully connected layers), the dropout rate during training (10–50%), the number of neurons in the fully connected layer (500–2500), and the initial learning rate. A more detailed explanation of the neural architecture search based on Bayesian optimization can be found in [42,49,50]. In order to optimize the hyperparameter, the default setup for the Bayesian optimization of Matlab is used for 50 trials. The optimization of the validation error is conducted only once with sensor 1. Afterward, the same parameters are used throughout the study, and all further tests are performed on the test data, which prevents the results from overfitting. The parameters found for this study are listed in Table 1. The derived parameters are given as follows: the number of filters is doubled every second convolutional layer; the striding size after the first two convolutional layers follows the pattern 1×2 then 1×1 , and the same is applied for the kernel size; and finally, the initial learning rate decays after every second epoch by a factor of 0.9.

Table 1. Values of all hyperparameters. The number of filters, striding size, and kernel size concern the first two layers, the number of neurons concerns the second to last fully connected layer, and the number of layers includes the convolutional layer and the last two fully connected layers.

Filters	Striding Size	Kernel Size	Layer	Number of Neurons	Initial Learning Rate	Dropout Rate
83	34	63	8	1312	$4.3 \cdot 10^{-4}$	13.83%

2.3. Calibration Transfer

Because of manufacturing tolerances, the responses of two sensors (same model) will always show different responses [51]. Therefore, calibration of every sensor is necessary to predict the target gas concentration. In our case, this calibration was carried out with the data recorded under laboratory conditions. However, many calibration samples are necessary before a suitable calibration is reached. Therefore, the idea is to reuse the calibration models of different sensors instead of building a new one every time (calibration transfer) [22,52,53]. The goal is to significantly reduce the number of samples needed for calibration.

The calibration transfer is usually performed based on a few transfer UGMs. In order to make the comparison as fair as possible, the transfer samples are always the same for every evaluation. However, they are chosen randomly (but static) from all available training and validation UGMs.

2.3.1. Signal Correction Algorithms

As described above, the goal is to use the same model for different sensors to reduce calibration time. However, because the differences between sensors are usually too sig-

nificant, it is impossible to use the same model immediately. One common approach is to match the signal of the new sensor to the sensors seen during training [21,25,27]. The sensor (or sensors) used for building the initial model is called the master sensor, and the new sensor, which is adapted to resemble the master sensor (or sensors), is called the slave. In the matching process, the signal of the slave sensor is corrected to resemble the signal of the master. This is usually done by taking multiple samples (transfer samples) where the master and slave sensors are under the exact same conditions and then calculating a correction matrix (C) that can be used to transform the slave signal to match that of the master also under different conditions.

Direct standardization is one of the most common methods used for calibration transfer in gas sensor applications [21,25,27]. The correction matrix is calculated for direct standardization, as shown in Equation (1) [25,54,55].

$$C = R_S^+ \cdot R_M \quad (1)$$

Here, C represents the correction Matrix, R_S^+ stands for the pseudoinverse of the response matrix of the slave sensor, and R_M resembles the response matrix of the master sensor. The response matrices are of the shape $\mathbb{R}^{n \times m}$, and n resembles the number of samples needed to apply for calibration transfer (e.g., 25 observations or 5 UGMs), and m stands for the length of one observation, e.g., 1440 for one sub-sensor. Therefore, the resulting Matrix C is of the size $\mathbb{R}^{m \times m}$ and is applied to new samples as given in Equation (2).

$$R_{S;C} = C \cdot R_S \quad (2)$$

Since the SGP40 consists of multiple sub-sensors, this approach is used for each sub-sensor independently. However, suppose various sensors (multiple SGP40) are used as the master sensors for signal correction. In that case, the slave responses are repeatedly stacked, and the different master sensors (all under the same condition) are stacked into one tall matrix.

As an example, the responses of two master sensors and one slave sensor under the same condition led to the correction matrix given in Equation (3).

$$C = \begin{bmatrix} R_S \\ R_S \end{bmatrix}^+ \cdot \begin{bmatrix} R_{M1} \\ R_{M2} \end{bmatrix} \quad (3)$$

The drawback of this method is that the construction of C requires the pseudoinverse of the response matrix, and the number of available transfer samples determines the quality. Since this study aims to reduce the number of transfer samples as much as possible, another signal correction algorithm is introduced. Piecewise direct standardization [55] uses the same approach as direct standardization, but the C parameter is calculated for small subsections of the raw signal. This means that before piecewise direct standardization (PDS) is applied, the signal is divided into z segments of length p .

Therefore, C can be calculated as shown in Equation (4) on small segments of length p .

$$C_p = R_{S;p \times n}^+ \cdot R_{M;n \times p} \quad (4)$$

C_p has the shape $\mathbb{R}^{p \times p}$, and the final C matrix is calculated by assembling those smaller Cs (total z different Cs) on the diagonal. This means that C for a small segment of length p is calculated based on Equation (5).

$$C = \begin{bmatrix} C_{p1} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & & 0 \\ 0 & \cdots & 0 & C_{pz} \end{bmatrix} \quad (5)$$

The final C is again of the shape $\mathbb{R}^{m \times m}$ and can be used as before. However, this leads to the conclusion that piecewise direct standardization has one hyperparameter that can be tuned. For this study, p is chosen to be 10. This was defined by testing a calibration model with one master and one slave sensor on multiple different window sizes (also two windows of different possible sizes) and choosing the window size with the smaller RMSE, as listed in Table 2.

Table 2. RMSE values for different window sizes for the piecewise direct standardization. Piecewise direct standardization was performed with five transfer samples. The RMSE was achieved by training the model with data from one master sensor, and testing was performed on the adapted data of the slave sensor. Entry 50;70 represents alternating window sizes to precisely cover the TCO shape.

Window width	5	10	20	50;70
RMSE in ppb TCOCNN	28.3	26.3	43.8	59.1
RMSE in ppb FESR	47.9	55.4	123.6	209.0

Although piecewise direct standardization is expected to achieve better results [25] as the calculation of C is more robust than direct standardization, both approaches are analyzed in this study. This is reasonable, as indicated by Figure 5, which illustrates the original signal of the master and slave sensor, together with the adapted (corrected) signal and the differential signal. Although the purple line (corrected signal PDS) follows the master signal more precisely, it is possible to spot small jumps that might influence the prediction quality. This is not visible for direct standardization, but in this case, the corrected signal is further apart from the master signal, especially when analyzing the peaks in the differential signal.

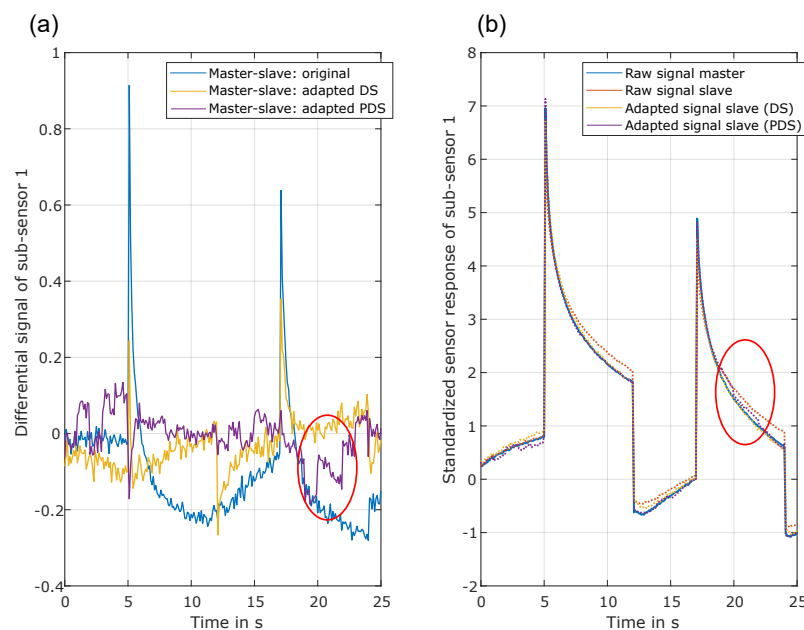


Figure 5. (a) Differential signal between original and adapted signal. (b) Sensor response of the master sensor, the initial sensor response from the slave sensor, and the adapted signal from the slave sensor (DS and PDS). Only a section (0 s–25 s) of one TC is shown for better visibility, and only the signal of sub-sensor 1 is shown.

A significant benefit of signal correction methods is that they are independent of the used model and can be applied to the FESR approach and the TCOCNN.

2.3.2. Transfer Learning for Deep Learning

Compared to the signal correction methods, the transfer learning method for deep learning can only be applied to the TCOCNN. This method adjusts the whole model to the new sensor instead of correcting the raw signal of the new (slave) sensor. Transfer learning is a common approach in deep learning, especially in computer vision [31–33]. Multiple works have shown that this approach can significantly reduce errors and speed up training [33,56]. In previous studies, it was demonstrated that transfer learning could also be used to transfer a model trained on gas sensor data based on many calibration samples to a different sensor with relatively few transfer samples [28,29,53] (calibration sample reduction by up to 97% (700 UGMs–20 UGMs)). An essential extension to previous studies is that the initial model is built with the help of multiple sensors, which should increase the performance even more.

The idea is illustrated in Figure 6. While the blue line resembles a model trained from scratch, the other two show the expected benefit when adjusting (retraining) an already working model to a new sensor. The modified model needs much fewer UGMs to get to a relatively low RMSE, and the improvement is much steeper. The hyperparameter to tune transfer learning is typically the learning rate. All hyperparameters of the TCOCNN are the same as before, and only the initial learning rate is set to the learning rate typically reached halfway through the training process. Of course, it would also be possible to tune this process with the help of Bayesian optimization to achieve even better results. However, this was not tested in this study, and the optimal value obtained in other studies is used [29].

A significant benefit compared to signal correction methods is that for this approach, the transfer can happen even if the sensors were never under the same condition, which makes even a transfer between datasets possible.

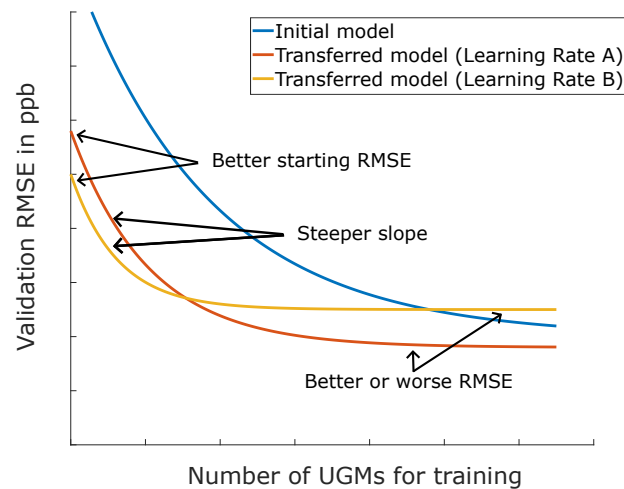


Figure 6. The effect of transfer learning for different hyperparameters (Reprinted with permission from Ref. [29]. 2022, Y. Robin).

2.4. Evaluation

After introducing the general methods used throughout this study, this section introduces the techniques to benchmark the different methods.

The first part will evaluate the performance of the FESR and TCOCNN approach regarding their ability to predict the target gas concentration. This will be done by using multiple sensors to build the models. The training and validation data of one to six sensors will be used to train six FESR and six TCOCNN models (increasing the number of sensors). Afterward, the models will be tested on the corresponding sensors' test data. This will then be used as a baseline for all further evaluations.

In the next step, the performance of a model trained with each of the available six sensors (trained independently) is tested with the test data of sensor 7. This is done to test the generalizability of a model trained with one sensor and tested with another sensor. Afterward, the models are trained on one to six sensors (same as baseline models), and after that, the generalizability is tested with the test data of sensor 7.

The last part then focuses on methods to improve generalizability. Therefore, multiple methods from the field of calibration transfer will be used. The initial models are again built with the training and validation data of sensors 1–6. This results in twelve initial models, which are used to test transfer learning, direct standardization, and piecewise direct standardization (six FESR models, and six TCOCNNs). After the initial models are built, transfer learning and the signal correction algorithms are applied as explained above with 5, 25, 100, and 600 transfer UGMs. In order to have a more sophisticated comparison, a global model is also trained on 1–6 sensors plus the transfer samples. This means the transfer data are already available during initial training to determine if that also improves the generalizability. Those results then allow a general comparison of the most promising methods.

For comparing the different methods, the root mean squared error (RMSE) is used as the metric to rate the performance of the various models. Also, other methods like R-squared, mean absolute error (MAE), or mean absolute percentage error (MAPE) can be used. However, the main goal in indoor air quality monitoring is to know if a certain threshold is exceeded and how far the estimation can be off the target value to account for a margin. Therefore, the RMSE as an interpretable metric is used. Furthermore, this study should mainly focus on the prediction quality's general trend rather than analyzing every aspect of the regression model. At the beginning of the results section, a scatter plot illustrating the target vs. the predicted value is shown, and the r-squared values are given to prove that the models work as intended.

As a final remark, the evaluations with the TCOCNN are repeated five times to consider the model's uncertainty.

3. Results

As described above, the first step is to create a baseline to interpret the following results. Figure 7 shows the results when training the initial model with 1–6 sensors (744 UGMs per sensor). For any number of sensors, the TCOCNN outperforms FESR. With an increasing number of sensors used to build the model, the RMSE value decreases for the TCONN, while it increases for FESR. This means the model can generalize and find a better model with more data from multiple sensors. The reason for the TCOCNN outperforming the FESR approach might be the more advanced feature extraction compared to the static extraction of the FESR. In order to give the RMSE values more context, the prediction on the test data for the FESR and TCOCNN models are shown in Figure 7b. There, it can be seen that despite the worse RMSE, the FESR approach still shows a suitable relationship between target and prediction (r -squared > 0.96). However, it must be mentioned that at high concentrations, the accuracy worsens for both models. This is because this region has fewer data points (extended concentration range). Nevertheless, this is not a problem since the threshold for the target gases is usually at smaller concentrations (more data points). It is essential to be very precise in lower regions, and beyond that point, it is sufficient to identify that the threshold is exceeded. Therefore, an RMSE of around 25 ppb can still be interpreted as a suitable model since the error is in an acceptable range, and the correlation is always (also for the upcoming results) clearly visible, like in Figure 7b (r -squared > 0.96).

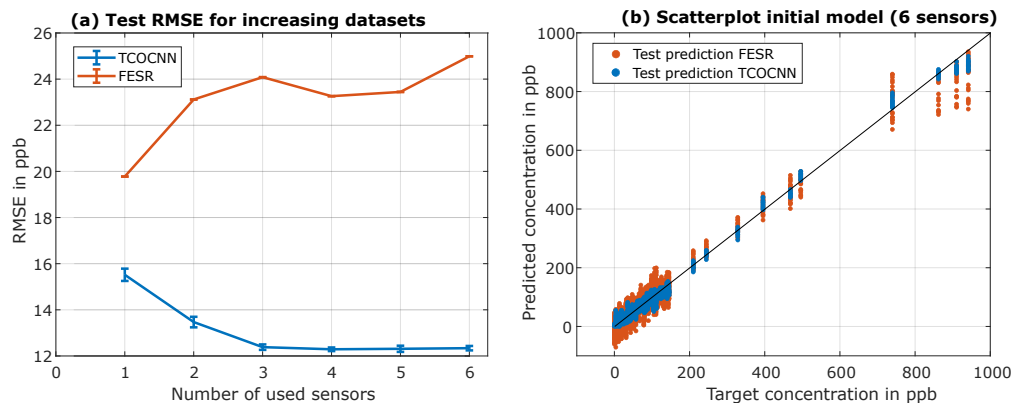


Figure 7. Achieved test RMSE values for the initial model trained with a different number of master sensors (1–6) and tested on the test data of the corresponding sensors. (a) RMSE over the number of sensors used for training and testing. (b) Scatter plot to illustrate target vs. prediction.

After analyzing the performance of the initial model on the test data of the corresponding sensors, the next step is to test the initial model on the test data of a completely different sensor. The first evaluation is carried out by training an independent model with one sensor each and testing the performance on the test data of sensor 7. The results are depicted in Figure 8a. It can be seen that it strongly depends on the sensor if the TCOCNN or FESR approach can find a general model to apply to multiple sensors. For example, the TCOCNN for sensor 6 achieves good results with sensor 7, while the model for sensor 1 applied to sensor 7 does not work. As seen by sensor 3, this also depends on the evaluation method. For example, sensors 3 and 7 are deemed similar by the FESR approach, while the TCOCNN indicates differently. This might be because both ways rely on different features. While the TCOCNN generates features independently, the FESR approach has fixed features based on the adaptive linear approximation. Since the scope of this article is not to highlight the different features used within the various methods, this will not be discussed in more detail. However, it was already shown in [51] that different methods are available (e.g., occlusion map) to identify the different feature sets used by the methods, depending on the sensor. Nevertheless, this does not mean that a model that is useful for multiple sensors can be applied to all SPG40 sensors—only to those similar. Therefore, Figure 8b illustrates the results that can be achieved with the initial models when trained with 1–6 sensors simultaneously. It can be seen that with increasing sensors, the TCOCNN model generalizes more and can be applied more successfully to sensor 7. However, the improvement does not directly correlate with the independent performance (Figure 8), which might be because the model needs to generalize more to suit all sensors, which then generalizes too much and causes the performance to drop (e.g., the TCOCNN with sensor 5).

However, the model trained with six sensors achieves an RMSE of 31 ppb, close to the suitable RMSE of 25 ppb from the baseline of the FESR method. In comparison, the TCOCNN achieves almost acceptable results without calibration transfer, while the FESR approach trained with multiple sensors struggles. Though the RMSE also generally shrinks in the case of sensor 7 when more sensors are used for training with the FESR approach, the results are worse than those of the TCOCNN. This can have multiple reasons. One reason could be that the approach of adaptive linear approximation, Pearson selection, and PLSR are not optimal for this task. A more sophisticated FESR approach based on a more sophisticated feature extraction and recursive feature elimination least squares as a feature selection might yield more promising results. However, because of the limited performance of the FESR approach for this specific setup in the baseline and the initial model

building, the remaining results will only cover the results achieved with the TCOCNN. The results of the FESR approach are listed in Appendix A.

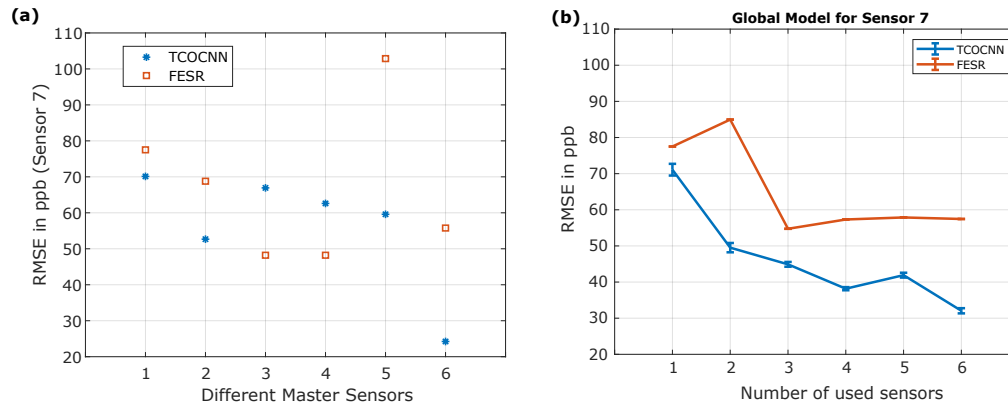


Figure 8. Achieved RMSE values for the TCOCNN and FESR approach if data from sensor seven are tested without any transfer. (a) Only one sensor is used to build the initial model. (b) Different number of master sensors are used to build the initial model.

After discussing the capability of the different machine learning methods to generalize across sensors, the next step is to evaluate the signal correction methods, transfer learning, and global model building (all available data used for training). Figure 9 depicts the results achieved with different initial models (built with 1–6 sensors) on the x-axis of each sub-figure, and it also shows the effect of the different number of transfer UGMs. In Figure 9a (five transfer UGMs), it can be seen that direct standardization does not achieve any reasonable results, which might be correlated with the problem of not having enough transfer samples to invert the matrix. As expected, the piecewise direct standardization performs much better as, from theory, the pseudoinverse should be much more manageable to calculate. However, the best method, in this case, is the transfer learning approach. While this approach does not perform exceptionally well if only one sensor is used to build the initial model, with six sensors for the initial model, an RMSE of 17.7 ppb can be achieved, which is better than the FESR baseline cf. Figure 9. That would mean a suitable model was created with only 5 UGMs (instead of 744). The reason that transfer learning can outperform the other method might be because of the advanced feature extraction that generalizes well across sensors and because only small adjustments inside the model are necessary. Similar but less impressive results can be observed for global model building and piecewise direct standardization (six sensors for the initial model); there, a reasonable RMSE of 24.3 ppb was achieved (again, smaller than the baseline FESR). The slightly worse performance compared to transfer learning can be attributed to the nonspecific model. While transfer learning generates a specific model for the new sensor, the global approach tries to find a model to fit all. Figure 9b (25 UGMs for transfer) indicates that if enough transfer samples are available, direct standardization can perform much better than piecewise direct standardization and achieves results similar to transfer learning. This might be because the pseudo inverse can now be calculated appropriately. However, with six sensors for the initial model, each method achieves an RMSE below 25 ppb, which is again better than the FESR approach's baseline, which indicates that all methods are suitable. Nevertheless, the best performance is again shown by transfer learning.

The two sub-figures at the bottom show the benefit of more transfer samples. Figure 9c (125 transfer samples) indicates that direct standardization and transfer learning perform similarly for this case, and that piecewise direct standardization does not improve significantly. Furthermore, global modeling and transfer learning has become ever so close. Moreover, it can be derived that the amount of transfer samples is now always sufficient

for the pseudo inverse of direct standardization. While 25 UGMs with one sensor is almost insufficient, the improvement between one and two sensors for 125 UGMs is much smaller. Figure 9d then concentrates on the results if 600 transfer samples (almost all training samples) are used. Global modeling and transfer learning perform more or less similar and now even achieve results smaller than the baseline of the TCOCNN from earlier, which was 12.1 ppb. This aligns with the baseline results of the TCOCNN as the RMSE also dropped by adding more sensors. Furthermore, more transfer samples do not improve the direct standardization and piecewise direct standardization results. This might be because it does not help to make the slave sensor more similar to the master sensors anymore (as already seen for 125 UGMs).

Since the sensor manufacturers are most interested in significantly reducing calibration time, the most suitable method seems to be transfer learning, as this method achieves a reduction in calibration UGMs of 99.3%. For small transfer sets, piecewise direct standardization and global model building also achieve good results. However, it has to be noted that global model building outperforms transfer learning and piecewise direct standardization regarding small initial datasets.

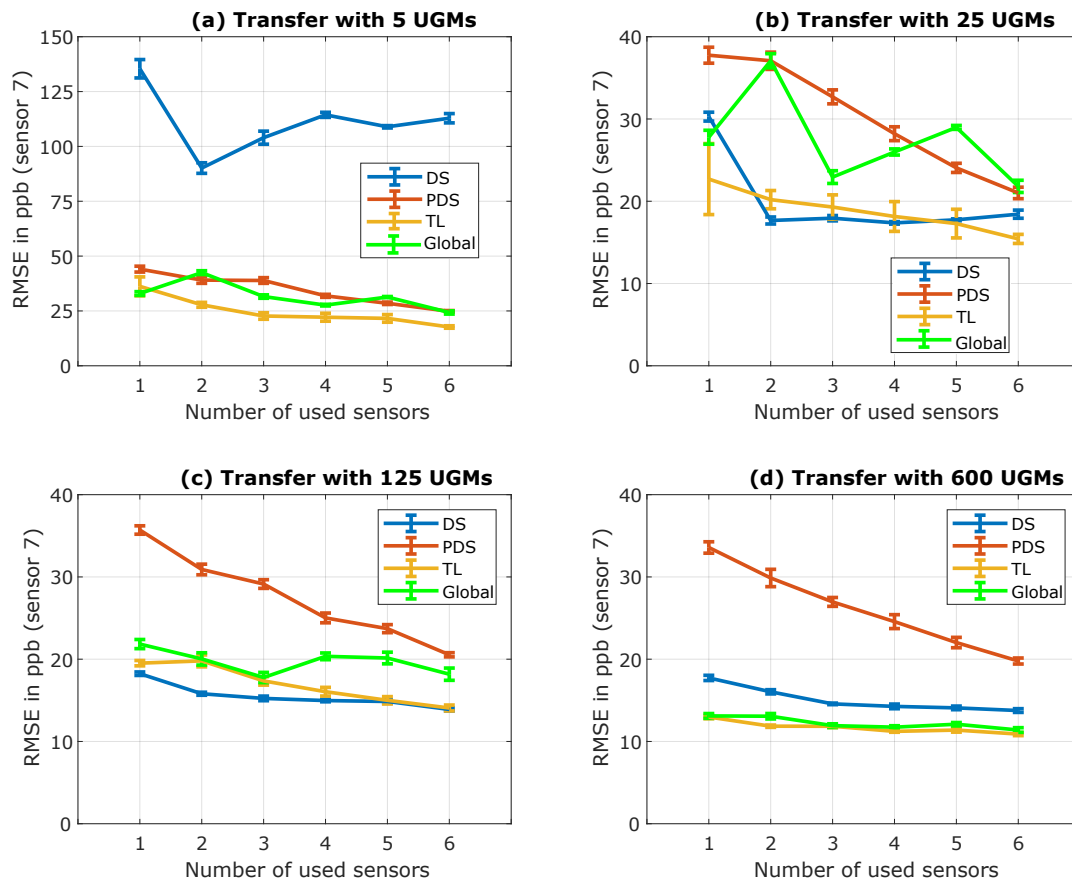


Figure 9. Comparison of direct standardization (DS), piecewise direct standardization (PDS), transfer learning for deep learning (TL), and global model building concerning the TCOCNN. Different numbers of UGMs for transfer learning are used in the different sub-plots.

