Association for Information Systems

AIS Electronic Library (AISeL)

Wirtschaftsinformatik 2021 Proceedings

Track 10: Design, management and impact of AI-based systems

Augmenting Humans in the Loop: Towards an Augmented Reality Object Labeling Application for Crowdsourcing Communities

Julian Schuir
Osnabrück University, Accounting and Information Systems, Osnabrück, Germany

René Brinkhege
Osnabrück University, Accounting and Information Systems, Osnabrück, Germany

Eduard Anton
Osnabrück University, Accounting and Information Systems, Osnabrück, Germany

Thuy Duong Oesterreich
Osnabrück University, Accounting and Information Systems, Osnabrück, Germany

Pascal Meier
Smart Enterprise Engineering, German Research Center for Artificial Intelligence, Osnabrück, Germany

See next page for additional authors

Schuir, Julian; Brinkhege, René; Anton, Eduard; Duong Oesterreich, Thuy; Meier, Pascal; and Teuteberg, Frank, "Augmenting Humans in the Loop: Towards an Augmented Reality Object Labeling Application for Crowdsourcing Communities" (2021). *Wirtschaftsinformatik 2021 Proceedings*. 5.
https://aisel.aisnet.org/wi2021/QDesign/Track10/5

Presenter Information

Julian Schuir, René Brinkhege, Eduard Anton, Thuy Duong Oesterreich, Pascal Meier, and Frank Teuteberg

Augmenting Humans in the Loop: Towards an Augmented Reality Object Labeling Application for Crowdsourcing Communities

Julian Schuir[1], René Brinkhege[1], Eduard Anton[1], Thuy Duong Oesterreich[1],
Pascal Meier[2], and Frank Teuteberg[1]

[1] Accounting and Information Systems, University of Osnabrück,
Osnabrück, Germany
{julian.schuir, rbrinkhege, eduard.anton, thuyduong.oesterreich,
frank.teuteberg}@uni-osnabrueck.de
[2] Smart Enterprise Engineering, German Research Center for Artificial Intelligence,
Osnabrück, Germany
pascal.meier@dfki.de

Abstract. Convolutional neural networks (CNNs) offer great potential for business applications because they enable real-time object recognition. However, their training requires structured data. Crowdsourcing constitutes a popular approach to obtain large databases of manually-labeled images. Yet, the process of labeling objects is a time-consuming and cost-intensive task. In this context, augmented reality provides promising solutions by allowing an end-to-end process of capturing objects, directly labeling them and immediately embedding the data in training processes. Consequently, this paper deals with the development of an object labeling application for crowdsourcing communities following the design science research paradigm. Based on seven issues and twelve corresponding meta-requirements, we developed an AR-based prototype and evaluated it in two evaluation cycles. The evaluation results reveal that the prototype facilitates the process of object detection, labeling and training of CNNs even for inexperienced participants. Thus, our prototype can help crowdsourcing communities to render labeling tasks more efficient.

Keywords: Crowdsourcing, Labeling, Object Recognition, Augmented Reality

1 Introduction

Data constitute the gasoline fueling artificial intelligence (AI) abilities [1, 2]. With cloud computing, the internet of things (IoT) and social media, data are increasingly abundant and accessible [3]. Yet, the availability of high-quality and structured training data is essential to leverage data for several supervised AI classifiers [4]. Given that up to 80% of corporate data are stored in an unstructured form [5], labeling data can be a costly and time-consuming endeavor [6]. As labeling is still mainly conducted by humans [7], many organizations rely on crowdsourcing platforms to render their labeling tasks more efficient [3]. Therefore, labeling represents a human-in-the-loop

16th International Conference on Wirtschaftsinformatik,
March 2021, Essen, Germany

approach, in which human skills are needed to gather training data for machine learning [8, 9].

Crowdsourcing platforms such as Amazon's MTurk enable organizations to outsource labeling as so-called “Human Intelligence Tasks” [10]. In this respect, the data type determines the complexity of the labeling job [6]. While high-level classification tasks (e.g. “cat” vs. “no cat” [11]) for images constitute straightforward and speedy annotation jobs, the complexity and duration increase with the requirements for visual perception within a video or image [12]. Consequently, labeling an object within an image is a challenging task that requires the capturing of additional position information within the observed frame [13]. In such cases, even in outsourcing scenarios the efficiency benefits are rather marginal [11]. Given these challenges, there is currently a lack of available solutions for labeling training data for use cases that enable efficient AI-based object recognition [14–16].

To remedy this shortcoming, researchers are increasingly focusing on providing tools that allow direct recognition and labeling of objects within a real-time environment leveraging augmented reality (AR) and convolutional neural networks (CNNs) [17]. AR involves the display of additional information in the user's field of vision and thus enables labeling tasks while capturing images [17, 18]. CNNs, meanwhile, are particularly performant for processing video or image data related to object recognition by utilizing three types of layers: the convolution layer, which generates the activation map enabling the identification of specific properties and defined spatial positions in a frame; the pooling layer, which reduces the dimensionality of the data; and the fully connected layer, which is responsible for linking the neurons from the previous layers [19]. Thus, the synergy of these technologies enables an end-to-end process of capturing objects, direct labeling and immediate embedding of captured information in CNNs’ training process [17]. Despite existing solutions for easing the labeling process of objects in images, to the best of our knowledge, there is no solution that is widely scalable to serve the crowdsourcing community. Previous solutions require either stationary hardware [12] or high processing power [17]. Considering this research gap, we derive the following research question (RQ):

> **RQ**: How can the process of capturing and labeling objects be designed and implemented as an AR application for the crowdsourcing community?

Therefore, the aim of this paper is to develop a mobile AR prototype for capturing, labeling and detecting objects based on training CNNs. Our solution is aimed at the crowdsourcing community as it provides the opportunity to capture labeled objects rather than to recruit thousands of workers to manually identify and label objects in images after they are captured.

In accordance with Gregor and Hevner [20], we organize our study as follows: Section 2 summarizes related work. Section 3 describes the incremental steps of the artifact development in line with the design science research (DSR) paradigm. This is followed by an explication of the artifact in Section 4 and a description of the evaluation in Section 5. Subsequently, we discuss our findings in Section 6. Finally, the paper concludes by summarizing the main findings.

