PROCEEDINGS OF SCIENCE

QUA^{3}CK
A Machine Learning Development Process

Jürgen Becker[*]
Karlsruhe Instiute of Technology
E-mail: Juergen.Becker@KIT.edu

Daniel Grimm
Karlsruhe Instiute of Technology
E-mail: Daniel.Grimm@KIT.edu

Tim Hotfilter
Karlsruhe Instiute of Technology
E-mail: Tim.Hotfilter@KIT.edu

Gabriela Molinar
Karlsruhe Instiute of Technology
E-mail: Gabriela.Molinar@KIT.edu

Marco Stang
Karlsruhe Instiute of Technology
E-mail: Marco.Stang@KIT.edu

Simon Stock[†]
Karlsruhe Instiute of Technology
E-mail: Simon.Stock@KIT.edu

Wilhelm Stork
Karlsruhe Instiute of Technology
E-mail: Wilhelm.Stork@KIT.edu

Machine learning and data processing are trending topics at the moment. However, there is still a lack of a standard process to support a fast, simple, and effective development of machine learning models for academia and industry combined. Processes such as KDD or CRISP-DM are highly specialized in data mining and business cases. Therefore, engineers often refer to individual approaches to solve a machine learning problem. Especially in teaching, the lack of a standard process is a challenge. Students typically get a better understanding if a systematic approach to solve problems is given to them. A challenge when formulating a machine learning development process is to provide standard actions that work on different use-cases. At the same time, it has to be simple. Complex processes often lead to the wrong approach.

The QUA^{3}CK process was created at the Karlsruhe Institute of Technology to fill the gap in research and industry for a machine learning development process. However, the main focus was to reach engineering students with an easy-to-remember, didactic way to solve machine learning problems. This five-stage process starts with a machine learning question (Q), a problem that has to be solved. Understanding the data (U) comes next. Then, the loop between selecting an Algorithm (A), Adapting the features (A), and Adjusting the hyperparameters (A) is executed until the system is ready for Conclude and compare (C). At last, the Knowledge transfer (K) of the given solution can be realized as deployment in hardware or as a documentation.

This paper describes the process and all individual steps in detail. Besides, we present several use-cases of QUA^{3}CK in academia and research projects.

Artificial Intelligence for Science, Industry and Society, AISIS2019
October 21-25, 2019
Universidad Nacional Autónoma de México, Mexico City, México

[*] Authors in alphabetical order

[†] Speaker at AISIS2019

1. Introduction

Machine learning and deep learning, in particular, have become increasingly important in engineering. The technology has been around for several decades already. Whereas data scientists in the past served only a niche in the job market, nowadays, all big companies bet on data science and machine learning. A reason might be that the trends of the Internet of Things (IoT) devices and the democratization of data are leading to the availability of more data than ever. Therefore, Machine Learning has become a demanded skill in the world of engineering.

Since many companies are now looking for *Machine Learning Engineers*, we developed a new laboratory for the Electrical Engineering Faculty at the Karlsruhe Institute of Technology (KIT, Germany). The novelty lies in teaching the students to recognize the kind of problems that can be solved with machine learning algorithms and how to find an initial solution for them as-fast-as-possible (see Section 4 for more details).

Engineering students usually are more motivated, understand faster, and perform better when having a systematic approach to solve problems. After researching the state-of-the-art standard processes for solving machine learning problems, we found that there is still a need for a generalized process. Engineers often refer to individual approaches to solve problems with machine learning algorithms. Processes, such as Knowledge Discovery in Databases (KDD) or the Cross-Industry Standard Process for Data Mining (CRISP-DM), are highly specialized in data mining and business cases. These, however, have not found general acceptance in academia yet and do not cover problems from diverse use-cases[1][2].

We introduce the QUA^{3}CK process to fill this need, in research and industry, for a standard machine learning development process. Its primary focus lies on reaching engineering students with an easy-to-remember, didactic way to solve machine learning problems. The process provides a set of standard actions for solving a variety of use-cases. For academia as well as industry, it supports fast, simple, and effective development of machine learning models. The process was first introduced to the scientific community at the Symposium on Artificial Intelligence for Science, Industry and Society in 2019[3].

QUA^{3}CK is an acronym for its steps. The method commences with **Q**uestion and **U**nderstanding the data, followed by the **A**-loop: Algorithm selection, data Adaption, and parameter Adjustment. Based on the results obtained, we **C**onclude and maybe also compare to other models. Finally, the **K**nowledge can be transferred into a product or documentation. Section 3 provides a detailed description of the process.

