ECONSTOR
Make Your Publications Visible.

A Service of

ZBW Leibniz-Informationszentrum Wirtschaft
Leibniz Information Centre for Economics

Rieder, Mathias; Verbeet, Richard

Conference Paper

Realization and validation of a collaborative automated picking system

Provided in Cooperation with:
Hamburg University of Technology (TUHH), Institute of Business Logistics and General Management

Suggested Citation: Rieder, Mathias; Verbeet, Richard (2020) : Realization and validation of a collaborative automated picking system, In: Kersten, Wolfgang Blecker, Thorsten Ringle, Christian M. (Ed.): Data Science and Innovation in Supply Chain Management: How Data Transforms the Value Chain. Proceedings of the Hamburg International Conference of Logistics (HICL), Vol. 29, ISBN 978-3-7531-2346-2, epubli GmbH, Berlin, pp. 521-558, https://doi.org/10.15480/882.3131

This Version is available at:
https://hdl.handle.net/10419/228932

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

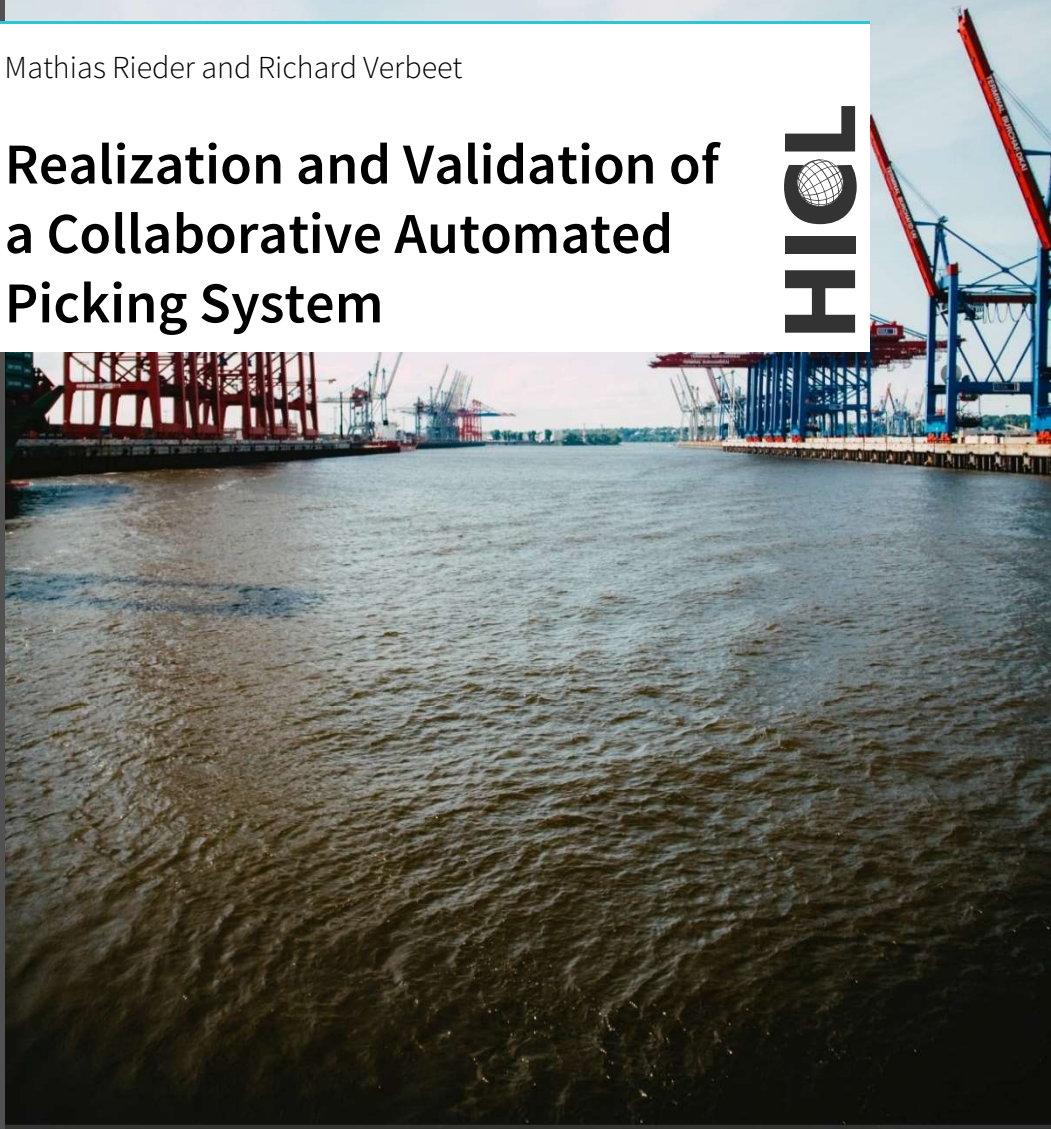
https://creativecommons.org/licenses/by-sa/4.0/

WWW.ECONSTOR.EU

Mitglied der Leibniz-Gemeinschaft

Mathias Rieder and Richard Verbeet

Realization and Validation of a Collaborative Automated Picking System



CC-BY-SA4.0

Published in: Data science and innovation in supply chain management
Wolfgang Kersten, Thorsten Blecker and Christian M. Ringle (Eds.)
ISBN: 978-3-753123-46-2 , September 2020, epubli

Realization and Validation of a Collaborative Automated Picking System

Mathias Rieder [1] and Richard Verbeet [1]

1 – Ulm University of Applied Sciences

Purpose: A picking system is presented ensuring order fulfilment and enabling transformation from manual to automated picking using a continuous learning process. It is based on Machine Learning for object detection and realized by a human-robot collaboration to meet requirements for flexibility and adaptability. A demonstrator is implemented to show cooperation and to evaluate the learning process.
Methodology: The collaborative process, system architecture, and an approach for evaluation and workload balancing for order fulfilment and learning of robots during picking have already been introduced. However, a practical application is still missing. A demonstrator is implemented using an agent-based architecture (JADEX) and a physical robot (UR5e) with a camera for object detection and first empirical data are evaluated.
Findings: Single components of the demonstrator are already developed, but a pending task is to implement their interaction to analyze overall system performance. This work focuses on human-robot-interaction (Emergency Call), automated generation of images extended by feedback information, and training of algorithms for object detection. Requirements of human-machine interface, technical evaluation of image recording, and effort of algorithm training are discussed.
Originality: Many approaches for automated picking assume a static range of objects. However, this approach considers a changing range as well as a concept for transformation of manual to automated picking enabled by human-robot cooperation and automated image recording while enabling reliable order fulfilment.