2 Related Work

With advances in the fields of computer vision and neuroinformatics, artificial neural networks (ANNs) are expected to be increasingly used in business operations [21]. Thereby, CNNs constitute the most commonly used type of ANN architectures applied for image classification [19]. A very promising application area for CNNs is real-time object detection [22]. While the training for this application constitutes a time-consuming task, the subsequent object detection enabled by the trained model is carried out within milliseconds [23]. In view of these capabilities, CNNs are frequently associated with various application scenarios of the IoT age [24]. For example, robots can immediately detect quality deviations in production by using CNNs [25].

However, a basic prerequisite for the effective recognition is the availability of labeled and structured data as well as pre-trained CNNs [4, 10]. To meet this need, several crowdsourcing tools have already been designed to label data for CNN training. For instance, Lionbridge.ai employs thousands of crowdworkers to label and annotate images, videos and audio recordings [26]. Moreover, various solutions for structuring image data in the fields of medicine, traffic and machinery have been developed in research [16]. However, these solutions require pre-defined sets of images that first must be provided to enable crowdworkers to perform the labeling [6, 10].

The use of AR applications for training neural networks in terms of gathering labeled training data and object detection has been a rarity so far, although AR user interfaces offer unique potential by guiding the user through visual and auditory stimuli [18, 27]. Combined with AR, CNNs have so far mainly been used for the recognition of markers (e.g. barcodes) that facilitate the recognition process [28, 29]. For instance, Dash et al. [30] developed an AR learning environment that identifies markers in the user's field of view, computes the geometric data and seamlessly displays the 3D content in the video stream. To date, however, multiple CNN architectures, like AlexNet and GoogLeNet, have been deployed to allow object recognition without markers [31, 32].

To the best of our knowledge, only one study has combined object labeling, real-time object detection and AR: Hoppenstedt et al. [17] implemented a prototype for labeling objects for the Microsoft HoloLens. The application allows to use voice commands for storing the metadata (e.g. label). Data input generated from the AR labeling is stored in a folding neural network. This network is then trained to classify the images along with the corresponding objects. However, the results of their evaluation indicate that the architecture is more suitable for small classification problems. Furthermore, the application does not provide feedback to the user, which could cause problems for novices. Finally, the use of AR headsets is still not prevalent.

In conclusion, companies, crowdsourcing communities and previous solutions suffer from several shortcomings, which we categorize as belonging to seven central issues (I): The shortage of structured data (I1) leads to high efforts for labeling images (I2), which in turn are often outsourced to crowdworkers. However, crowdworkers often lack the necessary domain knowledge (I3) [5, 16]. Even though a number of solutions have already been developed, they lack scalability (I4) [17]. Moreover, the missing domain knowledge of crowdworkers leads to poor data quality of labeled images and objects (I5), resulting in low accuracy of CNNs (I7) [16]. However, recent technological developments relating to mobile devices have created significant potential for the combination of data collection and labeling [33]. Furthermore, advancements in the field of CNNs are creating opportunities to accelerate training processes (I6) while achieving a comparatively high level of accuracy [16, 17, 34]. In spite of these potentials, research has so far failed to identify a solution that combines the advantages of CNNs, mobile devices and scalable architectures.

3 Research Approach

Given the problem statement outlined in the previous section, we initiated the artifact development and followed the DSR methodology proposed by Peffers et al. [35]. Figure 1 illustrates the research approach in six main stages.

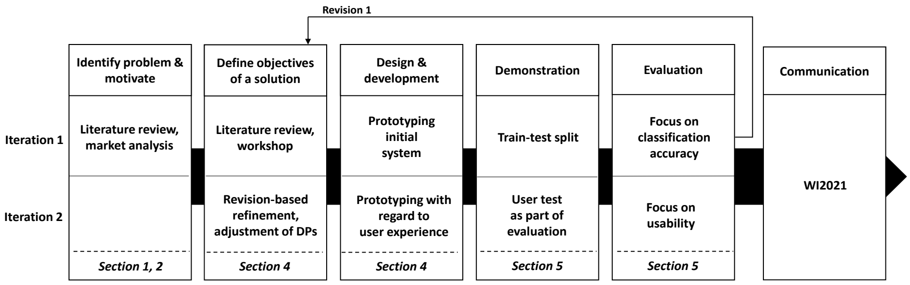

Revision 1
Identify problem & motivate
Define objectives of a solution
Design & development
Demonstration
Evaluation
Communication
Iteration 1
Iteration 2
Literature review, market analysis
Literature review, workshop
Prototyping initial system
Train-test split
Focus on classification accuracy
Revision-based refinement, adjustment of DPs
Prototyping with regard to user experience
User test as part of evaluation
Focus on usability
WI2021
Section 1, 2
Section 4
Section 4
Section 5
Section 5

Figure 1. Design science research approach based on Peffers et al. [35]

First, we examined the current state of practice and research by means of a market analysis and a literature review [36]. The former was conducted in the Apple App Store and the Google Play Store using search terms such as *object labeling* and *augmented reality labeling* [37, 38]. To identify relevant literature, we queried the scientific databases ScienceDirect, IEEE Explore, SpringerLink, ResearchGate and Google Scholar by applying the search string (*artificial neural networks OR connectionist models OR parallel distributed processing models OR convolutional neural networks) AND (augmented reality OR mixed reality) AND (label* OR training)*. This query yielded 43 research papers and two applications of particular importance for our project. To improve objectivity and validity, the screening process was conducted independently by two different researchers in line with the interrater agreement [39].

Second, we used a concept matrix according to Webster and Watson [40] for structuring the literature analysis. Thereby, we identified and categorized issues for the training of neural networks by means of a mobile application in the context of crowdsourcing. To subsequently deduce the meta-requirements (MRs) and design principles (DPs), we conducted a workshop with four researchers from the field of information systems (IS) and applied the anatomy proposed by Gregor et al. [41].

Third, we continued with the development of our artifact. Overall, we carried out two development cycles, each ending with an evaluation step to provide enhancements for the subsequent cycle. We employed two formative and naturalistic ex-post evaluations to examine the artifact's problem-solving ability in a real-world setting [42]. After the first design phase, we conducted a train-test split with 15 objects to validate the functionality of our artifact [43]. The second evaluation involved an experimental study and focused on the user experience. For this step, we applied the User Experience Questionnaire (UEQ) [44]. The two evaluation cycles are presented in Section 5.

4 Artifact Description

To address the observed real-world problem under consideration, we start by specifying the MRs, which describe the goals of our solution. These serve as a starting point for the derivation of DPs, which in turn guide the implementation of our artifact [20].

4.1 Meta-Requirements and Design Principles

Applying the research approach outlined in Section 3, we identified 12 MRs concerning *data labeling*, *system infrastructure* and *model development* (cf. Table 1).