In section 4, we present the Laboratory for Applied Machine learning Approaches (LAMA), as a use-case for the QUA^{3}CK process in teaching. This laboratory leads the students into three phases of learning. First, there is an introduction to the theoretical concepts of data science and machine learning. Then, the students work on guided, hands-on experiments. Finally, they take part in a creative phase called ‘Into-the-Wild’, where the students have to solve by their own real-life problem from scratch. This approach was received well by the students, igniting intrinsic motivation lasting the whole semester.

Chapter 5 demonstrates the feasibility of the QUA^{3}CK process in recent research projects. This paper will focus on the following three contributions:

- We introduce QUA^{3}CK as a machine learning development process, which presents an easy-to-remember, fast, simple, and effective set of standard actions for solving a variety of use-cases in both academia and industry.
- The process has been tested in a machine learning laboratory at the KIT. Insights and results are summarized in this paper.
- We also outline the use of this process in selected research projects.

2. Related Work

Adapting and self-learning computer systems are around since the 1960s. However, in the last decade, machine learning systems have been able to tackle more complex tasks due to the higher availability of computing resources and performance[4]. In the meanwhile, some standard processes to guide the development of such a system have established themselves [1]. Namely CRISP-DM [2], KDD [5] and SEMMA[6]. However, all of them specialize in data mining[1] rather than general machine learning. To the best of the authors' knowledge, standardized general processes for modern machine learning model development still represent a gap in the state-of-the-art.

One of the most popular processes when handling large amounts of data is the CRoss-Industry Standard Process for Data Mining (CRISP-DM) [2]. It was first introduced in the late 1990s by Shearer et al. and has been further developed from there. CRISP-DM has six major steps. The first step is *business understanding*, which is essential to map the data on the use case and the business context. Questions such as what information is relevant for the company may arise. Afterward, the data has to be understood in terms of how the data is arranged and what is missing. Furthermore, it has to be prepared accordingly. Preparation usually involves removing outlaws or removing incomplete samples. To some extent, we transferred these two steps into our model, since they are indispensable for a proper data handling process. As a next step, a data mining or machine learning model is built and then evaluated. If the model evaluation fails, the process guides back to the initial business understanding. If the model behaves as expected, it can be deployed. In the data-science community, CRISP-DM is one of the most prominent development processes. Because of a lack of specialized machine learning development processes, we used CRISP-DM for several data science experiments[7, 8]. The process has a strong empathy for business understanding. However, in science, we try to figure out what is possible and what is impossible without a business case. Furthermore, we can conclude from experience that CRISP-DM was not designed in hindsight towards deep learning approaches in research.

Another prominent example is Knowledge Discovery in Databases (KDD) from Fayyad et al. [5]. It is a simplified version of the CRISP-DM and consists of five steps. In the beginning, relevant data is selected, and after this pre-processed, which works similarly to CRISP-DM. The third step transforms the data in order to make it suitable for a given data mining application. The selection of the algorithm corresponds to the fourth step. In the end, the results are evaluated and interpreted.

This approach works well for the task of data-mining. However, deep learning introduced other aspects of data science challenges.

Sample, Explore, Modify, Model, Assess (SEMMA)[6] is another method for data mining and data handling tasks introduced by the American SAS Institute. It extends KDD with an 'Explore'-step, in which a deep understanding of the data is formed from yet unknown anomalies or correlations. Moreover, it features a 'Modify' stage that can introduce new variables or transformations based on the data in order to select relevant samples. This feature of SEMMA is useful for data mining but limited when applied to machine learning problems.