First received: 12. Mar 2020 Revised: 21. Jun 2020 Accepted: 12. Aug 2020

1 Introduction

Picking as a major logistic task is the customer order specific composition of a subset from a total assortment of goods (VDI, 1994). This composition often must be executed by exact number and therefore is often carried out by humans due to the impractical level of standardization of processes and the resulting needed flexibility (Müller, et al., 2019). This flexibility is based on the ability of humans to observe and understand their environment and to recognize and grip objects.

Order picking must often face steadily changing environments imposing high requirements on automated systems' flexibility. Recent developments in the field of Artificial Intelligence have reached a level of maturity to enable flexibility and adaptivity allowing the automation of processes only humans are able to execute so far (Gerke, 2015). But even considering further technical and conceptual progress in automated picking, robots will depend on humans in order picking systems. Therefore, an efficient setup of an operational human-robot picking system needs a reliable human-machine interaction (Azadeh, Koster and Roy, 2017). This kind of interaction must be integrated into planning and control of order picking systems fundamentally. A nearby approach is a cooperative picking environment to help robots to fulfil order picking and to support their ability of adaption. By developing sensor technologies to ensure the safety of humans the conditions for cooperative industrial applications have been created (Müller, et al., 2019). The cooperation of humans and robots is also of major importance in the context of the emergence of Cyber-Physical Systems in Industry 4.0 (Kamarul Bahrin, et al., 2016).

The picking environment must be implemented by an architecture enabling the realization of an adaptive robot system as well as the fulfilment of picking orders. Agent technology has proven to be a reliable approach by transforming heterogenous technical systems into Cyber-Physical Systems to realize integrated connectivity. These systems can also be equipped with local mechanisms for problem solving (Verbeet and Baumgärtel, 2020). (Verbeet, Rieder and Kies, 2019) introduce such an approach.

Contributions of this paper are the following: (1) definition and creation of a data set for training and testing of Convolutional Neural Networks (CNN) for object detection during automated picking, (2) training, evaluation and selection of CNN, (3) realization of a demonstrator consisting of the training environment described by Verbeet et al. (2019) and a virtual picking environment, which is implemented as multiagent-system, as well as (4) an analysis of a picking system by the demonstrator using selected CNN to evaluate the by Verbeet et al. (2019) proposed concept of a feedback loop (Emergency Call) in a cooperative human-robot picking system.

The remainder of this work is organized as follows. The second chapter describes related work about approaches and technical applications of automated picking. In chapter three the concept for a cooperative picking environment is described, which is used to design the demonstrator presented in chapter four to evaluate the resilience of this concept according to order fulfilment and improvement of object detection. Chapter five presents first empirical results describing the data set, training of CNN for object detection as well as first experiments. This paper concludes with a critical discussion and a prospect to future research.

2 Related Work

Many approaches exist to encounter challenges in order picking by automation of processes. In established applications and concepts robots often either transport or grasp goods, but rarely carry out the complete picking task (EHI Retail Institute, 2019).

(Wang, Chen and Wang, 2019) propose a heuristic for local routing of distributed units in a collaborative human-robot picking system. Many approaches focus on partwise automation by automating transportation within intralogistics (Zou, Zhang and Qi, 2019) (Valle and Beasley, 2019) because the tasks of recognizing and gripping objects during picking are too complex to be fulfilled by machines (Jansen, et al., 2018). Kugler and Gehlich (2013) propose a goods-to-person picking system by an agent-based conveyor system supplying human pickers. Salah et al. (2018) describe a human-robot system for apple harvest, in which robots transport apples as autonomous bins.

In contrary to transportation where some degree of standardization can be established by loading devices, gripping an object requires high flexibility (Müller, et al., 2019), because each object or at least each object class must be gripped in its own way (Liu, Huang and Huang, 2019). A common approach is the creation of a known and controlled picking environment. Martinez et al. (2015) propose a system for bin picking and Wahrman et al. (2019) for shelf picking. An industrial application for bin picking is provided by Photoneo (2020).

Only a few solutions exist to fulfill the whole picking task considering movement to shelves, picking objects and delivery to a transfer place. An application in a laboratory environment is provided by Bormann et al. (2019).

Magazino realizes an industrial picking robot capable of travelling to shelves, picking specific objects (cubic objects) and delivering them to a transfer station (Mester and Wahl, 2019). Such systems only provide a limited ability to adapt to changes of the picking environments or the range and look of stored objects concerning reliability of automated object detection (Wahrmann, et al., 2019). Bormann et al. (2019) and Thiel et al. (2018) confirm the need for an adequate amount of training samples in logistics environments to enable a reliable object detection.

One approach to face these dynamics in picking is a human-machine cooperation where robots support human activities and humans may compensate a robot's lack of adaptivity (Lee, Chang and Choe, 2018). Werner et al. (2017) describe the collaboration of humans and robots in a static assembly cell. Rieder and Verbeet (2019) present a process model to realize a cooperative picking system defining an Application-Phase and a Learning-Phase to ensure order fulfillment and continuous improvement of robots' ability for object detection. This model is extended by Verbeet et al. (2019) by an Adjustment-Phase and a Cooperation-Phase as well as by a conceptual picking system. Rieder and Verbeet (2020) show how this picking system can be evaluated using a capacity evaluation. They define an equilibrium between the requirements of order processing, the picking performance of humans and robots and the effort for improving object detection. By linear programming it can be used for strategic evaluation of the automated picking performance of robots, for tactical resource planning, and for operational workload balancing.

3 Cooperative Picking System

The model for a cooperative picking system proposed by Verbeet et al. (2019) contains two basics phases for learning and application. Within a Learning-Phase an algorithm for object detection is created and improved using image data recorded in a controlled environment as well as data from operational picking processes. This phase is decoupled from operational order picking within an Application-Phase where humans and robots work in parallel within a picking environment. A picking robot is supposed to successfully grip and withdraw from a storage location after a successful object detection, e.g. by a combination of images and depth information as discussed by Shao and Hu (2019). In case of an unsuccessful object detection a robot tries to find a solution on its own by predefined options during an Adjustment-Phase, e.g. by moving its camera to a different position. If this is not successful, a Cooperation-Phase is triggered calling a human picker to support the robot (Emergency Call).