Table 1. Meta-requirements

ID	Meta-requirements
Data labeling	
MR1	**Identification of unknown objects.** The system must help crowdworkers to identify previously unlabeled objects [45, 46].
MR2	**Highlighting the position of objects.** The application needs to enable crowdworkers to highlight the position of objects in the video stream in order to allow labelling [47].
MR3	**Recording multiple labeled data.** The system needs to be capable of recording multiple labeled training data within a short time [17].
MR4	**Intuitiveness.** Users without background knowledge need to be able to carry out the labeling process. Hence, the application needs to be intuitive to use [48].
System infrastructure	
MR5	**Scalability.** Given the need to train several models simultaneously, it is important to be able to train them in a parallel manner and thus enable scalable training [49, 50].
MR6	**Ubiquity of interaction device.** To enable crowdworkers to perform their tasks independent of location, a mobile device is required which functions as the user interface [51].
MR7	**Automation.** As outlined in Section 1, the training process requires an understanding of neural networks and does not constitute a trivial task [52]. Therefore, the training process is supposed to be automated to relieve the crowdworkers.
Model development	
MR8	**Processing of labeled training data.** To enable training, processing of camera data is required. Simultaneously, the recorded camera image needs to be visible to the user to be able to adjust the orientation of the camera [53].
MR9	**Diversified image data for an object.** To ensure the accuracy of the CNN, heterogeneous data need to be collected by recording the object from different perspectives [45].
MR10	**Time efficiency of training.** The training process needs to achieve useful results within the shortest possible time [34].
MR11	**Classification accuracy.** The CNN is intended to provide as few false positives as possible [54].
MR12	**Recognition and validation of previously trained objects.** To avoid redundant recordings by the user and verify the success of the trainings process, the application needs to notify the user of objects that have been recognized and highlight them [17].

Based on these MRs, we derived three initial DPs that guided us through the design process. In formulating each DP, we followed the anatomy proposed by Gregor et al. [41] to incorporate important elements like *aim*, *context* and *mechanism*.

Table 2. Design principles

ID	Design Principle Specification
Data labeling	
DP1	To allow crowdworkers to identify unlabeled objects in the environment and label them, provide a mobile application with capabilities for detecting and highlighting the objects to be labeled, because this intuitiveness facilitates the capture of objects for users without background knowledge in the domains of labeling and CNN.
System infrastructure	
DP2	To enable multiple crowdworkers to capture and label datasets, independently from their location, provide a mobile app that sends the captured data to a central server. This server, in turn, needs to be capable of automatically and simultaneously conducting the trainings process, because the centralization of training tasks enables the use of available resources as effectively as possible and crowdworkers lack the required background knowledge [55].
Model development	
DP3	To allow the system to train CNN algorithms with labeled input in a time-efficient and accurate manner, provide the CNN with heterogeneous, sufficient data and validate them against previously trained objects, because the storage capacity of mobile phones is limited, while neural networks require sufficient training data to maintain high accuracy.

Figure 1 visualizes the interrelation between the Is, MRs and DPs. Thus, for example, we address the issue of missing structured image data (I1) by enabling to identify objects that so far have not been labeled (MR1) [14, 15], thereby allowing users without background knowledge to identify and capture them in a structured manner (DP1).

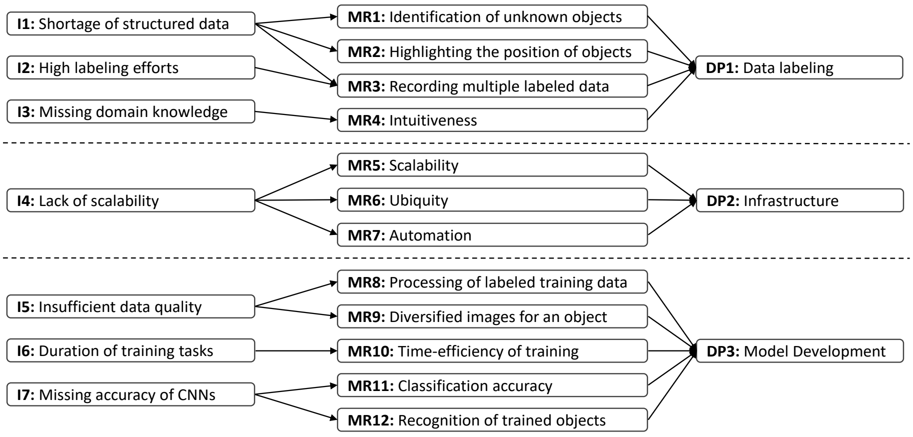

I1: Shortage of structured data
I2: High labeling efforts
I3: Missing domain knowledge
MR1: Identification of unknown objects
MR2: Highlighting the position of objects
MR3: Recording multiple labeled data
MR4: Intuitiveness
DP1: Data labeling
I4: Lack of scalability
MR5: Scalability
MR6: Ubiquity
MR7: Automation
DP2: Infrastructure
I5: Insufficient data quality
I6: Duration of training tasks
I7: Missing accuracy of CNNs
MR8: Processing of labeled training data
MR9: Diversified images for an object
MR10: Time-efficiency of training
MR11: Classification accuracy
MR12: Recognition of trained objects
DP3: Model Development

Figure 2. Issues, meta-requirements and design principles

To sum up, we identified seven Is that were translated into 12 MRs. Based on these, we derived three central DPs concerning *data labeling* (DP1), *infrastructure* (DP2) and *model development* (DP3).

4.2 Application

The design principles DP1, DP2 and DP3 governed the development of the application in the realms of data labeling, infrastructure and model development. The resulting system architecture is depicted in Figure 3.