3. The QUA^{3}CK Machine Learning Development Process

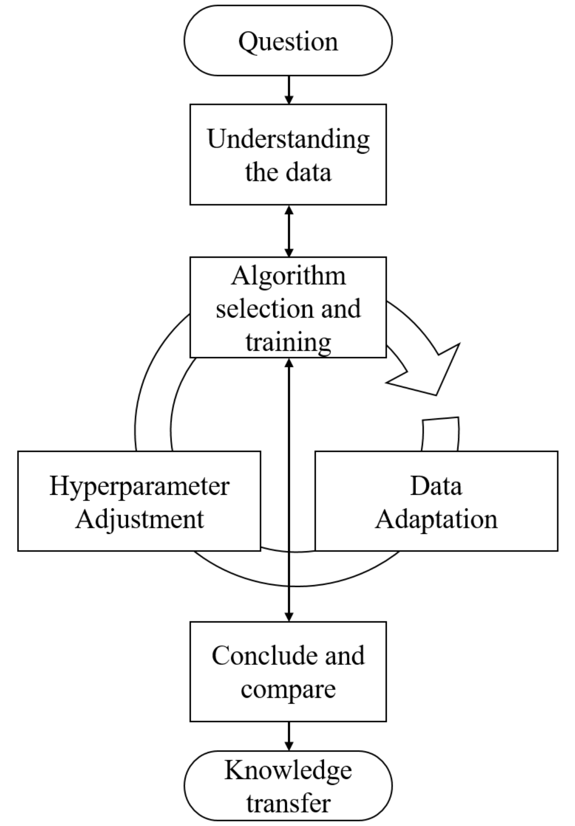

Figure 1: The QUA^{3}CK process starts with a general (scientific) Machine Learning related question. Then the data scientist has to understand the underlying principles of the data. Afterward, an iterative modeling step is performed. The results of this loop are regularly checked, concluded, and compared. The final results are transferred towards the next project, or into a publication or product.

The process is defined by its acronym: QUA^{3}CK. It refers to the steps of developing a machine learning solution. Figure 1 shows the structure of the process. In the following, we will describe the individual steps:

Question: The first step of the process is to specify a problem, which needs to be solved. The goals, requirements, and expected outcomes have to be stated. This step helps to avoid a common mistake of many students and young researchers of realizing too late that the conditions and specifications are not entirely defined. Once the requirements are precise, it is possible to continue analyzing the data.

Understanding the data: If a data set is not available from the beginning, the first step will be to formulate it. Therefore, an established database can be based on open-source data, which already fulfills a part of the predefined requirements. On the other hand, it can be collected from sensor measurements or by information retrieval. Once a data set is available, the raw values have to be analyzed to discover and understand possible hidden information, such as irrelevant data or unexpected trends. Sample measures to achieve this level of data understanding are the calculation of the mean, standard deviation, or correlation between features. Furthermore, it might be interesting to plot statistics in scatter-plots or probability distributions, to name a few examples.

Some general pre-processing has to be done at this stage as well. They involve, for instance, removing Not-A-Number (NAN) elements, filling them with interpolated values, or grouping the data according to some general assumptions. Data-augmentation might be executed as well to increase the diversity of the samples. In general, consistent formatting of the input data set and labels are crucial for the following steps.

Algorithm selection and training – Data Adaptation – Hyperparameter Adjustment: These three steps can be seen as a loop that can be executed as many times as necessary until the model fulfills the expected outcomes.

In the first step, an algorithm is selected based on the application. There are specific rules of thumb for this. For example, for images and videos, Convolutional Neural Networks are usually set in place; for time series forecast, Recurrent Neural Networks, for classification of numerical features, Feed-forward Neural Networks. However, some applications do not match these rules. In those cases, a first algorithm can be selected and then changed over time based on a benchmark.

Once an algorithm has been selected, the data set should be adapted correspondingly (Data Adaptation step). Each algorithm expects a particular input format. For example, the given data set represents a time series with a single feature, and the task is to predict the following ten steps. It is crucial to rearrange the data so that input (the single feature) and label (the following ten steps) are available to the machine learning model.

Hyperparameters, such as the number of training epochs or the architecture of a neural network, are present in most deep-learning algorithms. They control different aspects of the model and hence have a significant impact on its performance. However, they are not adapted by the learning process and have to be tuned either manually or automatically by another program (Hyperparameter Adjustment step).

As soon as a set of hyperparameters has been chosen, the model can be trained on the training data set. For validation, a separate, equally distributed dataset is extracted, called the validation set. The validation error is calculated for various hyperparameter configurations. The best performing selection is used to move to the next step, conclude, and compare.

Sometimes it might be necessary to return to the *Understanding the data* step, in case none of the algorithms selected is performing as expected. Changing the set of features or increasing the size of the database might be a solution. Problems, such as missing important information or interdependent variables, increase the complexity of the model but not its performance.

Conclude and compare: In this step, it is checked whether it is necessary to execute another round from the A^3 loop. Therefore, the performance of the trained model is tested on another, separated, equally distributed data set, called the test set. The results are compared to reference models, which might be taken from literature or other models created on the same QUA^{3}CK process.