As an extension to the original concept this support is organized in three levels. At first the human picker tries to modify the environment to enable the robot to detect the object, e.g. if it is covered or has fallen (Support Level 1). When the robot is still unable to detect the object, the human picker marks the object with a bounding box in an image recorded from current storage location using an interface provided by the robot. This bounding box is used for calculation of a gripping point and the recorded image is saved for retraining (Support Level 2). In case of a not successful calculation the human picker picks the object to fulfill the robot's picking order (Support Level 3). However, due to the assumption picking robots can success-

fully grip every object they can detect, this last support level is not considered in this paper. The complete process of an Emergency Call is shown in Figure 1.

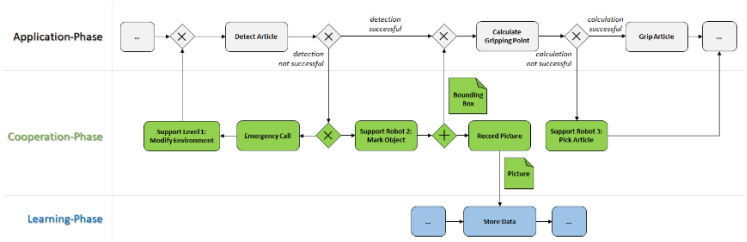


Figure 1: Support Level 1-3 during the process of an Emergency Call

Verbeet et al. (2019) propose the agent-based system architecture shown in Figure 2 to realize the process model. A Warehouse Management System (WMS) is responsible for administration of inventory data and initializing the allocation of picking orders. Human pickers and picking robots cooperatively process assigned orders. Furthermore, a Picture Recording Machine (PRM) is used for image recording in a controlled environment (Rieder and Verbeet, 2019). These images are stored on a data server and are used for training of algorithms for object detection by a computation cluster. Interaction patterns define the sequence of messages between components and embed it into picking processes. Assignment of orders to human pickers and picking robots is of major importance because it enables workload balancing. It is realized by a one-stage auction process arranged according to Contract Net Protocol (FIPA, 2019). The same mechanism is used for the assignment of an Emergency Call.

A picking robot can detect each object with a probability dependent on a trained algorithm for object detection. Rieder and Verbeet (2020) propose an evaluation of the picking system considering system parameters like working time (WT), picking capacity of humans (C_H) and robots (C_R) and a demand forecast (D_F) to define an equilibrium for system performance:

$$C_H \cdot WT - \frac{L_{EC,H,SR}}{WT} + C_R \cdot WT - \frac{L_{EC,R,SR}}{WT} \geq D_F \qquad (1)$$

$L_{EC,H,SR}$ and $L_{EC,R,SR}$ describe a time effort due to Emergency Calls for humans and robots considering a probability for object detection (P_{OD}) weighted with a demand for each object.

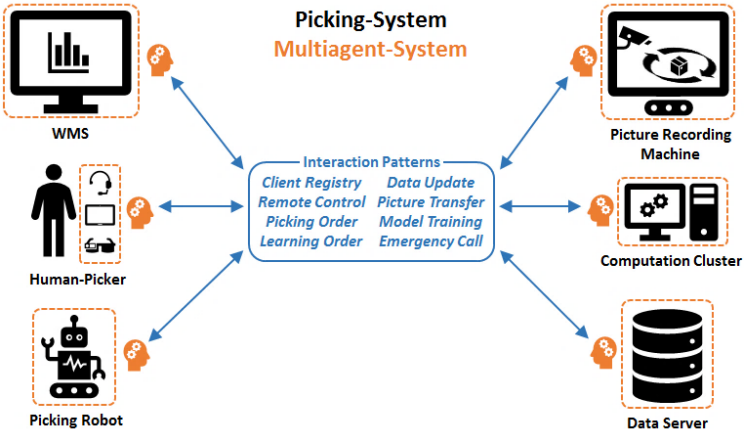


Figure 2: Agent-based architecture of a picking system realizing the process model according to (Verbeet, Rieder and Kies, 2019)

4 Demonstrator

An approach for a cooperative picking system has been developed and a concept how this system can be realized. An architecture has been proposed, its interactions are designed to realize the concept, and its performance in terms of picking and learning can be measured. A next step is the analysis of empirical data to show the resilience of the concept and to improve the mechanisms and processes.

Therefore, a demonstrator is created to illustrate the general functionality of the approach. It is implemented in two stages, which initially are supposed to work independently of one another: training environment and picking system. The training environment enables the recording of images and the training of a neural network for object detection. The picking system is an agent-based demonstrator in which the trained algorithms are used to evaluate the performance of object recognition and the effect of Emergency Calls.

4.1 Training Environment

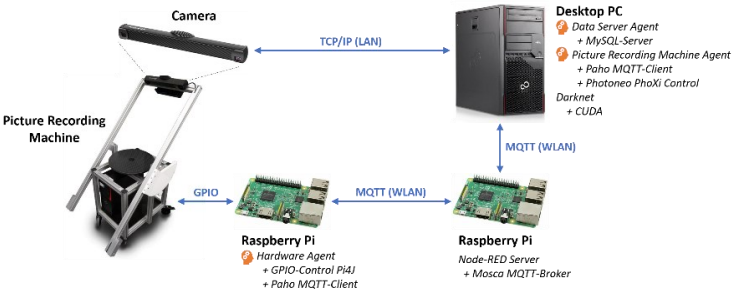


Figure 3: System architecture of the training environment

The system architecture to realize the interaction patterns "Picture Transfer", "Model Training" und "Data Update" is shown in Figure 3. For programming agents, the framework JADEX (Braubach, Pokahr and Lamersdorf, 2011) is used realizing agents following BDI concept (Bratman, 1987) basing on the FIPA compliant (FIPA, 2019) framework JADE (Braubach, Lamersdorf and Pokahr, 2003).

Image recording is done by PRM (Rieder and Verbeet, 2019). Its motors are controlled by a Hardware Agent implemented on a Raspberry Pi using GPIO to trigger motor drivers. Connection to an agent on a desktop PC is established by a Mosca MQTT broker (Noren and Müller, 2020). This allows control of PRM by the services "moveRocker" and "moveTurntable" using FIPA conform XML messages. Paho is used as MQTT client (Eclipse, 2020). The Photoneo 3D Scanner is connected to the desktop PC via LAN. Image recording of objects is coordinated by a Picture Recording Machine Agent using Photoneo software PhoXi Control and controlling object positioning by

services of PRM. A graphical user interface is provided to interact with this agent to control image recording. Recorded images are saved locally and referred to in a MySQL database managed by a Data Server Agent offering necessary services to select or change data.