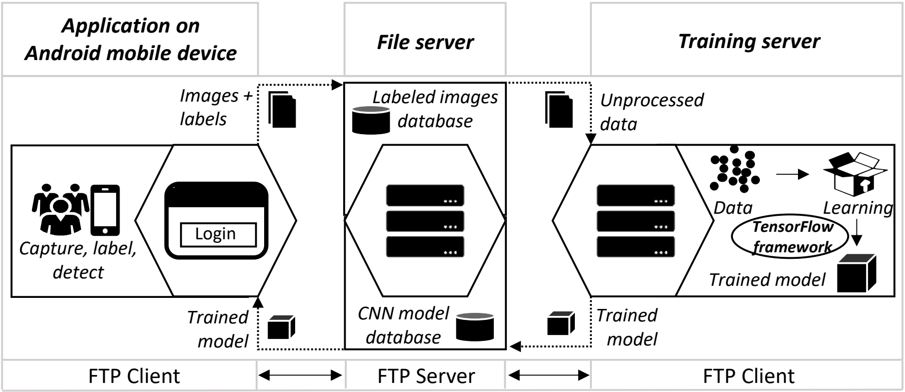
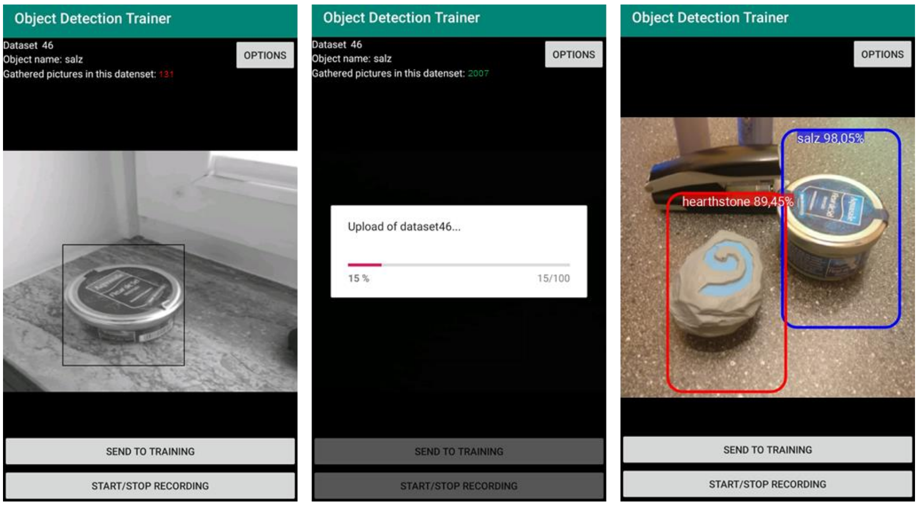


Figure 3. System architecture

To instantiate DP1, we developed an *application (app)* for mobile Android devices serving as the data collection component of the overall system to capture, label and detect objects within images. Since mobile devices usually do not have sufficient computing power for processing neural networks, we relied on the MobileNetV2 architecture integrated in Google's TensorFlow with regard to DP2 [56]. This resource-efficient architecture enables us to run CNNs on mobile devices [57] by incorporating the high-performance Single Shot MultiBox Detector (SSD), which handles the task of object detection, recognizing the object position in the image and its classification [58].

Once the user has completed the data collection process, the app transfers the information via file transfer protocol (FTP) to the data storage component and stores the data in a specific directory on a Linux server. Simultaneously, the *training server* monitors whether there are unprocessed data records on the *file server*. An implemented script downloads the identified unprocessed records and starts the training of a CNN model for a particular object class to incorporate DP3. Upon completion of the training, the resulting model is transferred back to the file server via FTP.

We developed an app for mobile Android devices on the client side in light of the operating system's corresponding smartphone market share [59]. The integrated camera enables users to capture and store images and the respective required spatial object information. When the user opens the app, the camera is activated, and the user is prompted to actively define a screen area by means of a bounding box in which the observed object is located in case the app does not recognize the object. The app automatically scans the object to check if it can be detected and recognized by previous capturing, labeling and training activities. If the object (e.g. the box of salt) can be detected, a rectangle appears around the object that is augmented on the camera screen with the presumed label and the accuracy in percent (cf. Figure 4, picture on the right). Otherwise, the user creates a new entry by clicking the button "create new object" and assigns a corresponding label (this would be necessary for the stapler in the right picture of Figure 3).



Object Detection Trainer

Dataset 46
Object name: salz
Gathered pictures in this datenset: 131

Object Detection Trainer

Dataset 46
Object name: salz
Gathered pictures in this datenset: 2007

Upload of dataset46...
15 %
15/100

Object Detection Trainer

Figure 4. Capture, label and detect object

Once the object area is marked and the label set (e.g. the box of salt, cf. Figure 4, picture on the left), the image capturing can be initialized. We enabled this procedure by deploying the CSR-DCF tracking method (CSRT) [60]. The first image is used as reference for the marked object area. The follow-up recordings are always validated by the CRST method by determining where the marked area is located on a new image. The CRST method corrects the marker and uses the corresponding input for the object detection. The image capturing is processed in black and white. The user receives meta-information at the top edge of the screen about the current capture and label process by the display of the selected label and the number of already captured images. The number is colored in green as a feedback function when the number of images reaches >2000 and in red when it is lower than this threshold (cf. Figure 4, picture on the left). The green color indicates that the amount of collected data is sufficient for a CNN training and that the user can proceed with the training process. The threshold for the image count was set at 2,000 because the first beta tests indicated satisfying results with this amount of data. The captured images are temporarily stored locally on the mobile device. To save the label and the information (width, height, xmin, ymin, xmax, ymax), the app also stores a CSV file for each image within the image folder. The coordinates of the object on the image are indicated by *xmin* and *ymin* for the lower left corner and *xmax* and *ymax* for the upper right corner of the bounding box; *width* and *height* refer to the overall image size. By clicking the button "send to training," the captured data is converted into a ZIP archive and transferred to the file server (cf. Figure 3, picture in the middle). After a record has been successfully sent to the file server, the associated data are deleted from the mobile device to free up local storage space. We further implemented several app functions to manage the end-to-end process (e.g. for monitoring the training status of a particular object class).

For the data processing component, we first installed the Python environment Anaconda 3.5 on the training server. This allows us to create independent Python environments without causing conflicts between them. We utilized several open source libraries and frameworks for building the training environment.

The training process starts by unpacking the downloaded ZIP archive and moving the images and labels to the designated locations in the environment. Subsequently, a script is executed that starts the training process. The training process ends when a predefined number of steps has been reached. Upon completion of the training, an implemented function converts the model into a format compatible with mobile devices (tflite) and sends the model via FTP to the file server. At this stage, the model can be used for object detection by displaying the label and accuracy of a detected object within the application.