If necessary, it is possible to go back to the Algorithm Selection step. Otherwise, if the model already satisfies the requirements to solve the initial problem, the next step is Knowledge Transfer.

Knowledge transfer: Once the results meet the requirements, the model can be considered as finished. In this step, it is of importance to prepare detailed documentation. The algorithm, starting values, the final hyperparameter set, the training, validation, and test sets have to be described in detail. Additionally, they should also be secured and stored as well as all other relevant information and data. This will ensure the replicability of the study in the future. The last step can involve the deployment of the algorithm on a target platform. This can be on embedded hardware, as an online solution in a data center, on a mobile device, or another system.

4. Use-case in Academia: The Laboratory for Applied Machine Learning Approaches (LAMA)

4.1 Background

Many courses thought at universities and schools are often focused on theoretical backgrounds. This also applies to the emerging field of machine learning. Especially for many engineering students, it appears to be difficult to get into this topic. However, in order to learn the proper use of machine learning methods, students have to apply them to real-world problems. Hence, the Institute for Information Processing Technologies (ITIV) at the KIT created the Laboratory for Applied Machine learning Approaches (LAMA). The laboratory strives for a wide audience of engineering students, independently of their previous machine learning experience.

4.2 Course structure

The laboratory is composed of three main parts: Introduction to the theoretical background, guided assignments, and the project-based *Into-the-Wild* phase. First, we teach students the basic concepts of frequently used machine-learning algorithms in theory and practice ranging from traditional approaches like decision trees to modern deep neural networks. We are using the widely adapted programming language *python* for the guided practical part of the laboratory. With *jupyter* *notebooks*, we can combine theoretical background documentation and its associated code interactively. The theoretical knowledge is backed by a few additional lectures on the QUA^{3}CK process and its implications for the particular lab appointment.

During the *Into-the-Wild* part, students can apply their learned skills to a real-world engineering challenge. This creative part aims to apply the QUA^{3}CK process. Groups of two to four students can seek individual challenges or use provided data sets by the institute. In the end, all teams present their results to the other students and tutors.

4.3 Results

There was a wide variety of students' projects during the *Into-the-Wild* phase in summer 2019. Some examples were a music composing neural network and the detection of sitting behavior in order to improve posture. Moreover, some projects were executed in collaboration with KIT-startups. Like a classification whether bees are affected by toxins.

The project results were very satisfying since the students were able to develop a suitable machine learning system for various problems in a short time of about five weeks. In addition to the promising practical results, we were able to see comprehensive documentation in the form of plots and industry-standard metrics. Moreover, the students proved their theoretical knowledge in an oral exam.

From our experience in LAMA, we can conclude that self-imposed goals lead to better intrinsic motivation and a higher identification with the project. We are sure that the gained knowledge in this broad field will help the students in their future careers. As a result of the high approval rate in the students' community, the LAMA-laboratory has earned the Teaching Award 2020 of the KIT Electrical Engineering faculty.

At the end of the winter term 2019/2020 LAMA, we asked the students to provide a quick resume of their *Into-the-Wild* projects. Those are presented in the following:

- Our Lama *Into-the-Wild*-project dealt with neural networks for object detection. With the help of these, meals can be detected and located on photos.
- With the goal of aiding harmonic mixing of music, we developed multiple neural networks for key, mode and beat detection in music and assessed different input data formats regarding their effects on the accuracy achievable by the neural nets.
- With OTO (Online Training Optimization), we have started an attempt to adjust the learning rate during the training so that the error is minimized.

5. Use-cases in research

LAMA, outlined in section 4, was the starting point for the creation of the QUA^{3}CK machine learning development process. The initial focus was to teach young engineers a systematic approach to solve machine learning problems. The process, however, has been also useful in applied-research. This section presents example projects, which have successfully made use of the QUA^{3}CK machine learning process.

Prognonetz The main goal of the PrognoNetz project is the development of an artificial intelligent meteorological network for an efficient electrical system operation, under high renewable energy generation. This represents the question (Q) in the development process.

Nowadays, transmission system operators need to utilize more efficiently their existing infrastructure, increasing the power transmission capacity of overhead lines. The increasing amount of wind parks has clear advantages, producing clean energy, but also means a stronger instability in power generation (represented by sudden peaks, related to changes in wind speed and direction). These peaks surpass today the fixed conservative limit to the maximum current capacity of overhead lines (OL), the so-called ampacity. The solution to this is either to construct new OL or to increment the fixed conservative ampacity limit. The latter can be achieved monitoring and predicting the conductor temperature dynamically.