The images must be annotated before used for training. The software Yolo Mark (Bochkovskiy, 2020b) is used for annotation displaying images and object classes. A user marks the objects with a bounding box after choosing the appropriate object class. Selected class as well as coordinates of the bounding box are saved to a text file.

A Convolutional Neuronal Network (CNN) is used for object detection within the demonstrator. It is being trained with recorded and annotated images by the framework Darknet (Redmon, 2016). This training bases on a neural network that was pretrained using MS COCO data set (Lin, et al., 2014) to reduce training duration (He, Girshick and Dollár, 2018). This network is trained using Darknet and CUDA support (NVIDIA Corporation, 2020) by a special adaption of YOLO algorithm version 3 (Redmon and Farhadi, 2018) for Windows (Bochkovskiy, 2020a). Windows 10 (64-Bit) is used as operating system running on a commercial desktop PC equipped with 32GB RAM and NVIDIA GeForce GTX 745. The trained CNN is saved locally and is referenced in the database.

Complete automation is not realized yet, i.e. recording, annotation and training must be controlled manually. Automated placement of an object on PRM is not possible and reliable automation of image annotation within such a well controllable recording environment must be evaluated. Annotation must be done highly accurate to ensure training success in follow up processes.

4.2 Picking System

The structure of a warehouse's picking zone and the agent types creating the multiagent-system to realize the interaction patterns "Picking Order" und "Emergency Call" are shown in Figure 4. Again, agents are implemented by JADEX. Their communication is realized by a framework specific service-oriented architecture (SoA) based on service interfaces. Physical processes are simulated by an idle function within the agents.

Human and Robot Picker Agents do not have a physical representation yet, i.e. it is a completely virtual realization of the picking system. However, the agents can be understood as digital twins of the components with their capabilities and can be extended by hardware interaction in a subsequent step. Using a SoA makes the interaction pattern "Client Registry" not necessary, appropriate mechanisms for registry, discovery and invocation are provided by the JADEX framework. Furthermore, there is no need for "Remote Control" as there is no hardware. To simplify data evaluation an additional Monitor Agent is implemented. The interaction pattern "Learning Order" is not realized at this stage of work.

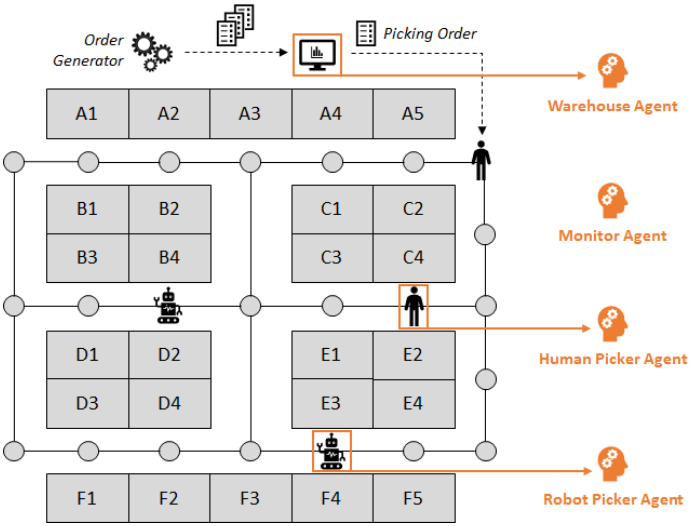


Figure 4: Warehouse structure and agent types of the virtual demonstrator

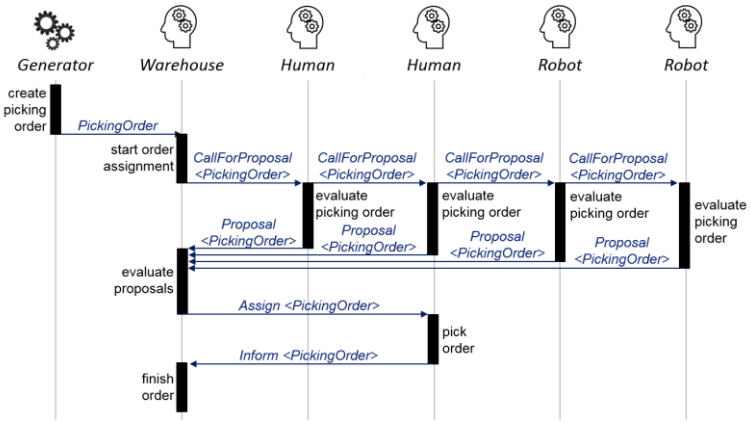


Figure 5: Auction for assignment of a picking order

Picking orders are generated by an order generator within a Warehouse Agent and are distributed to human pickers and picking robots using the auctioning process shown in Figure 5. For simplicity, picking orders consist of one order line containing a single object with quantity one. During this one stage auction a call for proposal is sent to all human pickers and picking robots requesting their current workload as well as their estimation to execute the transmitted picking order. The Warehouse Agent selects the picker with the lowest response value and assigns the order (Verbeet, Rieder and Kies, 2019).

Prerecorded images are used to simulate operational object detection within the demonstrator. These images are not included in the data set the CNN is trained with and are not used for follow up retraining. For each object detection an image is randomly chosen and is handed over to the CNN. Each agent contains a list of assigned picking orders which are processed one after another in sequence of assignment. To fulfill a picking order, an agent moves through the picking zone on shortest path to the waypoint assigned to the storage location of a corresponding object. After reaching this storage location Human Picker Agents start gripping instantly while a Robot Picker Agent starts an object detection using Darknet. A successful object detection is followed by gripping the object and completing the order. If the prediction accuracy is below the certain threshold of 95% a Robot Picker Agent tries to achieve sufficient accuracy by recording and testing a new image of the current shelf with a maximum of two attempts (Adjustment-Phase).