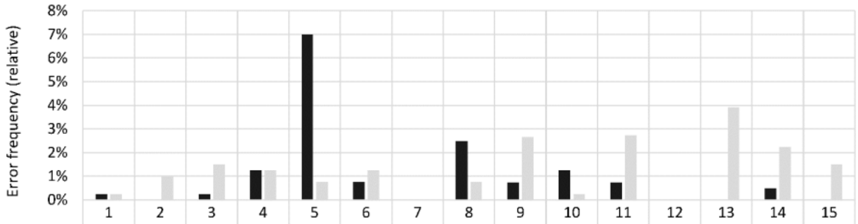
5 Evaluation

The prototype results from two build-evaluate cycles that enabled us to validate and improve our application through constant feedback. Given our objective was to develop a socio-technical artifact with user-oriented design risks, the FEDS framework by Venable et al. [61] inspired us to pursue a human-risk and effectiveness strategy.

The first evaluation cycle involved an assessment of the classification accuracy within a train-test split, whereas in the second evaluation cycle, we conducted an experiment with real end users to assess usability. Accordingly, in cycle 2, the application was first given to the volunteers to perform three tasks with the artifact: First, the environment had to be scanned for an unknown object. Second, the object had to be captured and labeled. Third, the captured object from the previous step needed to be validated using the application.

5.1 Cycle 1: Classification Accuracy

The first evaluation cycle involved examining the classification accuracy of the machine learning component by means of a train-test split [43]. To this end, 15 individual objects were captured and labeled using the mobile application. Each dataset comprised 2,000 images, with 80% of randomly selected images being used in training. To determine the accuracy, we subsequently analyzed these images by using the trained models and documenting the number of errors. We distinguished between two types of errors: undetected objects (1) and false positives (2). The former refer to errors that occur in cases where the object is in the camera image but is not recognized (type 1 errors), whereas the latter occur once the system indicates having recognized an object even though it is not in the camera frame (type 2 errors). We chose 50% as the baseline for a correctly detected object. Thus, an object is considered as detected if the model estimates the likelihood of being the targeted object to be 50% or higher. Figure 5 summarizes the frequency of the errors that occured during classification.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
■ Not detected (%)	0.2%	0.0%	0.2%	1.3%	7.0%	0.7%	0.0%	2.5%	0.7%	1.3%	0.7%	0.0%	0.0%	0.5%	0.0%
■ False positives (%)	0.2%	1.0%	1.5%	1.3%	0.8%	1.2%	0.0%	0.7%	2.7%	0.3%	2.7%	0.0%	3.9%	2.2%	1.5%

Figure 5. Error occurrence within classification per object

The average percentage of images with type 1 errors was 1.01%, whereas the corresponding average percentage for type 2 errors amounted to 1.34%. Hence, the share of incorrectly analyzed images can be considered low [17]. As shown in Figure 5, only the data set for object 5 constitutes an explicit outlier with a share of 7% for the type 1 errors, and we thus examined it in greater depth. Upon inspecting the dataset, we noticed that a number of images were taken by mistake. As the training process cannot independently separate such defective images from high-quality images, those images were also used for training the CNN.

In summary, the CNNs can detect objects at a low error rate. Upon completion of the train-split evaluation, we tested all models with regard to their operability on a mobile device for ensuring the functionality of the object detection functions before proceeding with the experimental evaluation in cycle 2.

5.2 Cycle 2: Usability

To assess the usability of our artifact and derive future research avenues, we adopted the User Experience Questionnaire (UEQ) developed by Laugwitz et al. [44] and supplemented it with an open question section. The participants received 26 word couples (e.g., unpleasant vs. pleasant, inefficient vs. efficient) and applied a 7-point Likert scale to rate the interaction with the technology in a range from -3 to +3. Apart from the UEQ questions, 15 participants were asked to submit feedback on the overall quality of the system and potential areas for improvement. Most of them were male (80%) while all of them were between 17 and 50 years old (with an average age of 32.4). One out of three (33.3%) were familiar with the concept of neural networks, and the remaining two thirds had no domain knowledge (66.7%). Nevertheless, all participants succeeded in completing the tasks, with an average duration of 30.42 minutes. Upon completion, the participants were asked to rate the interaction with the mobile application using the UEQ. Figure 6 illustrates the results of the UEQ survey in accordance with Laugwitz et al. [44].

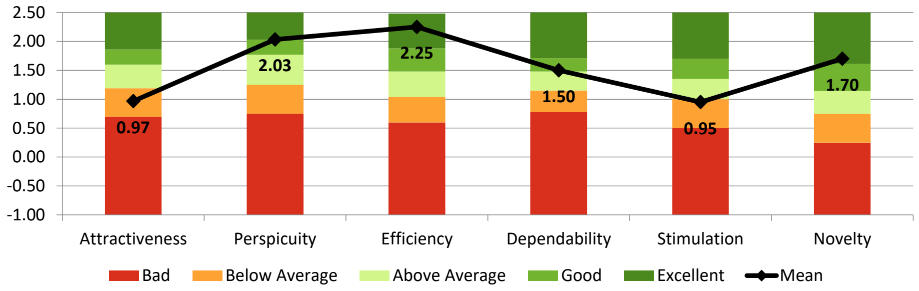


Figure 6. User Experience Questionnaire results

Overall, the average rating of all 26 items was positive by exceeding the critical mark of 0.8 (mean: 1.5). As proposed by Laugwitz et al. [44], the pre-defined items were aggregated into the six categories of *attractiveness*, *perspicuity*, *efficiency*, *dependability*, *stimulation* and *novelty*. These six categories achieved a mean value between 0.95 and 2.25 and in all cases a standard deviation below 1 for all six categories, which confirms a homogeneous positive impression of the system. We obtained the highest score for *efficiency* (2.25), a finding that reveals that users can accomplish labeling tasks within a short time. Furthermore, with high means in the categories of *perspicuity* (2.03) and *novelty* (1.70), the interaction with the prototype was on average perceived as “understandable” (2.10), “easy to learn” (2.30) and “clear” (1.80) while being regarded as a rather “creative” (1.40) and “innovative” (2.00) solution. The lowest values were given in the categories *stimulation* (0.95) and *attractiveness* (0.97), resulting from the negative ratings of the requested word couples “attractive” vs. “unattractive” (0.10) and “motivating” vs. “demotivating” (0.10).

Apart from small visual adjustments to the user interface design (e.g., integration of icons and a more user-friendly arrangement of buttons), the participants proposed to integrate a tutorial to guide users through the initial labeling process and thus avoid preventable errors. Another suggested major improvement concerned the highlighting of objects; according to the volunteers, the rectangular shape of the bounding box limits the quality and flexibility of the capturing process. An integration of a customizable shape to adjust the object position within the camera frame would be an enhancement to capture the object from different distances (e.g. by scaling). Moreover, the shape itself needs an indication by means of a striking color (e.g. green instead of black) to increase its visibility during the capturing process (e.g. within dark environments). Further improvement suggestions relate to the image capturing process: first, the user should be instructed on how to change the camera angle to improve the quality of the training data by providing different visual contexts for more heterogeneous images. This instruction can be achieved by displaying arrows that indicate the direction to rotate the camera. To adjust for poor-quality inputs, a function for deleting the last 50 images during the process must be provided.

