Towards AI-based Solutions in the System Development Lifecycle

Stephan Jüngling, Martin Peraic, Andreas Martin
FHNW University of Applied Sciences and Arts Northwestern Switzerland, School of Business
Peter Merian-Strasse 86, 4052 Basel, Switzerland
stephan.juengling@fhnw.ch, martin.peraic@bvb.ch, andreas.martin@fhnw.ch

Abstract

Many teams across different industries and organizations explicitly apply agile methodologies such as Scrum in their system development lifecycle (SDLC). The choice of the technology stack, the programming language, or the decision whether AI solutions could be incorporated into the system design either is given by corporate guidelines or is chosen by the project team based on their individual skill set. The paper describes the business case of implementing an AI-based automatic passenger counting system for public transportation, shows preliminary results of the prototype using anonymous passenger recognition on the edge with the help of Google Coral devices. It shows how different solutions could be integrated with the help of rule base systems and how AI-based solutions could be established in the SDLC as valid and cost-saving alternatives to traditionally programmed software components.

Introduction

AI in general and deep learning, in particular, are amongst the current hot topics in computer science research, and many universities create new bachelor and master programs in data science. The hype is also visualized by the Gartner hype cycle for AI (2019), where edge AI, deep neuronal networks, and machine learning are at the peak of inflected expectations. Nevertheless, AI technologies are already used and well established in a wide area of different business domains. However, most of the time, these are isolated and specialized applications, where they are clearly the sole possible IT solution, e.g., machine translation or image processing for cancer detection. In such cases, ML capabilities are mostly predominantly compared to human skills. There are not yet many situations where AI-based components are compared to traditional software or hardware components. But with the current state of the art neuronal network models and new emerging infrastructure around TensorFlow (Tensorflow Hub, n.d.), where pre-trained models can be re-used as modules, and with the help of transfer learning be adapted to a variety of similar applications, AI components will slowly be deployed in many more business cases. In the case of semiconductor manufacturing, it was demonstrated that a portable image classifier could be embedded in offline edge devices to detect defects on laser chips with an accuracy of 97% (Hou, Liu, Pan, & Hou, 2019). However, what would be the impact on the software development process within companies, given the fact that AI will be increasingly used as part of their IT systems, products, and solutions?

Application Domains for AI-Based solutions

Smith and Eckroth (2017) provide a comprehensive insight into lessons learned from building AI applications during the last three decades. In one of their key insights, they mention that the ease of use delivered by the human interface is the “license to operate”. This statement, which has been made focusing on the client and user perspective, most probably will hold true during the entire software development lifecycle (SDLC). The “new” discipline of data scientists needs to be incorporated into the SDLC. Traditional skills of people that are involved in software design need to be extended with AI topics such as machine learning and knowledge engineering.

In many practical situations, the choice of the best suitable system development methodology is actively discussed and explicitly decided at the project start. Arguments for different methodologies such as Scrum, Kanban, SAFE, or even traditional water methods are sought, and different suitable tools are evaluated and selected. However, all these methods still take the predominant traditional design and implementation processes of software components into account. Also, with agile methods, some sort of requirements engineering, be it in the form of use cases or user stories, is conducted. Maybe it is followed by test-driven feature implementation using continuous integration as a core idea, and customers can, at any time, verify the current implementation state of the application and provide immediate feedback. Given the availability of state-of-the-art AI-based video analysis algorithms and edge computing capabilities, the idea of building an AI-based passenger counting system for public transit was reasonable and could be conducted during a bachelor thesis in cooperation with Basler Verkehrs-Betriebe, a leading public transport provider in the area of Basel (Peraic, 2019).

Business Case of an AI-based Passenger Counting System for Public Transit

With almost 29000 kilometers, Switzerland has one of the densest public transport networks in Europe (VÖV, 2017). This public service is offered by numerous private transport companies throughout Switzerland. In order to make it easier for customers to use these services, a variety of regional and national fare associations exist in different geographical areas. They ensure that public transport services can be accessed with a single subscription across Switzerland. Consequently, all public transport operators are obliged to record passenger data of onboarding and alighting passengers. Based on these figures, the percentage of the subscription income of the respective fare association is calculated for the participating companies.

Automatic Passenger Counting Systems (APC), especially developed for public transport, are applied to collect such data. Since the early 1970s, various manufacturers have been offering solutions for passenger counting (Siebert & Ellenberger, 2019), and many different approaches and measuring methods are applied. Some operators rely on the measurement of boarding and alighting motion with the help of light barrier sensors in the infrared spectrum, and some systems measure weight changes in the boarding area based on associated spring movements of the vehicle suspension. All of these “classic” methods are complex systems that must be installed at every door of the vehicle. This requires additional hardware components, which are expensive to purchase or to upgrade. This poses financial problems, especially for smaller public transport operators. Such companies use manual counting methods instead, which are frequently based on insufficient customer surveys regarding driving behavior and do not provide reliable figures (Siebert & Ellenberger, 2019).