To emphasize the benefit of transfer learning compared to global modeling, Figure 10 illustrates the side-by-side comparison of both approaches over the different number of transfer UGMs regarding an initial model built with one sensor, and one where the initial model was constructed with all six sensors. The most important part is in relation to the five transfer UGMs. While the benefit of transfer learning compared to global model building is not apparent when the initial model is built with only one sensor, the effect can be seen when six sensors are used. Figure 10b indicates that transfer learning shows its full potential when trained with more sensors. While global model building achieves an RMSE of only 24.9 ppb, transfer learning can get as low as 17.7 ppb. This is in accordance with the theory that a model trained simultaneously with the initial and transfer data cannot adapt to the new sensor like the specifically tailored model obtained by transfer learning.

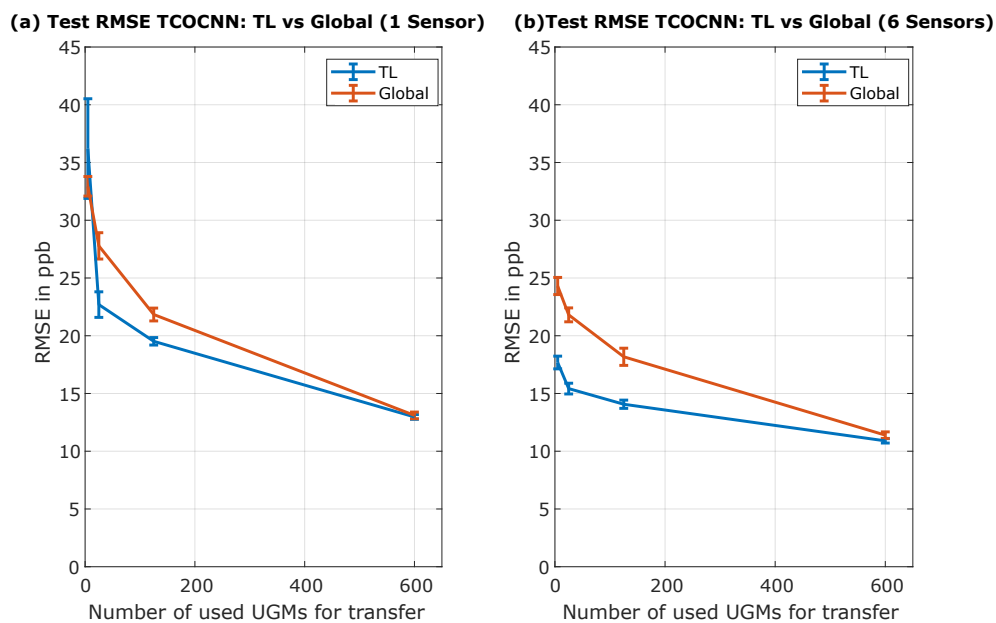


Figure 10. Comparison of transfer learning (in blue), and global model building (in red) with respect to the TCOCNN. (a) shows the results if only one sensor is used to build the initial model. (b) shows the results if six sensors are used to build the initial model.

After showing that transfer learning is a very promising method to reduce the calibration time significantly, it can be seen in Appendix A that for the FESR approach, the same phenomena as explained above can be observed. However, the results of the FESR approach are not as good as those of the TCOCNN since the baseline is worse. Furthermore, it seems that the FESR approach does not work well with piecewise direct standardization, possibly because of the small edges in the adapted signal.

4. Discussion

After analyzing the baseline results and the calibration transfer methods, the TCOCNN shows the most promising result when it comes to generalizability. Furthermore, it was shown that, especially with the TCOCNN, using multiple sensors for the initial model building could be beneficial. Even without calibration transfer methods, applying a model trained with six sensors to a new sensor was possible, and suitable results of around 32 ppb were achieved. This can be attributed to the more flexible feature extraction of the TCOCNN, which allows for better generalizability. Furthermore, different sensors' effects on the initial model building were investigated. Here, it was shown that it makes a

significant difference which sensors are used to build the initial model. It was shown that when only one sensor is used for model building, the results can differ by up to 45 ppb concerning the new sensor (for difference in the sensor response, see [51]). This might be interesting to investigate in future experiments. Nevertheless, it was shown that the most effective way to achieve the lowest RMSE values possible is to use calibration transfer. Transfer learning proved to be the best option since this method outperformed every other approach when the initial model was trained with many sensors and only a few transfer samples were available. It was shown that with less than 99.3% of the calibration UGMs, results of 18 ppb are still possible (better than the FESR baseline). Compared to the other methods, the exceptional performance can be attributed to the specifically retrained network. However, the other methods showed decent results as well. As expected, piecewise direct standardization performs well for minimal transfer sets and can even outperform direct standardization since the calculation of the pseudo-inverse is more straightforward. Direct standardization showed the full potential if 25 transfer UGMs were available (manageable pseudo inverse) and surpassed transfer learning if smaller initial datasets were investigated. Global model building performed very similarly, although transfer learning outperformed global model building significantly when large initial and small transfer datasets were concerned. This might be because a more general model is appropriate for this task. Moreover, the calibration methods for signal correction and global model building also worked for the FESR approach, although further improvements need to be made to be compatible with transfer learning.

5. Conclusions

This study's results allow the conclusion that transfer learning is a powerful method to reduce the calibration time by up to 99.3%. It was shown that transfer learning could outperform the other techniques, especially with small transfer sets and initial models trained on multiple sensors. Furthermore, it was shown that the other calibration transfer methods are comparable, especially for the most important case of 5 transfer UGMs. Piecewise direct standardization or global model building with many sensors for initial model building also achieved decent results with 5 UGMs for transfer (24.3 ppb). In comparison, direct standardization needed at least 25 transfer UGMs. The FESR approach did not show optimal results, but this might be possible if a method combination is found that is more tailored to calibration transfer. This would be beneficial because the computational effort would be much smaller.

For further research, it would be exciting to see how the TCOCNN performs in combination with (piecewise) direct standardization and transfer learning. Furthermore, it was not investigated if something similar is possible if two different datasets with different gases (same target gas) are used. One interesting extension of this work is to analyze how the models differ (explainable AI) when using multiple sensors and whether it is possible to generate FESR methods based on insights gained with techniques from explainable AI. It is also possible to build an error model based on multiple sensors' raw signals to apply data augmentation and further improve the results. It should also be determined in future work if transfer learning can be used to compensate for drift. Furthermore, this study only covers the specific case of indoor air quality monitoring. Future research should also extend this approach to breath analysis, outdoor air quality monitoring, and other sensor calibration tasks.

Author Contributions: Conceptualization, Y.R., C.B. and A.S.; methodology, Y.R. and J.A.; software, Y.R.; validation, Y.R.; formal analysis, Y.R.; investigation, Y.R.; resources, A.S.; data curation, J.A.; writing—original draft preparation, Y.R.; writing—review and editing, Y.R., C.B., T.S. and A.S.; visualization, Y.R.; supervision, Y.R., C.B., T.S. and A.S.; project administration, A.S. All authors have read and agreed to the published version of the manuscript.

Funding: We acknowledge support by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) and Saarland University within the funding program Open Access Publishing. Part of this research was performed within the project “VOC4IAQ” funded by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) through the program Industrial Collective Research (AiF-iGF) under the grant number 22084N/1.

Data Availability Statement: Data and code are available on request.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- AI artificial intelligence
- CNN Convolutional Neural Network
- DS direct standardization
- PDS piecewise direct standardization
- FE Feature Extraction
- FESR Feature Extraction Selection Regression
- FS Feature Selection
- IAQ Indoor Air Quality
- MDPI Multidisciplinary Digital Publishing Institute
- ML Machine Learning
- MOS Metal Oxide Semiconductor
- PLSR Partial Least Squares Regression
- RH Relative Humidity
- RMSE Root Mean Square Error
- TCO Temperature Cycled Operation
- UGM unique gas mixtures
- VOC Volatile Organic Compounds

Appendix A

FESR

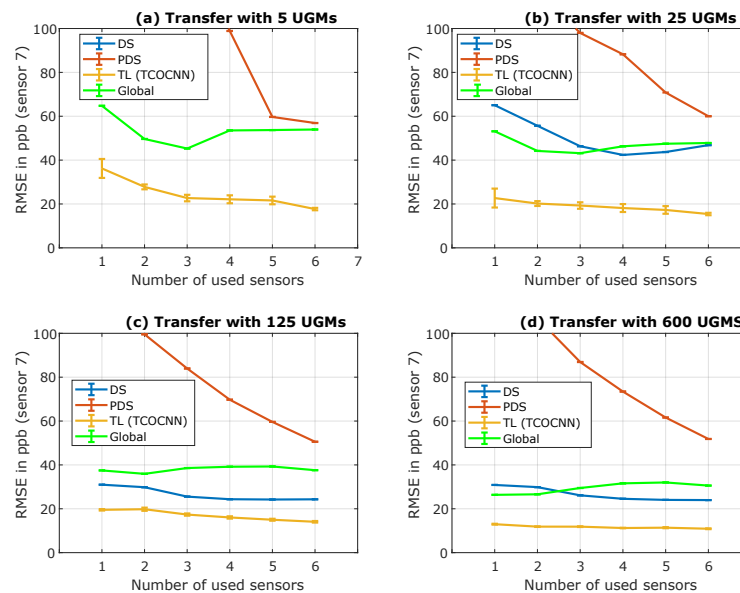


Figure A1. Comparison of direct standardization (DS), piecewise direct standardization (PDS), transfer learning for deep learning (TL), and global model building concerning FESR. Different numbers of UGMs for transfer learning are used in the different sub-plots.

References

1. Brasche, S.; Bischof, W. Daily time spent indoors in German homes—Baseline data for the assessment of indoor exposure of German occupants. *Int. J. Hyg. Environ. Health* **2005**, *208*, 247–253. [CrossRef]
2. United States Environmental Protection Agency. Indoor Air Quality. 2021. Available online: <https://www.epa.gov/report-environment/indoor-air-quality> (accessed on 15 November 2022).
3. Hauptmann, M.; Lubin, J.H.; Stewart, P.A.; Hayes, R.B.; Blair, A. Mortality from Solid Cancers among Workers in Formaldehyde Industries. *Am. J. Epidemiol.* **2004**, *159*, 1117–1130. [CrossRef]
4. Sarigiannis, D.A.; Karakitsios, S.P.; Gotti, A.; Liakos, I.L.; Katsoyiannis, A. Exposure to major volatile organic compounds and carbonyls in European indoor environments and associated health risk. *Environ. Int.* **2011**, *37*, 743–765. [CrossRef]
5. WHO Regional Office for Europe. *WHO Guidelines for Indoor Air Quality: Selected Pollutants*; World Health Organization: Copenhagen, Denmark, 2010. [CrossRef]
6. Salthammer, T. Very volatile organic compounds: An understudied class of indoor air pollutants. *Indoor Air* **2014**, *26*, 25–38. [CrossRef]
7. Pettenkofer, M. *Über den Luftwechsel in Wohngebäuden*; Literarisch-Artistische Anstalt der J.G. Cotta'schen Buchhandlung: Munich, Germany, 1858.
8. Mølhave, L. Indoor air pollution due to organic gases and vapours of solvents in building materials. *Environ. Int.* **1982**, *8*, 117–127. [CrossRef]
9. Marzorati, D.; Mainardi, L.; Sedda, G.; Gasparri, R.; Spaggiari, L.; Cerveri, P. MOS Sensors Array for the Discrimination of Lung Cancer and At-Risk Subjects with Exhaled Breath Analysis. *Chemosensors* **2021**, *9*, 209. [CrossRef]
10. Dong, H.; Qian, L.; Cui, Y.; Zheng, X.; Cheng, C.; Cao, Q.; Xu, F.; Wang, J.; Chen, X.; Wang, D. Online Accurate Detection of Breath Acetone Using Metal Oxide Semiconductor Gas Sensor and Diffusive Gas Separation. *Front. Bioeng. Biotechnol.* **2022**, *10*, 861950. [CrossRef]
11. Sofia, D.; Giuliano, A.; Gioiella, F.; Barletta, D.; Poletto, M. Modeling of an air quality monitoring network with high space-time resolution. In *Computer Aided Chemical Engineering*; Elsevier: Amsterdam, The Netherlands, 2018; pp. 193–198. [CrossRef]
12. Lotrecchiano, N.; Sofia, D.; Giuliano, A.; Barletta, D.; Poletto, M. Pollution Dispersion from a Fire Using a Gaussian Plume Model. *Int. J. Saf. Secur. Eng.* **2020**, *10*, 431–439. [CrossRef]
13. Baur, T.; Amann, J.; Schultealbert, C.; Schütze, A. Field Study of Metal Oxide Semiconductor Gas Sensors in Temperature Cycled Operation for Selective VOC Monitoring in Indoor Air. *Atmosphere* **2021**, *12*, 647. [CrossRef]
14. Goel, N.; Kunal, K.; Kushwaha, A.; Kumar, M. Metal oxide semiconductors for gas sensing. *Eng. Rep.* **2022**, *5*, e12604. [CrossRef]
15. Müller, G.; Sberveglieri, G. Origin of Baseline Drift in Metal Oxide Gas Sensors: Effects of Bulk Equilibration. *Chemosensors* **2022**, *10*, 171. [CrossRef]
16. Krutzler, C.; Unger, A.; Marhold, H.; Fricke, T.; Conrad, T.; Schütze, A. Influence of MOS Gas-Sensor Production Tolerances on Pattern Recognition Techniques in Electronic Noses. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 276–283. [CrossRef]
17. Schütze, A.; Baur, T.; Leidinger, M.; Reimringer, W.; Jung, R.; Conrad, T.; Sauerwald, T. Highly Sensitive and Selective VOC Sensor Systems Based on Semiconductor Gas Sensors: How to? *Environments* **2017**, *4*, 20. [CrossRef]
18. Schütze, A.; Sauerwald, T. Dynamic operation of semiconductor sensors. In *Semiconductor Gas Sensors*, 2nd ed.; Jaaniso, R., Tan, O.K., Eds.; Woodhead Publishing: Amsterdam, The Netherlands, 2020; pp. 385–412. [CrossRef]
19. Artursson, T.; Eklöv, T.; Lundström, I.; Martensson, P.; Sjöström, M.; Holmberg, M. Drift correction for gas sensors using multivariate methods. *J. Chemom.* **2000**, *14*, 711–723. [CrossRef]
20. Bur, C.; Engel, M.; Horras, S.; Schütze, A. Drift compensation of virtual multisensor systems based on extended calibration. In Proceedings of the IMCS2014—the 15th International Meeting on Chemical Sensors (Poster Presentation), Buenos Aires, Argentina, 16–19 March 2014.
21. Fonollosa, J.; Fernández, L.; Gutiérrez-Gálvez, A.; Huerta, R.; Marco, S. Calibration transfer and drift counteraction in chemical sensor arrays using Direct Standardization. *Sens. Actuators B Chem.* **2016**, *236*, 1044–1053. [CrossRef]
22. Laref, R.; Losson, E.; Sava, A.; Siadat, M. Calibration Transfer to Address the Long Term Drift of Gas Sensors for in Field NO₂ Monitoring. In Proceedings of the 2021 International Conference on Control, Automation and Diagnosis (ICCAD), Grenoble, France, 3–5 November 2021; IEEE: Piscataway, NJ, USA, 2021. [CrossRef]
23. Vito, S.D.; D'Elia, G.; Francia, G.D. Global calibration models match ad-hoc calibrations field performances in low cost particulate matter sensors. In Proceedings of the 2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May–1 June 2022; IEEE: Piscataway, NJ, USA, 2022. [CrossRef]
24. Miquel-Ibarz, A.; Burgués, J.; Marco, S. Global calibration models for temperature-modulated metal oxide gas sensors: A strategy to reduce calibration costs. *Sens. Actuators B Chem.* **2022**, *350*, 130769. [CrossRef]
25. Fernandez, L.; Guney, S.; Gutierrez-Galvez, A.; Marco, S. Calibration transfer in temperature modulated gas sensor arrays. *Sens. Actuators B Chem.* **2016**, *231*, 276–284. [CrossRef]
26. Vito, S.D.; Fattoruso, G.; Pardo, M.; Tortorella, F.; Francia, G.D. Semi-Supervised Learning Techniques in Artificial Olfaction: A Novel Approach to Classification Problems and Drift Counteraction. *IEEE Sens. J.* **2012**, *12*, 3215–3224. [CrossRef]

27. Rudnitskaya, A. Calibration Update and Drift Correction for Electronic Noses and Tongues. *Front. Chem.* **2018**, *6*, 433. [[CrossRef](#)]
28. Robin, Y.; Amann, J.; Goodarzi, P.; Schütze, A.; Bur, C. Transfer Learning to Significantly Reduce the Calibration Time of MOS Gas Sensors. In Proceedings of the 2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May–1 June 2022; IEEE: Piscataway, NJ, USA, 2022. [[CrossRef](#)]
29. Robin, Y.; Amann, J.; Goodarzi, P.; Schneider, T.; Schütze, A.; Bur, C. Deep Learning Based Calibration Time Reduction for MOS Gas Sensors with Transfer Learning. *Atmosphere* **2022**, *13*, 1614. [[CrossRef](#)]
30. Plested, J.; Gedeon, T. Deep transfer learning for image classification: A survey. *arXiv* **2022**, arXiv:2205.09904.
31. Weiss, K.; Khoshgoftaar, T.M.; Wang, D. A survey of transfer learning. *J. Big Data* **2016**, *3*, 9. [[CrossRef](#)]
32. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A Survey on Deep Transfer Learning. In Proceedings of the Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018.
33. Bozinovski, S. Reminder of the First Paper on Transfer Learning in Neural Networks, 1976. *Informatica* **2020**, *44*, 291–302. [[CrossRef](#)]
34. Arendes, D.; Lensch, H.; Amann, J.; Schütze, A.; Baur, T. P13.1—Modular design of a gas mixing apparatus for complex trace gas mixtures. In *Proceedings of the Poster*; AMA Service GmbH: Wunstorf, Germany, 2021. [[CrossRef](#)]
35. Helwig, N.; Schüler, M.; Bur, C.; Schütze, A.; Sauerwald, T. Gas mixing apparatus for automated gas sensor characterization. *Meas. Sci. Technol.* **2014**, *25*, 055903. [[CrossRef](#)]
36. Leidinger, M.; Schultealbert, C.; Neu, J.; Schütze, A.; Sauerwald, T. Characterization and calibration of gas sensor systems at ppb level—A versatile test gas generation system. *Meas. Sci. Technol.* **2017**, *29*, 015901. [[CrossRef](#)]
37. Arendes, D.; Amann, J.; Brieger, O.; Bur, C.; Schütze, A. P35—Qualification of a Gas Mixing Apparatus for Complex Trace Gas Mixtures. In *Proceedings of the Poster*; AMA Service GmbH: Wunstorf, Germany, 2022. [[CrossRef](#)]
38. Loh, W.L. On Latin hypercube sampling. *Ann. Stat.* **1996**, *24*, 2058–2080. [[CrossRef](#)]
39. Baur, T.; Bastuck, M.; Schultealbert, C.; Sauerwald, T.; Schütze, A. Random gas mixtures for efficient gas sensor calibration. *J. Sens. Syst.* **2020**, *9*, 411–424. [[CrossRef](#)]
40. Baur, T.; Schütze, A.; Sauerwald, T. Optimierung des temperaturzyklischen Betriebs von Halbleitersensoren (Optimization of temperature cycled operation of semiconductor gas sensors). *Tm-Tech. Mess.* **2015**, *82*, 187–195. [[CrossRef](#)]
41. Burgués, J.; Marco, S. Feature Extraction for Transient Chemical Sensor Signals in Response to Turbulent Plumes: Application to Chemical Source Distance Prediction. *Sens. Actuators B Chem.* **2020**, *320*, 128235. [[CrossRef](#)]
42. Robin, Y.; Amann, J.; Baur, T.; Goodarzi, P.; Schultealbert, C.; Schneider, T.; Schütze, A. High-Performance VOC Quantification for IAQ Monitoring Using Advanced Sensor Systems and Deep Learning. *Atmosphere* **2021**, *12*, 1487. [[CrossRef](#)]
43. Dorst, T.; Schneider, T.; Schütze, A.; Eichstädt, S. D1.1 GUM2ALA—Uncertainty Propagation Algorithm for the Adaptive Linear Approximation According to the GUM. In *Proceedings of the SMSI 2021—System of Units and Metrological Infrastructure*; AMA Service GmbH: Wunstorf, Germany, 2021. [[CrossRef](#)]
44. Schneider, T.; Helwig, N.; Schütze, A. Industrial condition monitoring with smart sensors using automated feature extraction and selection. *Meas. Sci. Technol.* **2018**, *29*, 094002. [[CrossRef](#)]
45. de Jong, S. SIMPLS: An alternative approach to partial least squares regression. *Chemom. Intell. Lab. Syst.* **1993**, *18*, 251–263. [[CrossRef](#)]
46. Dorst, T.; Schneider, T.; Eichstädt, S.; Schütze, A. Influence of measurement uncertainty on machine learning results demonstrated for a smart gas sensor. *J. Sens. Syst.* **2023**, *12*, 45–60. [[CrossRef](#)]
47. Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, L.; Wang, G.; et al. Recent Advances in Convolutional Neural Networks. *arXiv* **2017**, arXiv:1512.07108.
48. Robin, Y.; Amann, J.; Goodarzi, P.; Baur, T.; Schultealbert, C.; Schneider, T.; Schütze, A. Überwachung der Luftqualität in Innenräumen mittels komplexer Sensorsysteme und Deep Learning Ansätzen. In *Proceedings of the Vorträge*; AMA Service GmbH: Wunstorf, Germany, 2021. [[CrossRef](#)]
49. White, C.; Neiswanger, W.; Savani, Y. BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search. *arXiv* **2020**, arXiv:1910.11858.
50. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian Optimization of Machine Learning Algorithms. *arXiv* **2012**, arXiv:1206.2944.
51. Robin, Y.; Amann, J.; Goodarzi, P.; Schneider, T.; Schütze, A.; Bur, C. Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays. In Proceedings of the I2MTC, Kuala Lumpur, Malaysia, 22–25 May 2023.
52. Fonollosa, J.; Neftci, E.; Huerta, R.; Marco, S. Evaluation of calibration transfer strategies between Metal Oxide gas sensor arrays. *Procedia Eng.* **2015**, *120*, 261–264. [[CrossRef](#)]
53. Yadav, K.; Arora, V.; Jha, S.K.; Kumar, M.; Tripathi, S.N. Few-shot calibration of low-cost air pollution (PM2.5) sensors using meta-learning. *arXiv* **2021**, arXiv:2108.00640.
54. Brown, S.D.; Tauler, R.; Walczak, B. *Comprehensive Chemometrics: Chemical and Biochemical Data Analysis*; Elsevier: Amsterdam, The Netherlands, 2020.

55. Wang, Y.; Lysaght, M.J.; Kowalski, B.R. Improvement of multivariate calibration through instrument standardization. *Anal. Chem.* **1992**, *64*, 562–564. [[CrossRef](#)]
56. Zhuang, F.; Qi, Z.; Duan, K.; Xi, D.; Zhu, Y.; Zhu, H.; Xiong, H.; He, Q. A Comprehensive Survey on Transfer Learning. *arXiv* **2020**, arXiv:1911.02685.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

3.5 Paper A – Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays

Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, C. Bur

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)

The original paper can be found in the online version at <https://ieeexplore.ieee.org/document/10175975> or DOI: <https://doi.org/10.1109/I2MTC53148.2023.10175975>

© 2023 IEEE. Reprinted, with permission, from Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, C. Bur; Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays, July/2023.