We used the provided feedback from the second evaluation cycle for further improvement of the artifact. For example, we revised the arrangement of the user interface to provide the user with a more intuitive interaction. In addition, we improved the performance of the application by intensively modifying the source code.

6 Discussion, Limitations and Future Research

The process of labeling objects is a time-consuming and cost-intensive task [12] that is still mainly conducted by humans [7]. Many organizations rely on crowdsourcing platforms to outsource their labeling tasks [3]. As an alternative to manual labeling methods, tools are needed for the direct detection and labeling of objects within a real-environment. Responding to this need, we developed a mobile AR-based prototype for the object recognition, labeling and training of CNNs in three steps. First, we identified and derived the main issues, MRs and DPs based on a thorough literature review and a workshop. Interestingly, most MRs are concerned with model development (MR8-MR12), which underlines the major role of data processing and object recognition within the entire process. Second, we developed a mobile AR-based prototype that consists of three subsystems. Third, the prototype was evaluated in two iterations through an accuracy assessment and a UEQ-based survey conducted among 15 participants. The evaluation results reveal that the artifact facilitates the described process of object detection, labeling and training of CNNs even for inexperienced participants with no prior knowledge in this field. Against this background, we conclude that AR-based labeling constitutes a promising alternative or complement to the manual labeling of pre-defined data sets.

Given these findings, our research is of interest for practitioners for several reasons. First, crowdsourcing platforms and crowdworkers can be informed through our findings about the capabilities of AR-based systems for enhancing object labeling processes. In a similar manner, the proposed system architecture consisting of three interacting subsystems (cf. Section 4.2) is expected to be a more practical alternative compared to conventional system architectures with respect to system resources, system performance and scalability. Thus, we provide a scalable approach to the manual labeling methods of images (of videos) in the crowdsourcing context. For crowdworkers responsible for the tasks, the system can help to avoid cognitive overload and mental stress by facilitating the labeling process. Moreover, for developers, the proposed MRs and DPs can serve as a starting point when attempting to develop similar prototypes for object detection, labeling and training of neural networks. In addition, the mobile-based AR prototype and the corresponding infrastructure can be valuable for companies that are planning to implement AI-based image recognition systems as it facilitates the data entry step required for CNN training. By implementing the system, companies can thus collect structured data and train neural networks in a facilitated manner, thereby enabling real-time object recognition. One promising application area is the domain of logistics, where high-level object recognition can be employed for quality control of picking processes [62].

Apart from the practical relevance, the scientific contributions of this paper are manifold. First, the DPs contribute to the IS discipline by providing high-level guidance for researchers and developers in designing similar prototypes for object detection, labeling and training [35]. In doing so, our study aligns with prior IS research efforts on the interplay between humans and AI-based machines in the context of human-in-the-loop approaches (cf. [4, 9]). We encourage researchers from the IS discipline to critically examine our DPs with regards to modifications and extensions. Second, our findings expand the growing research stream on crowdsourcing human intelligence tasks by providing a mobile AR-based prototype as a substitute for the manual labeling of images [10]. However, the results of the second evaluation round based on the survey of 15 participants indicate major areas for improvement. For example, we found that the factors of attractiveness and stimulation displayed the lowest ratings in the UEQ survey, the latter being a consequence of the workers' lower cognitive loads due to the increase in repetitive tasks. Hence, the design of the user interface is subject to further improvements, along with considerations for how to redesign the user interface such that a well-balanced task-technology fit can be achieved. Therefore, researchers must find a trade-off between an attractive and stimulating design and a level of complexity for workers that is suited to their cognitive abilities [63]. For instance, recent research revealed that the integration of gamification elements represents a suitable instrument to enhance the user experience in terms of enjoyment with regard to labeling tasks [64].

Despite the promising results, our solution is subject to several limitations that highlight worthwhile avenues for future research. First, the MRs and DPs are based on a limited literature sample. Since we searched for literature in a limited number of databases by applying a limited set of search phrases, studies may have been overlooked that could be relevant for our research. Furthermore, the MRs and DPs are mainly literature-based. A possible extension of the requirements engineering step is to triangulate and complement the requirements with insights from experts to form a more practice-oriented view. Another limitation relates to the evaluation conducted to test the practicability and functionality of the prototype. Although we have evaluated the developed artifact, it has not been implemented and tested in a real business setting to date. A deployment of the prototype in a real case study, for example in cooperation with a crowdsourcing provider, constitutes the next step to further examine the impact of such a system on the contractors' and customers' work processes and organization as well as the associated social and economic implications. An important aspect to be considered is the impact of the system's use on the crowdworkers' skills requirements and cognitive performance, since AI-based systems facilitate the entire process of detecting and labeling objects and thereby render the workflows monotonous. Thus, the use of AI-based systems does not necessarily only lead to positive effects such as increased efficiency, but may also have negative consequences for humans in the loop (i.e. crowdworkers). At the same time, the human as an integral part of our socio-technical system constitutes an inherent source of vulnerability since capturing faulty data sets may lead to a decrease in the accuracy of the trained models, as shown in the first evaluation. Since our solution does not yet integrate any quality control mechanisms, the fully automated training could thus result in incorrectly trained models, thereby eliminating the advantage in terms of efficiency compared to existing solutions like Liongbridge.ai [26]. Future research could focus on answering the question of how these negative consequences can be avoided. Finally, our implementation does only concern Android devices. Thus, the use of other mobile devices (i.e. iOS) or devices such as AR glasses is not within the scope of this research and should be considered as a worthwhile avenue for future research. Likewise, conversational agents could be integrated into the system to facilitate the data entry step, especially when using AR glasses to enable hands-free working.