Previous classic APC models are rarely deployed fleet-wide due to high costs. Instead, the systems are distributed across all existing routes and randomly measure the number of passengers, which are later extrapolated to estimate the total number of passengers in the entire transport network. As such, existing systems do not provide real-time data, and the measurements must first be collected and processed for subsequent use. In addition to the high acquisition costs, this also results in ongoing costs for storing and processing the recorded data, which is costly, inflexible, and no longer appropriate. New methods for passenger counting must be studied. In particular, it would be a good idea if existing systems in the vehicles, such as the cameras installed for security purposes, could be re-used. It is therefore suitable to apply the latest achievements in the field of AI-based object recognition. Such an APC method could re-use existing systems, collect data in real-time, and directly process the data without further manual operations. In order to demonstrate the feasibility of an AI-based counting system and to describe first benefits and limitations, the following research questions have to be answered:

- **RQ1**: Is AI-based object detection competitive against traditional infrared-based measurement systems under realistic conditions?
- **RQ2**: Is it possible to perform offline object recognition on edge devices from a legal perspective?
- **RQ3**: Can existing vehicle systems, such as surveillance cameras, be re-used?
- **RQ4:** What are the benefits of an AI-based automated passenger counting system?

Implemented Prototype

The Basler Verkehrs-Betriebe strives to continuously improve its core business, the transport of passengers, in order to offer the best possible service to its 350,000 daily customers. More and more processes are being enhanced or even replaced by modern software solutions. The need for tailor-made and in-house developed applications is increasing and consequently the demand for optimized conditions for the development of software. Currently, this new corporate focus is taking place, thus no company-wide SDLC for development projects could be applied for the prototype. This led to challenges before and during the project. Development and test environments first had to be created from scratch. Alongside development, the Hermes project methodology practiced by public companies in Switzerland could not be applied, as this methodology follows little to no agile approaches. However, an agile approach was crucial for the development of an AI-based APC. Without a flexible delimitation of the objectives, the continuous testing of the prototype versions under real conditions and the subsequent optimizations, a runnable prototype could not have been implemented in such a short timeframe.

The AI-based APC was realized with Google Coral devices (Google Coral, n.d.), which are optimized for TensorFlow Lite (TFL, n.d.) AI models such as the mobile Single Shot

Detector (SSD). The TFL mobile SSD object recognition is supported by the correlation tracking provided by DLIB (Rosebrock, n.d.). With this combination, the SSD model recognizes all objects (position coordinates) and their object class (human, bicycle, dog, etc.) in predefined frame intervals. Subsequently, position and class type are stored temporarily and are further tracked during the consecutive frames by the less computationally intensive correlation tracker. Thereby the Coral Edge device can be relieved by the reduced utilization of the object recognition model to keep resources free for additional operations. That capacity is used for the ID-based tracking of captured and stored objects. For this purpose, Centroid Tracking is being used. A primitive yet efficient tracking algorithm. This feature provides the ability to observe objects across multiple frames and measure motion into or out of the vehicle.
The setup is specially designed for offline operations to enable onboard computation. It allows bypassing possible future data protection obstacles of object recognition in public space by avoiding the transmission of sensitive data to a remotely located datacenter.