3.5.1 Synopsis

Since DL is constantly faced with skepticism, it is necessary to use XAI techniques to understand the TCOCNN better. With XAI, it is possible to learn from the inner workings of the network and understand the sensor, comprehend the Temperature-Cycle (TC), as well as build simpler models with more advanced features based on the FESR approach. As before, it is vital to introduce the dataset. The dataset used in this publication was the same as in Paper 1. All evaluations were based on 1200 UGMs of two SGP40 sensors with four sub-sensors in TCO. While the data was recorded within the same measurement as the data of Paper 1, different sensors were used. The TC was the same as in Paper 1; the only exception was sub-sensor 3, which was only modulated between 200 °C and 300 °C. The dataset was split into training and testing. The last 60 UGMs were used for testing, and the remaining UGMs for training. The target gas was formaldehyde, essential for IAQ monitoring.

This publication presented two XAI techniques to analyze the TCOCNN and the SGP40 sensor in more detail. The XAI techniques used in this publication were the occlusion and gradient maps from computer vision. Those methods identify the most crucial section within a TC by training a model on the training data while using the respective method and test data to obtain an importance score for each input pixel. Afterward, the mean importance scores across multiple test observations were calculated to highlight the most critical sections of the TC. However, it is challenging to verify if the selected sections contain meaningful information (cf. Figure 3.7). Therefore, a new approach was introduced to confirm the selected regions. This approach trains new models with and without the most essential part of the signal. Suppose the selected section contains the most important information; the prediction quality should decrease drastically for the model trained without the most important sections. Thereby, it was possible to validate the importance score and rate the different XAI methods.

As a result, it was demonstrated that the occlusion map provides a more helpful importance score. However, both methods showed similar results. Furthermore, it was possible to demonstrate that only 7 % of the TC was sufficient to build a model almost as good as the original (RMSE: 15.8 ppb vs. 19.3 ppb). This indicates that the occlusion map can be used to identify which sub-sensor, and even which temperature step in the TC, is most important for the target gas. Those results were further validated with a second sensor, which indicates those methods also allow for analyzing the differences between sensors. The results also suggest it might be possible to reduce the calibration

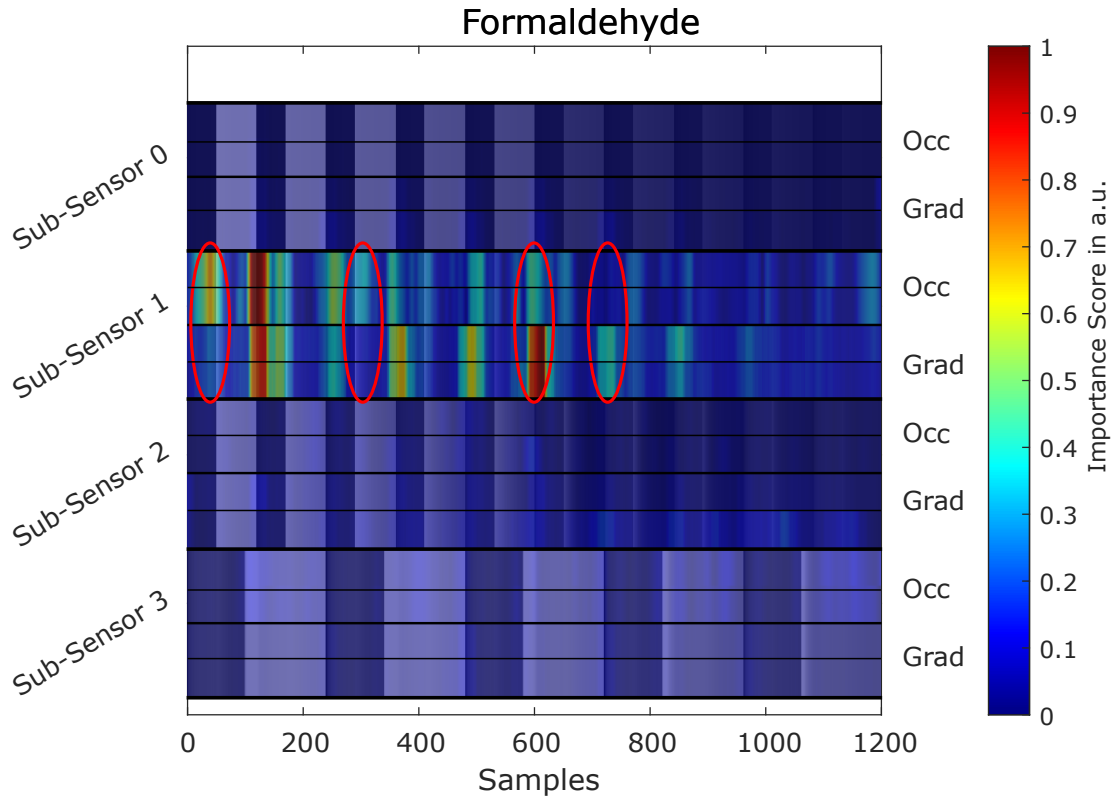


Figure 3.7: Comparison of importance scores occlusion map (Occ) vs. gradient map (Grad) standardized and rescaled from 0 to 1 for sensor A (two evaluations per method). Marked areas indicate regions with significant differences between methods (see text for details). Reprinted with permission of Ref. Paper A. Y. Robin, © 2023 IEEE.

time by an additional 50 % by removing unnecessary temperature steps. However, those results must be verified with actual modified TCs and more XAI methods. It is important to mention that, specifically, the occlusion map can also be used together with the FESR approach to make the same observations.

In conclusion, this newly developed validation scheme, together with the XAI techniques, demonstrated that it is possible to understand the TC and which temperature steps are essential for predicting a specific target gas. Furthermore, extracting the optimal duration of the temperature steps and which sub-sensor contains the most critical information was possible. Similarly, XAI opens up the possibility of analyzing the sensor and the differences between sensors. Moreover, it allows future evaluations to study the effect of different gases at different temperature steps to explore which mixtures can be measured with the sensor. However, it should be mentioned that this

was only one experiment with one gas on one dataset. More tests must be conducted to understand better which XAI method works best and which one is applicable for real-world evaluation. This paper concludes the application of the TCOCNN for gas sensing applications by showing that the TCCOCNN, although complex, can still be verified and understood. Paper B now analyses the performance of the TCOCNN for different data-driven tasks.

The main takeaways of this publication are:

- A new method for validating XAI techniques for gas sensor calibration was introduced.
- XAI techniques can be used to understand the sensor and the TCO.
- XAI can be used to estimate the capability of a sensor to analyze specific gases.
- It is possible to highlight differences between sensors.
- Occlusion map performs slightly better. However, this has to be further validated.
- The TC could be reduced by up to 50 % while maintaining performance.

Open questions/tasks are:

- Can the TCOCNN be applied to different fields (e.g., condition monitoring)?

This full text paper was peer-reviewed at the direction of IEEE Instrumentation and Measurement Society prior to the acceptance and publication.

Comparison of Explainable Machine Learning Algorithms for Optimization of Virtual Gas Sensor Arrays

1st Yannick Robin

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
y.robin@lmt.uni-saarland.de

2nd Johannes Amann

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
j.amann@lmt.uni-saarland.de

3rd Payman Goodarzi

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
p.goodarzi@lmt.uni-saarland.de

4th Tizian Schneider

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
t.schneider@lmt.uni-saarland.de

5th Andreas Schütze

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
schuetze@lmt.uni-saarland.de

6th Christian Bur

Lab for Measurement Technology
Saarland University
Saarbruecken, Germany
c.bur@lmt.uni-saarland.de

Abstract—Metal oxide semiconductor (MOS) gas sensors operated in temperature cycled operation (TCO) and calibrated with machine learning algorithms are increasingly promising for indoor air quality (IAQ) assessments. This can be attributed to the cost-efficient sensors, with a broad sensitivity spectrum and the possibility of continuous measurements. However, with the ever-increasing complexity of data-driven models used to calibrate the MOS gas sensors, understanding the connection between the raw input and the predicted gas concentration is especially important. In this work, two methods from the field of explainable AI are applied to our custom neural network (TCOCNN) and compared regarding their capability to identify essential parts of the raw input signal. For this purpose, a validation scheme is introduced to rate the explanation methods. Finally, it is shown that with only 7 % of the original raw input, root-mean-squared error (RMSE) values for formaldehyde that are only 22 % worse compared to the absolute best (15.8 ppb vs. 19.3 ppb) can be achieved. This more profound understanding of the sensor can then be used to show differences between sensors, allow more accessible models to be built, and optimize the temperature-cycled operation regarding the number of temperature steps.

Index Terms—volatile organic compounds, indoor air quality, deep neural networks, temperature cycled operation, explainable machine learning algorithms.

I. INTRODUCTION

With people spending almost 90 % of their time indoors [1], [2], it is ever more critical to ensure suitable indoor air quality to maintain good health conditions. The U.S. Environmental Protection Agency states that indoor air pollutants can cause a wide range of health problems, from mild symptoms like headaches to severe illnesses like cancer if contaminated with carcinogenic volatile organic compounds (VOCs) like formaldehyde [2], [3]. However, continuously monitoring indoor air quality concerning all dangerous VOCs, most prominent formaldehyde and benzene [4], is difficult as there are

many different interfering gases present (e.g., ethanol or CO₂) [5]. It is still an open research field where much progress must be made. The current problem with accurate indoor air quality monitoring is that the systems capable of monitoring indoor air concerning single dangerous VOCs (e.g., GC/MS or PTR-MS) are expensive, require expert knowledge to operate, or do not provide online monitoring. Metal oxide semiconductor (MOS) gas sensors are one promising solution for accurate, real-time quantification of single dangerous VOCs that would allow advanced indoor air quality assessment systems to be widely used. They have proven affordable, easy to operate, and sensitive to various VOCs [6]. Nevertheless, those systems suffer from significant drawbacks, like lack of selectivity, manufacturing tolerances, long calibration times, and drift over time. Those problems must first be overcome before they can be universally used for predicting individual VOCs accurately. Previous studies showed that deep learning models could be used to calibrate a MOS gas sensor to accurately predict individual gases with only a few calibration samples to overcome long calibration times of multiple weeks [6], [7]. Furthermore, advances in deep neural networks could be utilized to overcome other challenges, like manufacturing tolerances [8], [9]. Nevertheless, the drawback of increasingly complex evaluation methods is that the origin of the prediction is lost. In our case, the MOS gas sensor is operated dynamically (temperature cycle operation (TCO)), and one entire cycle is used to predict the specific gas concentrations. Therefore, it is not apparent which temperature steps are the most important. That is because the computation path is too complex for a human to understand. Thus, this work shows two methods from the field of deep learning that can be used to gain this information and allow the user to optimize the data-driven model or the operation mode of the gas sensor. It also allows

gaining other insights into how the model works and allows the user to understand the sensor itself better. Furthermore, a validation scheme is introduced to rate different explanation methods to find the best possible method for virtual gas sensor arrays. There is already some work trying to explain deep neural networks for gas sensor data like [10], [11]. This work extends those studies because deep convolutional neural networks are used, the focus is on comparing different explanation methods, and the raw signal is used instead of extracted features. Specifically, formaldehyde was chosen as a reference because of its carcinogenic properties [4], making it of utmost importance for indoor air quality assessments.

II. MATERIALS AND METHODS

A. Dataset

The dataset consists of multiple sensors and their response to complex gas mixtures (simulate indoor air), which are used to calibrate the sensors (build a data-driven model). For this work, we use the data of two SGP40 sensors (Sensirion AG, Stäfa, Switzerland), sensors A and B, consisting of four sub-sensors each. During data recording, the sensors are operated in temperature cycled operation (TCO), resulting in a combination of real and virtual sensor arrays to improve the selectivity and sensitivity [5], [12]. Each sub-sensor is individually heated with the underlying micro-hotplate and is sampled at 10 Hz. This operation mode was possible through specific commands given by Sensirion under a non-disclosure agreement. The temperature cycle (TC) has a duration of 120 s and comprises high and low-temperature phases specific to the independent sub-sensors. For sub-sensors 0-2, the ten high-temperature phases last 5 seconds and are set at 400 °C, and the corresponding ten low-temperature phases are increased from 100 °C to 375 °C in 25 °C steps where the steps at 225 °C, 250 °C, and 275 °C are left out. For sub-sensor 3, the high-temperature state is set at 300 °C, and the low temperature is increased in 5 steps from 100 °C to 200 °C (25 °C). The temperature setpoints were chosen based on previous experiments to be selective to dangerous VOCs [5]. The temperature cycle is depicted in Figure 1 together with gas sensor data recorded in synthetic air.

The two SGP40 sensors operated in TCO are exposed to various gas mixtures simulating indoor air (laboratory conditions) and natural environments with release tests. The first part of the dataset is recorded in the laboratory for approximately one week. Thereafter, the sensors are operated for four weeks in the field and, afterwards, moved back to the laboratory to recalibrate for drift (one week). Subsequently, the sensor is moved in the field for three weeks to analyze the stability again, and then a final calibration dataset is recorded in the laboratory (one week). During calibration, the sensors are exposed to complex and randomized gas mixtures [13] consisting of four VOCs (acetone, ethanol, formaldehyde, and toluene) together with two background gases (carbon monoxide and hydrogen), and the relative humidity (varied between 25 % and 70 % (at 20 °C)). In order to generate a unique gas mixture (UGM), each gas concentration and the relative humidity was

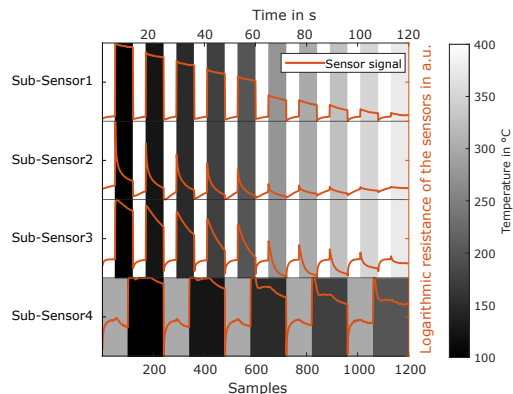


Fig. 1. One sensor response during temperature cycled operation of the SGP40 in synthetic air for all sub-sensors is given in orange together with the temperature setpoint in grayscale intervals (sampled at 10 Hz).

TABLE I
CONCENTRATION RANGES FOR THE UNIQUE GAS MIXTURES DURING LABORATORY MEASUREMENTS [6], [16].

Substance	Minimum	Maximum	Extended
Carbon monoxide	150 ppb	2000 ppb	-
Hydrogen	400 ppb	2000 ppb	4000 ppb
Humidity	25 % RH	70 % RH	-
Acetone	14 ppb	300 ppb	1000 ppb
Toluene	4 ppb	300 ppb	1000 ppb
Formaldehyde	1 ppb	400 ppb	-
Ethanol	4 ppb	300 ppb	1000 ppb
TVOCsens	300 ppb	1200 ppb	

randomly picked from a predefined uniform distribution (latin hypercube sampling) [14]. In total, 1200 UGMs are measured (500 UGMs first calibration, 500 UGMs second calibration, 200 UGMs last calibration), with ten TCs recorded per gas mixture. However, because of synchronization errors between the gas mixing apparatus (GMA) and the recording system, together with the delay of the GMA until a stable gas mixture is applied to the gas sensors, only the five core temperature cycles (those in the middle) per UGM are used for calibrating the sensor (model-building). The concentration ranges for the single gases are based on studies for indoor air performed with analytical methods [15] and can be found in Table I. An in-depth explanation of the dataset and the experimental setup, albeit with data recorded for a different sensor, can be found in [6], [16]. In this work, only the highly reliable data from the three calibration phases are used to build the data-driven models and evaluate the deep learning methods.

B. Methods

In order to calibrate single sensors to predict a specific gas, the data recorded with the sensor is passed through a machine-

learning model, which learns the dependencies between the raw input signal and target gas concentration. The model used in this work is the TCOCNN [16], a custom convolutional neural network (cf. Figure 2) consisting of ten convolutional layers followed by two fully connected layers [16]. The input for the TCOCNN is a 4×1200 matrix where the four represents the four sub-sensors of one SGP40 and 1200 represents the number of samples in one full TC (resulting in 4800 samples or pixels). The resulting data was then standardized (zscore: subtract mean and divide by standard deviation for each sub-sensor in each observation) to have a similar value range for each sub-sensor before it was provided to the TCOCNN. The hyperparameters of the neural network are optimized for every target gas independently based on Bayesian optimization as explained in some previous work [16] to guarantee optimal network performance. The training (only laboratory samples) is always performed with the first 1140 UGMS, and testing (only laboratory samples) is done with the last 60 UGMs. This approach is reasonable because of the randomized UGMs.

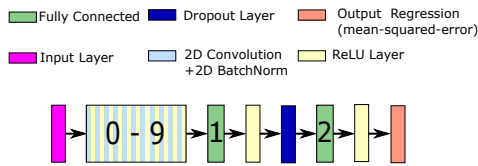


Fig. 2. Architecture of the TCOCNN (adapted from [7]).

After introducing the TCOCNN to calibrate the gas sensor for a specific gas, the methods to explain those models are introduced. First, a standard method for explaining the TCOCNN from the field of deep learning called occlusion map [17] is introduced. This method operates as a black-box explainer, where only the model's input and output are required, and no internal access is needed. For this method, the model needs to be trained first. Afterwards, one data sample (one full TCO) is fed through the network. The observation is passed multiple times through the network, while different parts of the network are occluded with a reference sample in each iteration. The occluded area is specified similarly to a convolutional layer in a neural network. It has a striding and kernel size, which can be of any rectangular shape. The kernel size specifies the size of the occluded area, while the striding size specifies the distance the kernel is moved over the matrix after each iteration. For this use case, the kernel size was set to 1×30 and the striding size to 1×10 . The striding size was chosen so that the frame fits in the 4×1200 array during evaluation, and the kernel size is smaller than the high-temperature phases of sub-sensors 0-2 (50 samples). Every sample point (or pixel) is occluded at least once with a reference sample during all iterations. For image processing, the occluded pixels are often replaced with a grey-valued pixel. However, this is not possible for gas sensor data. Therefore, a reference observation for each sample point in the input

matrix (4×1200), based on the mean value across the full training data, is built. This allows a mean sensor response of 4×1200 to be created and used for occluding the observation evaluated. After evaluating an occluded frame, the difference between the actual output and the new output is calculated and repeated for all iterations (calculating the error). This results in an error matrix smaller than the original input matrix (due to the fact that the striding size is larger than one), which is then interpolated to reproduce the same size as the input. The resulting array is then referred to as the occlusion map. Each sample within the observation has a corresponding value from the occlusion map that resembles an importance score. The higher this value is, the more critical that sample is since occluding this sample causes a more significant deviation from the original prediction. Investigating only one sample has the problem that this might only represent parts of the dataset as only a local explanation is gained. In this work, multiple samples are tested to obtain a global explanation and to gain insights into the overall dependencies between input data and output prediction [18], [19]. To be able to compare the resulting importance scores between multiple runs, the absolute mean is calculated across all samples, and the resulting 4×1200 matrix is then standardized (zscore).

The second method also originates from the field of deep learning and is called a gradient map [20]. A gradient map is a white box model that requires access to the model's inner workings and is also instance-based (local explanation). This algorithm calculates the gradient of the input layer (I) to the regression output (RO) concerning the analyzed sensor response (S), which then resembles the importance score (IC) for the 4×1200 matrix [20]. The higher the absolute value of the gradient, the more critical this pixel is for the TCOCNN.

$$IC = \left. \frac{\partial RO}{\partial I} \right|_S \quad (1)$$

A more detailed explanation can be found in [20]. As for the occlusion map, this provides only a local explanation as it examines one instance at a time. Therefore, this approach is also extended to a global model by analyzing multiple instances [18], [19] and calculating the mean absolute response. Furthermore, the resulting IC is smoothed by a sliding window to obtain regions instead of single vital pixels. The window size for the sliding window was set to 30 pixels across each sub-sensor independently. The value of 30 was chosen to be comparable to the occlusion map and much smaller than the high-temperature phases of sub-sensors 0-2 (50 samples). These two methods (occlusion map and gradient map) have been chosen as they are complementary concerning white and black box access and are fast to calculate. Since both require samples to be operated on, the results are based on the first 60 UGMs and the last 60 UGMS (test data). Using a reduced number of UGMs from the training and test set for evaluating the two methods was possible because the randomized UGMs guaranteed similar distributions. Furthermore, a reduction was necessary to speed up the evaluation process, which, although

fast methods were used, still required a long computation time (multiple hours).

C. Evaluation

In order to test those two methods together with the TCOCNN, multiple evaluations are performed. First, it was tested if the TCOCNN achieves stable results in multiple runs (reproducibility). After that, the two methods are tested to determine if they are consistent regarding the regions of importance. Therefore, a model is trained on Sensor A, and the importance scores from the two methods are compared. In order to verify those results and show reproducibility and generalizability, this procedure is repeated four times. The repetition is done with the same data to consider the variation caused by the neural network. After verifying the reproducibility and generalizability and comparing the two approaches, the next step is to verify that the selected samples are indeed the most important parts of the input signal. For pictures, that can often be done by analyzing the selected region [21]. Since this is not possible for gas sensors, where it is unknown where the most crucial section is, a new validation scheme is introduced. This is done by training the TCOCNN on the data where the most crucial 7 % are occluded with a reference frame (93 % of data still available). The value of 7 % was selected after multiple trials because this was the minimum number of samples needed for the TCOCNN to successfully train with the X % most important samples and outperform the model trained with the remaining part. Occluding the 7 % of the most important samples ensures that all the essential information can not be used to train and test the model. Furthermore, the reference frame is the mean frame across the training set (4x1200). Thereafter, this approach is reversed. This means that for this experiment, only the 7 % best samples are used for training, and the remaining 93 % are occluded with a reference frame (do not contain any information). This is done based on the importance score of the occlusion map and gradient map, respectively. Afterward, the achieved root-mean-squared error (RMSE) on the test set is analyzed and compared to the accuracy when the whole frame is used. In order to make a reliable comparison, each evaluation is repeated four times on the same data, and the mean RMSE and the standard deviations are compared. This was done to consider the variation a random initialization of the TCOCNN and the adam solver causes.

III. RESULTS

As stated before, the first step is to compare the reproducibility based on the RMSE over multiple runs. Table II shows that for sensors A and B during multiple runs (full dataset 4x1200), the achieved RMSE on the test data varies only slightly (RMSE: 15.8 ppb and 18.8 ppb; standard deviation: 0.3 ppb and 0.6 ppb for sensors A and B, respectively). Those RMSE values are suitable for the indoor air quality task if considering the range for formaldehyde (1 - 400 ppb), the complex background, and the limit of 80 ppb set by the WHO [22]. Furthermore, the minor variation ensures that

multiple runs result in reasonably similar models, which is supported by Figure 3. This figure shows the importance score obtained with the occlusion and gradient maps. Each sub-sensor is plotted on top of the other, and each sub-sensor plot, in turn, consists of four importance scores (two for each method). The two importance scores per method result from repeated evaluations (at least twice). Comparing the importance scores of the occlusion maps and gradient maps for the two independent runs, they show that similar features across multiple evaluations are obtained. This confirms that the TCOCNN uses similar features from the same sensor (only sub-sensor 2) in every run. In the next step, the importance scores between the two methods are compared based on Figure 3. In order to make the comparison more accessible, the scores are rescaled from 0 to 1. It is shown that both methods highlight the same sub-sensor as the most critical sensor. However, when analyzing in more detail, the gradient map highlights a different set of samples as the most crucial section (dark red). Furthermore, it can be observed that in some cases, they even highlight entirely different sections of the input. With 7 % of the essential samples selected, both methods only agree on 57 % of the most critical samples. Each method provides 43 % different ones, which shows that the two methods do not entirely agree.

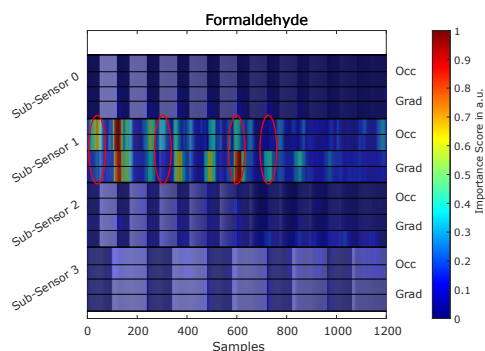


Fig. 3. Comparison of importance scores occlusion map (Occ) vs. gradient map (Grad) standardized and rescaled from 0 to 1 for sensor A (two evaluations per method). Marked areas indicate regions with significant differences between methods (see text for details).