7 Conclusion

This paper presents a mobile AR-based prototype for capturing, labeling and detecting objects based on training CNNs following the design science research paradigm. Based on seven issues, we derived initial meta-requirements and design considerations from the scientific literature, that were translated into three design principles. We subsequently instantiated these design principles to develop a mobile AR-based prototype that consists of three subsystems. We evaluated and re-designed the artifact in two iterations though a train-test split and a usability assessment with 15 test users. The findings of the evaluations reveal that the proposed mobile-based AR prototype enables novices to detect objects and label them. A central server allows CNNs to be trained using the labeled data, generating models with a high degree of classification accuracy. Against this background, our research provides researchers and practitioners with a mobile application as a scalable alternative to the manual labeling methods of images in the context of crowdsourced labeling. The derived design principles serve as a higher-level guidance for system designers and IS researchers in the realm of AI-based assistance systems with regards to object labeling and recognition. Future studies should investigate the influence of AR-based labeling on crowdworkers' skill requirements and the integration of control mechanisms to ensure data quality.

Acknowledgements

The authors would like to thank the reviewers for their positive and constructive feedback. This contribution was prepared within the research training group "va-eva: Vertrauen und Akzeptanz in erweiterten und virtuellen Arbeitswelten" at the University of Osnabrück.

References

1. He, J., Baxter, S.L., Xu, J., Xu, J., Zhou, X., Zhang, K.: The practical implementation of artificial intelligence technologies in medicine. Nat. Med. 25, 30–36 (2019)
2. Sun, T.Q., Medaglia, R.: Mapping the challenges of Artificial Intelligence in the public sector: Evidence from public healthcare. Gov. Inf. Q. 36, 368–383 (2019)
3. Gu, Y., Leroy, G.: Mechanisms for automatic training data labeling for machine learning. In: 40th Int. Conf. Inf. Syst. ICIS 2019. München, Germany (2019)
4. Maedche, A., Legner, C., Benlian, A., Berger, B., Gimpel, H., Hess, T., Hinz, O., Morana, S., Söllner, M.: AI-Based Digital Assistants. Bus. Inf. Syst. Eng. 61, 535–544 (2019)
5. Accenture: Natural Language Processing Applications in Business. (2019)
6. Haq, R.: Enterprise Artificial Intelligence Transformation. John Wiley & Sons, Inc., Hoboken, New Jersey (2020)
7. Sun, Y., Lank, E., Terry, M.: Label-And-learn: Visualizing the likelihood of machine learning classifier's success during data labeling. In: Proc. of the 22nd International Conference on IUI, pp. 523–534. USA (2017)

8. Anton, E., Behne, A., Teuteberg, F.: The Humans behind Artificial Intelligence-an operationalisation of AI Competencies. In: 28th Eur. Conf. Inf. Syst. ECIS 2020. Marrakech, Morocco (2020)
9. Traumer, F., Oeste-Reiß, S., Leimeister, J.M.: Towards a Future Reallocation of Work between Humans and Machines – Taxonomy of Tasks and Interaction Types in the Context of Machine Learning. In: 38th Int. Conf. Inf. Syst. ICIS 2017. Seoul, Korea (2017)
10. Kauker, F., Hau, K., Iannello, J.: An Exploration of Crowdwork, Machine Learning and Experts for Extracting Information from Data. In: Lecture Notes in Computer Science, Vol. 10904, pp. 643–657. Springer, Heidelberg (2018)
11. Chang, J.C., Amershi, S., Kamar, E.: Revolt: Collaborative crowdsourcing for labeling machine learning datasets. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. pp. 2334–2346. ACM, New York, USA (2017)
12. Ramirez, P.Z., Paternesi, C., De Gregorio, D., Di Stefano, L.: Shooting Labels: 3D Semantic Labeling by Virtual Reality. arXiv preprint arXiv:1910.05021. (2019)
13. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2014. USA (2014)
14. Gao, P., Sun, X., Wang, W.: Moving object detection based on Kirsch operator combined with optical flow. In: IASP 10 - 2010 International Conference on Image Analysis and Signal Processing. USA (2010)
15. Rangel, J.C., Martínez-Gómez, J., Romero-González, C., García-Varea, I., Cazorla, M.: Semi-supervised 3D object recognition through CNN labeling. Appl. Soft Comput. 65, 603–613 (2018)
16. Zhang, J., Wu, X., Sheng, V.S.: Learning from crowdsourced labeled data: a survey. Artif. Intell. Rev. 46, 543–576 (2016)
17. Hoppenstedt, B., Kammerer, K., Reichert, M., Spiliopoulou, M., Pryss, R.: Convolutional Neural Networks for Image Recognition in Mixed Reality Using Voice Command Labeling. In: Lecture Notes in Computer Science, Vol. 11614, pp. 63–70. Springer, Heidelberg (2019)
18. Milgram, P., Kishino, F.: A taxonomy of mixed reality visual displays. IEICE Trans. Inf. Syst. 77, 1321–1329 (1994)
19. O'Shea, K., Nash, R.: An introduction to convolutional neural networks. arXiv Prepr. arXiv1511.08458. (2015)
20. Gregor, S., Hevner, A.R.: Positioning and presenting design science research for maximum impact. MIS Q. Manag. Inf. Syst. 37, 337–355 (2013)
21. Wäldchen, J., Mäder, P.: Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review. Arch. Comput. Methods Eng. 25, 507–543 (2018)
22. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149 (2017)
23. Pezeshk, A., Hamidian, S., Petrick, N., Sahiner, B.: 3D convolutional neural networks for automatic detection of pulmonary nodules in chest CT. IEEE J. Biomed. Heal. Informatics. 23, 2080–2090 (2018)
24. Jain, S.K., Rajankar, S.O.: Real-Time Object Detection and Recognition Using Internet of Things Paradigm. Int. J. Image, Graph. Signal Process. 1, 18–26 (2017)
25. Quack, T., Bay, H., Van Gool, L.: Object recognition for the internet of things. In: The Internet of Things. pp. 230–246. Springer, Heidelberg (2008)
26. Lionbridge Technologies: Lionbridge, https://lionbridge.ai/ (Accessed: 14.12.2020)
27. Chen, C.H., Wu, C.L., Lo, C.C., Hwang, F.J.: An Augmented Reality Question Answering System Based on Ensemble Neural Networks. IEEE Access. 5, 17425–17435 (2017)