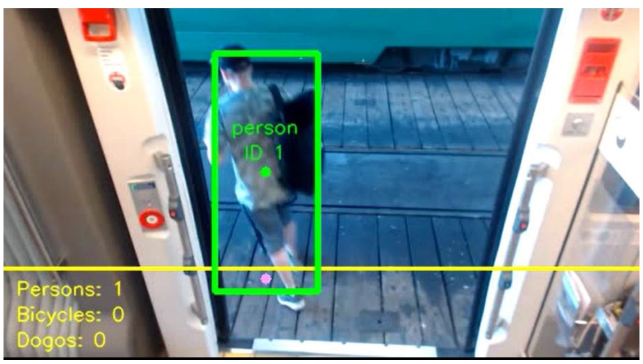

Figure 1 - Displayed Detection and Tracking Information

After the development of the prototype, a first comparison of the counting accuracy between the already deployed APC, the infrared measurement, and the newly developed AI-based system was conducted. In this respect, real test scenarios from day-to-day life were executed under laboratory conditions. The number of passengers and their measurement complexity was continuously increased. Extreme cases, such as simultaneous and dense boarding of a larger crowd, were considered. These test scenarios were created by observations from past experiments and real-life situations. This combination of knowledge engineering for the optimization of the measurement scenarios proved to be a great advantage for the development of realistic and meaningful test cases.
In order to obtain reliable results and to exclude any phenomena, each test case was performed three times. All measurements were recorded with a webcam directed at the door of the vehicle. This video feed was as well as the data source for the AI-based prototype. Subsequently, the measurements could be compared and evaluated using the timestamps of both measurement systems. The AI-based prototype correctly determined all passengers in 54% of the cases. The traditional infrared APC solution achieved a result of 72% correct measurements.

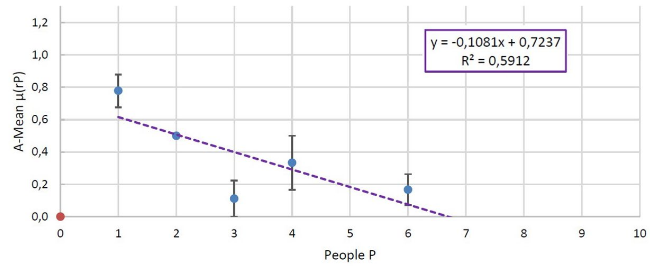

Figure 2 - Arithmetic Mean of Success in Relation to People involved (AI System)

RQ1: First results measured under real conditions demonstrate convincingly that with a small number of passengers, the AI-based APC can compete with the infrared system in terms of accuracy. Figure 2 reveals how only a few counting errors could be observed during experiments with one or two passengers. As complexity increases, so does the error rate. The object tracking mostly causes this increase. With a growing number of passengers or increasing complexity of movements, the basic Centroid Tracker can hardly distinguish between objects. They are either swapped or mistakenly re-detected, causing erroneous measurements.
Conversely, the infrared system proved to be acceptable for multiple passengers, as shown in Figure 3.

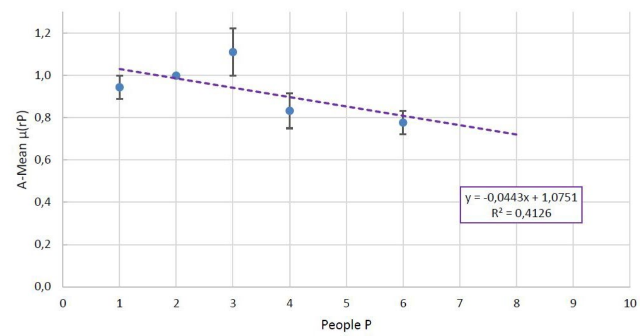

Figure 3 - Arithmetic Mean of Success in Relation to People involved (IR System)

Further erroneous measurements are attributed to the AI-based object recognition. In realistic circumstances, object recognition can be misled by background noise or disturbance of passing objects. Since for performance reasons the AI-based recognition is not applied at every frame, objects can pass the image without being detected. Possibilities for improving the AI-based APC have already been explored. Improved tracking algorithms like Deep SORT (Wojke, Bewley, & Paulus, 2017) or the utilization of recent RGB-video or Depth-video approaches could provide the required performance improvement. In particular, the use of depth sensors, as applied by Sun et al. (2019), theoretically outperform infrared sensors regarding accuracy. However, when facing problems with ML methods, this could be the point to improve the results by combination with rule-based systems as described in the section about combining machine learning with knowledge engineering.

RQ2: The legal use of high-resolution video cameras as a sensor has already been discussed for autonomous driving. In particular, the storage of video data was defined as a breach of data protection (Kunnert, 2017) if the image data is stored on the device or an external database (e.g., cloud storage). The AI-based prototype is therefore designed to process the image data in real-time – storage-free. Anonymous counting results could subsequently be sent to service providers for further processing. Such an architecture requires further security considerations. It is therefore essential to prevent third parties from gaining access to the edge devices. These could collect sensitive image data or manipulate counting results, which would cause manipulation of the entire service (Zhang, Chen, Zhao, Cheng, & Hu, 2018).

RQ3: Existing security cameras can theoretically be leveraged for AI-based passenger counting. Both the resolution and the quality of the images are sufficient for object recognition. Only the position and the associated viewing angle of the cameras are decisive for re-use. New vehicle procurements, though, could take this into account without additional efforts and costs. For already procured vehicles, the camera would have to be repositioned in the event of misplacement.