After comparing the occlusion and gradient maps, the next step is to compare the importance scores between the TCOCNN for different sensors. Figure 4 and Figure 5 show the importance score for the occlusion map and gradient map for two separate runs on sensors A and B. As stated above, the models trained on the same sensor have similar importance scores and differ only slightly (sensors A and B). Furthermore, it can be seen that the importance score (occlusion map and gradient map) between sensors differ slightly. This is reasonable since it is impossible to use the same model on multiple sensors as, for example, shown in [7], [9] without recalibration.

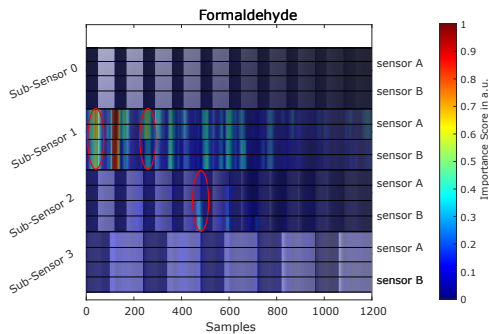


Fig. 4. Comparison of the calculated importance scores based on occlusion map during two evaluations for two sensors (rescaled from 0 to 1). Marked areas indicate regions with significant differences between sensors (see text for details).

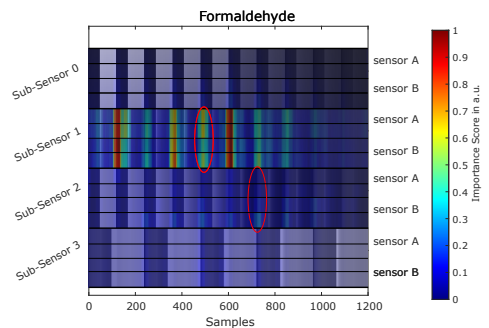


Fig. 5. Comparison of the calculated importance scores based on gradient map during two evaluations for two sensors (rescaled from 0 to 1). Marked areas indicate regions with significant differences between sensors (see text for details).

After analyzing the two methods for calculating the importance score based on their reproducibility, generalizability, and stability across sensors and multiple evaluations, the next step is to analyze the performance of the different methods. The goal is to investigate which method provides the most accurate importance score. As explained above, the TCOCNN is evaluated multiple times without the most critical samples (93 % of the input signal (TC)) and with only the most critical samples (7 %). The results are listed in Table II. For the occlusion map, it can be observed that without the most critical samples for sensor A, the mean RMSE increases significantly from 15.8 ppb to 23.8 ppb. The next row then indicates the mean RMSE achieved with only 7 % of the samples in one full TCO. For the occlusion map, it is shown that the mean RMSE only increases from 15.8 ppb to 19.3 ppb, which is much smaller compared to training with 93 % of the TC. This confirms that this part of the TC contains the essential information and that the occlusion map provides significant insights into the sensor and the TC. This insight

could, for example, be used to shorten the TC and thereby shorten the calibration time or extract specific features from a critical area for a model with reduced complexity (e.g., feature extraction, selection, and regression (FESR) [16]). This is slightly different for the gradient map. Although the importance score is similar to the occlusion map, it shows a different performance. For this use case, the dataset with only the most critical 7 % performs similarly to the dataset without the most critical samples (21.8 ppb vs. 21.2 ppb). This shows that the not overlapping 43 % (occlusion map vs. gradient map) of the most critical samples is the cause of the drop in performance. The main factor might be the difference at the beginning of the TCO in sub-sensor one, cf. Figure 3. Therefore, the gradient map cannot be used as successfully as the occlusion map in its most basic form. The reason might be that the gradient map does not consider the variation the input samples can have. This is different from the occlusion map, where this is taken into account with the reference sample. Therefore, it is necessary to use advanced versions of this method like guided backpropagation [23] or gradCam [24] to maybe reach a similar performance, which will be done in future work. All evaluations are repeated for sensor B, cf. Table II, in order to verify the results. Similar results for sensor B were achieved, which means that the occlusion map outperforms the gradient map for all the other evaluations. However, the gradient map shows a slightly better performance for sensor B, but it is still worse than the occlusion map. Furthermore, this indicates that if more advanced methods are used, and the difference is smaller than in this comparison, multiple virtual sensor arrays are necessary to identify the best method.

TABLE II
THE ACHIEVED RMSE VALUES WITH THE FULL DATASET AND THE DATASET WITH A REDUCED NUMBER OF FEATURES BASED ON IMPORTANCE SCORES FROM SENSORS A AND B FOR THE OCCLUSION MAP AND THE GRADIENT MAP.

		mean RMSE in ppb ± standard deviation in ppb	
		occlusion map	gradient map
sensor A	training data set	15.8 ± 0.3	
	all data	15.8 ± 0.3	
	w/o most important 7 %	23.8 ± 1.0	21.2 ± 1.3
	most important 7 % only	19.3 ± 1.0	21.8 ± 0.7
sensor B	all data	18.8 ± 0.6	
	w/o most important 7 %	26.3 ± 0.5	26.2 ± 1.1
	most important 7 % only	19.9 ± 1.0	24.3 ± 1.0

IV. DISCUSSION AND CONCLUSION

This study showed two approaches for analyzing a neural network for gas sensor data. The goal was to understand which samples of one full TC are most significant for the TCOCNN and which methods provide the most reliable information based on the validation scheme. It was possible to show that both methods provide reproducible results and highlight the

same sub-sensor as the most important. The results were reproduced within multiple runs (same data) across two different sensors. It was shown that the most critical samples differ slightly according to the importance score. For both methods, the most important 7 % of samples showed an overlap of only 57 %. In order to analyze which method is better at identifying the most critical samples, a new validation scheme was introduced. Here, the TCOCNN was evaluated multiple times without the most critical samples (93 % of the original TC) and with only the most critical samples (7 % of the original TC). This evaluation showed that the occlusion map is better suited for calculating the importance score since even with only 7 % of the actual TC, better results were achieved than with the remaining 93 % (19.8 ppb vs. 23.8 ppb). This was not possible for the gradient map, where the most critical 7 % for sensor A showed slightly worse performance than the dataset with 93 % of the TC (21.8 ppb vs. 21.2 ppb). These results enable the TCOCNN to be used to gain insights into the sensor, sensor differences (sensor A vs. sensor B), and the TC. For example, these results could reduce the number of temperature steps in the TC by up to 50 % for formaldehyde and still obtain similar results. The 50 % of TC is needed to cover almost the complete 7 %. Additionally, the essential sub-sensors can indicate which sensors are sensitive to which gas. Furthermore, new and more specific features could be extracted from the most critical sections of the TC to build a less complex model based on feature extraction, selection, and regression (FESR), where it is not even necessary to recalibrate the model for different sensors. These methods and the validation scheme could also be used in future work to understand the effect of different features regarding drift compensation and find robust features against sensor poisoning [25]. This is possible because the TCOCNN will adapt to the new situation, and by comparing the resulting importance score with the importance score from the sensor in its basic form, insights can be gained. Furthermore, more methods like gradCam or guided backpropagation with more complex datasets should be tested to gain even more precise insights.

ACKNOWLEDGMENT

This work was supported by the German ministry for education and research (BMBF) in the project KI-PREDICT (16ME0030K).

REFERENCES

- [1] S. Brasche and W. Bischof, "Daily time spent indoors in german homes – baseline data for the assessment of indoor exposure of german occupants," *International Journal of Hygiene and Environmental Health*, vol. 208, no. 4, pp. 247–253, jul 2005.
- [2] "Indoor air quality," <https://www.epa.gov/report-environment/indoor-air-quality>, united States Environmental Protection Agency, Sep. 2021, [Online; accessed 15-November-2022].
- [3] M. Hauptmann, J. H. Lubin, P. A. Stewart, R. B. Hayes, and A. Blair, "Mortality from Solid Cancers among Workers in Formaldehyde Industries," *American Journal of Epidemiology*, vol. 159, no. 12, pp. 1117–1130, 06 2004.
- [4] D. A. Sarigiannis, S. P. Karakitsios, A. Gotti, I. L. Liakos, and A. Katsoyiannis, "Exposure to major volatile organic compounds and carbonyls in european indoor environments and associated health risk," *Environment International*, vol. 37, no. 4, pp. 743–765, may 2011.
- [5] A. Schütze and T. Sauerwald, "Dynamic operation of semiconductor sensors," in *Semiconductor Gas Sensors (Second Edition)*. Woodhead Publishing, 2020, pp. 385–412.
- [6] T. Baur, J. Amann, C. Schultealbert, and A. Schütze, "Field study of metal oxide semiconductor gas sensors in temperature cycled operation for selective VOC monitoring in indoor air," *Atmosphere*, vol. 12, no. 5, p. 647, may 2021.
- [7] Y. Robin, J. Amann, P. Goodarzi, T. Schneider, A. Schütze, and C. Bur, "Deep learning based calibration time reduction for MOS gas sensors with transfer learning," *Atmosphere*, vol. 13, no. 10, p. 1614, oct 2022.
- [8] S. Feng, F. Farha, Q. Li, Y. Wan, Y. Xu, T. Zhang, and H. Ning, "Insight on smart gas sensing technology," *Sensors*, vol. 19, no. 17, p. 3760, aug 2019.
- [9] Y. Robin, J. Amann, P. Goodarzi, A. Schütze, and C. Bur, "Transfer learning to significantly reduce the calibration time of MOS gas sensors," in *2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN)*. IEEE, may 2022.
- [10] S. A. Schober, Y. Bahri, C. Carbonelli, and R. Wille, "Neural network robustness analysis using sensor simulations for a graphene-based semiconductor gas sensor," *Chemosensors*, vol. 10, no. 5, p. 152, apr 2022.
- [11] Y. Robin, J. Morsch, T. Schneider, A. Schütze, and C. Bur, "Insight in dynamically operated gas sensor arrays with shapley values for data segments," in: *Micro and Nano Engineering- Euroensors (MNE-ES) (Leuven, Belgium, 2022-09-19)*, 2022.
- [12] P. Reimann and A. Schütze, *Sensor Arrays, Virtual Multisensors, Data Fusion, and Gas Sensor Data Evaluation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 67–107.
- [13] T. Baur, M. Bastuck, C. Schultealbert, T. Sauerwald, and A. Schütze, "Random gas mixtures for efficient gas sensor calibration," *Journal of Sensors and Sensor Systems*, vol. 9, no. 2, pp. 411–424, nov 2020. [Online]. Available: <https://jsss.copernicus.org/articles/9/411/2020/>
- [14] W.-L. Loh, "On Latin hypercube sampling," *The Annals of Statistics*, vol. 24, no. 5, pp. 2058 – 2080, 1996.
- [15] H. Hofmann and P. Plieninger, "Bereitstellung einer datenbank zum vorkommen von flüchtigen organischen verbindungen in der raumlufte. forschungsbericht 205 61 243. hrsg.: Arbeitsgemeinschaft ökologischer forschungsinstitute (agöf) e. v.," 2008.
- [16] Y. Robin, J. Amann, T. Baur, P. Goodarzi, C. Schultealbert, T. Schneider, and A. Schütze, "High-performance VOC quantification for IAQ monitoring using advanced sensor systems and deep learning," *Atmosphere*, vol. 12, no. 11, p. 1487, nov 2021.
- [17] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 818–833.
- [18] M. Setzu, R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, and F. Giannotti, "Glocalx – from local to global explanations of black box ai models," *Journal of Artificial Intelligence, Volume 294*, 2021.
- [19] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "Explainable ai for trees: From local explanations to global understanding," *Nat Mach Intell* 2, 56–67, 2020.
- [20] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," 2013. [Online]. Available: <https://arxiv.org/abs/1312.6034>
- [21] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable AI: A review of machine learning interpretability methods," *Entropy*, vol. 23, no. 1, p. 18, dec 2020.
- [22] WHO Regional Office for Europe, *WHO guidelines for indoor air quality: selected pollutants*. Copenhagen: World Health Organization, 2010.
- [23] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014. [Online]. Available: <https://arxiv.org/abs/1412.6806>
- [24] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 336–359, oct 2019. [Online]. Available: <https://doi.org/10.1007%2Fs11263-019-01228-7>
- [25] V. Palmisano, E. Weidner, L. Boon-Brett, C. Bonato, F. Harskamp, P. Moretto, M. Post, R. Burgess, C. Rivkin, and W. Buttner, "Selectivity and resistance to poisons of commercial hydrogen sensors," *International Journal of Hydrogen Energy*, vol. 40, no. 35, pp. 11 740–11 747, sep 2015.

3.6 Paper B – Deep convolutional neural networks for cyclic sensor data

P. Goodarzi, Y. Robin, A. Schütze, T. Schneider

Lab for Measurement Technology, Saarland University, Campus A5 1, 66123 Saarbrücken, Germany

ArXiv 2308.06987

The original paper can be found in the online version at <https://arxiv.org/abs/2308.06987> or DOI: <https://doi.org/10.48550/arXiv.2308.06987>

© 2023 by the authors. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). Reprinted, with permission, from P. Goodarzi, Y. Robin, A. Schütze, T. Schneider; Deep convolutional neural networks for cyclic sensor data.

3.6.1 Synopsis

The previous papers introduced the whole processing chain for gas sensor calibration. Different ML solutions were introduced, and several improvements were achieved with the help of DL. Furthermore, XAI was introduced to provide an in-depth explanation of those models and allow the optimization of several parts of sensor calibration. However, sensor calibration is only one of many domains where ML can be used with cyclic sensor data. One additional domain is condition monitoring, which is applied to predict the current system state of a complex machine. Usually, data-driven condition monitoring is applied if the state of a system cannot be measured directly or if frequent maintenance is too costly. Most of the time, sensor data is used together with an ML model to predict the system's condition [120, 133]. Examples of such systems and the corresponding ML solution can be found in [120, 133, 264, 265]. In [264], it was already shown that DL could tackle condition monitoring problems, as introduced above. Paper B aims to use the newly obtained knowledge regarding the TCOCNN and tries to transfer this knowledge to the condition monitoring domain. The system analyzed in Paper B was a hydraulic testbed [133] capable of producing different pressure levels. Within the system, various components could wear down and cause system failure. This could be the cooler for the system, the main valve within the system, the pump that generates the pressure, or the accumulator used for pre-charging. The specific target for this work was to identify the state of the accumulator.³¹ Since the condition of every component could be controlled in this specific testbed, a pre-defined set of system states was recorded. In total, 144 system states were recorded (Paper B Figure 1) with ten working cycles each. This dataset's working cycle (similar to the TC) consisted of six different pressure levels varied within 60 seconds (10 seconds duration each), representing one observation. The working cycle was monitored with 17 sensors from various domains (e.g., pressure, vibration, temperature). Although the different sensors varied according to their sampling rate, an interpolation was performed to combine all sensors in a matrix of 17×6000 (60 s @ 100 Hz). The goal was to train a classifier capable of identifying the accumulator state correctly. In the first step of this paper, a baseline was generated with the help of the FESC method to rate the TCOCNN performance. The FESC toolbox achieved excellent results with an error rate of only 1 %. In contrast, the TCOCNN achieved an error rate of 20 % with all 17 sensors. To better understand the TCOCNN's poor performance and analyze if a general problem is the reason the model does not learn the intended dependencies, the TCOCNN was retrained with only one sensor (1×6000

³¹Optimal pressure, severely reduced pressure, close to total failure.

input). With this test, it was demonstrated that the TCOCNN could achieve an error rate of 1.7 % while only using one of the system's pressure sensors. However, if the TCOCNN was trained with multiple sensors from different domains, this result could not be reached again. The conclusion is that because of the significant difference in the input pattern, the TCOCNN could not select the sensor with the most information and was distracted by others. This effect is similar to the normalization of Paper 2. There, the normalization made learning features from every sub-sensor easier because the raw signals were in a similar range after normalization. Further experiments showed that combining the best sensor with a simulated static sensor with the same shape but a higher amplitude already worsened the results. This very poor performance was traced back to the early sensor fusion. While the FESC approach extracts features sensor-wise and selects the most promising features afterward, the TCOCNN tries to find a feature extraction that applies to all sensors simultaneously. This leads to the conclusion that for condition monitoring with signals from multiple domains, it is impossible to use the TCOCNN without any modifications to achieve compatible results. A possible solution to achieve reasonable results could be the multi-lane input network that independently performs feature extraction (convolutional layer) per sensor. However, more research should be conducted to create a model that can outperform the FESC approach for this task.

In conclusion, it can be stated that the TCOCNN cannot be easily transferred to different applications. Therefore, it is still necessary to have the domain knowledge to analyze the data and the problem to build an appropriate model for the task. Furthermore, it is recommended to test simpler models for the first evaluation as they naturally provide system insights without XAI methods.

The main takeaways of this publication are:

- It is impossible to use the TCOCNN for condition monitoring if the sensors have significantly different characteristics.
- The TCOCNN struggles to build a feature extraction that focuses on the features of the sensors that contain the information.
- Multi-lane CNNs might be a possible solution.
- The FESR approach works best for condition monitoring.

Open questions/tasks that are not covered in the following are:

- Find neural networks that are better suited for condition monitoring.
- Apply also other methods used for the TCOCNN (e.g., XAI, transfer learning) to condition monitoring to achieve better and more understandable solutions.
- One of the biggest problems in condition monitoring is domain shift. Can methods like meta-learning from the field of DL help to solve those problems?

Deep convolutional neural networks for cyclic sensor data

1st Payman Goodarzi
Lab for Measurement Technology
 Saarland University
 Saarbruecken, Germany
 p.goodarzi@lmt.uni-saarland.de

2nd Yannick Robin
Lab for Measurement Technology
 Saarland University
 Saarbruecken, Germany
 y.robin@lmt.uni-saarland.de

3th Andreas Schütze
Lab for Measurement Technology
 Saarland University
 Saarbruecken, Germany
 schuetze@lmt.uni-saarland.de

4th Tizian Schneider
Lab for Measurement Technology
 Saarland University
 Saarbruecken, Germany
 t.schneider@lmt.uni-saarland.de

Abstract—Predictive maintenance plays a critical role in ensuring the uninterrupted operation of industrial systems and mitigating the potential risks associated with system failures. This study focuses on sensor-based condition monitoring and explores the application of deep learning techniques using a hydraulic system testbed dataset. Our investigation involves comparing the performance of three models: a baseline model employing conventional methods, a single CNN model with early sensor fusion, and a two-lane CNN model (2L-CNN) with late sensor fusion. The baseline model achieves an impressive test error rate of 1% by employing late sensor fusion, where feature extraction is performed individually for each sensor. However, the CNN model encounters challenges due to the diverse sensor characteristics, resulting in an error rate of 20.5%. To further investigate this issue, we conduct separate training for each sensor and observe variations in accuracy. Additionally, we evaluate the performance of the 2L-CNN model, which demonstrates significant improvement by reducing the error rate by 33% when considering the combination of the least and most optimal sensors. This study underscores the importance of effectively addressing the complexities posed by multi-sensor systems in sensor-based condition monitoring.

Index Terms—Predictive maintenance, hydraulic system, deep learning, convolutional neural network

I. INTRODUCTION

Industrial systems and factories operate continuously, necessitating uninterrupted performance to avoid process downtime, significant financial losses, and potential safety hazards. To mitigate these risks, companies employ various maintenance approaches, including corrective maintenance, preventive maintenance, and predictive maintenance. Predictive maintenance (PdM) heavily relies on monitored signals from diverse sensors, with machine learning methods playing a pivotal role in data-driven PdM. These methods can be categorized into two groups: conventional approaches and deep learning techniques. Conventional methods involve preprocessing, feature extraction (FE), feature selection (FS), and the subsequent application of classification or regression algorithms [1], commonly referred to as FESC/FESR in this study. In contrast,

modern deep neural networks have demonstrated exceptional performance across various applications, including PdM [2]–[4].

In line with these advancements, gas mixture measurement has emerged as a promising application for deep neural networks in recent research. Notably, Robin et al. [5] introduced a convolutional neural network (CNN) specifically designed for indoor air quality monitoring (TCOCNN), accurately predicting volatile organic compounds using temperature-cycled operation sensors. The proposed method surpassed existing data evaluation techniques, underscoring the effectiveness of CNNs in this domain. It is worth noting that the signals utilized in their study bear resemblance to the typical data encountered in condition monitoring and predictive maintenance applications, i.e. multiple sensors with periodic or cyclic data.

Motivated by the aforementioned findings, the objective of our study is to compare our previously published method, TCOCNN, with a benchmark method in a different application context. To achieve this, we utilize a publicly available dataset from a hydraulic system testbed [6]. Recent research has applied various deep learning techniques to the dataset under investigation [7]–[11]. Prakash et al. [7] employed a 1D CNN model to analyze the pressure difference between two pressure sensors. Huang et al. [12] took a parallel approach by utilizing multiple independent convolutional neural networks to extract features from individual sensors. Furthermore, Berghout et al. [10] introduced a novel neural network model specifically designed to process the extracted features. In a distinct approach, Zhang et al. [9] demonstrated the application of a Transformer model with self-attention, originally trained on natural language, to the task of sensor fusion. Collectively, these studies contribute to the exploration of diverse methodologies for analyzing sensor data and extracting meaningful insights. The primary goal of our comprehensive evaluation is to assess the performance of the air quality model when applied to the field of condition monitoring. In doing so, we aim to address potential challenges and difficulties associated with multiple

arXiv:2308.06987v1 [eess.SP] 14 Aug 2023

sensors of different types, thereby providing valuable insights for future research in this area. The remaining structure of this paper is outlined as follows: Section II describes the materials and methods, including details about the dataset and the utilized convolutional neural network. Section III reports the results, and finally, Section IV presents the conclusions derived from this study.

II. MATERIALS AND METHODS

A. Dataset

This study utilizes a dataset that captures the behavior of a hydraulic system (HS) testbed, which has been specifically designed to simulate various common faults encountered in such systems [6]. The ZeMA¹ dataset includes simulated faults such as decreased cooler performance, main valve switching degradation, internal pump leakage, and accumulator pre-charge pressure reduction with the control system enabling independent adjustment of each fault condition. Fig. 1a provides an illustration of the conditions of the cooler, valve, pump, and accumulator within the dataset, which consists of recordings from 17 sensors over a constant operating cycle lasting 60 seconds as would be typical, e.g. for a hydraulic press operation. These sensors measure process values, including pressure (PS1 - PS6), flow (FS1, FS2), temperature (TS1 - TS5), electrical power (EPS1), and vibration (VS1). Additionally, the dataset includes three virtual sensors, namely cooling efficiency (CE), cooling power (CP), and system efficiency (SE). These virtual sensors are calculated using a physical model that combines various measured values. The dataset comprises sensors of various types, and their sampling rates vary based on the measured parameter. The sampling frequencies range from 1 to 100 Hz, resulting in observations with 60 to 6000 data samples per sensor per cycle. Fig. 1b effectively showcases the distinct characteristics exhibited by two sensors through three cycles, highlighting the multimodality of the sensor data [13]. In the context of the present study, the target is to predict the condition of the accumulator. Specifically, the model aims to classify the hydraulic accumulator pre-charge pressure into categories, i.e., "optimal pressure," "lightly reduced pressure," "severely reduced pressure," and "close to total failure."

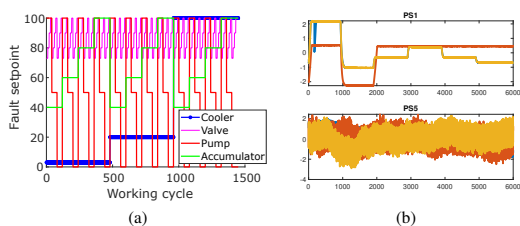


Fig. 1: ZeMA dataset control variables (a), and three cycles of two selected sensors (b).

¹Zentrum für Mechatronik und Automatisierungstechnik gemeinnützige GmbH

B. Algorithms

Conventional ML: The baseline model for this study is selected from [14], where an AutoML toolbox is employed to analyze the dataset. The AutoML toolbox explores various combinations of FESC methods to identify the most effective approach. In this study, we utilize the method identified by the toolbox, which achieved the highest cross-validation accuracy. The selected method involves extracting statistical moments (mean, standard deviation, skewness, and kurtosis) from the raw data as the features. Pearson correlation is used as the feature selector, and linear discriminant analysis with Mahalanobis distance serves as the classifier.

Deep learning methods: Deep learning techniques were employed as the second approach to construct the models. Specifically, two CNN models were utilized: TCOCNN and 2L-CNN, as depicted in Figure 2. TCOCNN is a deep network comprising 10 convolutional layers, and we also evaluated the late fusion version of TCOCNN, which has demonstrated its effectiveness in various multimodal systems [12], [15]. In the 2L-CNN model, each sensor was assigned its own set of convolutional filters. The resulting feature maps from each convolutional lane were concatenated, and a fully connected layer was then applied.