28. Billinghurst, M., Clark, A., Lee, G.: A survey of augmented reality. Found. Trends Hum.-Comput. Interact. 8, 73–272 (2014)
29. Neges, M., Koch, C., König, M., Abramovici, M.: Combining visual natural markers and IMU for improved AR based indoor navigation. Adv. Eng. Informatics. 31, 18–31 (2017)
30. Dash, A.K., Behera, S.K., Dogra, D.P., Roy, P.P.: Designing of marker-based augmented reality learning environment for kids using convolutional neural network architecture. Displays. 55, 46–54 (2018)
31. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Commun. ACM. 60, 84–90 (2017)
32. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–9. USA (2015)
33. Vakharia, D., Lease, M.: Beyond Mechanical Turk: An Analysis of Paid Crowd Work Platforms University of Texas at Austin. In: Proc. iConference 2015. pp. 1–17. USA (2015)
34. Holzinger, A., Plass, M., Kickmeier-Rust, M., Holzinger, K., Crişan, G.C., Pintea, C.M., Palade, V.: Interactive machine learning: experimental evidence for the human in the algorithmic loop: A case study on Ant Colony Optimization. Appl. Intell. 49, 2401–2414 (2019)
35. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. J. Manag. Inf. Syst. 24, 45–77 (2007)
36. vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., Cleven, A.: Reconstructing the giant: On the importance of rigour in documenting the literature search process. 17th Eur. Conf. Inf. Syst. ECIS 2009. Verona, Italy (2009)
37. Google: Google Play Store, https://play.google.com/ (Accessed: 14.12.2020)
38. Apple: Apple App Store, https://www.apple.com/ios/app-store/ (Accessed: 14.12.2020)
39. LeBreton, J.M., Senter, J.L.: Answers to 20 questions about interrater reliability and interrater agreement. Organ. Res. Methods. 11, 815–852 (2008)
40. Webster, J., Watson, R.T.: Analyzing the Past to Prepare for the Future: Writing a Literature Review. MIS Q. Manag. Inf. Syst. 26, xiii–xxiii (2002)
41. Gregor, S., Kruse, L.C., Seidel, S.: The Anatomy of a Design Principle. J. Assoc. Inf. Syst. 21. 1622–1652 (2020)
42. Venable, J., Pries-Heje, J., Baskerville, R.: A comprehensive framework for evaluation in design science research. In: International Conference on Design Science Research in Information Systems. pp. 423–438. Springer, Heidelberg (2012)
43. Bronshtein, A.: Train/test split and cross validation in python. Underst. Mach. Learn. (2017)
44. Laugwitz, B., Held, T., Schrepp, M.: Construction and evaluation of a user experience questionnaire. In: Symposium of the Austrian HCI and usability engineering group. pp. 63–76. Springer, Heidelberg (2008)
45. Kent, D., Behrooz, M., Chernova, S.: Crowdsourcing the construction of a 3D object recognition database for robotic grasping. In: Proceedings - IEEE International Conference on Robotics and Automation, pp. 3347–3352. IEEE (2014)
46. Valdenegro-Toro, M.: End-to-end object detection and recognition in forward-looking sonar images with convolutional neural networks. In: Proc. IEEE/OES Auton. Underwater Vehicles (AUV) 2016. pp. 144–150. Tokyo, Japan (2016)
47. Li, C., Parikh, D., Chen, T.: Extracting adaptive contextual cues from unlabeled regions. In: Proc. of the ICCV 2011. Barcelona, Spain (2011)
48. Chatzimilioudis, G., Konstantinidis, A., Laoudias, C., Zeinalipour-Yazti, D.: Crowdsourcing with smartphones. IEEE Internet Comput. 16, 36–44. IEEE (2012)

49. Lee, S., Kang, Q., Madireddy, S., Balaprakash, P., Agrawal, A., Choudhary, A., Archibald, R., Liao, W.K.: Improving Scalability of Parallel CNN Training by Adjusting Mini-Batch Size at Run-Time. In: 2019 IEEE Int. Conf. on Big Data 2019. pp. 830–839, IEEE (2019)
50. Radovic, M., Adarkwa, O., Wang, Q.: Object recognition in aerial images using convolutional neural networks. J. Imaging. 3, 1–9 (2017)
51. Goncalves, J., Hosio, S., Rogstadius, J., Karapanos, E., Kostakos, V.: Motivating participation and improving quality of contribution in ubiquitous crowdsourcing. Comput. networks. 90, 34–48 (2015)
52. Cui, Y., Zhou, F., Lin, Y., Belongie, S.: Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop. In: Proc. of the IEEE conf. on computer vision and pattern recognition. pp. 1153–1162, IEEE (2016)
53. Kawano, Y., Yanai, K.: FoodCam-256: A large-scale real-time mobile food recognition system employing high-dimensional features and compression of classifier weights. In: MM 2014 - Proc. of the 2014 ACM Conference on Multimedia. pp. 761–762, ACM (2014)
54. Navalpakkam, V., Itti, L.: Sharing resources: Buy attention, get object recognition. Int. Work. Atten. Perform. Comput. Vis. WAPCV. 73–79 (2003)
55. Briese, C., Schlüter, M., Lehr, J., Maurer, K., Krüger, J.: Towards Deep Learning in Industrial Applications Taking Advantage of Service-Oriented Architectures. Procedia Manuf. 43, 503–510 (2020)
56. Abandi, M., Agarwal, A., Barham, P., Al., E.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. ArXiv preprint arXiv:1603.04467. (2015)
57. Sandler, M., Howard, A.: MobileNetV2: The Next Generation of On-Device Computer Vision Networks. https://ai.googleblog.com/2018/04/mobilenetv2-next-generation-of-on.html (Accessed: 16.12.2020)
58. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: SSD: Single shot multibox detector. In: Lecture Notes in Computer Science, Vol. 9905, pp. 21–37. Springer, Heidelberg (2016)
59. Statista: Mobile operating systems' market share worldwide from January 2012 to December 2019. (2020)
60. OpenCV: OpenCV: cv: TrackerCSRT Class Reference. (2000)
61. Venable, J., Pries-Heje, J., Baskerville, R.: FEDS: A Framework for Evaluation in Design Science Research. Eur. J. Inf. Syst. 25, 77–89 (2016)
62. Stoltz, M.H., Giannikas, V., McFarlane, D., Strachan, J., Um, J., Srinivasan, R.: Augmented Reality in Warehouse Operations: Opportunities and Barriers. IFAC-PapersOnLine 50. 12979–12984 (2017)
63. Goodhue, D.L., Thompson, R.L.: Task-technology fit and individual performance. MIS Q. Manag. Inf. Syst. 19, 213–236 (1995)
64. Spatharioti, S.E., Fatehi, B., Smith, M., Rosenbloom, A., Miller, J.A., Seif El-Nasr, M., Wylie, S., Cooper, S.: Tile-o-Scope AR: An Augmented Reality Tabletop Image Labeling Game Toolkit. In: FDG 2020 Proc. pp. 1–4. USA (2020)