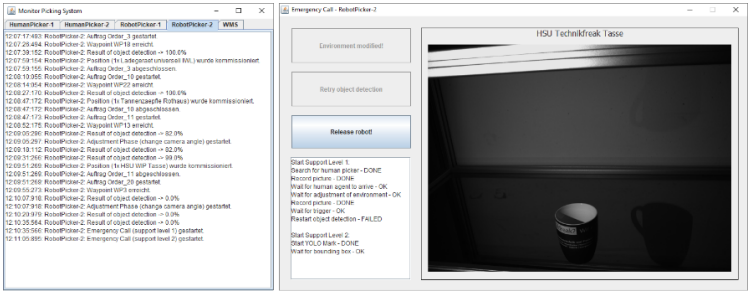


Figure 6: Monitor Agent showing processing of a picking order by a Robot Picker Agent (left) and the GUI of an Emergency Call (right)

In case of still failing, an Emergency Call is triggered calling a nearby Human Picker Agent. This agent interrupts its current activity and moves to the Robot Picker Agent's position (waypoint) which enables interaction by a GUI. Therefore, an Emergency Call is handled by the user modelling the support levels "Modify Environment" and "Mark Object" proposed in chapter three. The modification of the environment is simulated by loading another image followed by an object detection. If the predicted accuracy is below the threshold Yolo Mark is started enabling the user to mark the searched object. After finishing marking the robot can be released to grip the object and complete the picking order. Figure 6 shows a Robot Picker Agent's status updates during order processing (left) and the GUI for an Emergency Call (right).

5 Empirical Data

This chapter introduces the data set of images used for training of the neural network, the network itself and empirical results of object detection. Furthermore, neural networks in different training stages are evaluated by the demonstrator to estimate their impact on the overall picking system.

5.1 Data Set

A data set containing the following 12 objects of different materials is generated to be used by the demonstrator: a bucket, 4 different ceramic cups, a glass, 2 different plastic cups, a key, a charger, a plastic bottle, and a glass bottle. These objects are supposed to represent an object set in a realistic logistics environment with partially similar shape. Images of this data set are grouped into the three categories "PRM-Data" (images created by PRM), "EC-Data" (images based on Emergency Calls), and "D-Data" (images to simulate object detection). These images are recorded using a Photoneo PhoXi 3D Scanner M and are saved as black and white PNG files with 2064 x 1544 pixels.

PRM-Data – Using PRM images are recorded for each object using 18°-steps for object rotation (turntable) and 10°-steps for camera angle (rocker) resulting in 200 images for each object. The images are taken as texture images without additional lighting. They are saved to the data base extended

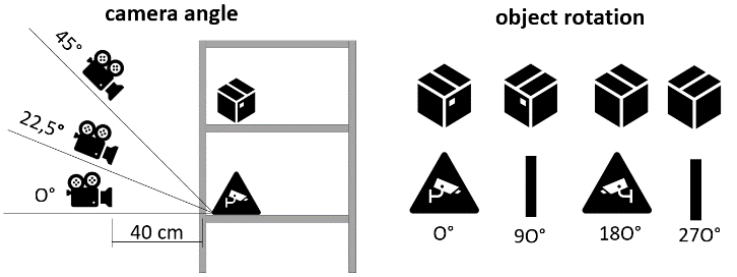


Figure 7: Setup for camera and objects for image recording of EC- and D-

by object information (ID, name, group, …) and the recording setup (camera angle, object rotation, …). Each image is annotated by YOLO Mark. PRM needs about 21 minutes to record a set of 200 images of one object and manual annotation of these images takes about 15 minutes.

EC-Data – These images are recorded from a demonstrating rack in a logistics laboratory using the same camera model mounted at PRM to simulate an operational shelf of a warehouse. Each object is recorded in 90°-steps for object rotation and with camera angles at 0°, 22.5° and 45° resulting in 12 images of each object. The specifications for camera angle and object rotation are approximations and are not adjusted exactly for image recording to simulate a realistic logistics environment in which objects are normally not stored to a specific view. The recording setup is shown in Figure 7.

Table 1: Data set categories and recording specifications

Data set	*Camera angel*	*Object rotation*	*No. of images*
PRM-Data	*Range: 0°-90°* *Step: 10°*	*Range: 0°-342°* *Step: 18°*	*2400*
EC-Data	*Range: 0°-45°* *Step: 22.5°*	*Range: 0°-270°* *Step: 90°*	*144*
D-Data	*22.5°*	*Range: 0°-330°* *Step: 30°*	*144*

D-Data − These images are recorded with the same setup as EC-Data but at a fix camera angle of 22.5°. This seems to be the most realistic angle of view considering shelfs of racks. Objects are rotated in 30°-steps leading to 12 images of each object. Similarity to EC-Data is desired. Table 1 gives an overview of applied camera angles and object rotations as well as the number of resulting images of the data sets. Figure 8 shows exemplary images from the different data sets for one object class.

Figure 8: Object "ceramic cup 2" in PRM-Data with different camera angles and 90°-step rotation (top), EC- and D-Data recorded with camera angle of 22,5° and 90°-step rotation (middle, bottom)

5.2 Training and Testing of Neural Network

This section describes training, testing, and evaluation of the neural network. During testing an image is given to a CNN. It analyzes the image by means of the 12 trained object classes resulting in several detections each with a probability of object detection greater than 25%. Figure 9 shows the different steps of training and testing as well as used data sets and resulting neural networks.

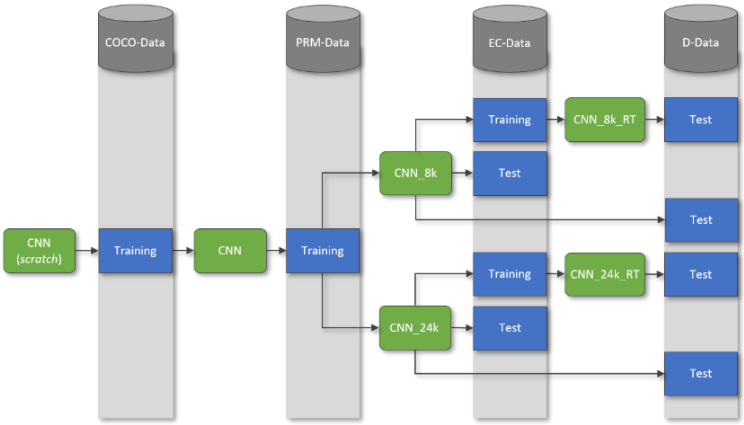


Figure 9: Training and testing of neural networks

Predictions of a neural network for an image are classified into True Positives (TP) (correct prediction: correct object class and location), False Positives (FP) (false prediction: false object or incorrect located) and False Negatives (no prediction but image contains searched object). A TP is a correct result leading to a successful pick whereas FP can provoke errors during operations, e.g. by damaging an object or by picking a wrong object. FN give a hint that a specific object is not trained in a satisfying manner and for further data must be collected by Emergency Calls.