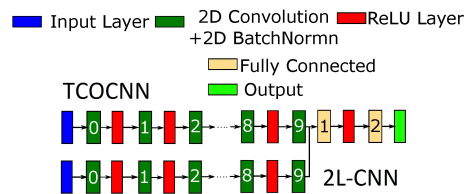


Fig. 2: Architecture of the TCOCNN and 2L-CNN.

In a previous study [5], the employed model was utilized to predict multiple gas concentrations, such as acetone, ethanol, formaldehyde, toluene, the total concentration of all volatile organic compounds (VOCsum), as well as the inorganic gases carbon monoxide and hydrogen. However, in the present study, a different approach is adopted as it focuses on a classification task. Consequently, the last fully connected layer and output function differ from those used in [5]. To determine the network's hyperparameters, a hyperparameter (HP) tuning process is employed. Due to computational resource limitations, the HP tuning involves conducting 50 iterations within a predefined search space for the HPs, as detailed in Table I.

To address the issue of multiple sensors having different sampling rates and, thus, lengths, a preprocessing step is employed where all sensors are upsampled to 6000 data points, ensuring the same number of raw data for each sensor. These upsampled sensor readings are then combined to form a 2D matrix, enabling the application of a 2D convolutional network. Following the preprocessing step, a random split is performed on the dataset. The data is divided into three subsets: 70% of the data is allocated for training, 10% for

TABLE I: The hyperparameter ranges for the CNN.

Initial Learning Rate (Log Scale)	Number of Filters (First Two Layers)	Kernel Size (First Two Layers)	Stride Size (First Layer)	Dropout Number of Neurons (FC)	Number of Neurons (FC)
$1 \times 10^{-7} - 1 \times 10^{-4}$	10-100	100-300	100-175	30-50%	500-2500

validation, and 20% for testing. This partitioning ensures that the model is trained on a substantial portion of the data while having separate datasets for validation and testing. The HP tuning is then exclusively performed on the training data, enabling the identification of optimal HPs based on minimizing the validation loss. This ensures the model is fine-tuned and optimized for performance on unseen data.

III. RESULTS

The results obtained from the baseline model, which utilizes conventional methods, showcased excellent performance for the defined task. When all 17 sensors were used as input, the model achieved a test error rate of 1%. The conventional method applies feature extraction to each sensor individually, thereby avoiding any challenges associated with multiple variant sensors.

In contrast, the CNN model incorporating all 17 sensors and the mentioned preprocessing steps exhibited lower performance, achieving an error rate of 20.5%. This outcome highlighted the challenges arising from the multimodal characteristics of the sensors. To highlight this issue, we trained the network separately for each sensor and observed varying error rates. Fig. 3a displays the error results, with PS1 achieving the best error rate of 1.7% and PS2 obtaining the highest error of 73.4%.

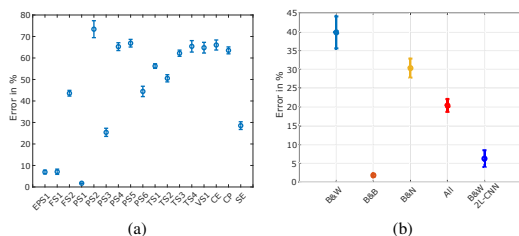


Fig. 3: Test error when only single sensors are used (a), and when combinations of sensors are used (b). Each HP tuning is repeated 5 times, and the error bars represent the range for the standard deviation of the results.

To further investigate the impact of dissimilar sensors on the deep neural network model, we conducted a series of tests using different combinations of sensors. In order to keep the experiments manageable, we focused on two sensors and trained the network using all possible combinations involving the best sensor (PS1) and the remaining sensors. The results are illustrated in Figure 1, where the most notable findings are highlighted. The worst combination, involving PS1 and PS5, exhibited a significantly higher error rate of 39.8% (B&W).

Moreover, when comparing the performance of using the best sensor alone to using it twice (B&B), we observed that increasing the input size (2x6000) did not have a significant impact on the results. However, introducing a second sensor with uniform noise (B&N) as an input resulted in a substantial increase in the error rate, reaching 30.3%.

Fig. 3b provides an illustration of the result achieved by this new model, denoted as "2L-CNN," using the best and worst sensors as input. The new model demonstrated improved performance compared to the normal single-lane network, achieving error rates of 39.8% and 6.5% respectively. This finding provides additional support for the importance of effectively addressing the challenges associated with dissimilar sensors and multimodal learning in order to maintain and improve performance.

IV. CONCLUSION

In conclusion, this study aimed to assess the performance of a network originally designed for temperature-cycled gas sensors in a different application, namely the fault classification of a hydraulic system, and compare it with conventional methods. The conventional baseline model demonstrated impressive performance in handling the 17 diverse sensors, achieving an outstanding error rate of 1%. This method utilizes independent late data fusion after extracting features individually from each sensor.

In contrast, the CNN model that incorporated all 17 sensors and utilized preprocessing to achieve the same raw data length demonstrated significantly lower performance, resulting in an error rate of 20.5%. This outcome highlighted the challenges arising from the dissimilar characteristics of the sensors, which hindered the network's ability to effectively handle them. Notably, training the network separately for each sensor revealed substantial variations in error rates, ranging from 1.7% for PS1 to 73.4% for PS2.

To emphasize the importance of pertinent input data, we performed experiments using various sensor combinations. These tests provided clear evidence that incorporating irrelevant sensors in the input data significantly compromised the results. Additionally, we evaluated the performance of a 2L-CNN model that utilizes late-sensor fusion, which proved to be a viable strategy for addressing the issues caused by irrelevant sensors. This discovery underscores the crucial significance of meticulously choosing and prioritizing relevant sensors to enhance the model's performance.

Overall, this study underscores the challenges posed by the multimodal nature of sensor data. It emphasizes the significance of effectively addressing these challenges to unlock the full potential of sensor-based applications and enhance their overall performance. Future research endeavors can focus on

exploring advanced techniques to overcome these obstacles and further enhance the accuracy of fault detection models.

REFERENCES

- [1] T. Schneider, S. Klein, and A. Schütze, "Machine learning in industrial measurement technology for detection of known and unknown faults of equipment and sensors," *Technisches Messen*, vol. 86, pp. 706–718, 11 2019.
- [2] S. Namuduri, B. N. Narayanan, V. S. P. Davuluru, L. Burton, and S. Bhansali, "Review—deep learning methods for sensor based predictive maintenance and future perspectives for electrochemical sensors," *Journal of The Electrochemical Society*, vol. 167, p. 037552, 2 2020.
- [3] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0888327018303108>
- [4] R. Magar, L. Ghule, J. Li, Y. Zhao, and A. B. Farimani, "Faultnet: A deep convolutional neural network for bearing fault classification," *IEEE Access*, vol. 9, pp. 25 189–25 199, 2021.
- [5] Y. Robin, J. Amann, T. Baur, P. Goodarzi, C. Schultealbert, T. Schneider, and A. Schütze, "High-performance voc quantification for iaq monitoring using advanced sensor systems and deep learning," *Atmosphere*, vol. 12, p. 1487, 11 2021.
- [6] N. Helwig, E. Pignanelli, and A. Schutze, "Condition monitoring of a complex hydraulic system using multivariate statistics." IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, 5 2015, pp. 210–215.
- [7] J. Prakash, S. Singh, A. Miglani, and P. K. Kankar, "Pressure signal-based analysis of anomalies in switching behavior of a two-way directional control valve," *ASME Open Journal of Engineering*, vol. 2, 1 2023.
- [8] S. Pillai and P. Vadakkepat, "Deep learning for machine health prognostics using kernel-based feature transformation," *Journal of Intelligent Manufacturing*, vol. 33, pp. 1665–1680, 8 2022.
- [9] Z. Zhang, M. Farnsworth, B. Song, D. Tiwari, and A. Tiwari, "Deep transfer learning with self-attention for industry sensor fusion tasks," *IEEE Sensors Journal*, vol. 22, pp. 15 235–15 247, 8 2022.
- [10] T. Berghout, M. Benbouzid, S. M. Muyeen, T. Bentrchia, and L.-H. Mouss, "Auto-nahl: A neural network approach for condition-based maintenance of complex industrial systems," *IEEE Access*, vol. 9, pp. 152 829–152 840, 2021.
- [11] X. Ma, P. Wang, B. Zhang, and M. Sun, "A multirate sensor information fusion strategy for multitask fault diagnosis based on convolutional neural network," *Journal of Sensors*, vol. 2021, pp. 1–17, 6 2021.
- [12] K. Huang, S. Wu, F. Li, C. Yang, and W. Gui, "Fault diagnosis of hydraulic systems based on deep learning model with multirate data samples," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, pp. 6789–6801, 11 2022.
- [13] T. Baltrusaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, pp. 423–443, 2 2019.
- [14] T. Schneider, N. Helwig, and A. Schütze, "Industrial condition monitoring with smart sensors using automated feature extraction and selection," *Measurement Science and Technology*, vol. 29, p. 94002, 8 2018. [Online]. Available: <https://doi.org/10.1088/1361-6501/aad1d4>
- [15] H. R. V. Joze, A. Shaban, M. L. Iuzzolino, and K. Koishida, "Mmtm: Multimodal transfer module for cnn fusion." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13 289–13 299.

4 Conclusion

This work aimed to develop a new DL-based calibration strategy for MOS gas sensors. The main focus was to show the capabilities of the DL model TCOCNN regarding laboratory and field tests and tackle long calibration times. A significant reduction of the calibration time is necessary to commercialize MOS gas sensors for selectively measuring harmful VOCs for IAQ monitoring. Within the first publication, the TCOCNN was introduced for gas sensor calibration. The TCOCNN showed superior performance over classic ML (e.g., FESR) regarding prediction quality under laboratory conditions and field tests. This was demonstrated by achieving much smaller RMSE values under laboratory conditions and predicting the gas concentration with less noise and a more realistic baseline during the field tests. This superiority was further improved with the help of transfer learning. The second paper showed that suitable models can be built with only 93 % of the data. As an additional result, it was revealed that this process can be optimized with additional hyperparameters that arise from the application of this method. Hyperparameters can be the transfer learning method (e.g., fine-tuning, freezing), the initial model, the number of UGMs, and the specific UGMs themselves. Subsequently, the third paper demonstrated that the results could be further improved with the help of global initial modeling. With this approach, it was even possible to reduce the calibration time by up to 99.3 %, which indicates that only five UGMs (~ 120 minutes) are sufficient to find a model that provides similar results to the models trained with 600 - 700 UGMs ($\sim 1 - 2$ weeks).³² Finally, those results were compared with state-of-the-art classic ML and calibration transfer methods. It was shown that the TCOCNN can outperform classic ML regarding sensor-to-sensor generalization. Furthermore, transfer learning showed that it is the best approach for calibration transfer by surpassing state-of-the-art methods like Direct Standardization (DS), Piecewise Direct Standardization (PDS), and global model building. In the additional fourth paper, a common issue of DL was addressed. Usually, DL is faced with skepticism if classical ML achieves similar results,

³²The model trained with 700 UGMs without transfer learning achieved an RMSE of 15 ppb, while the model with transfer learning (5 UGMs) achieved an RMSE of 17.7 ppb.

as FESR methods tend to be easier to understand. Therefore, different approaches from XAI were introduced to tackle this problem. It was shown that with advanced XAI methods, it is likely to obtain deep insights into the model. It was possible to gain a more profound understanding of which parts of the TC and sub-sensor are essential for which gas and even allow for verification. This paper showed that optimizing the TC for specific gases can reduce the calibration time by around 50 % if the TC is shortened. However, it should be mentioned that some methods from XAI likewise work for the FESR approach and are, therefore, universally applicable. The last paper showed how the TCOCNN can be adapted for other use cases like condition monitoring. Within this publication, it was discovered that although similar problems are faced, more than a simple one-by-one adaptation is needed. This was obvious if sensors from different domains were mixed to create an input image for the TCOCNN. It was shown that it is necessary to make further improvements to the TCOCNN to apply to condition monitoring and compete with state-of-the-art approaches (e.g., FESC).

Concentrating on the results of the core papers regarding gas sensor data, the summary above shows that it is possible, with the help of advanced DL techniques, to calibrate MOS gas sensors successfully and to significantly tackle the drawback of long calibration times for IAQ assessments. Transfer learning enables calibration transfer between sensors, simultaneously allowing for shorter calibration times. The newly developed method surpasses state-of-the-art approaches and suggests that it is possible to calibrate a large batch of sensors quickly (120 minutes or five gas tests). Two hours or five gas tests can be deemed suitable for industry as two hours are adequate for commercial applications, and with five test gases, no complex UGMs are needed (no GMA needed). With the necessary global initial models, it is possible to quickly deploy the MOS gas sensors and the trained TCOCNNs to a wide range of use cases where VOCs are the primary target. This is interesting for the industry as it allows for a wider field of customers, ranging from monitoring outdoor air quality in industrial applications to consumer use cases like IAQ monitoring and health monitoring. Furthermore, the transfer learning approach can be used to transfer between datasets, which would leverage DL-based MOS gas sensor calibration even further. However, this thesis only builds the basis for future development of gas sensor calibration. Many open questions still need to be discussed, like the effect of unknown interfering gases, drift over time, and sensor poisoning. Possible extensions are discussed in the following chapter.

5 Outlook

Although this thesis already showed the impact that DL can have on the field of MOS gas sensor applications, it only scraped roughly on the surface of what is possible with the help of DL for MOS gas sensor calibration. In the following, the open research task will be elaborated. One open question from the first paper is the absolute accuracy of the FESR and TCOCNN methods in real-world applications. Therefore, a new dataset with laboratory calibration and field tests is required. During the field test, calibrated reference instruments are mandatory to be able to rate the ML models. From the first paper, another open question is whether detecting multiple gases from one chemical group with a model trained only on one representative (unknown interfering gases) is possible. This has currently only been done with two examples with inconsistent results.

The second and third papers leave the optimal method for selecting transfer samples open. For this task, it is suggested to test different methods, like random sub-sampling and the Kennard-Stone algorithm. Likewise, the full potential of hyperparameter selection for transfer learning, e.g., learning rate, freezing vs. fine-tuning, and learning drop factor, must be further explored. Another open question from those two papers is whether achieving similar transfer results between GMAs and datasets is possible and whether the calibration model would work with data recorded in a different laboratory with additional unknown interfering gases. Cross-GMA and laboratory tests are necessary to validate the effectiveness of the newly developed methods. Although papers 2 and 3 already compared the newly developed method with state-of-the-art approaches, more models and techniques should be compared to establish a rating between the different strategies. Another point that was not thoroughly covered in this thesis was drift compensation. It should be investigated if transfer learning can reduce this effect as well. Therefore, an initial model should be trained on data containing drift. Afterward, the transfer should happen with a few transfer samples that do not contain drift. After multiple weeks, the model performance can be evaluated with data collected from the new sensor. Similarly, sensor poisoning should be analyzed to prevent undetected sensor

failure. To solve this problem, it might be possible to apply novelty detection that detects a sudden and unexpected change in the sensor response pattern.

Regarding the fourth paper, only the general concept of XAI was introduced, and therefore, much room for research is present. Possible continuations can be to generate new TCs based on the insights obtained by XAI and test their performance compared to the long cycles used in this publication. Another possibility would be to generate new features for FESR based on the importance scores. Furthermore, a comprehensive study can be conducted on the effectiveness of different XAI methods for the optimization of gas sensor calibration. In this context, the XAI methods can be used to understand the sensor, which might also help to understand the different responses of the gas sensor to different environments (e.g., different gases from the same chemical group, drift).

In the fifth paper, the realm of condition monitoring was introduced. Neural networks are already widely used in this field. However, they also suffer from domain shifts similar to sensor-to-sensor variance. A possible continuation will be to analyze if it is possible to use methods developed within this thesis to tackle significant problems within condition monitoring.

The following discusses a few DL methods that can help to improve the ML models. A data augmentation approach is often used for DL and describes the generation of additional samples with the help of an error model. An example for gas sensor data could be to model the variance in sensitivity between sensors and generate new samples that contain this variation [261]. Although some different neural networks have already been used for gas sensor calibration, no study has been conducted that analyzes different neural network architectures for gas sensor calibration. This can be done using one of the introduced datasets and testing different architectures (e.g., LSTMs, ResNet, Transformer architecture, MLPs). Simultaneously, the performance of decision trees and transfer learning should be tested. The computational complexity might be one of the most critical challenges for deploying the TCOCNN. It needs to be analyzed if it is possible to deploy the TCOCNN on a microcontroller and how powerful the microcontroller needs to be. In this context, the truncation and quantization of weights need to be analyzed.

A general open topic, not yet thoroughly analyzed, is the usable concentration range for the MOS gas sensors in combination with the different calibration schemes and ML models.

References

- [1] GBD 2019 Risk Factors Collaborators. “Global burden of 87 risk factors in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019.” In: *The Lancet* (Oct. 2020), pp. 1223–1249. ISSN: 01406736. DOI: 10.1016/S0140-6736(20)30752-2.
- [2] Department of Economic and Sustainable Development United Nations Social Affairs. *Ensure healthy lives and promote well-being for all at all ages. (accessed on 15 October 2021)*. URL: <https://sdgs.un.org/goals/goal3>.
- [3] United States Environmental Protection Agency. *Indoor Air Quality. (accessed on 20 April 2023)*. URL: <https://www.epa.gov/report-environment/indoor-air-quality>.
- [4] S. Brasche and W. Bischof. “Daily time spent indoors in German homes – Baseline data for the assessment of indoor exposure of German occupants.” In: *International Journal of Hygiene and Environmental Health* 208 (4 July 2005), pp. 247–253. ISSN: 14384639. DOI: 10.1016/j.ijheh.2005.03.003.
- [5] M. Mannan and S. Al-Ghamdi. “Indoor Air Quality in Buildings: A Comprehensive Review on the Factors Influencing Air Pollution in Residential and Commercial Structure.” In: *International Journal of Environmental Research and Public Health* 18 (6 Mar. 2021), p. 3276. ISSN: 1660-4601. DOI: 10.3390/ijerph18063276.
- [6] D. Sarigiannis et al. “Exposure to major volatile organic compounds and carbonyls in European indoor environments and associated health risk.” In: *Environment International* 37 (4 May 2011), pp. 743–765. ISSN: 01604120. DOI: 10.1016/j.envint.2011.01.005.
- [7] M. Pettenkofer. “Über den Luftwechsel in Wohngebäuden.” Literarisch-Artistische Anstalt der J.G. Cotta’schen Buchhandlung, 2018. (accessed on 20 April 2023). URL: <https://opacplus.bsb-muenchen.de/title/BV013009721>.

- [8] T. Salthammer. “Very volatile organic compounds: an understudied class of indoor air pollutants.” In: *Indoor Air* 26 (1 Feb. 2016), pp. 25–38. ISSN: 0905-6947. DOI: 10.1111/ina.12173.
- [9] United States Environmental Protection Agency. *The Inside Story: A Guide to Indoor Air Quality* (accessed on 29 August 2023). URL: <https://www.epa.gov/indoor-air-quality-iaq/inside-story-guide-indoor-air-quality>.
- [10] P. Pandey and R. Yadav. “A Review on Volatile Organic Compounds (VOCs) as Environmental Pollutants: Fate and Distribution.” In: *International Journal Of Plant And Environment* 4 (02 July 2018), pp. 14–26. ISSN: 2455-202X. DOI: 10.18811/ijpen.v4i02.2.
- [11] H. Wang and J. Xiong. “Very Volatile Organic Compounds (VVOCs).” In: *Zhang, Y., Hopke, P.K., Mandin, C. (eds) Handbook of Indoor Air Quality. Springer, Singapore* (2022), pp. 37–69. DOI: 10.1007/978-981-16-7680-2_3.
- [12] L. Lucattini et al. “A review of semi-volatile organic compounds (SVOCs) in the indoor environment: occurrence in consumer products, indoor air and dust.” In: *Chemosphere* 201 (June 2018), pp. 466–482. ISSN: 00456535. DOI: 10.1016/j.chemosphere.2018.02.161.
- [13] M. Hauptmann et al. “Mortality from Solid Cancers among Workers in Formaldehyde Industries.” In: *American Journal of Epidemiology* 159 (12 June 2004), pp. 1117–1130. ISSN: 0002-9262. DOI: 10.1093/aje/kwh174.
- [14] M. Yeoman et al. “Simplified speciation and atmospheric volatile organic compound emission rates from non-aerosol personal care products.” In: *Indoor Air* 30 (3 May 2020), pp. 459–472. ISSN: 0905-6947. DOI: 10.1111/ina.12652.
- [15] T. Baur et al. “Field Study of Metal Oxide Semiconductor Gas Sensors in Temperature Cycled Operation for Selective VOC Monitoring in Indoor Air.” In: *Atmosphere* 12 (5 May 2021), p. 647. ISSN: 2073-4433. DOI: 10.3390/atmos12050647.
- [16] S. Feng et al. “Review on Smart Gas Sensing Technology.” In: *Sensors* 19 (17 Aug. 2019), p. 3760. ISSN: 1424-8220. DOI: 10.3390/s19173760.
- [17] P. Reimann and A. Schütze. “Sensor Arrays, Virtual Multisensors, Data Fusion, and Gas Sensor Data Evaluation.” In: *Kohl, CD., Wagner, T. (eds) Gas Sensing Fundamentals. Springer Series on Chemical Sensors and Biosensors, vol 15. Springer, Berlin, Heidelberg* (2013), pp. 67–107. DOI: 10.1007/5346_2013_52.

-
- [18] C. Schultealbert et al. “A novel approach towards calibrated measurement of trace gases using metal oxide semiconductor sensors.” In: *Sensors and Actuators B: Chemical* 239 (Feb. 2017), pp. 390–396. ISSN: 09254005. DOI: 10.1016/j.snb.2016.08.002.
- [19] A. Schütze et al. “Highly Sensitive and Selective VOC Sensor Systems Based on Semiconductor Gas Sensors: How to?” In: *Environments* 4 (1 Mar. 2017), p. 20. ISSN: 2076-3298. DOI: 10.3390/environments4010020.
- [20] R. Laref et al. “Calibration Transfer to Address the Long Term Drift of Gas Sensors for in Field NO₂ Monitoring.” In: *2021 International Conference on Control, Automation and Diagnosis (ICCAD), Grenoble, France, 03-05 November (2021)*, pp. 1–6. DOI: 10.1109/ICCAD52417.2021.9638737.
- [21] C. Bur et al. “Drift compensation of virtual multisensor systems based on extended calibration.” In: *IMCS2014 - the 15th International Meeting on Chemical Sensors (poster presentation), Buenos Aires, Argentina, 16-19 March (2014)*.
- [22] T. Artursson et al. “Drift correction for gas sensors using multivariate methods.” In: *Journal of Chemometrics* 14 (5-6 Sept. 2000), pp. 711–723. ISSN: 0886-9383. DOI: 10.1002/1099-128X(200009/12)14:5/6<711::AID-CEM607>3.0.CO;2-4.
- [23] T. Baur, A. Schütze, and T. Sauerwald. “Optimierung des temperaturzyklischen Betriebs von Halbleitergassensoren.” In: *tm - Technisches Messen* 82 (4 Apr. 2015), pp. 187–195. ISSN: 0171-8096. DOI: 10.1515/teme-2014-0007.
- [24] T. Baur et al. “Random gas mixtures for efficient gas sensor calibration.” In: *Journal of Sensors and Sensor Systems* 9 (2 Nov. 2020), pp. 411–424. ISSN: 2194-878X. DOI: 10.5194/jsss-9-411-2020.
- [25] A. Rudnitskaya. “Calibration Update and Drift Correction for Electronic Noses and Tongues.” In: *Frontiers in Chemistry* 6 (Sept. 2018). ISSN: 2296-2646. DOI: 10.3389/fchem.2018.00433.
- [26] N. Goel et al. “Metal oxide semiconductors for gas sensing.” In: *Engineering Reports* 5 (6 June 2023). ISSN: 2577-8196. DOI: 10.1002/eng2.12604.
- [27] A. Wilson. “Review of Electronic-nose Technologies and Algorithms to Detect Hazardous Chemicals in the Environment.” In: *Procedia Technology* 1 (2012), pp. 453–463. ISSN: 22120173. DOI: 10.1016/j.protcy.2012.02.101.
- [28] X. Liu et al. “A Survey on Gas Sensing Technology.” In: *Sensors* 12 (7 July 2012), pp. 9635–9665. ISSN: 1424-8220. DOI: 10.3390/s120709635.
-

- [29] T. Seiyama et al. “A New Detector for Gaseous Components Using Semiconductive Thin Films.” In: *Analytical Chemistry* 34 (11 Oct. 1962), pp. 1502–1503. ISSN: 0003-2700. DOI: 10.1021/ac60191a001.
- [30] A. Tricoli, M. Righettoni, and S. Pratsinis. “Minimal cross-sensitivity to humidity during ethanol detection by SnO₂–TiO₂ solid solutions.” In: *Nanotechnology* 20 (31 Aug. 2009), p. 315502. ISSN: 0957-4484. DOI: 10.1088/0957-4484/20/31/315502.
- [31] G. Lei et al. “Thin films of tungsten oxide materials for advanced gas sensors.” In: *Sensors and Actuators B: Chemical* 341 (Aug. 2021), p. 129996. ISSN: 09254005. DOI: 10.1016/j.snb.2021.129996.
- [32] H. Ogawa, M. Nishikawa, and A. Abe. “Hall measurement studies and an electrical conduction model of tin oxide ultrafine particle films.” In: *Journal of Applied Physics* 53 (6 June 1982), pp. 4448–4455. ISSN: 0021-8979. DOI: 10.1063/1.331230.
- [33] R. Singh et al. “Synthesis of zinc oxide nanorods and nanoparticles by chemical route and their comparative study as ethanol sensors.” In: *Sensors and Actuators B: Chemical* 135 (1 Dec. 2008), pp. 352–357. ISSN: 09254005. DOI: 10.1016/j.snb.2008.09.004.
- [34] T. Kim et al. “Drastic Gas Sensing Selectivity in 2-Dimensional MoS₂ Nanoflakes by Noble Metal Decoration.” In: *ACS Nano* 17 (5 Mar. 2023), pp. 4404–4413. ISSN: 1936-0851. DOI: 10.1021/acsnano.2c09733.
- [35] C. Na et al. “Transformation of ZnO Nanobelts into Single-Crystalline Mn₃O₄ Nanowires.” In: *ACS Applied Materials and Interfaces* 4 (12 Dec. 2012), pp. 6565–6572. ISSN: 1944-8244. DOI: 10.1021/am301670x.
- [36] R. Faleh et al. “A Transient Signal Extraction Method of WO₃ Gas Sensors Array to Identify Pollutant Gases.” In: *IEEE Sensors Journal* 16 (9 May 2016), pp. 3123–3130. ISSN: 1530-437X. DOI: 10.1109/JSEN.2016.2521578.
- [37] A. Gramm and A. Schütze. “High performance solvent vapor identification with a two sensor array using temperature cycling and pattern classification.” In: *Sensors and Actuators B: Chemical* 95 (Oct. 2003), pp. 58–65. ISSN: 09254005. DOI: 10.1016/S0925-4005(03)00404-0.