RQ4: The edge device architecture offers the ability to transmit real-time count data. Furthermore, the data could be automatically merged and analyzed, allowing real-time measurement of passenger flows using the fleet-wide deployment of the AI-based APCs. This information provides the control center or emergency services, with valuable information in a variety of situations. For example, in the case of a traffic accident, the number of people in the vehicle can be evaluated immediately and required emergency resources can be notified accordingly. On the other hand, conventional APC systems are usually entirely isolated and require complex and sometimes labor-intensive processes to evaluate the data. Sensor data must be converted into human-readable datasets and merged by skilled professionals. By eliminating these time-consuming and resource-intensive tasks, major financial savings can be achieved. Re-using already installed hardware such as surveillance cameras is an additional advantage of AI-performed passenger counting. Significantly fewer hardware components must be installed for the AI-based APC. This results in further cost savings in the procurement and maintenance of the system. Ultimately, edge devices can be continuously updated with new and enhanced AI models. Other applications such as the automatic detection of dangers in the vehicle such as violence or emergencies are just a few examples that support a switch to this new technology.

SDLC with Embedded AI-Based Solutions

Given the assumption of continuous improvements in the area of object detection with low-cost edge devices, the initial prototype could be improved within the next couple of months, but there will be additional issues, which must be considered for the organization.

According to a recent survey in Japan (Ishikawa & Yoshioka, 2019) software engineering professionals report difficulties in system engineering when ML-based components are incorporated in the engineering process. Many of the existing principles and best practices need to be enlarged with additional domain knowledge on how machine learning and knowledge engineering can be incorporated into the software development lifecycle of companies.

Worth mentioning is also the performance difference when SDLC methodologies known from software development are applied in data science projects, where little standardized process methodologies exist. By comparing the efficiency of different data science student teams working with different SDLC methodologies, Saltz, Shamshurin, & Crowston (2017) reported an improved performance and efficiency of CRISP-DM and agile Kanban, while agile Scrum was even less efficient than using no methodology at all.

Further differences need to be considered when existing code or models will be re-used. While object-oriented programming existing functionality can be extended using well-known inheritance mechanisms, new concepts such as transfer learning need to be researched in more depth-first. Adding additional functionality would not only mean adding additional code but also retraining some of the models and redeploy it to the edge devices.

Besides the additional skills required for the design and implementation of hybrid solutions, characteristics of AI-based solutions will most probably also have an impact on the skill sets in the field of requirements engineering as well as testing. As stated by Jüngling & Hofer (2019), AI components could even become active parts of business scenarios and represented as actors in UML use case diagrams.

Overall, companies that have internal software development skills will need either additional staff with appropriate experience in implementing AI technologies or educate internal employees and enable them to gain experience in order to establish new best practices for SDLC with hybrid solutions. With more experience from practical situations, all different disciplines, from requirements engineering to testing and deployment will have to develop new insights into how these two different kinds of systems engineering can be combined. It could well be, that it is best to split up the entire development into two parallel or serial sub-projects, one for the development of the AI components, the other for the traditional software components, where the SDLC of both components can be decoupled, having different life cycles, independent regression tests, automated build pipelines and integration-tests in the end. Alternatively, both disciplines could be tighter integrated, such as we have seen from certain database engineering frameworks, where the persistence layer is automatically generated based on the given design of the business layer.

Combining Machine Learning with Knowledge Engineering

Since different ML-based AI components are used as sensors on edge devices, rule-based systems could be handy to orchestrate their results. Given that multiple ML-based sensors are installed, their results could be combined with results from traditional sensors. Rule-based systems could be used to aggregate multiple inputs and decide about the most suitable estimates of a number of passengers. Similar to Grangström, Baum, & Reuter (2017), where they compile and overview of different approaches combining various sensor types for object tracking, or improvements reported by Tian et. al (Tian, Luo, Wang, & Tang, 2015), where they improved pedestrian detection by adding semantics based on additional pedestrians and scene attributes, sensors within the tram could be combined with sensors outside to determine the most appropriate result.

Considering the management of existing models that are trained for particular purposes, knowledge bases incorporating an ontology could be established describing the context in which the models can be applied. Facts could be collected from the results of different AI-based modules in combination with the run-time-environment they are deployed, and rules can be learned, in which cases, which modules work best. Typical situations of runtime configurations could be tracked, and the models could even be updated or deployed to the edge devices based on decisions from a knowledge-based system.

Also, for the general management of the SDLC and the design-time of ML-based components, best practice rules could be established and stored in appropriate knowledge bases about the most successful transfer learning modifications of re-usable pre-trained modules. Such knowledge bases could be established not only for the given case of passenger counting. Such a concept could generally be applied to different business application scenarios.

