The future of machine vision: AI software designed with users in mind

Markus Schatzl[1], Jonas Meier[1], Henning Frechen[2], and Katrin Götzer[3]

[1] senswork GmbH Innovation Lab,
Friedenstr. 18, 81671 München,
[2] Fraunhofer-Institut für Integrierte Schaltungen IIS,
Nordostpark 84, 90411 Nürnberg
[3] eresult GmbH
Friedrichstraße 3-4, 37073 Göttingen

Abstract Machine learning by means of neural networks has developed an indispensable method to solve intricate challenges in optical quality control for manufacturing. When the technology became usable for inline inspection tasks first, neural network architectures themselves were at focus. However, it has become increasingly obvious that the degree of success in implementing vision AI systems is highly dependent on a well-structured and reliable infrastructure. These aspects are commonly summarised under the terms of machine learning operations (MLOps) and human centered design (HCD). Our experiments are conducted using the industrial AI software Neuralyze®, which has served as a basis for several research projects starting in 2019 to test new approaches to machine learning in manufacturing. In our research, we introduce approaches on how to ideally integrate those methods into AI software concepts to derive an optimum benefit. It is a key goal to retain standardized handling semantics despite the variety of model architectures and use cases.

Keywords Machine vision, industrial imaging, image processing, image analysis, machine learning, deep learning, machine learning operations, user centered design, human centered design, context analysis, quality assurance

DOI: 10.58895/ksp/1000174496-4 erschienen in:
Forum Bildverarbeitung 2024
DOI: 10.58895/ksp/1000174496 | https://www.ksp.kit.edu/site/books/m/10.58895/ksp/1000174496/

1 Introduction

As the applied counterpart to computer vision, machine vision has been putting academic knowledge of image processing into practice almost continuously for over 40 years. This is also the case for machine learning based on convolutional neural networks (CNNs), which emerged as a novel approach to analyze camera data in actual applications about a decade ago. The discipline, usually referred as DL/ML (Deep Learning / Machine Learning) developed into a major research topic since then.

The related transfer into practical application however was subject to significant obstacles in its early phase - it's primary reason being the lack of computing power to be of any value in industrial production use cases. This barrier successively lowered by the fast development of GPU hardware and the related increase of GPU power. It also benefited from the growing interest of the scientific community, going along with the implementation of libraries offering abstractions for fundamental mathematical operations as well as transparent access to computing resources from high level programming languages, like Keras [1].

2017 marks a change with the release of Keras 2.0 as a hugely improved toolset to enable easy access for experimentation with CNNs. The top-level-library TensorFlow [2] added additional capabilities to the point of automated image set downloads. It further decreased the threshold to access deep learning technology, also for non-computer scientists. This also marks about the point where desktop GPU power had developed accordingly to enable first machine vision applications, yet still on very small image sizes.

Since then, major model architectures have evolved which focus on image analysis [3]. Their constant development has lead to a number of core applications that have emerged in the process. The central categories comprise classification, object detection and semantic segmentation, with a number of distinctive forms such as anomaly detection or combinations as in instance segmentation. In the majority of cases, the descendants of these architectures are capable of solving even the most complex machine vision problems when used appropriately.

This suggests that in terms of technical feasibility, as of 2024 almost any conventional image analysis task can be solved fast enough for inline processing in industry. This holds even more true as machine vision also encompasses the entire vertical design of the image acquisition and processing pipeline, and thus also has control over data generation.

However, experience in industry-grade development of those systems shows that the exclusive focus on the technical solution leaves key aspects of the deployment unaddressed. This can lead to poorly performing and and unsustainable vision AI solutions in the field. From a practice-oriented point of view, it becomes apparent that a consistent and well-structured development environment has a crucial impact on the operational success of machine learning systems. Our research seeks to explore ways to standardize these methods and make them more accessible.