This is not only a very important prerequisite for the expansion of renewable energy sources, but also will allow saving money in systems operations. Through a forecast of the maximum carrying capacity, TSOs are able to plan the optimum power transmission, instead of conducting short-notice expensive re-dispatching.

The project is based on a data set of distributed weather observations along the electrical network. The data is pre-processed (U) in a central server and stored in a data base. The A-phase of QUA^{3}CK is executed at this main computer; machine learning models are trained and tested predicting the cooling effect of weather on the conductors of the electrical network[9]. The performance of several models are compared (C) and the best solution is considered for deployment (K), in order to make sure that the system is always up-to-date.

DeepBees In collaboration with the Karlsruhe start-up apic.ai[10], we did a machine learning project, where we were able to apply the QUA^{3}CK process on a real-world problem successfully. The self-set mission of the start-up is to save the bees with the help of artificial intelligence.

In the project called 'DeepBees', we build an algorithm focusing on near-time monitoring of beehives using machine learning techniques. In the project, we used QUA^{3}CK to plan the current and next steps in algorithm development. The question (Q) was, 'how can we track bees with the help with machine learning and what kind of information can we obtain from bees?'.

We used data provided by apic.ai, tried to understand (U) the underlying principles, and modified the data to make them useable for machine learning techniques. In the algorithm step (A), we used deep convolutional neural networks incorporating a multi-task-learning-loss. We adapted the data for this kind of machine learning approach and finally used Bayesian Optimization for hyperparameter adjustment. Enabling the system to perform bee classification successfully, pose estimation, and pollen detection. We compared (C) and further inspired our work with the help of the current state of the art in bee and insect monitoring research. In the final step of QUA^{3}CK (K), we published our work at the Computer Vision for Wildlife Conservation workshop of the International Conference on Computer Vision[11].

Automotive Cyber Threat Detection One of our current research topics in automotive cyber security is the identification of attacks on vehicles using Intrusion Detection Systems. Therefore, we analyze vehicular Ethernet data using a machine learning based system. The system answers the question (Q) if the captured in-vehicle data is normal or anomalous. Anomalous communication data could indicate a cyber attack. The requirements in this project differ from other application domains of artificial intelligence, as the system shall be deployed on an embedded micro-controller with limited memory and computing capabilities.

Data Understanding (U) is a critical part, as no data of vehicular Ethernet is publicly available. Hence, we simulated the data and developed different sets of features that characterize the communication. On our simulated data set, we trained various algorithms that fulfill the requirements of limited computation power and used a grid-search with cross-validation to get a suitable model (A).

To draw conclusions, we evaluated our system with synthetically generated attacks on the vehicular communication (C). The knowledge transfer (K) was initiated with the first publication on the 4th International Conference on Vehicle Technology and Intelligent Transport Systems [12].

Neuro-visual-function assessment In an ongoing, particularly exciting project, we are researching objective measurements of human visual function. Our initial research question (Q) was whether we could measure the perimetry (visual field) using Brain-Computer Interfaces, Virtual Reality, and Deep Learning. There is not much publicly available data for such task. Therefore, in order to train a model, we needed sufficient data. For data acquisition, we use our specially developed software platform, which enables us to synchronize Virtual Reality-stimuli and labels. As the algorithm (A), we examined the use of Convolutional Neural Networks and ResNet-Architectures. Furthermore, Bayesian Parameter-Optimization was used to obtain a good hyperparameter set. We concluded that it is indeed possible to measure a person's visual field objectively (C). All findings are being published (K) in-depth at the SPIE Photonics Europe - Neurophotonics Conference [13].

Embedded hardware design for neural network In a recent work by our research group we apply the QUACK process to the design of dedicated hardware accelerators for neural networks [14]. This highlights again the flexibility and possibilities of our introduced process as it is not limited to machine learning algorithms themselves.

In this case the Question (Q) in the beginning is how to make CNNs faster and more efficient in embedded and area-restricted systems. Even there is not a classic data understanding, it is still important to know how the input to the hardware looks like and how it is organized (U). The algorithm selection is based on the motivating use case, which is face detection for automotive realized by a CNN (A). From an initial hardware architecture we are able to optimize it similar to the iterative A^3 loop. However, we have to apply different metrics in the compare step (C) such as power consumption, silicon area or throughput. Our architecture is optimized towards a low power consumption. The knowledge (K) transfer is done by the given publication.