-
- [38] Y. Kato, K. Yoshikawa, and M. Kitora. “Temperature-dependent dynamic response enables the qualification and quantification of gases by a single sensor.” In: *Sensors and Actuators B: Chemical* 40 (1 May 1997), pp. 33–37. ISSN: 09254005. DOI: 10.1016/S0925-4005(97)80196-7.
- [39] G. Neri. “First Fifty Years of Chemosensitive Gas Sensors.” In: *Chemosensors* 3 (1 Jan. 2015), pp. 1–20. ISSN: 2227-9040. DOI: 10.3390/chemosensors3010001.
- [40] H. Kim and J. Lee. “Highly sensitive and selective gas sensors using p-type oxide semiconductors: Overview.” In: *Sensors and Actuators B: Chemical* 192 (Mar. 2014), pp. 607–627. ISSN: 09254005. DOI: 10.1016/j.snb.2013.11.005.
- [41] P. Raju and Q. Li. “Review—Semiconductor Materials and Devices for Gas Sensors.” In: *Journal of The Electrochemical Society* 169 (5 May 2022), p. 057518. ISSN: 0013-4651. DOI: 10.1149/1945-7111/ac6e0a.
- [42] U. Nakate et al. “Hydrothermal synthesis of p-type nanocrystalline NiO nanoplates for high response and low concentration hydrogen gas sensor application.” In: *Ceramics International* 44 (13 Sept. 2018), pp. 15721–15729. ISSN: 02728842. DOI: 10.1016/j.ceramint.2018.05.246.
- [43] K. Christmann. *Introduction to Surface Physical Chemistry*. Vol. 1. Steinkopff, 1991. ISBN: 978-3-7985-0858-3. DOI: 10.1007/978-3-662-08009-2.
- [44] A. Adamson and A. Gast. *Physical chemistry of surfaces*. Vol. 150. Interscience publishers New York, 1967.
- [45] N. Sui et al. “The effect of different crystalline phases of In₂O₃ on the ozone sensing performance.” In: *Journal of Hazardous Materials* 418 (Sept. 2021), p. 126290. ISSN: 03043894. DOI: 10.1016/j.jhazmat.2021.126290.
- [46] T. Sauerwald, T. Baur, and A. Schütze. “Strategien zur Optimierung des temperaturzyklischen Betriebs von Halbleitersensoren.” In: *Tagungsband, AHMT 2014 - Symposium des Arbeitskreises der Hochschullehrer für Messtechnik, Saarbrücken, Germany, 18-20 September* (2014), pp. 65–74. DOI: 10.5162/AHMT2014/3.1.
- [47] P. Peterson et al. “Practical Use of Metal Oxide Semiconductor Gas Sensors for Measuring Nitrogen Dioxide and Ozone in Urban Environments.” In: *Sensors* 17 (7 July 2017), p. 1653. ISSN: 1424-8220. DOI: 10.3390/s17071653.
- [48] J. Gardner. “A diffusion-reaction model of electrical conduction in tin oxide gas sensors.” In: *Semiconductor Science and Technology* 4 (5 May 1989), pp. 345–350. ISSN: 0268-1242. DOI: 10.1088/0268-1242/4/5/003.
-

- [49] M. Bastuck, T. Baur, and A. Schütze. “DAV3E – a MATLAB toolbox for multivariate sensor data evaluation.” In: *Journal of Sensors and Sensor Systems* 7 (2 Sept. 2018), pp. 489–506. ISSN: 2194-878X. DOI: 10.5194/jsss-7-489-2018.
- [50] C. Schultealbert et al. “Facile Quantification and Identification Techniques for Reducing Gases over a Wide Concentration Range Using a MOS Sensor in Temperature-Cycled Operation.” In: *Sensors* 18 (3 Mar. 2018), p. 744. ISSN: 1424-8220. DOI: 10.3390/s18030744.
- [51] S. Chiu and K. Tang. “Towards a Chemiresistive Sensor-Integrated Electronic Nose: A Review.” In: *Sensors* 13 (10 Oct. 2013), pp. 14214–14247. ISSN: 1424-8220. DOI: 10.3390/s131014214.
- [52] A. Schütze. “Präparation und Charakterisierung von Phthalocyanin-Schichten zum Nachweis oxidierender und reduzierender Gase.” Doctoral Dissertation, Shaker Verlag GmbH: Berichte aus der Physik, Aachen Germany, 1995. ISBN: 3826504860.
- [53] J. Yan et al. “Electronic Nose Feature Extraction Methods: A Review.” In: *Sensors* 15 (11 Nov. 2015), pp. 27804–27831. ISSN: 1424-8220. DOI: 10.3390/s151127804.
- [54] A. Schütze, N. Pieper, and J. Zacheja. “Quantitative ozone measurement using a Pc thin film sensor and dynamic signal evaluation.” In: *Sensors and Actuators B* 23 (1995) 215 - 218. presented at: *International Workshop "New Developments in Semiconducting Gas Sensors", Castro Marina, Italy, 13. - 14. September (1993)*.
- [55] S. De Vito, G. D’Elia, and G. Di Francia. “Global calibration models match ad-hoc calibrations field performances in low cost particulate matter sensors.” In: *2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN) 29 May - 01 June (2022)*. DOI: 10.1109/ISOEN54820.2022.9789669.
- [56] B. Shi et al. “Optimization of electronic nose sensor array by genetic algorithms in Xihu-Longjing Tea quality analysis.” In: *Mathematical and Computer Modelling* 58 (Aug. 2013), pp. 752–758. ISSN: 08957177. DOI: 10.1016/j.mcm.2012.12.029.
- [57] H. Vine. “Method and apparatus for operating a gas sensor.” In: *Patent: US3906473A, Figaro Engineering Inc Matheson Gas Products Inc, accessed 20 July 2023 (1974)*. URL: <https://patents.google.com/patent/US3906473A/en#patentCitations>.

-
- [58] K. Ngo, P. Lauque, and K. Aguir. “High performance of a gas identification system using sensor array and temperature modulation.” In: *Sensors and Actuators B: Chemical* 124 (1 June 2007), pp. 209–216. ISSN: 09254005. DOI: 10.1016/j.snb.2006.12.028.
- [59] A. Schütze and T. Sauerwald. “Dynamic operation of semiconductor sensors.” In: *Semiconductor Gas Sensors (Second Edition)*, In Woodhead Publishing Series in Electronic and Optical Materials (2020), pp. 385–412. DOI: 10.1016/B978-0-08-102559-8.00012-4.
- [60] M. Leidinger et al. “Selective detection of hazardous VOCs for indoor air quality applications using a virtual gas sensor array.” In: *Journal of Sensors and Sensor Systems* 3 (2 Oct. 2014), pp. 253–263. ISSN: 2194-878X. DOI: 10.5194/jsss-3-253-2014.
- [61] M. Bastuck et al. “Improved quantification of naphthalene using non-linear Partial Least Squares Regression.” In: *ISOEN 2015, 16th International Symposium on Olfaction and Electronic Noses, Dijon, France, 28 June - 01 July* (2015).
- [62] J. Radhakrishnan and M. Kumara. “Effect of temperature modulation, on the gas sensing characteristics of ZnO nanostructures, for gases O₂, CO and CO₂.” In: *Sensors International* 2 (2021), p. 100059. ISSN: 26663511. DOI: 10.1016/j.sintl.2020.100059.
- [63] A. Kobald, U. Weimar, and N. Barsan. “Regression Model for the Prediction of Pollutant Gas Concentrations with Temperature Modulated Gas Sensors.” In: *2022 IEEE International Symposium on Olfaction and Electronic Nose (ISOEN), Aveiro, Portugal, 29 May - 01 June* (2022). DOI: 10.1109/ISOEN54820.2022.9789650.
- [64] J. Cervera Gómez, J. Pelegri-Sebastia, and R. Lajara. “Circuit Topologies for MOS-Type Gas Sensor.” In: *Electronics* 9 (3 Mar. 2020), p. 525. ISSN: 2079-9292. DOI: 10.3390/electronics9030525.
- [65] C. Fuchs et al. “Concept and realization of a modular and versatile platform for metal oxide semiconductor gas sensors.” In: *tm - Technisches Messen* 89 (12 Dec. 2022), pp. 859–874. ISSN: 2196-7113. DOI: 10.1515/teme-2022-0046.
- [66] H. Endres, H. Jander, and W. Göttler. “A test system for gas sensors.” In: *Sensors and Actuators B: Chemical* 23 (Feb. 1995), pp. 163–172. ISSN: 09254005. DOI: 10.1016/0925-4005(94)01272-J.
-

- [67] L. Spinelle, M. Aleixandre, and M. Gerboles. *Protocol of evaluation and calibration of low-cost gas sensors for the monitoring of air pollution*. EUR 26112. Luxembourg (Luxembourg): Publications Office of the European Union; 2013. JRC83791, 2013.
- [68] M. Bastuck. “Improving the performance of gas sensor systems with advanced data evaluation, operation, and calibration methods.” Doctoral Dissertation, Saarland University, Lab for Measurement Technology, 2019. DOI: doi:10.22028/D291-29811.
- [69] S. De Vito et al. “On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario.” In: *Sensors and Actuators B: Chemical* 129 (2 Feb. 2008), pp. 750–757. ISSN: 09254005. DOI: 10.1016/j.snb.2007.09.060.
- [70] WHO Team Guidelines Review Committee. *WHO guidelines for indoor air quality: selected pollutants*. Ed. by World Health Organization (accessed on 17 October 2023). 2010. URL: <https://iris.who.int/bitstream/handle/10665/260127/9789289002134-eng.pdf?sequence=1>.
- [71] T. Sauerwald et al. “Highly sensitive benzene detection with metal oxide semiconductor gas sensors – an inter-laboratory comparison.” In: *Journal of Sensors and Sensor Systems* 7 (1 Apr. 2018), pp. 235–243. ISSN: 2194-878X. DOI: 10.5194/jsss-7-235-2018.
- [72] J. Gómez et al. “Design and Implementation of a Gas Generating System for Complex Gas Mixtures and Calibration Gases.” In: *Chemie Ingenieur Technik* 92 (10 Oct. 2020), pp. 1574–1585. ISSN: 0009-286X. DOI: 10.1002/cite.202000110.
- [73] J. Fonollosa et al. “Calibration transfer and drift counteraction in chemical sensor arrays using Direct Standardization.” In: *Sensors and Actuators B: Chemical* 236 (Nov. 2016), pp. 1044–1053. ISSN: 09254005. DOI: 10.1016/j.snb.2016.05.089.
- [74] ISO 6145-1:2019. “Gas analysis - Preparation of calibration gas mixtures using dynamic methods - Part 1: General aspects.” In: *International Organization for Standardization, Geneva, Switzerland* (2019), pp. 1–23. URL: <https://www.iso.org/standard/53598.html>.
- [75] N. Helwig et al. “Gas mixing apparatus for automated gas sensor characterization.” In: *Measurement Science and Technology* 25 (5 May 2014), p. 055903. ISSN: 0957-0233. DOI: 10.1088/0957-0233/25/5/055903.

-
- [76] M. Leidinger et al. “Characterization and calibration of gas sensor systems at ppb level—a versatile test gas generation system.” In: *Measurement Science and Technology* 29 (1 Jan. 2018), p. 015901. ISSN: 0957-0233. DOI: 10.1088/1361-6501/aa91da.
- [77] D. Arendes et al. “Modular design of a gas mixing apparatus for complex trace gas mixtures.” In: *15. Dresdner Sensor-Symposium, AMA Service GmbH, Dresden, Germany, 06 - 08 December (2021)*. DOI: 10.5162/15dss2021/P13.1.
- [78] D. Arendes et al. “Qualification of a Gas Mixing Apparatus for Complex Trace Gas Mixtures.” In: *16. Dresdner Sensor-Symposium AMA Service GmbH, Dresden, Germany, 05 - 07 December (2022)*, pp. 183–188. DOI: 10.5162/16dss2022/P35.
- [79] Schmidlin Labor + Service GmbH. *Gt-plus-30000-ultra-zero-air-generator (accessed on 19 July 2023)*. URL: <https://www.vici-dbs.com/products/gt-plus-30000-ultra-zero-air-generator>.
- [80] F. Viana. “A Tutorial on Latin Hypercube Design of Experiments.” In: *Quality and Reliability Engineering International* 32 (5 July 2016), pp. 1975–1985. ISSN: 07488017. DOI: 10.1002/qre.1924.
- [81] Y. Choi et al. “Comparison of Factorial and Latin Hypercube Sampling Designs for Meta-Models of Building Heating and Cooling Loads.” In: *Energies* 14 (2 Jan. 2021), p. 512. ISSN: 1996-1073. DOI: 10.3390/en14020512.
- [82] C. Schnur, S. Klein, and A. Blum. *Checklist – Measurement and data planning for machine learning in assembly (version 7)*. Zenodo, 2023. DOI: 10.5281/zenodo.7556876.
- [83] A. Jain et al. “Overview and Importance of Data Quality for Machine Learning Tasks.” In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 2020), pp. 3561–3562. DOI: 10.1145/3394486.3406477.
- [84] C. Bur et al. “On the Detection of Propofol in Exhaled Air by MOS Gas Sensors.” In: *tm - Technisches Messen* 89 (Sept. 2022), pp. 66–71. ISSN: 2196-7113. DOI: 10.1515/teme-2022-0060.
- [85] H. Hofmann and P. Plieninger. “Bereitstellung einer Datenbank zum Vorkommen von flüchtigen organischen Verbindungen in der Raumluft.” In: *Forschungsbericht 205 61 243*. Hrsg.: *Arbeitsgemeinschaft ökologischer Forschungsinstitute (AGÖF)*
-

- e. V, (accessed on 16 October 2023) (2008). URL: <https://www.umweltbundesamt.de/publikationen/bereitstellung-einer-datenbank-vorkommen-von>.
- [86] H. Hofmann and G. Erdmann. “Zielkonflikt energieeffiziente Bauweise und gute Raumluftqualität–Datenerhebung für flüchtige organische Verbindungen in der Innenraumluft von Wohn- und Bürogebäuden (Lösungswege).” In: *Hrsg.: Arbeitsgemeinschaft ökologischer Forschungsinstitute (AGÖF) e. V, Abschlussbericht AGÖF-Forschungsprojekt, (accessed on 16 October 2023)* (2015). URL: <https://www.agoeff.de/forschung/fue-11-voc-datenerhebung/abschlussbericht.html>.
- [87] M. McKay, R. Beckman, and W. Conover. “A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code.” In: *Technometrics* 21 (2 May 1979), p. 239. ISSN: 00401706. DOI: 10.2307/1268522.
- [88] R. Iman, J. Helton, and J. Campbell. “An Approach to Sensitivity Analysis of Computer Models: Part I—Introduction, Input Variable Selection and Preliminary Variable Assessment.” In: *Journal of Quality Technology* 13 (3 July 1981), pp. 174–183. ISSN: 0022-4065. DOI: 10.1080/00224065.1981.11978748.
- [89] B. Tang. “Orthogonal Array-Based Latin Hypercubes.” In: *Journal of the American Statistical Association* 88 (424 1993), pp. 1392–1397. DOI: 10.1080/01621459.1993.10476423. URL: <https://www.tandfonline.com/doi/abs/10.1080/01621459.1993.10476423>.
- [90] G. Baudic et al. “Using emulation to validate applications on opportunistic networks.” In: *Advances in Delay-Tolerant Networks (DTNs)* (2021), pp. 273–280. DOI: 10.1016/B978-0-08-102793-6.00014-X.
- [91] P. Bellan. “Analytic Model for the Time-dependent Electromagnetic Field of an Astrophysical Jet.” In: *The Astrophysical Journal* 888 (2 Jan. 2020), p. 69. ISSN: 1538-4357. DOI: 10.3847/1538-4357/ab5f0d.
- [92] C. Janiesch, P. Zschech, and K. Heinrich. “Machine learning and deep learning.” In: *Electronic Markets* 31 (3 Sept. 2021), pp. 685–695. ISSN: 1019-6781. DOI: 10.1007/s12525-021-00475-2.
- [93] I. Sarker. “Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions.” In: *SN Computer Science* 2 (6 Nov. 2021), p. 420. ISSN: 2662-995X. DOI: 10.1007/s42979-021-00815-1.

-
- [94] S. Sah. “Machine Learning: A Review of Learning Types.” In: *Preprints 2020, 2020070230* (2020). DOI: <https://doi.org/10.20944/preprints202007.0230.v1>.
- [95] J. Yan and X. Wang. “Unsupervised and semi-supervised learning: the next frontier in machine learning for plant systems biology.” In: *The Plant Journal* 111 (6 Sept. 2022), pp. 1527–1538. ISSN: 0960-7412. DOI: [10.1111/tpj.15905](https://doi.org/10.1111/tpj.15905).
- [96] B. Mahesh. “Machine Learning Algorithms -A Review.” In: *International Journal of Science and Research (IJSR)* 9 (1 2019). ISSN: 2319-7064.
- [97] K. Arulkumaran et al. “A Brief Survey of Deep Reinforcement Learning.” In: *IEEE Signal Processing Magazine, Special Issue on Deep Learning for Image Understanding (arXiv extended version)* (Aug. 2017). DOI: [10.1109/MSP.2017.2743240](https://doi.org/10.1109/MSP.2017.2743240).
- [98] L. Alzubaidi et al. “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions.” In: *Journal of Big Data* 8 (1 Mar. 2021), p. 53. ISSN: 2196-1115. DOI: [10.1186/s40537-021-00444-8](https://doi.org/10.1186/s40537-021-00444-8).
- [99] J. van Engelen and H. Hoos. “A survey on semi-supervised learning.” In: *Machine Learning* 109 (2 Feb. 2020), pp. 373–440. ISSN: 0885-6125. DOI: [10.1007/s10994-019-05855-6](https://doi.org/10.1007/s10994-019-05855-6).
- [100] I. Thoidis, M. Giouvanakis, and G. Papanikolaou. “Semi-Supervised Machine Condition Monitoring by Learning Deep Discriminative Audio Features.” In: *Electronics* 10 (20 Oct. 2021), p. 2471. ISSN: 2079-9292. DOI: [10.3390/electronics10202471](https://doi.org/10.3390/electronics10202471).
- [101] M. Delgado-Prieto et al. “Novelty Detection based Condition Monitoring Scheme Applied to Electromechanical Systems.” In: *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), Turin, Italy, 04-07 September* (2018), pp. 1213–1216. DOI: [10.1109/ETFA.2018.8502503](https://doi.org/10.1109/ETFA.2018.8502503).
- [102] Z. Ghahramani. “Unsupervised Learning.” In: *Bousquet, O., von Luxburg, U., Rätsch, G. (eds) Advanced Lectures on Machine Learning. ML 2003. Lecture Notes in Computer Science, vol 3176. Springer, Berlin, Heidelberg* (2004), pp. 72–112. DOI: [10.1007/978-3-540-28650-9_5](https://doi.org/10.1007/978-3-540-28650-9_5).

- [103] D. Pfitzner, R. Leibbrandt, and D. Powers. “Characterization and evaluation of similarity measures for pairs of clusterings.” In: *Knowledge and Information Systems* 19 (3 June 2009), pp. 361–394. ISSN: 0219-1377. DOI: 10.1007/s10115-008-0150-6.
- [104] V. Mnih et al. “Playing Atari with Deep Reinforcement Learning.” In: *NIPS Deep Learning Workshop, Stateline, USA, 05 - 10 December* (2013). DOI: <https://doi.org/10.48550/arXiv.1312.5602>.
- [105] K. Weinberger. “Lecture 1: Supervised Learning.” In: *Online Lecture, (accessed on 16 August 2023)* (2018). URL: http://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote01_MLsetup.html.
- [106] F. Maleki et al. “Machine Learning Algorithm Validation.” In: *Neuroimaging Clinics of North America* 30 (4 Nov. 2020), pp. 433–445. ISSN: 10525149. DOI: 10.1016/j.nic.2020.08.004.
- [107] M. Borg et al. “Safely Entering the Deep: A Review of Verification and Validation for Machine Learning and a Challenge Elicitation in the Automotive Industry.” In: *Journal of Automotive Software Engineering* (Dec. 2018). DOI: <https://doi.org/10.48550/arXiv.1812.05389Focustolearnmore>.
- [108] M. Krousel-Wood, R. Chambers, and P. Muntner. “Clinicians’ guide to statistics for medical practice and research: part I.” In: *Ochsner journal* 6 (2 2006), pp. 68–83. ISSN: 1524-5012.
- [109] T. Wong. “Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation.” In: *Pattern Recognition* 48 (9 Sept. 2015), pp. 2839–2846. ISSN: 00313203. DOI: 10.1016/j.patcog.2015.03.009.
- [110] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer New York, 2009. ISBN: 978-0-387-84857-0. DOI: 10.1007/978-0-387-84858-7.
- [111] X. Ying. “An Overview of Overfitting and its Solutions.” In: *Journal of Physics: Conference Series* 1168 (Feb. 2019), p. 022022. ISSN: 1742-6588. DOI: 10.1088/1742-6596/1168/2/022022.
- [112] R. Mukhamediev et al. “From Classical Machine Learning to Deep Neural Networks: A Simplified Scientometric Review.” In: *Applied Sciences* 11 (12 June 2021), p. 5541. ISSN: 2076-3417. DOI: 10.3390/app11125541.