2 Related work

Several prior works have already addressed the importance of a unified approach to cope with the complexity of AI applications. In general, the accuracy and performance of ML systems and in particular Vision AI systems depends on three main factors

1. the chosen model type
2. the model implementation
3. as well as the quality of the input data,

which implies a high complexity of these systems [4]. The interdependence of these three factors requires a high level of care already in the development phase with respect to versioning and reproducibility of the entire ML pipeline. During operations (MLOps), the data quality and quality of the models must also be continuously monitored in order to detect malfunctions at an early stage.

One particular challenge is the large landscape of tools for specific tasks, often developed on a small scale by start-ups or communities. The widely different operating paradigms encountered turned out to be an obstacle in constructing seamless workflows. In addition, the market for MLOps software is currently very dynamic due to the permanent release of new solutions. As one example, ”Tensorflow Extended” offers a generic platform for the development of ML systems that maps the complete ML lifecycle ”end-to-end” [5]. However, specially trained personnel such as data scientists, ML engineers and infrastructure teams are required to set up and operate such platforms.

In order to simplify access to ML systems for domain experts without AI expertise, first technical steps are already being taken by developing explanation methods to understand ML model predictions. Yet these methods are still mainly aimed at data scientists. In addition, ”best practices” from classical software development are increasingly being adopted and adapted to increase confidence in the development process [6] [7].

Due to the often probabilistic nature of ML systems, a key factor of good usability, expectation conformance according to ISO 9241-110:2020, is not given. This means that a system does not always behave similarly, and in particular predictably, even in repeated, identical interactions [8]. This complicates user experience (UX) design in the context of AI systems, since different misbehavior in particular cannot be predicted before model implementation is complete. Here, the use-case-specific development of ”AI playbooks” for designers and developers, which collect typical errors in the operation of ML systems can provide a remedy [8]. In addition, a comprehensive meta-study on guidelines for the development and design of AI systems has already summarised initial guidelines for the design of human-AI interaction. The derived 18 core design principles for human-centered design of AI systems are bundled in the Microsoft HAX Toolkit [9]. However, both of the described guidelines have been evaluated only on publicly available AI products for end users, but not yet on ”critical” applications as found in industry.

Finally, to further democratize ML, recent research suggests the notion of ”human-centric machine learning.” AI systems are now conceived as a symbiosis between humans and machines, and a shift in perspective from ”human-in-the-loop” to ”ML-in-the-loop” is called for [10] [11] [12].

3 Methodology

The initial ideas of the methods we target do not originally arise from a machine learning context. They evolved from good practice in adjacent fields, like Development Operations (DevOps) as a practice to unify and streamline all processes that are necessary to manage and build software code. Based on this ideal, MLOps emerged to achieve something analogous for the development of machine learning applications. Human Centered Design (HCD) originated from experimental psychology in the first half of the last century, expanding to a large range of fields since then. [13].

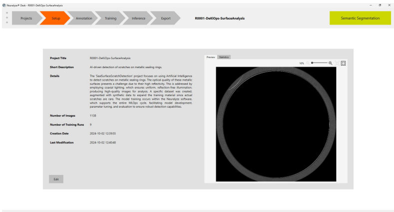
3.1 Machine Learning Operations (MLOps)

Organizations are confronted with many obstacles when optimizing machine learning systems within their technical lifecycles. Versioning of models, result repeatability, and preserving constant performance across different environments are among the key operational concerns [14]. Cross-functional cooperation, handling a variety of tools, and incorporating ML workflows with current procedures are organizational hurdles [14]. Issues with data quality, resource constraints, and model deployment challenges are the main concerns in industrial settings. We have devised a general method to address these problems in the beforementioned sectors.

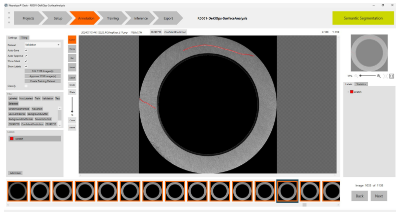
We illustrate our efforts with Neuralyze®, a software framework developed by senswork for AI-based image analysis, which puts the above tasks into practice. Figure 1(a) shows the general overview of a project in Neuralyze®. It serves as an informational entry point to provide any user in cross-functional collaboration projects with insights into the development process. This is of high importance for all involved personas of a vision AI project.