6. Summary

In this paper, we shared an approach to plan, build, and deploy machine learning algorithms efficiently. We introduced the steps of the QUA^{3}CK Machine Learning development process in detail. The process was evaluated in an academic setting within the Laboratory for Applied Machine Learning Algorithms. Additionally, we have outlined use cases in science. We hope that QUA^{3}CK will be broadly adapted in research projects, student theses, and maybe industry. Machine learning has much potential, but it is still difficult for beginners to gain a foothold in this exciting field. By introducing QUA^{3}CK, we want to contribute along the way towards more natural machine learning algorithm development. In the future, we plan to adapt further and improve QUA^{3}CK and teach it through LAMA to a broad audience of students. Lastly, we have one recommendation for everyone who is struggling with processes: A good process should always support and not restrict the engineers and scientists using it.

Acknowledgments

We want to thank the Karlsruhe Institute of Technology and especially the faculty for Electrical Engineering and Information Technology, for providing us with an environment in which we can grow and flourish. We want to thank everyone involved in the set up of LAMA and QUA^{3}CK. Furthermore, we would like to thank the research assistants at ITIV, the students who have taken the LAMA-course, our industry partners, and everyone else whose participation helped to improve the QUA^{3}CK process.

References

[1] Azevedo, A. and Santos, M., “KDD, SEMMA and CRISP-DM: A parallel overview,” 182–185 (01 2008).

[2] Shearer, C., “The CRISP-DM model: the new blueprint for data mining,” *Journal of data warehousing* **5**(4), 13–22 (2000).

[3] Becker, J., Grimm, D., Hotfilter, T., Meier, C., Molinar, G., Stang, M., Stock, S., and Stork, W., “The QUA^{3}CK Machine Learning Development Process and the Laboratory for Applied Machine Learning Approaches (LAMA).” Presentation given at Symposium Artificial Intelligence for Science, Industry and Society (AISIS 2019), Mexico City, Mexico, 20–25 October 2019 (2019).

[4] Jordan, M. I. and Mitchell, T. M., “Machine learning: Trends, perspectives, and prospects,” *Science* **349**(6245), 255–260 (2015).

[5] Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P., “From data mining to knowledge discovery in databases,” *AI magazine* **17**(3), 37–37 (1996).

[6] SAS Institute Inc., “Introduction to SEMMA.” https://documentation.sas.com/ (Last update: August 30, 2017).

[7] Stang, M., Boehme, M., and Eric, S., “Applied machine learning: Reconstruction of spectral data for the classification of oil-quality levels,” *The Eurasia Proceedings of Science Technology Engineering and Mathematics* (5), 1–13 (2019).

[8] Molinar, G., Popovic, N., and Stork, W., “From data points to ampacity forecasting: Gated Recurrent Unit networks,” in [*2018 IEEE Fourth International Conference on Big Data Computing Service and Applications (BigDataService)*], 200–207, IEEE (2018).

[9] Molinar, G., Fan, L. T., and Stork, W., “Ampacity forecasting: an approach using Quantile Regression Forests,” in [*IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT), Washington, DC, USA, 18-21 Feb. 2019*], 1–5, IEEE, Piscataway (NJ) (2019).

[10] apic.ai, “Website of apic.ai.” http://apic.ai (Last accessed: January 2020).

[11] Marstaller, J., Tausch, F., and Stock, S., “Deepbees-building and scaling convolutional neuronal nets for fast and large-scale visual monitoring of bee hives,” in [*Proceedings of the IEEE International Conference on Computer Vision Workshops*], (2019).

[12] Grimm, D., Weber, M., and Sax, E., “An extended hybrid anomaly detection system for automotive electronic control units communicating via ethernet - efficient and effective analysis using a specification- and machine learning-based approach,” in [*Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems - Volume 1: VEHITS,*], 462–473, INSTICC, SciTePress (2018).

[13] Stock, S. C., Kovacs, B., Maier, H., Gerdes, M., Stork, W., Armengol-Urpi, A., and Sarma, S. E., “A system approach for closed loop assessment of neuro-visual-function based on convolutional neural network analysis of EEG signals,” in [*Neurophotonics*], International Society for Optics and Photonics (in press, 2020).

[14] Hotfilter, T., Kempf, F., Reinhardt, D., Baili, I., and Becker, J., “Embedded image processing the european way: A new platform for the future automotive market,” (forthcoming, 2020).