5.2.1 Training with PRM Data

The initial training of the CNN with PRM-Data (2400 images) lasts for about 14.5 days. For this training, the data set is randomly divided into 75% training and 25% validation images. It is evaluated and steered by an average loss being calculated after an iteration each using a random subset of 64

images. Bochkovskiy (2020a) recommends the number of iterations to be calculated by 2000 times the number of object classes within the data set, i.e. training with PRM-Data terminates after 24000th iteration. Training with PRM data starts with an average loss greater than 1400, decreasing exponentially, and reaching an average loss of 1 at about 500 and 0.5 at about 1000 iterations. It gets steady at about 5000 iterations with an average loss of 0.05 and stops with 0.014 after 24000 iterations.

Table 2: Detections of CNN trained with PRM-Data (1k to 24k) tested against EC-Data

Camera angle	*Number of tests*	*Detections = 0 (negative)*		*Detections > 0 (positive)*		*TP*		*FP*	
0°	*1152*	*511*	*44.4%*	*641*	*55.6%*	320	36.0%	570	64.0%
22.5°	*1152*	*240*	*20.8%*	*912*	*79.2%*	682	58.6%	481	41.4%
45°	*1152*	*289*	*25.1%*	*863*	*74.9%*	612	57.9%	445	42.1%

A CNN is saved after each 1000th training iteration and is tested against EC-Data. The number of detections can be greater than the actual number of images as there can be more than one prediction for an image. However, only one prediction can be correct because an image of EC-Data contains only one object. The results of the analysis are shown in Table 2 and reveals the CNN work best on images recorded from a camera position of 22.5° showing higher rates of TP than angles of 0° and 45°. Furthermore, the rate of FN is almost the same. This accompanies a camera angle of 22.5° to be

used in operational processes enabling an unblocked view on a stored object in a shelf. The steeper this angle the less insight into the shelf.
To evaluate which stage of training should be used for object detection the CNN after each 1000th iteration is tested against EC-Data at 22.5°. The results are shown in Figure 10. After 8000 iterations the gap between TP and FP implies a reliable object detection. Because the levels of TP and FP are also better than after 24000 iterations, in addition to the recommended CNN after 24000 iterations (CNN_24k) the training stage after 8.000 iteration is also considered during the following evaluation (CNN_8k).

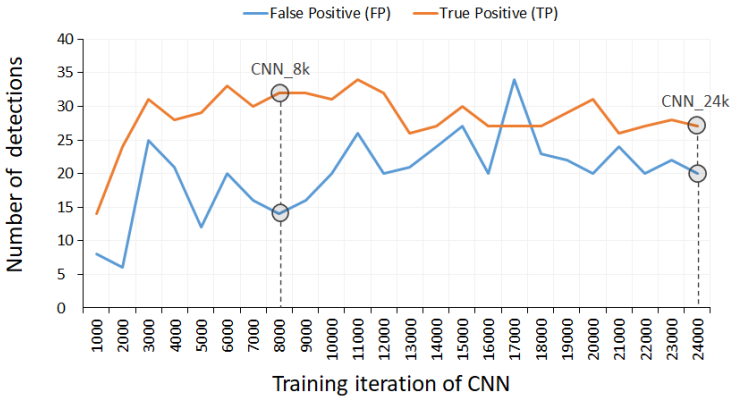


Figure 10: Test of training stages against EC-Data at a camera angle of 22.5° (48 images)

5.2.2 Retaining with EC Data

CNN_8k and CNN_24k are retrained with EC-Data to evaluate the impact of training an existing network with a few operational images initially trained

with laboratory data. Retraining starts with an average loss value smaller than 0.7 resulting from the initial training with PRM-Data. During retraining about 70 iterations are calculated per hour which is a similar performance as during initial training with PRM-Data. After 200 iterations the number of TP increases and the number of FP declines significantly.

Table 3: Test of CNN 8k, 24k, 8k_RT, and 24k_RT against D-Data

	CNN_8k		*CNN_8k_RT*		*CNN_24k*		*CNN_24k_RT*	
True Positive	*98*	*52.4%*	*138*	*92.0%*	*78*	*46.7%*	*137*	*89.5%*
False Positive	*73*	*39.0%*	*10*	*6.7%*	*75*	*44.9%*	*14*	*9.2%*
False Negative	*16*	*8.6%*	*2*	*1.3%*	*14*	*8.4%*	*2*	*1.3%*
∑	*187*	*100%*	*150*	*100%*	*167*	*100%*	*153*	*100%*

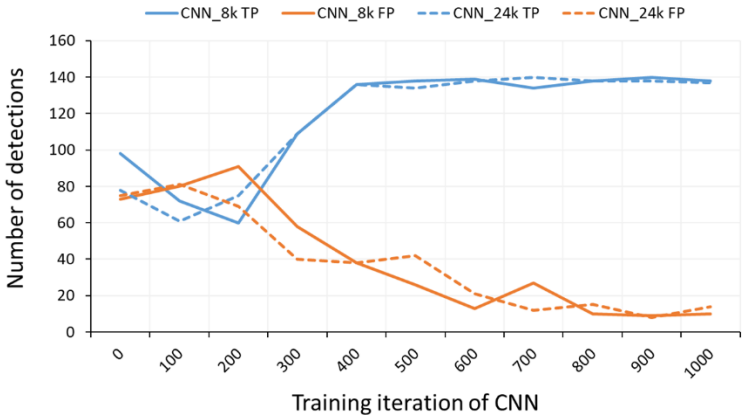


Figure 11: Test of CNN 8k, 24k, 8k_RT, and 24k_RT against D-Data

Because this trend converges at about 400 iterations only the first 1000 iterations are considered. The retraining for 1000 iterations lasts 14.3 hours for CNN_8k and 15.3 hours for CNN_24k. The resulting CNN "CNN_8k_RT" and "CNN_24k_RT" (RT = retrained) are each tested against D-Data. The results are shown in Figure 11. After 1000 iterations of retraining CNN_8k_RT and CNN_24_RT show similar behaviour. In Table 3 the results from testing all selected CNN against D-Data is shown. Comparing CNN_8k and CNN_24k reveals CNN_8k performing better by reaching a higher number of TP.