Figure 1(b) shows the data management step in the annotation tab. Users are provided with tools to handle data, which includes data cleaning, sorting, and tagging, gaining insights on the metadata, labeling the data, and finally creating datasets.

The subsequent step in an MLOps cycle is model development. The availability of ready-to-use datasets on sites like Kaggle [15] causes a significant change in focus toward model development. Many academic articles similarly emphasize getting high performance scores on benchmark datasets, with the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) being one of the most well-known examples [16]. Industrial experience has however shown that a more balanced strategy that incorporates data-centric strategies frequently produces superior long-term results.

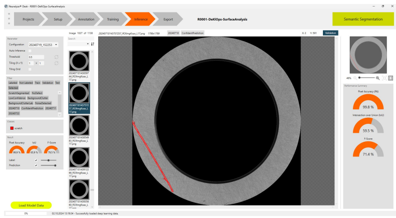


(a) The setup tab gives an overview of a complete AI project.



(b) The annotation tab provides tools to work on data.

(c) The training tab offers facilities to train computer vision models.



(d) The inference tab enables model testing and evaluation based on metrics.

Figure 1: Human centered interface of Neuralyze® Desk.

Neuralyze® follows the paradigm of data-centric AI (DC-AI). Data-centric AI is an emerging paradigm that emphasizes enhancing data quality and quantity to improve AI systems, complementing the traditional model-centric approach [17, 18].

Figure 1(c) shows the training tab in Neuralyze®. In this working area, users can develop a machine vision model based on the dataset created before. Users have the option to select predefined model architectures like ResNet [19]. The focus of this selection is put on model architectures that have proven effective in the field. Combined with a data-centric AI approach, this allows for the efficient and rapid development of models ready for production. Furthermore, the most important hyper-parameters, with default values based on empirical experience from industrial practice, are accessible to the user. Among them are the input size, batch size, number of epochs, learning rate, and a predefined selection of loss functions.

Figure 1(d) shows how models are evaluated in Neuralyze®. Various performance indicators and error metrics have been proposed for both regression and classification algorithms in engineering and sciences [20]. These metrics are often dependent on the dataset and the specific application of the model [21]. In Neuralyze® we have implemented the most important metrics for this task. These metrics are visualized in Neuralyze® so that users can easily evaluate the trained models based on them.

3.2 Human Centered Design (HCD)

Human-Centered Design (HCD) is an interdisciplinary approach that focuses on optimizing products towards user-friendlyness. As the design and implementation of engineering software is generally profound, and these systems are often highly complex, they require seamless interaction between humans and technology. We will outline the critical role of HCD in creating effective, efficient and satisfying engineering software solutions in relation to the scope of our work.

HCD places users at the center of the design process by iteratively involving them through prototyping, testing, and feedback collection at every stage of development [22]. In engineering software, usability issues can lead to reduced productivity or costly mistakes in high-stakes environments like aerospace, healthcare, and manufacturing [23].

In order to create engineering software with optimum usability, it is necessary to align the design with user needs. This requires knowledge of their characteristics, goals, tasks, environment and resources. The findings are collected by means of a user context analysis. The examination of these findings leads to requirements for the information architecture, system design and interaction design.

The industry-grade systems investigated in the research project represent processes that involve both manual activities, e.g. in production, and pure information work. This results in a wide range of potential requirements. Their identification requires the participation of various groups of stakeholders. Stakeholders in industry (quality assurance, production, technology deployment planning) as well as domain experts in relation to machine learning and AI applications must be involved.

Both in-depth and contextual interviews were used to collect data that served as the basis for the creation of proto-personas and task models. These easily understandable and communicable artefacts have been iteratively discussed and adapted with the respective stakeholders. For the DeKIOps project, 12 interview partners from both the industrial context and machine vision experts were surveyed.

It went apparent that tasks within the field of data exploration and feature engineering, belonging to the domain of data experimentation are difficult to define and cannot be fully mapped by engineering software. It is likely that these tasks have to be further performed by human experts in the future, using auxiliary tools that are closely tailored to the tasks. For machine learning on image data (vision AI), tasks relating to the creation and continuous, iterative improvement of neural models (training, retraining, monitoring) have been identified as key topics.