-
- [113] M. Hearst et al. “Support vector machines.” In: *IEEE Intelligent Systems and their Applications* 13 (4 July 1998), pp. 18–28. ISSN: 1094-7167. DOI: 10.1109/5254.708428.
- [114] M. Seeger. “Gaussian Processes For Machine Learning.” In: *International Journal of Neural Systems* 14 (02 Apr. 2004), pp. 69–106. ISSN: 0129-0657. DOI: 10.1142/S0129065704001899.
- [115] H. Seal. “Studies in the History of Probability and Statistics. XV The historical development of the Gauss linear model.” In: *Biometrika* 54 (1-2 1967), pp. 1–24. ISSN: 0006-3444. DOI: 10.1093/biomet/54.1-2.1.
- [116] P. Utgoff. “Incremental Induction of Decision Trees. Machine Learning.” In: *Machine Learning* 4 (2 1989), pp. 161–186. ISSN: 08856125. DOI: 10.1023/A:1022699900025.
- [117] P. Bühlmann. “Bagging, Boosting and Ensemble Methods.” In: *Gentle, J., Härdle, W., Mori, Y. (eds) Handbook of Computational Statistics. Springer Handbooks of Computational Statistics. Springer, Berlin, Heidelberg* (2012), pp. 985–1022. DOI: 10.1007/978-3-642-21551-3_33.
- [118] R. Shwartz-Ziv and A. Armon. “Tabular Data: Deep Learning is Not All You Need.” In: *Information Fusion* 81 (June 2021). ISSN: 1566-2535. DOI: <https://doi.org/10.1016/j.inffus.2021.11.011>.
- [119] R. Bellman. *Adaptive Control Processes*. Princeton University Press, Princeton, USA, Dec. 1961. ISBN: 9781400874668. DOI: 10.1515/9781400874668.
- [120] T. Schneider, N. Helwig, and A. Schütze. “Industrial condition monitoring with smart sensors using automated feature extraction and selection.” In: *Measurement Science and Technology* 29 (9 Sept. 2018), p. 094002. ISSN: 0957-0233. DOI: 10.1088/1361-6501/aad1d4.
- [121] S. Sagar et al. “Review—Modern Data Analysis in Gas Sensors.” In: *Journal of The Electrochemical Society* 169 (12 Dec. 2022), p. 127512. ISSN: 0013-4651. DOI: 10.1149/1945-7111/aca839.
- [122] J. Heaton. “An Empirical Analysis of Feature Engineering for Predictive Modeling.” In: *SoutheastCon 2016, Norfolk, VA, USA, 30 March - 03 April* (2016). DOI: 10.1109/SECON.2016.7506650.

- [123] A. Jovic, K. Brkic, and N. Bogunovic. “A review of feature selection methods with applications.” In: *38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 25-29 May* (2015). DOI: 10.1109/MIPRO.2015.7160458.
- [124] A. Mangal and E. Holm. “A comparative study of feature selection methods for stress hotspot classification in materials.” In: *Integrating Materials and Manufacturing Innovation* 7 (2018), pp. 87–95. DOI: 10.1007/s40192-018-0109-8.
- [125] T. Dorst et al. “Automated ML Toolbox for Cyclic Sensor Data.” In: *Joint Virtual Workshop of ENBIS and MATHMET Mathematical and Statistical Methods for Metrology MSMM, online, 31 May – 1 June* (2021).
- [126] Y. Meyer. *Wavelets and Operators*. Cambridge University Press, Cambridge, USA, Apr. 1993. ISBN: 9780521420006. DOI: 10.1017/CB09780511623820.
- [127] M. Gruber et al. “Discrete wavelet transform on uncertain data: Efficient online implementation for practical applications.” In: *Advanced Mathematical and Computational Tools in Metrology and Testing XII (Book Chapter)* (2022), pp. 249–261. DOI: 10.1142/9789811242380_0014.
- [128] J. Lange and T. Lange. *Fourier-Transformation zur Signal- und Systembeschreibung*. Springer Fachmedien Wiesbaden, 2019. ISBN: 978-3-658-24849-9. DOI: 10.1007/978-3-658-24850-5.
- [129] R. Olszewski. “Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data.” Doctoral Dissertation, School of Computer Science Carnegie Mellon University, Pittsburgh USA, 2001.
- [130] T. Dorst et al. “GUM2ALA – Uncertainty Propagation Algorithm for the Adaptive Linear Approximation According to the GUM.” In: *SMSI 2021 - System of Units and Metrological Infrastructure, digital, 03 - 06 May* (2021), pp. 314–315. DOI: 10.5162/SMSI2021/D1.1.
- [131] H. Sanz et al. “SVM-RFE: selection and visualization of the most relevant features through non-linear kernels.” In: *BMC Bioinformatics* 19 (1 Dec. 2018), p. 432. ISSN: 1471-2105. DOI: 10.1186/s12859-018-2451-4.
- [132] R. Urbanowicz et al. “Relief-based feature selection: Introduction and review.” In: *Journal of Biomedical Informatics* 85 (Sept. 2018), pp. 189–203. ISSN: 15320464. DOI: 10.1016/j.jbi.2018.07.014.

-
- [133] T. Schneider, N. Helwig, and A. Schutze. “Automatic feature extraction and selection for condition monitoring and related datasets.” In: *IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Houston, TX, USA, 14-17 May* (2018). DOI: 10.1109/I2MTC.2018.8409763.
- [134] R. Fisher. “THE USE OF MULTIPLE MEASUREMENTS IN TAXONOMIC PROBLEMS.” In: *Annals of Eugenics* 7 (2 Sept. 1936), pp. 179–188. ISSN: 2050-1420. DOI: 10.1111/j.1469-1809.1936.tb02137.x.
- [135] S. Wold, M. Sjöström, and L. Eriksson. “PLS-regression: a basic tool of chemometrics.” In: *Chemometrics and Intelligent Laboratory Systems* 58 (2 Oct. 2001), pp. 109–130. ISSN: 01697439. DOI: 10.1016/S0169-7439(01)00155-1.
- [136] M. Andersson. “A comparison of nine PLS1 algorithms.” In: *Journal of Chemometrics* 23 (10 Oct. 2009), pp. 518–529. ISSN: 0886-9383. DOI: 10.1002/cem.1248.
- [137] S. de Jong. “SIMPLS: An alternative approach to partial least squares regression.” In: *Chemometrics and Intelligent Laboratory Systems* 18 (3 Mar. 1993), pp. 251–263. ISSN: 01697439. DOI: 10.1016/0169-7439(93)85002-X.
- [138] F. Rosenblatt. “The perceptron: A probabilistic model for information storage and organization in the brain.” In: *Psychological Review* 65 (6 1958), pp. 386–408. ISSN: 1939-1471. DOI: 10.1037/h0042519.
- [139] J. Schmidhuber. “Annotated History of Modern AI and Deep Learning.” In: *eprint arXiv:2212.11279, Neural and Evolutionary Computing* (2022). DOI: <https://doi.org/10.48550/arXiv.2212.11279>.
- [140] J. McClelland et al. *Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises*. 2nd ed. MIT-Press, Cambridge, USA, 2015. DOI: <https://doi.org/10.7551/mitpress/5617.001.0001>.
- [141] H. Ide and T. Kurita. “Improvement of learning for CNN with ReLU activation by sparse regularization.” In: *2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14-19 May* (2017), pp. 2684–2691. DOI: 10.1109/IJCNN.2017.7966185.
- [142] L. Lu et al. “Dying ReLU and Initialization: Theory and Numerical Examples.” In: *Communications in Computational Physics* 28 (5 Mar. 2019). DOI: 10.4208/cicp.0A-2020-0165.

- [143] T. Szandała. “Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks.” In: *Bhoi, A., Mallick, P., Liu, CM., Balas, V. (eds) Bio-inspired Neurocomputing. Studies in Computational Intelligence, vol 903. Springer, Singapore.* (2021). DOI: 10.1007/978-981-15-5495-7.
- [144] F. Murtagh. “Multilayer perceptrons for classification and regression.” In: *Neurocomputing 2* (1991), pp. 183–197. ISSN: 09252312. DOI: 10.1016/0925-2312(91)90023-5.
- [145] J. Terven et al. “Loss Functions and Metrics in Deep Learning. A Review.” In: *ArXiv Preprint 2307.02694, Machine Learning* (July 2023). DOI: <https://doi.org/10.48550/arXiv.2307.02694>.
- [146] S. Amari. “Backpropagation and stochastic gradient descent method.” In: *Neurocomputing 5* (June 1993), pp. 185–196. ISSN: 09252312. DOI: 10.1016/0925-2312(93)90006-0.
- [147] S. Ruder. “An overview of gradient descent optimization algorithms.” In: *ArXiv Preprint, 1609.04747, Machine Learning* (Sept. 2016). DOI: <https://doi.org/10.48550/arXiv.1609.04747>.
- [148] R. Hecht-Nielsen. “Theory of the backpropagation neural network.” In: *International Joint Conference on Neural Networks, Washington, DC, USA* (1989), 593–605 vol.1. DOI: 10.1109/IJCNN.1989.118638.
- [149] D. Rumelhart and J. McClelland. *Parallel Distributed Processing*. The MIT Press, Cambridge, USA, 1986. ISBN: 9780262291408. DOI: 10.7551/mitpress/5236.001.0001.
- [150] P. Werbos. “Generalization of backpropagation with application to a recurrent gas market model.” In: *Neural Networks 1* (4 Jan. 1988), pp. 339–356. ISSN: 08936080. DOI: 10.1016/0893-6080(88)90007-X.
- [151] D. Rumelhart, G. Hinton, and R. Williams. “Learning representations by backpropagating errors.” In: *Nature 323* (6088 Oct. 1986), pp. 533–536. ISSN: 0028-0836. DOI: 10.1038/323533a0.
- [152] E. Bisong. “Optimization for Machine Learning: Gradient Descent.” In: *Building Machine Learning and Deep Learning Models on Google Cloud Platform, Apress, Berkeley, CA.* (2019), pp. 203–207. DOI: 10.1007/978-1-4842-4470-8_16.

-
- [153] X. Glorot and Y. Bengio. “Understanding the difficulty of training deep feed-forward neural networks.” In: *Appearing in Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W and CP 9*. 9 (Aug. 2010). Ed. by Yee Whye Teh and Mike Titterton, pp. 249–256. URL: <https://proceedings.mlr.press/v9/glorot10a.html>.
- [154] M. Gardner and S. Dorling. “Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences.” In: *Atmospheric Environment* 32 (14-15 Aug. 1998), pp. 2627–2636. ISSN: 13522310. DOI: 10.1016/S1352-2310(97)00447-0.
- [155] H. Robbins and S. Monro. “A Stochastic Approximation Method.” In: *The Annals of Mathematical Statistics* 22 (3 Sept. 1951), pp. 400–407. ISSN: 0003-4851. DOI: 10.1214/aoms/1177729586.
- [156] L. Bottou, F. Curtis, and J. Nocedal. “Optimization Methods for Large-Scale Machine Learning.” In: *SIAM Review* 60 (June 2016), pp. 223–311. DOI: <https://doi.org/10.48550/arXiv.1606.04838>.
- [157] L. Bottou. “Online Learning and Stochastic Approximations.” In: *D. Saad (Ed.), On-Line Learning in Neural Networks (Publications of the Newton Institute, pp. 9-42)*. Cambridge: Cambridge University Press. (1998). DOI: doi:10.1017/CB09780511569920.003.
- [158] G. Hinton, N. Srivastava, and K. Swersky. “Neural Networks for Machine Learning Lecture 6.” In: *Lecture Slides (accessed on 27 July 2023)* (2012). URL: https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.
- [159] A. Graves. “Generating Sequences With Recurrent Neural Networks.” In: *Eprint arXiv:1308.0850, Neural and Evolutionary Computing* (Aug. 2013). DOI: <https://doi.org/10.48550/arXiv.1308.0850>.
- [160] N. Qian. “On the momentum term in gradient descent learning algorithms.” In: *Neural Networks* 12 (1 Jan. 1999), pp. 145–151. ISSN: 08936080. DOI: 10.1016/S0893-6080(98)00116-6.
- [161] D. Kingma and J. Ba. “Adam: A Method for Stochastic Optimization.” In: *3rd International Conference on Learning Representations, (ICLR) 2015, Conference Track Proceedings, San Diego, CA, USA, 7 - 9 May* (2015). DOI: <https://doi.org/10.48550/arXiv.1412.6980>.

- [162] S. Haji and A. Abdulazeez. “Comparison Of Optimization Techniques Based On Gradient Descent Algorithm: A Review.” In: *PalArch’s Journal of Archaeology of Egypt / Egyptology* (2021), pp. 2715–2743. URL: <https://archives.palarch.nl/index.php/jae/article/view/6705>.
- [163] D. Wolpert and W. Macready. “No free lunch theorems for optimization.” In: *IEEE Transactions on Evolutionary Computation* 1 (1 Apr. 1997), pp. 67–82. ISSN: 1089778X. DOI: 10.1109/4235.585893.
- [164] D. Wolpert. “The Lack of A Priori Distinctions Between Learning Algorithms.” In: *Neural Computation* 8 (7 Oct. 1996), pp. 1341–1390. ISSN: 0899-7667. DOI: 10.1162/neco.1996.8.7.1341.
- [165] S. Basha et al. “Impact of Fully Connected Layers on Performance of Convolutional Neural Networks for Image Classification.” In: *Neurocomputing* 378 (Jan. 2019). DOI: 10.1016/j.neucom.2019.10.008.
- [166] A. Rana et al. “Application of Multi Layer (Perceptron) Artificial Neural Network in the Diagnosis System: A Systematic Review.” In: *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), San Salvador, El Salvador, 22-24 August* (2018), pp. 1–6. DOI: 10.1109/RICE.2018.8509069.
- [167] M. Zeiler and R. Fergus. “Visualizing and Understanding Convolutional Networks.” In: *Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8689. Springer, Cham.* (Nov. 2013). DOI: https://doi.org/10.1007/978-3-319-10590-1_53.
- [168] J. Gu et al. “Recent Advances in Convolutional Neural Networks.” In: *Pattern Recognition* 77 (Dec. 2015), pp. 354–377. DOI: <https://doi.org/10.1016/j.patcog.2017.10.013>.
- [169] A. Krizhevsky, I. Sutskever, and G. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks.” In: *Advances in Neural Information Processing Systems* 25 (2012). Ed. by F Pereira et al. URL: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [170] R. Yamashita et al. “Convolutional neural networks: an overview and application in radiology.” In: *Insights into Imaging* 9 (4 Aug. 2018), pp. 611–629. ISSN: 1869-4101. DOI: 10.1007/s13244-018-0639-9.

-
- [171] A. Alsobhani, H. ALabboodi, and H. Mahdi. “Speech Recognition using Convolution Deep Neural Networks.” In: *Journal of Physics: Conference Series* 1973 (1 Aug. 2021), p. 012166. ISSN: 1742-6588. DOI: 10.1088/1742-6596/1973/1/012166.
- [172] M. Hashemi. “Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation.” In: *Journal of Big Data* 6 (1 Dec. 2019), p. 98. ISSN: 2196-1115. DOI: 10.1186/s40537-019-0263-7.
- [173] Keras. “BatchNormalization layer.” In: *Documentation*, (accessed on 28 July 2023) (2023). URL: https://keras.io/api/layers/normalization_layers/batch_normalization/.
- [174] MathWorks. “batchNormalizationLayer.” In: *Documentation*, (accessed on 28 July 2023) (2023). URL: <https://de.mathworks.com/help/deeplearning/ref/nnet.cnn.layer.batchnormalizationlayer.html>.
- [175] C. Santos and J. Papa. “Avoiding Overfitting: A Survey on Regularization Methods for Convolutional Neural Networks.” In: *ACM Computing Surveys* 54 (10 Jan. 2022). DOI: 10.1145/3510413.
- [176] S. Ioffe and C. Szegedy. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.” In: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, Lille, France* (Feb. 2015). DOI: <https://doi.org/10.48550/arXiv.1502.03167>.
- [177] G. Philipp, D. Song, and J. Carbonell. “The exploding gradient problem demystified - definition, prevalence, impact, origin, tradeoffs, and solutions.” In: *Eprint ArXiv: Learning 1712.05577, Machine Learning* (Dec. 2017). DOI: <https://doi.org/10.48550/arXiv.1712.05577>.
- [178] A. Rehmer and A. Kroll. “On the vanishing and exploding gradient problem in Gated Recurrent Units.” In: *IFAC-PapersOnLine* 53 (2020), pp. 1243–1248. ISSN: 24058963. DOI: 10.1016/j.ifacol.2020.12.1342.
- [179] S. Basodi et al. “Gradient amplification: An efficient way to train deep neural networks.” In: *Big Data Mining and Analytics* 3 (3 Sept. 2020), pp. 196–207. ISSN: 2096-0654. DOI: 10.26599/BDMA.2020.9020004.

- [180] N. Srivastava et al. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting.” In: *Journal of Machine Learning Research* 15 (1 Jan. 2014), pp. 1929–1958. ISSN: 1532-4435.
- [181] Keras. “Keras layers API.” In: *Documentation*, (accessed 28 July 2023) (2023). URL: <https://keras.io/api/layers/>.
- [182] MathWorks. “List of Deep Learning Layers.” In: *Documentation*, (accessed 28 July 2023) (2023). URL: <https://de.mathworks.com/help/deeplearning/ug/list-of-deep-learning-layers.html>.
- [183] A. Ng. “Feature selection, L1 vs.2 regularization, and rotational invariance.” In: ACM Press, 2004, p. 78. ISBN: 1581138285. DOI: 10.1145/1015330.1015435.
- [184] D. Maulud and A. Abdulazeez. “A Review on Linear Regression Comprehensive in Machine Learning.” In: *Journal of Applied Science and Technology Trends* 1 (4 Dec. 2020), pp. 140–147. ISSN: 2708-0757. DOI: 10.38094/jastt1457.
- [185] Y. LeCun, Y. Bengio, and G. Hinton. “Deep learning.” In: *Nature* 521 (7553 May 2015), pp. 436–444. ISSN: 0028-0836. DOI: 10.1038/nature14539.
- [186] D. Baymurzina, E. Golikov, and M. Burtsev. “A review of neural architecture search.” In: *Neurocomputing* 474 (Feb. 2022), pp. 82–93. ISSN: 09252312. DOI: 10.1016/j.neucom.2021.12.014.
- [187] P. Ren et al. “A Comprehensive Survey of Neural Architecture Search: Challenges and Solutions.” In: *ACM Computing Surveys* 54 (4 June 2020). DOI: <https://doi.org/10.48550/arXiv.2006.02903>.
- [188] T. Elsken, J. Metzen, and F. Hutter. “Neural Architecture Search: A Survey.” In: *Journal of Machine Learning Research* 20 (Aug. 2018). DOI: <https://doi.org/10.48550/arXiv.1808.05377>.
- [189] C. White, W. Neiswanger, and Y. Savani. “BANANAS: Bayesian Optimization with Neural Architectures for Neural Architecture Search.” In: *Proceedings of the AAAI Conference on Artificial Intelligence*, online (Oct. 2021). DOI: 10.1609/aaai.v35i12.17233.
- [190] J. Snoek, H. Larochelle, and R. Adams. “Practical Bayesian Optimization of Machine Learning Algorithms.” In: *Advances in Neural Information Processing Systems* 25 (June 2012). DOI: <https://doi.org/10.48550/arXiv.1206.2944>.

-
- [191] V. Nguyen. “Bayesian Optimization for Accelerating Hyper-Parameter Tuning.” In: *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), Sardinia, Italy, 03 - 05 June (2019)*, pp. 302–305. DOI: 10.1109/AIKE.2019.00060.
- [192] J. Mockus. *Bayesian Approach to Global Optimization*. Vol. 37. Springer Netherlands, 1989. ISBN: 978-94-010-6898-7. DOI: 10.1007/978-94-009-0909-0.
- [193] MathWorks. “Bayesian Optimization Algorithm.” In: *Documentation, (accessed 02 August 2023)* (2023). URL: <https://de.mathworks.com/help/stats/bayesian-optimization-algorithm.html>.
- [194] M. Ebden. “Gaussian Processes: A Quick Introduction.” In: *Eprint arXiv:1505.02965, Statistics Theory* (May 2015). DOI: <https://doi.org/10.48550/arXiv.1505.02965>.
- [195] C. Williams and C. Rasmussen. “Gaussian Processes for Regression.” In: *Advances in Neural Information Processing Systems, MIT Press, Cambridge USA* 8 (1995). Ed. by D Touretzky, M C Mozer, and M Hasselmo. URL: https://proceedings.neurips.cc/paper_files/paper/1995/file/7cce53cf90577442771720a370c3c723-Paper.pdf.
- [196] K. Weinberger. “Lecture 15: Gaussian Processes.” In: *Online Lecture, (accessed 02 August 2023)* (2018). URL: <https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote15.html>.
- [197] MathWorks. “Gaussian Process Regression Models.” In: *Documentation, (accessed on 28 July 2023)* (2023). URL: <https://de.mathworks.com/help/stats/gaussian-process-regression-models.html>.
- [198] Y. Gorishniy et al. “Revisiting Deep Learning Models for Tabular Data.” In: *Advances in Neural Information Processing Systems* (June 2021). DOI: <https://doi.org/10.48550/arXiv.2106.11959>.
- [199] R. Levin et al. “Transfer Learning with Deep Tabular Models.” In: *International Conference on Learning Representations (ICLR), Kigali Rwanda, 01 - 05 May (2023)*. DOI: <https://doi.org/10.48550/arXiv.2206.15306>.
- [200] K. Weiss, T. Khoshgoftaar, and D. Wang. “A survey of transfer learning.” In: *Journal of Big Data* 3 (1 Dec. 2016), p. 9. ISSN: 2196-1115. DOI: 10.1186/s40537-016-0043-6.
-

- [201] F. Zhuang et al. “A Comprehensive Survey on Transfer Learning.” In: *Proceedings of the IEEE* 109 (1 Nov. 2019). DOI: 10.1109/JPROC.2020.3004555.
- [202] C. Tan et al. “A Survey on Deep Transfer Learning.” In: *Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (eds) Artificial Neural Networks and Machine Learning – ICANN 2018. ICANN 2018. Lecture Notes in Computer Science, vol 11141. Springer, Cham.* (Aug. 2018). DOI: https://doi.org/10.1007/978-3-030-01424-7_27.
- [203] W. Dai et al. “Boosting for transfer learning.” In: *Proceedings of the 24th international conference on Machine learning* (June 2007), pp. 193–200. DOI: 10.1145/1273496.1273521.
- [204] F. Tambon et al. “How to Certify Machine Learning Based Safety-critical Systems? A Systematic Literature Review.” In: *Autom Softw Eng* 29 (July 2022). DOI: 10.1007/s10515-022-00337-x.
- [205] S. Grigorescu et al. “A Survey of Deep Learning Techniques for Autonomous Driving.” In: *Journal of Field Robotics* 37 (Oct. 2020). DOI: 10.1002/rob.21918.
- [206] G. Varoquaux and V. Cheplygina. “Machine learning for medical imaging: methodological failures and recommendations for the future.” In: *npj Digital Medicine* 5 (1 Apr. 2022), p. 48. ISSN: 2398-6352. DOI: 10.1038/s41746-022-00592-y.
- [207] O. Loyola-Gonzalez. “Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View.” In: *IEEE Access* 7 (2019), pp. 154096–154113. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2949286.
- [208] M. Ribeiro, S. Singh, and C. Guestrin. ““Why Should I Trust You?”: Explaining the Predictions of Any Classifier.” In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, San Diego, California* (Feb. 2016). DOI: <https://doi.org/10.48550/arXiv.1602.04938>.
- [209] B. Zhou et al. “Learning Deep Features for Discriminative Localization.” In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27 - 30 June* (2016). DOI: 10.1109/CVPR.2016.319.
- [210] M. Ancona et al. “Towards better understanding of gradient-based attribution methods for Deep Neural Networks.” In: *International Conference on Learning Representations, Vancouver Convention Center, Vancouver, BC, Canada, 30 April - 03 May* (2018). DOI: <https://doi.org/10.48550/arXiv.1711.06104>.