Conclusion and Outlook

Although the given potential of AI in general and image recognition and object tracking in particular, practical AI-based business applications are still rare. Openness for new solutions and gaining practical experience within companies will be key. Given the different nature of both ML and KE solutions compared to traditional SDLC, it seems important for companies to develop and deploy prototypes of AI-based solutions into production. It will be necessary to manage these components over the entire lifecycle and optimize the current SDLC accordingly. It will be important to gain experience with low-risk type of applications such as a passenger counting system first, before thinking about more advanced application scenarios such as self-driving trams or trains, which, compared to self-driving cars seem to be feasible earlier due to a lesser degree of freedom.

References

Gartner Hype Cycle for AI, 2019. (n.d.). Retrieved from https://www.gartner.com/smarterwithgartner/top-trends-on-the-gartner-hype-cycle-for-artificial-intelligence-2019/

Google Coral. (n.d.). Retrieved November 12, 2019, from https://coral.withgoogle.com/

Hou, D., Liu, T., Pan, Y. T., & Hou, J. (2019). AI on edge device for laser chip defect detection. 2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019, 247–251. https://doi.org/10.1109/CCWC.2019.8666503

Ishikawa, F., & Yoshioka, N. (2019). How Do Engineers Perceive Difficulties in Engineering of Machine-Learning Systems? - Questionnaire Survey. 2–9. https://doi.org/10.1109/cesser-ip.2019.00009

Jüngling, S., & Hofer, A. (2019). Leverage White-Collar Workers with AI. Proceedings of the AAAI 2019 Spring Symposium on Combining Machine Learning with Knowledge Engineering (AAAI-MAKE 2019). Retrieved from http://ceur-ws.org/Vol-2350/paper2.pdf

Kunnert, G. (2017). “Autonomes Fahren” aus datenschutzrechtlicher Sicht (I. Eisenberger, K. Lachmayer, & G. Eisenberger, Eds.). MANZ’sche Verlags- und Universitätsbuchhandlung.

Peraic,M. (2019). “Feasibility Study: Al-based Passenger Counting System for Public Transit” Assistant. Bachelor

thesis. Bachelor of Science in Business Information Technology, School of Business, University of Applied Sciences and Arts Northwestern Switzerland, Basel, Switzerland

Rosebrock, A. (n.d.). pyimagesearch. Retrieved November 14, 2019, from https://www.pyimagesearch.com/2018/07/23/simple-object-tracking-with-opencv/

Saltz, J., Shamshurin, I., & Crowston, K. (2017). Comparing Data Science Project Management Methodologies via a Controlled Experiment. Proceedings of the 50th Hawaii International Conference on System Sciences (2017), 1013–1022. https://doi.org/10.24251/hicss.2017.120

Siebert, M., & Ellenberger, D. (2019). Validation of automatic passenger counting: introducing the t-test-induced equivalence test. Transportation, (0123456789). https://doi.org/10.1007/s11116-019-09991-9

Smith, R. G., & Eckroth, J. (2017). Building AI applications: Yesterday, today, and tomorrow. AI Magazine, 38(1), 6–22. https://doi.org/10.1609/aimag.v38i1.2709

Sun, S., Akhtar, N., Song, H., Zhang, C., Li, J., & Mian, A. (2019). Benchmark Data and Method for Real-Time People Counting in Cluttered Scenes Using Depth Sensors. IEEE Transactions on Intelligent Transportation Systems, 20(10), 3599–3612. https://doi.org/10.1109/tits.2019.2911128

TensorFlow Hub. (n.d.). Retrieved from https://www.tensorflow.org/hub

TensorFlow Lite. (n.d.). Retrieved from https://www.tensorflow.org/lite

Tian, Y., Luo, P., Wang, X., & Tang, X. (2015). Pedestrian detection aided by deep learning semantic tasks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 07-12-June-2015, 5079–5087. https://doi.org/10.1109/CVPR.2015.7299143

VÖV. (2017). Facts & Figures Swiss Public Transport 2016/2017 (Vol. 41).

Wojke, N., Bewley, A., & Paulus, D. (2018). Simple online and realtime tracking with a deep association metric. Proceedings - International Conference on Image Processing, ICIP, 2017-September, 3645–3649. https://doi.org/10.1109/ICIP.2017.8296962

Zhang, J., Chen, B., Zhao, Y., Cheng, X., & Hu, F. (2018). Data Security and Privacy-Preserving in Edge Computing Paradigm: Survey and Open Issues. IEEE Access, 6(Idc), 18209–18237. https://doi.org/10.1109/ACCESS.2018.2820162