5.2.3 Evaluation of object classes

Further analysis evaluates single object classes. P_{OD} is computed by calculating an average probability for a true positive object detection for each object class. Some objects already show a P_{OD} greater than 90% after testing CNN_8k and CNN_24k against D-Data, but some do not get even one single positive detection (0%). By using CNN_8k_RT and CNN_24k_RT a great improvement can be reached for all classes except ceramic cup 3. The results for all objects classes are shown in Table 4.

Table 4: P_{OD} for all object classes resulting from testing against D-data

Object class	*CNN_8k*	*CNN_8k_RT*	*CNN_24k*	*CNN_24k_RT*
bucket	*23,5%*	*99,8%*	*6,1%*	*100,0%*
ceramic cup 1	*19,3%*	*97,3%*	*0%*	*99,7%*
ceramic cup 2	*0%*	*83,3%*	*0%*	*91,5%*
ceramic cup 3	*90,8%*	*70,8%*	*86,9%*	*60,1%*
ceramic cup 4	*87,2%*	*100,0%*	*89,3%*	*91,7%*
glass	*54,4%*	*99,9%*	*22,1%*	*99,9%*
plastic cup 1	*65,8%*	*99,7%*	*29,7%*	*100,0%*
plastic cup 2	*30,6%*	*99,8%*	*11,8%*	*100,0%*
key	*0%*	*86,4%*	*31,3%*	*93,4%*
charger	*73,3%*	*91,2%*	*80,9%*	*91,6%*
plastic bottle	*76,9%*	*99,4%*	*48,6%*	*99,0%*
glass bottle	*92,8%*	*100,0%*	*91,0%*	*99,8%*

Analyzing FP data reveals occurring misdetections between certain objects. Figure 12 shows a group of objects having a very similar appearance causing 9 of 12 misdetections of CNN_8k_RT and 9 of 14 for CNN_24k_RT. The images are taken from D-Data with an object rotation of 60° and 180°. The object "ceramic cup 1" has a slightly different form and a darker surface, therefore it is not affected by misdetections. These images show the challenge to distinguish between these object classes.

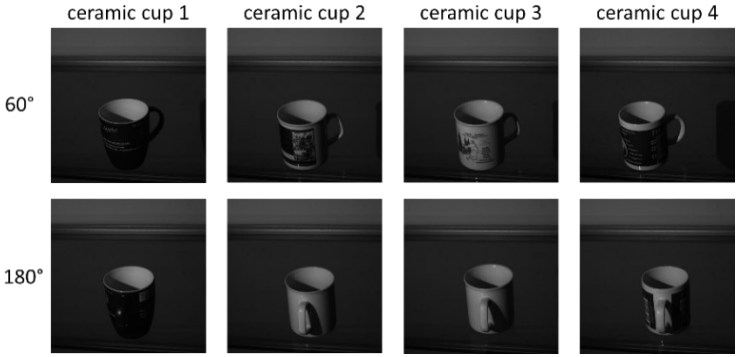


Figure 12: Similarity of ceramic cups with 60° and 180° (images enlarged)

5.3 System Evaluation

All 12 objects of the data set are assigned randomly to a shelf within the picking zone described in section 4.2 to evaluate the efficiency of the trained CNN. 720 predefined picking orders are generated by the order generator containing each one order line with a single object with quantity one uniformly distributed over all objects as a forecast for one hour. Picking orders of real systems may contain more than one order line but this configuration simplifies parameter setting and evaluation. However, multi order line orders can be interpreted as a sequence of several single order line orders. Orders are fulfilled by two human pickers and two picking robots using CNN_24K for object detection.

At first this scenario is evaluated by the capacity planning according to Rieder and Verbeet (2020) shown in chapter three. The calculation of $L_{EC,R}$ is extended by a duration for object detection t_4 during Adjustment-Phase. Used parameters and formulas are shown in Figure 13. The optimization (CPLEX OPL Studio 12.8.0.0) considers picking capacity, expected object detection performance by CNN_24k and the time effort caused by Emergency Calls. It shows that fulfilment of all picking orders of the forecast is possible. Capacities are almost used completely with C_H= 552 and C_R= 182.

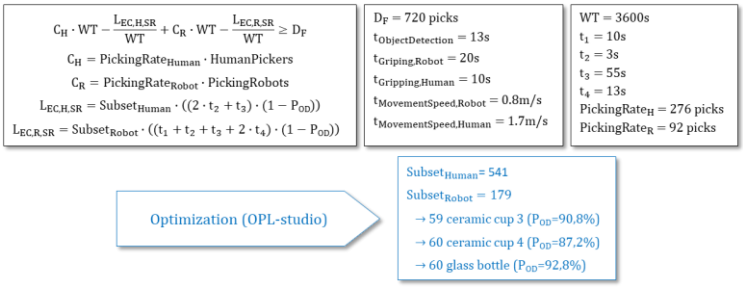


Figure 13: Formulas and parameters for system evaluation

As $Subset_{Robot}$ contains only objects with a high P_{OD} dynamic order assignment during simulation will not result in complete order fulfillment within one hour. System performance is evaluated by measuring the degree of order fulfilment as well as current workload of human pickers and picking robots.

The progress of order fulfilment during simulation is shown in Figure 14. Confirming expectations released orders cannot be fulfilled completely within one hour due to 55 triggered Emergency Calls leading to an order backlog of 176 orders. Using CNN_24k_RT in a second run almost every order can be completed successfully while only a single Emergency Call is triggered. The backlog of 7 orders after one hour results from the delay between order release and order processing.