4 Discussion

MLOps frameworks are becoming more and more necessary as the complexity of implementing machine learning models in industrial systems increases. This is particularly important in machine vision, where productivity expectations require tasks like segmentation, classification, and object recognition to be improved. In this work, we utilized a prototype platform to show how MLOps concepts, such as automated monitoring and continuous integration/discovery, might simplify model construction for users who are technically inclined but may not be machine learning experts. This technique covers critical difficulties including model versioning, scalability, and performance monitoring [4].

4.1 Findings and Interpretation

Academically trained data scientists have historically been key roles in industrial AI model development. Our prototype, however, seeks to transfer this accountability to users who have received technical vocational training. The system enables such users to iteratively update and upgrade models as new product features or flaws develop. Though it was intended to someday be usable by non-academic users, those with an academic background now dominate the platform.

The platform's user-friendly interface effectively promotes communication and interaction among the various stakeholders in an organization, such as technicians, project managers, sales staff, and even non-technical personnel. This significantly increases the number of people who can enhance, manage, and optimize AI systems without requiring traditional AI development experts like data scientists. This type of cross-disciplinary collaboration is absolutely necessary to ensure that AI systems function reliably under various constraints, such as real-time processing, strict compliance with safety regulations, and scalability for large-scale operations [5].

The integration of Human-Centered Design (HCD) principles ensures that users without deep ML knowledge can interact effectively with the platform via simplified interfaces. This inclusion of HCD ensures that the system not only performs technically but is also usable and efficient for the end-users [22].

4.2 Limitations and Future Work

Even while the platform makes model construction easier, data scientists and machine learning experts are still needed for specialized solutions when dealing with demanding tasks. Furthermore, it is still difficult to define a terminology that unites ML experts and non-experts. Subsequent investigations will concentrate on creating a common lexicon and verifying how successfully non-technical users can utilize the system, finding difficulties they encounter.

An interesting question is whether the methods and processes of HCD can be applied for machine learning and AI systems on a general basis. Analysing the context of use in the DeKIOps project, it became clear that some HCD methods pose new challenges. The extension of HCD towards the scope of machine learning is a field of research to be further explored in the future.

5 Conclusion

In conclusion, scalable and dependable machine learning model deployment in industrial contexts requires the integration of MLOps frameworks, as the Neuralyze® platform demonstrates. Through the prioritization of data-centric AI and the facilitation of cross-functional cooperation, MLOps guarantees that models are resilient, replicable, and condition-adaptive. Simultaneously, an adoption of Human-Centered Design principles improves the platform's usability, making it accessible to both AI professionals and non-experts. The successful use of MLOps and HCD in difficult operational situations will be crucial for industrial AI systems as they develop further.

Acknowledgement

This research is accomplished within the project DeKIOps (AKZ DIK0451). We acknowledge the financial support for the project by the Bavarian Joint Research Programme (BayVFP) of the Free State of Bavaria.

References

1. "Introducing Keras 1.0," 2016, accessed on September 29, 2024. [Online]. Available: https://blog.keras.io/introducing-keras-10.html
2. "Announcing TensorFlow 1.0," 2017, accessed on September 29, 2024. [Online]. Available: https://developers.googleblog.com/en/announcing-tensorflow-10/
3. L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, cnn architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, Mar 2021.
4. D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, and D. Dennison, "Hidden technical debt in machine learning systems," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28, Advances in Neural Information Processing Systems. Curran Associates, Inc., 2015.