-
- [211] I. Kakogeorgiou and K. Karantzalos. “Evaluating explainable artificial intelligence methods for multi-label deep learning classification tasks in remote sensing.” In: *International Journal of Applied Earth Observation and Geoinformation* 103 (Dec. 2021), p. 102520. ISSN: 15698432. DOI: 10.1016/j.jag.2021.102520.
- [212] MathWorks. “occlusionSensitivity.” In: *Documentation*, (accessed on 28 July 2023) (2023). URL: <https://de.mathworks.com/help/deeplearning/ref/occlusionsensitivity.html>.
- [213] K. Simonyan, A. Vedaldi, and A. Zisserman. “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps.” In: *Workshop at International Conference on Learning Representations, Scottsdale, Arizona USA, 02 - 04 May* (2013). DOI: <https://doi.org/10.48550/arXiv.1312.6034>.
- [214] J. Springenberg et al. “Striving for Simplicity: The All Convolutional Net.” In: *ICLR 2015 - International Conference on Learning Representations, San Diego, USA, 07 - 09 May* (2015). DOI: <https://doi.org/10.48550/arXiv.1412.6806>.
- [215] R. Selvaraju et al. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization.” In: *International Journal of Computer Vision* 128 (2 Feb. 2020), pp. 336–359. ISSN: 0920-5691. DOI: 10.1007/s11263-019-01228-7.
- [216] S. Lundberg et al. “Explainable AI for Trees: From Local Explanations to Global Understanding.” In: *Nat Mach Intell* 2 (May 2019), pp. 56–57. DOI: <https://doi.org/10.1038/s42256-019-0138-9>.
- [217] M. Setzu et al. “GLocalX – From Local to Global Explanations of Black Box AI Models.” In: *Artificial Intelligence* 294 (Jan. 2021). DOI: 10.1016/j.artint.2021.103457.
- [218] Switzerland Sensirion AG Stäfa. “Datashet SGP40.” In: *Online Documentation* (accessed on 07 January 2024) (2022). URL: https://sensirion.com/media/documents/296373BB/6203C5DF/Sensirion_Gas_Sensors_Datasheet_SGP40.pdf.
- [219] V. Palmisano et al. “Selectivity and resistance to poisons of commercial hydrogen sensors.” In: *International Journal of Hydrogen Energy* 40 (35 Sept. 2015), pp. 11740–11747. ISSN: 03603199. DOI: 10.1016/j.ijhydene.2015.02.120.
- [220] H. Chai et al. “Stability of Metal Oxide Semiconductor Gas Sensors: A Review.” In: *IEEE Sensors Journal* 22 (6 Mar. 2022), pp. 5470–5481. ISSN: 1530-437X. DOI: 10.1109/JSEN.2022.3148264.
-

- [221] Y. Robin et al. “Machine Learning based calibration time reduction for Gas Sensors in Temperature Cycled Operation.” In: *Conference Record - IEEE Instrumentation and Measurement Technology Conference, Glasgow, United Kingdom, 17-20 May (2021)*. ISSN: 10915281. DOI: 10.1109/I2MTC50364.2021.9459919.
- [222] C. Krutzler et al. “Influence of MOS Gas-Sensor Production Tolerances on Pattern Recognition Techniques in Electronic Noses.” In: *IEEE Transactions on Instrumentation and Measurement* 61 (1 Jan. 2012), pp. 276–283. ISSN: 0018-9456. DOI: 10.1109/TIM.2011.2161015.
- [223] Y. Robin et al. “Insight in Dynamically Operated Gas Sensor Arrays with Shapley Values for Data Segments.” In: *Micro and Nano Engineering- Eurosensors (MNE-ES), Leuven, Belgium, 19 September (2022)*.
- [224] G. Müller and G. Sberveglieri. “Origin of Baseline Drift in Metal Oxide Gas Sensors: Effects of Bulk Equilibration.” In: *Chemosensors* 10 (5 May 2022), p. 171. ISSN: 2227-9040. DOI: 10.3390/chemosensors10050171.
- [225] R. Yi et al. “Improving the performance of drifted/shifted electronic nose systems by cross-domain transfer using common transfer samples.” In: *Sensors and Actuators B: Chemical* 329 (Feb. 2021), p. 129162. ISSN: 0925-4005. DOI: 10.1016/J.SNB.2020.129162.
- [226] J. Amann. “Möglichkeiten und Grenzen des Einsatzes von Halbleitersensoren im temperaturzyklischen Betrieb für die Messung der Innenraumluftqualität–Kalibrierung, Feldtest, Validierung.” Masterthesis, Saarland University Lab for Measurement Technology, Saarbrücken Germany, 2021.
- [227] Ltd Zhengzhou Winsen Electronics Technology Co. *MEMS Alcohol Gas Sensor (Model No.:GM-302B) Manual 2021 (accessed on 08 August 2023)*. URL: <https://www.cnwinsen.com/wp-content/uploads/2021/08/MEMS-GM-302B-Manual-V1.1.pdf>.
- [228] C. Schultealbert. “Siloxanvergiftung von Metalloxid-Gassensoren im temperaturzyklischen Betrieb – Effekte, Erkennung, Optimierung.” Doctoral Dissertation, Saarland University Lab for Measurement Technology, Saarbrücken Germany, 2021. DOI: <http://dx.doi.org/10.22028/D291-35481>.
- [229] C. Schultealbert et al. “Siloxane treatment of metal oxide semiconductor gas sensors in temperature-cycled operation – sensitivity and selectivity.” In: *Journal*

-
- of Sensors and Sensor Systems* 9 (2 Aug. 2020), pp. 283–292. ISSN: 2194-878X. DOI: 10.5194/jsss-9-283-2020.
- [230] M. Schüler et al. “Detecting poisoning of metal oxide gas sensors at an early stage by temperature cycled operation.” In: *Proceedings SENSOR 2015, AMA Conferences, Nürnberg, Germany, 19 - 21 May* (2015), pp. 735–739. DOI: 10.5162/sensor2015/E8.4.
- [231] M. Schüler et al. “Impedance based detection of HMDSO poisoning in metal oxide gas sensors.” In: *tm - Technisches Messen* 84 (11 Nov. 2017), pp. 697–705. ISSN: 2196-7113. DOI: 10.1515/teme-2017-0002.
- [232] M. Schüler, T. Sauerwald, and A. Schütze. “A novel approach for detecting HMDSO poisoning of metal oxide gas sensors and improving their stability by temperature cycled operation.” In: *Journal of Sensors and Sensor Systems* 4 (2 Oct. 2015), pp. 305–311. ISSN: 2194-878X. DOI: 10.5194/jsss-4-305-2015.
- [233] A. Miquel-Ibarz, J. Burgués, and S. Marco. “Global calibration models for temperature-modulated metal oxide gas sensors: A strategy to reduce calibration costs.” In: *Sensors and Actuators B: Chemical* 350 (Jan. 2022), p. 130769. ISSN: 09254005. DOI: 10.1016/j.snb.2021.130769.
- [234] T. Dorst et al. “Influence of measurement uncertainty on machine learning results demonstrated for a smart gas sensor.” In: *Journal of Sensors and Sensor Systems* 12 (1 Jan. 2023), pp. 45–60. ISSN: 2194-878X. DOI: 10.5194/jsss-12-45-2023.
- [235] S. Marco and A. Gutierrez-Galvez. “Signal and Data Processing for Machine Olfaction and Chemical Sensing: A Review.” In: *IEEE Sensors Journal* 12 (11 Nov. 2012), pp. 3189–3214. ISSN: 1530-437X. DOI: 10.1109/JSEN.2012.2192920.
- [236] N. Ma et al. “Comparison of Machine Learning Algorithms for Natural Gas Identification with Mixed Potential Electrochemical Sensor Arrays.” In: *ECS Sensors Plus* 2 (1 Mar. 2023), p. 011402. ISSN: 2754-2726. DOI: 10.1149/2754-2726/acbe0c.
- [237] L. Fernandez et al. “Calibration transfer in temperature modulated gas sensor arrays.” In: *Sensors and Actuators B: Chemical* 231 (Aug. 2016), pp. 276–284. ISSN: 09254005. DOI: 10.1016/j.snb.2016.02.131.
- [238] J. Workman. “A Review of Calibration Transfer Practices and Instrument Differences in Spectroscopy.” In: *Applied Spectroscopy* 72 (3 Mar. 2018), pp. 340–365. ISSN: 0003-7028. DOI: 10.1177/0003702817736064.
-

- [239] J. Fonollosa et al. “Evaluation of calibration transfer strategies between Metal Oxide gas sensor arrays.” In: *Procedia Engineering* 120 (2015), pp. 261–264. ISSN: 18777058. DOI: 10.1016/j.proeng.2015.08.601.
- [240] Y. Wang, D. Veltkamp, and B. Kowalski. “Multivariate instrument standardization.” In: *Analytical Chemistry* 63 (23 Dec. 1991). doi: 10.1021/ac00023a016, pp. 2750–2756. ISSN: 0003-2700. DOI: 10.1021/ac00023a016.
- [241] Y. Wang, M. Lysaght, and B. Kowalski. “Improvement of multivariate calibration through instrument standardization.” In: *Analytical Chemistry* 64 (5 Mar. 1992), pp. 562–564. ISSN: 0003-2700. DOI: 10.1021/ac00029a021.
- [242] K. Yan and D. Zhang. “Improving the transfer ability of prediction models for electronic noses.” In: *Sensors and Actuators B: Chemical* 220 (Dec. 2015), pp. 115–124. ISSN: 09254005. DOI: 10.1016/j.snb.2015.05.060.
- [243] O. Tomic, H. Ulmer, and J. Haugen. “Standardization methods for handling instrument related signal shift in gas-sensor array measurement data.” In: *Analytica Chimica Acta* 472 (1-2 Nov. 2002), pp. 99–111. ISSN: 00032670. DOI: 10.1016/S0003-2670(02)00936-4.
- [244] O. Shaham, L. Carmel, and D. Harel. “On mappings between electronic noses.” In: *Sensors and Actuators B: Chemical* 106 (1 Apr. 2005), pp. 76–82. ISSN: 09254005. DOI: 10.1016/j.snb.2004.05.039.
- [245] T. Fearn. “On orthogonal signal correction.” In: *Chemometrics and Intelligent Laboratory Systems* 50 (1 Jan. 2000), pp. 47–52. ISSN: 01697439. DOI: 10.1016/S0169-7439(99)00045-3.
- [246] R. Gutierrez-Osuna. “Drift Reduction For Metal-Oxide Sensor Arrays Using Canonical Correlation Regression And Partial Least Squares.” In: *Proceedings of the 7th International Symp. On Olfaction and Electronic Nose, Brighton, UK, 20-24 July* (2000). URL: https://www.researchgate.net/publication/268297393_Drift_Reduction_For_Metal-Oxide_Sensor_Arrays_Using_Canonical_Correlation_Regression_And_Partial_Least_Squares.
- [247] K. Yan and D. Zhang. “Calibration transfer and drift compensation of e-noses via coupled task learning.” In: *Sensors and Actuators B: Chemical* 225 (Mar. 2016), pp. 288–297. ISSN: 09254005. DOI: 10.1016/j.snb.2015.11.058.

-
- [248] M. Vauhkonen et al. “Tikhonov regularization and prior information in electrical impedance tomography.” In: *IEEE Transactions on Medical Imaging* 17 (2 Apr. 1998), pp. 285–293. ISSN: 02780062. DOI: 10.1109/42.700740.
- [249] S. Garcia-Munoz, J. MacGregor, and T. Kourti. “Product transfer between sites using Joint-Y PLS.” In: *Chemometrics and Intelligent Laboratory Systems* 79 (Oct. 2005), pp. 101–114. DOI: 10.1016/j.chemolab.2005.04.009.
- [250] Z. Liang et al. “A correlated information removing based interference suppression technique in electronic nose for detection of bacteria.” In: *Analytica Chimica Acta* 986 (Sept. 2017), pp. 145–152. ISSN: 00032670. DOI: 10.1016/j.aca.2017.07.028.
- [251] L. Zhang and X. Peng. “Time series estimation of gas sensor baseline drift using ARMA and Kalman based models.” In: *Sensor Review* 36 (1 Jan. 2016), pp. 34–39. ISSN: 0260-2288. DOI: 10.1108/SR-05-2015-0073.
- [252] X. Zhao et al. “Sensor Drift Compensation Based on the Improved LSTM and SVM Multi-Class Ensemble Learning Models.” In: *Sensors* 19 (18 Sept. 2019), p. 3844. ISSN: 1424-8220. DOI: 10.3390/s19183844.
- [253] Y. Bahri et al. “Temperature Stability Investigations of Neural Network Models for Graphene-Based Gas Sensor Devices.” In: *Engineering Proceedings* 10 (Nov. 2021), p. 19. DOI: 10.3390/ecsa-8-11250.
- [254] V. Pareek and S. Chaudhury. “Deep learning-based gas identification and quantification with auto-tuning of hyper-parameters.” In: *Soft Computing* 25 (22 Nov. 2021), pp. 14155–14170. ISSN: 1432-7643. DOI: 10.1007/s00500-021-06222-1.
- [255] J. Oh et al. “Machine Learning-Assisted Gas-Specific Fingerprint Detection/-Classification Strategy Based on Mutually Interactive Features of Semiconductor Gas Sensor Arrays.” In: *Electronics* 11 (23 Nov. 2022), p. 3884. ISSN: 2079-9292. DOI: 10.3390/electronics11233884.
- [256] S. Wang et al. “Prediction of Gas Concentration Using Gated Recurrent Neural Networks.” In: *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Genova, Italy, 31 August - 02 September (2020)*, pp. 178–182. DOI: 10.1109/AICAS48895.2020.9073806.

- [257] S. Arik and T. Pfister. “TabNet: Attentive Interpretable Tabular Learning.” In: *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8), 6679–6687, online 02 - 09 February (2019). DOI: <https://doi.org/10.1609/aaai.v35i8.16826>.
- [258] Y. Zhang et al. “TDACNN: Target-domain-free domain adaptation convolutional neural network for drift compensation in gas sensors.” In: *Sensors and Actuators B: Chemical* 361 (June 2022), p. 131739. ISSN: 09254005. DOI: 10.1016/j.snb.2022.131739.
- [259] Q. Liu et al. “Gas Recognition under Sensor Drift by Using Deep Learning.” In: *International Journal of Intelligent Systems* 30 (8 Aug. 2015), pp. 907–922. ISSN: 08848173. DOI: 10.1002/int.21731.
- [260] B. Mahesh et al. “Few-Shot Time-Series Classification of Chemosensor Data.” In: *ECS Meeting, online, May 30 - June 03* (2021), pp. 1310–1310. ISSN: 2151-2043. DOI: 10.1149/MA2021-01541310mtgabs.
- [261] S. Schober et al. “Neural Network Robustness Analysis Using Sensor Simulations for a Graphene-Based Semiconductor Gas Sensor.” In: *Chemosensors* 10 (5 Apr. 2022), p. 152. ISSN: 2227-9040. DOI: 10.3390/chemosensors10050152.
- [262] Y. Robin et al. “Überwachung der Luftqualität in Innenräumen mittels komplexer Sensorsysteme und Deep Learning Ansätzen.” In: *15. Dresdner Sensor-Symposium, Dresden, Germany 06 - 08 December* (2021), pp. 85–90. DOI: 10.5162/15dss2021/5.3.
- [263] R. Schleyer, E. Bieber, and M. Wallasch. “Das Luftmessnetz des Umweltbundesamtes.” In: *Umweltbundesamt: Dessau-Roßlau, Germany (accessed on 18 January 2024)* (2023). URL: <https://www.umweltbundesamt.de/themen/luft/messe-nbeobachteneuberwachen/luftmessnetz-des-umweltbundesamtes#aufgabedes-umweltbundesamtes>.
- [264] J. Miettinen. “Deep learning applications for condition monitoring of rotating systems.” Doctoral Dissertation, Aalto University, Espoo Finland, 2023. ISBN: 978-952-64-1432-4.
- [265] N. Helwig, E. Pignanelli, and A. Schütze. “Condition Monitoring of a Complex Hydraulic System Using Multivariate Statistics.” In: *I2MTC-2015 - 2015 IEEE International Instrumentation and Measurement Technology Conference, paper PPS1-39, Pisa, Italy, 11-14 May* (2015). DOI: 10.1109/I2MTC.2015.7151267.

- [266] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge USA, 2006. DOI: <https://doi.org/10.7551/mitpress/3206.001.0001>.

A Appendix: Gaussian Process

In ML, especially in linear regression, the task is to find a projection of the input to the output that minimizes some loss function. This can be described as shown in Equation A.1 with w being the model parameter, \hat{w} all possible w , D the training data, and L a loss function (e.g., Equation 2.1).

$$w = \operatorname{argmin}_{\hat{w}} L_{\hat{w}}(D) \tag{A.1}$$

One way to get to this point is by maximizing the conditional probability of w given the training data D ($P(W|D)$). However, the drawback of this approach is that the result from Equation A.1 returns only a single value (the most probable value given the most probable w). Therefore, Gaussian processes are introduced on a fundamental level. A more profound derivation can be found in [266], and the core elements of this derivation are based on the sources [114, 194–196]. In Gaussian processes, we do not want to obtain the most probable w ; instead, the goal is to directly predict the distribution of the test point, as shown in Equation A.2. This can be done by integrating over every possible w (D : Dataset, x : test point, y : predicted label).

$$P(y_{test}|x_{test}, D) = \int_w P(y_{test}|x_{test}, w) * P(w|D) dw \tag{A.2}$$

This means it is necessary to integrate all possible projections or functions (infinite dimensional) to get the probability of y given x and D . Therefore, not a single w is calculated; instead, every possible w gets a probability. The real benefit now from a Gaussian process is that it is assumed that $P(y_{test}|x_{test}, W)$ and $P(w|D)$ are Gaussian distributed, allowing the assumption that $P(y_{test}|x_{test}, D)$ is Gaussian distributed. It is no longer necessary to compute the integral as shown in Equation A.3. Instead, it is possible to model the probability of y directly since the probability has to be Gaussian distributed, as shown in Equation A.1.

$$P(y_{test}|x_{test}, D) \sim \mathcal{N}(\mu_{test}, \Sigma_{test}) \quad (\text{A.3})$$

This can be interpreted that for each test point, it is possible to calculate a Gaussian distribution with μ_{test} and Σ_{test} . Before showing how to calculate those values, it is necessary to go one step back and look at where this formula is derived from. For Gaussian processes, it is assumed that all labels are drawn from a Gaussian distribution with $\mu = 0$ (can be constructed) and Σ . This is reasonable since $P(y_{test}|x_{test}, D)$ is Gaussian as shown above. Therefore, it can be stated that :

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{test} \end{bmatrix} \sim \mathcal{N}(0, \Sigma_{all}) \quad (\text{A.4})$$

Where Σ_{all} defines the correlation of every combination of y's. Furthermore, Σ is constructed as follows:

$$\Sigma_{all} = \begin{bmatrix} K_{train,train} & K_{train,test} \\ K_{test,train} & K_{test,test} \end{bmatrix} \quad (\text{A.5})$$

In this case, K is the kernel function that takes two x-inputs and outputs the correlation between two y's (returns a matrix). The user defines this kernel function that makes a statement about the correlation and can be interpreted as the user-defined function of what correlation should be assigned to similar or dissimilar points. More abstractly, the kernel can be interpreted as a definition of the shape of all possible fitted functions (all possible w). The RBF kernel is the most popular for Bayesian optimization as it allows for relatively smooth functions with not overly high peaks. Furthermore, it assigns a high correlation to two y's if their x's are similar. This means that similar points should have a similar label. With the help of this Σ_{all} for all drawn samples, it is now possible to calculate μ_{test} and Σ_{test} of the joint probability of $P(y_{test}|x_{test}, D)$. The derivation to obtain μ_{test} and Σ_{test} can be found in [266]. The results are given as follows:

$$\mu_{test} = K_{test,train}^T * (K_{train,train} + \sigma * I)^{-1} * y_{train} \quad (\text{A.6})$$

$$\Sigma_{test} = K_{test,test} - K_{test,train}^T * (K_{train,train} + \sigma * I)^{-1} * K_{test,train} \quad (\text{A.7})$$

After the general functions are introduced, a few remarks are made. Although a Gaussian process is, in theory, parameter-less, the kernel still has a few parameters that need to be learned with the training data to fit the training points optimally, or σ is defined as noise to guarantee that K is invertible.

List of Figures

2.1	a) Basic MOS sensor model (adapted from [42]). b) Model for grain boundary effect during adsorption and reduction of oxygen. The change in resistance is indicated by the width of the conduction channel between grains illustrated in red (adapted from [42, 46]).	6
2.2	Double logarithmic dependency between sensor response and gas concentration. Reprinted with permission of [52], © 1995 Shaker.	7
2.3	Working principle of the temperature-cycled operation regarding oxygen coverage. During heating, the general resistance of the sensor decreases. Afterward, the oxygen molecules adsorb on the sensor, increasing the resistance. Then, the sensor is cooled down, which generally increases the resistance. Additionally, reducing gases recombine with O^- during the low-temperature phases, lowering the overall resistance (adapted from [18, 41, 46]).	8
2.4	Example configuration for a readout circuit (adapted from [64]).	9
2.5	Schematic of the most complex GMA at LMT today (adapted from [77]).	12
2.6	Example for a possible design of experiments concerning the selection of gas concentrations. Sequential subsampling, random subsampling, and Latin hypercube subsampling for calibration are shown.	15
2.7	Example of one neuron.	24
2.8	Example of one neural network consisting of two fully connected hidden layers.	25
2.9	Illustration of gradient descent to find the optimum (adapted from [152]).	27
2.10	Example for a convolutional layer with an input image with size 4×4 , a kernel size of 2×2 , a striding size of 1×1 , and without padding.	31
2.11	Example of Bayesian optimization with a Gaussian Process Regression (GPR) for multiple steps with lower confidence bound (adapted from [197]).	34

2.12	Transfer Learning: The effect of transfer learning for different hyperparameters and number of training samples. Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.	36
2.13	Transfer Learning: The effect of transfer learning for different dataset sizes. The optimization function, if infinite data is available, is illustrated together with the slightly different optimization functions that result from incomplete datasets. A comparison of the optimization is shown between a large dataset (from scratch), a small dataset (from scratch), and a small dataset with transfer learning. For transfer learning, the learning rate is reduced, and only limited data is available.	36
2.14	Example of an occlusion map for computer vision. The task was to identify the object in the picture and to identify the most important sections for that prediction (adapted from [211, 212]).	38
2.15	Overview of Grad-CAM that illustrates the calculation of the importance scores (adapted from [215]).	39
2.16	Example of the sensor signals from an SGP40 (Sensirion AG, Stäfa, Switzerland [218]) with four sensor pixels sampled at 10 Hz. Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.	41
2.17	Example of the sensor signals displayed as an image from an SGP40 (Sensirion AG, Stäfa, Switzerland [218]) with four sensor pixels sampled at 10 Hz (adapted from Paper A).	41
2.18	Evaluation pipeline for ML algorithms.	42
2.19	Example of drift for the four sub-sensors of an SGP40 over 70 days. Different sub-sensors show a different severity of drift over time (@ 400 °C). Data used for visualization from [226].	43
2.20	Example of how drift over time can influence the prediction accuracy; offset and slope can be altered (adapted from [Paper 1, 15]).	44
2.21	a) Differential signal between original and adapted signal. b) Sensor response of the master sensor, the initial sensor response from the slave sensor, and the adapted signal from the slave sensor (Direct Standardization (DS) and Piecewise Direct Standardization (PDS)). Only a section (0 s - 25 s) of one Temperature-Cycle (TC) is shown for better visibility, and only the signal of sub-sensor 1 is shown. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.	47

3.1	Neural network architecture of the TCOCNN (adapted from [262]). An example configuration with ten convolutional layers (later optimized). Reprinted with permission from Ref. Paper 2. Y. Robin, 2023.	54
3.2	Comparison of the results obtained during field tests with the FESR and TCOCNN models for formaldehyde and hydrogen. Reprinted with the permission of Ref. Paper 1. Y. Robin, 2023.	60
3.3	Prediction of gas concentrations during release tests 5 and 6 (acetone and toluene) showing the various models trained compared to the analytical measurements (adapted from [15]). Reprinted with permission of Ref. Paper 1. Y. Robin, 2023.	61
3.4	Comparison of different transfer methods for carbon monoxide and formaldehyde. Transfer method 1: learning rate set to the value reached at the end of original training. Transfer method 2: learning rate set to the value reached halfway through original training. Transfer method 3: learning rate set to the original value. Transfer method 4: implicit feature extraction fixed, only fully connected layer can be trained. Reprinted with permission of Ref. Paper 2. Y. Robin, 2023.	92
3.5	Overview of the gases included in the randomized calibration. Each UGM contains all of the shown gases. (a) The composition of the different UGMs (adapted from [24, Paper 2]). (b) All the maximum concentrations during recording. The lowest concentration for all VOCs during the measurement is 0 ppb, for carbon monoxide 200 ppb, and for hydrogen 400 ppb. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.	118
3.6	Comparison of Direct Standardization (DS), Piecewise Direct Standardization (PDS), transfer learning for DL (TL), and global model building concerning the TCOCNN. Different numbers of UGMs for transfer learning are used in the different sub-plots. Reprinted with permission of Ref. Paper 3. Y. Robin, 2023.	120
3.7	Comparison of importance scores occlusion map (Occ) vs. gradient map (Grad) standardized and rescaled from 0 to 1 for sensor A (two evaluations per method). Marked areas indicate regions with significant differences between methods (see text for details). Reprinted with permission of Ref. Paper A. Y. Robin, © 2023 IEEE.	145

List of Tables

2.1	The eight most important chemical classes for VOCs extracted from analytical studies [85, 86] together with representatives, P90, and P95 quantiles for reference measurements. Reprinted with permission of Ref. [15]. T. Baur, 2023.	14
3.1	One example for possible hyperparameter ranges for the TCOCNN (adapted from Paper 1).	54
3.2	Rough overview of the complexity of the datasets used in the respective paper [Paper 1, Paper 2, Paper 3].	55
3.3	Concentration ranges for all gases within gas mixtures during the calibration phases. Reprinted with permission of Ref. [15, Paper 1]. Y. Robin and T. Baur, 2023.	58
3.4	Concentration ranges for all target gases and the number of unique gas mixtures (UGMs) for each range within the dataset. Relative Humidity (RH) was varied between 25 % and 80 %. Reprinted with permission of Ref. Paper 2. Y. Robin, 2023.	91