5. A. N. Modi, C. Y. Koo, C. Y. Foo, C. Mewald, D. M. Baylor, E. Breck, H.-T. Cheng, J. Wilkiewicz, L. Koc, L. Lew, M. A. Zinkevich, M. Wicke, M. Ispir, N. Polyzotis, N. Fiedel, S. E. Haykal, S. Whang, S. Roy, S. Ramesh, V. Jain, X. Zhang, and Z. Haque, "Tfx: A tensorflow-based production-scale machine learning platform," in *KDD 2017*. KDD 2017, 2017.
6. A. Serban, K. van der Blom, H. Hoos, and J. Visser, "Practices for engineering trustworthy machine learning applications," in *2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN)*, 2021 IEEE/ACM 1st Workshop on AI Engineering - Software Engineering for AI (WAIN). IEEE, May 2021.
7. S. R. Kaminwar, J. Goschenhofer, J. Thomas, I. Thon, and B. Bischl, "Structured verification of machine learning models in industrial settings," *Big Data*, Dec. 2021.
8. M. K. Hong, A. Fourney, D. DeBellis, and S. Amershi, "Planning for natural language failures with the ai playbook," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. ACM, May 2021.
9. S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen, J. Teevan, R. Kikin-Gil, and E. Horvitz, "Guidelines for human-ai interaction," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. ACM, May 2019.
10. Y. Liu and H. Shen, "Human-centric machine learning - a human-machine collaboration perspective," München, Tech. Rep., 2021.
11. B. Shneiderman, "Human-centered artificial intelligence: Three fresh ideas," *AIS Transactions on Human-Computer Interaction*, p. 109–124, 2020.
12. M. K. Lee, N. Grgić-Hlača, M. C. Tschantz, R. Binns, A. Weller, M. Carney, and K. Inkpen, "Human-centered approaches to fair and responsible ai," in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems. ACM, Apr. 2020.
13. J. Auernhammer, M. Zallio, L. Domingo, and L. Leifer, "Facets of Human-Centered Design: The Evolution of Designing by, with, and for People," in *Design Thinking Research*, ser. Understanding Innovation, C. Meinel and L. Leifer, Eds. Springer, November 2022, pp. 227–245.
14. A. Singla, "Machine Learning Operations (MLOps): Challenges and Strategies," *Journal of Knowledge Learning and Science Technology ISSN:*

2959-6386 (online), vol. 2, no. 3, pp. 333–340, Aug. 2023. [Online]. Available: https://jklst.org/index.php/home/article/view/107

15. "Kaggle documentation," 2024, accessed on September 29, 2024. [Online]. Available: https://www.kaggle.com/docs
16. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.
17. J. Jakubik, M. Vössing, N. Kühl, J. Walk, and G. Satzger, "Data-Centric Artificial Intelligence," Jan. 2024, arXiv:2212.11854 [cs]. [Online]. Available: http://arxiv.org/abs/2212.11854
18. A. Majeed and S. O. Hwang, "Towards Unlocking the Hidden Potentials of the Data-Centric AI Paradigm in the Modern Era," *Applied System Innovation*, vol. 7, no. 4, p. 54, Jun. 2024. [Online]. Available: https://www.mdpi.com/2571-5577/7/4/54
19. S. Targ, D. Almeida, and K. Lyman, "Resnet in Resnet: Generalizing Residual Architectures," Mar. 2016, arXiv:1603.08029 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1603.08029
20. M. Z. Naser and A. H. Alavi, "Error Metrics and Performance Fitness Indicators for Artificial Intelligence and Machine Learning in Engineering and Sciences," *Architecture, Structures and Construction*, vol. 3, no. 4, pp. 499–517, Dec. 2023. [Online]. Available: https://link.springer.com/10.1007/s44150-021-00015-8
21. R. S. Tiwari, "Model Evaluation," in *Fundamentals and Methods of Machine and Deep Learning*, 1st ed., P. Singh, Ed. Wiley, Feb. 2022, pp. 33–100. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/9781119821908.ch3
22. J. Hehn and D. Mendez, "Combining design thinking and software requirements engineering to create human-centered software-intensive systems," *arXiv preprint arXiv:2112.05549*, 2021. [Online]. Available: https://doi.org/10.48550/arXiv.2112.05549
23. A. Seffah, J. Gulliksen, and M. C. Desmarais, *Human-Centered Software Engineering: Software Engineering Models, Patterns and Architectures for HCI*. Dordrecht: Springer, 2005.