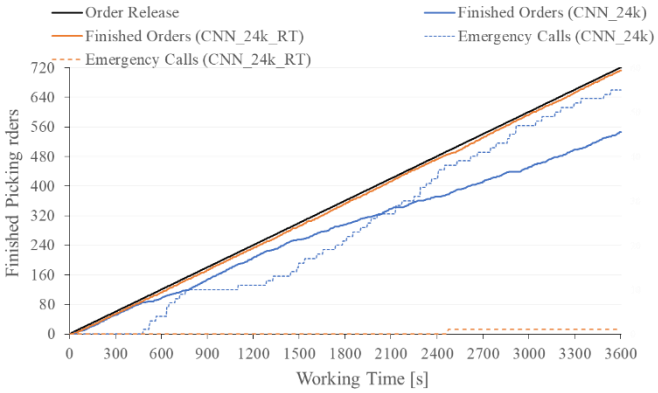


Figure 14: Results of simulation using CNN_24k and CNN_24_RT

6 Discussion

Contributions of this paper are the definition and creation of a data set containing data for training and testing CNN for object detection, the training and evaluation of these CNN to enable picking, and the implementation of an agent-based demonstrator to create empirical data to evaluate the concept of a feedback loop within a cooperative human-robot picking system. 12 objects of different material are selected to evaluate the cooperative picking system representing objects of a realistic picking zone with partially similar objects. For an initial training 2400 images are recorded, for retraining and testing each 144. Because no additional lighting is used during recording it results in quite dark images showing barely recognizable objects. The intention is to create realistic images showing the impact of a small data set during initialization of an automated system. Even if the number is quite small and some images are bad illuminated object detection provides good results. Images are saved as black and white data not containing as much information as RGB data, but the used 3D-scanner only provides texture information in a black and white mode. Furthermore, each image contains a single object to simplify image recording and evaluation. The operational images the trained CNN are tested against are recorded with a camera angle of 22.5°. Nevertheless, images recorded by PRM between 0° and 90° are also useful because the effective angle of operational object detection may match these angles due to positioning of the object, e.g. images from above (0°) become relevant if the object is on side. The presented results encourage to pursue the presented approach. Evaluating industrial applications might need to extend data set with images containing more objects from different classes as well as fallen or damaged objects enabling

development of error handling strategies for misclassification during object detection.

The authors of this work support the thesis images from a laboratory environment can be used to train a CNN for object detection in real picking environments. It is shown a well performing CNN can be trained with only a few iterations and a small data set. However, during training each interim stage must be analyzed to select the best performing CNN while ensuring the ability to adapt to new objects by retraining, which is a necessary ability in logistics environments with a dynamic object range. Furthermore, only a few images from a close to reality setting is necessary to adapt a CNN by retraining with laboratory generated images to significantly improve performance. This training approach seems reliable for logistics applications and should be evaluated by further research to confirm the promising results for object detection. The handling of false positive predictions provided by a CNN must also be considered in further developments by implementation another double check mechanism to avoid false friend picking.

Simulation of the picking system by the virtual demonstrator using the trained CNN reveals weaknesses within the basic mechanisms of order release and order assignment. A predictive capacity utilization is prevented by the evenly distributed order release. Therefore, performance of order picking is limited by released orders, which means a later drop in performance due to Emergency Calls is difficult or even impossible to make up for. This effect can be observed particularly during CNN_24k simulation. The previous calculation of a possible order distribution by optimization does not adequately consider system dynamics since the subsets are de-

fined pure in sorts using objects with a high P_{OD}. With additional time constraints the resilience of the subsets could be increased. An evenly distribution of picking orders between pickers by the auction mechanism is prevented by triggered Emergency Calls. The reason is the immediate distribution using a local order list within each agent. This leads to the effect human pickers are running out of picking orders while picking robots still must process several picking orders at the end of working time during simulation. Therefore, agents should either be given the option to take over picking orders from other agents or the local order lists must be restricted to ensure a balanced distribution. An alternative approach is a mechanism for a preselected order assignment described by Verbeet et al. (2019) to control order distribution considering critical objects with a low P_{OD}.

7 Conclusion

Automation of transport, object detection and gripping within picking processes are major challenges for robots. One approach is a cooperative human-robot picking system to ensure a fallback for order fulfillment and to support robots in a dynamic learning process by a feedback loop (Emergency Call). An agent-based demonstrator should examine the resilience of this approach. The proposed system is divided into an environment for controlled image recording and training of a Convolutional Neural Network (CNN) for object detection as well as a picking environment for order fulfillment and operational image recording. Currently the picking environment is implemented virtually as multiagent-system, but a connection to physical components is possible and planned. A data set consisting of laboratory images (PRM-Data) and operative images (EC-Data) for the training of the CNN is created. The CNN is trained with this data set, is tested against additional operational data (D-Data), and is evaluated by a simulation using the virtual picking environment. The results show that even with a few images gathered by a feedback loop and low training effort good results can be achieved in automated object detection. For an industrial application, however, further development is necessary.

The evaluation of trained CNN shows the challenge to find a best fitting configuration and training stage considering all objects. This problem may increase in a real setup considering thousands of objects. One possible approach is the training of many specialized CNN with a single object or a group of similar objects. Because the system knows which objects are stored at which shelf a specific CNN can be chosen for object detection. The results gained from these special CNN can also be marshaled and evaluated

by an additional mechanism to provide an even more reliable object detection. Thereby, the effort for initial training and retraining may be reduced while retaining the ability to adapt to a changing object range. The demonstrator is planned to be extended by a physical picking station realized by a UR5E robotic manipulator embedded into the multiagent-system, i.e. a new agent type will be implemented as digital representation to control the robot. Recording of images using PRM cannot be automated, but the training environment and the demonstrator can be linked to automate retraining of CNN. Both systems are already realized as multiagent-systems. During writing this paper version 4 of YOLO was published (Bochkovskiy, Wang and Liao, 2020). Assuming this new version improves performance of object detection the demonstrator should be updated and tested.

Acknowledgments

This work is part of the project “ZAFH Intralogistik”, funded by the European Regional Development Fund and the Ministry of Science, Research and Arts of Baden Württemberg, Germany (F.No. 32-7545.24-17/3/1).

References

Azadeh, K., Koster, M. B. M. de and Roy, D., 2017. Robotized Warehouse Systems: Developments and Research Opportunities. SSRN Electronic Journal. http://dx.doi.org/10.2139/ssrn.2977779.

Bochkovskiy, A., 2020a. darknet: Yolo-v4 and Yolo-v3/v2 for Windows and Linux. [online] Available at: <https://github.com/AlexeyAB/darknet>.

Bochkovskiy, A., 2020b. Yolo_mark: Windows & Linux GUI for marking bounded boxes of objects in images for training Yolo v3 and v2. [online] Available at: <https://github.com/AlexeyAB/Yolo_mark>.

Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y. M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. <http://arxiv.org/pdf/2004.10934v1>.

Bormann, R., Brito, B. F. de, Lindermayr, J., Omainska, M. and Patel, M., 2019. Towards Automated Order Picking Robots for Warehouses and Retail. In: D. Tzovaras, D. Giakoumis, M. Vincze, and A. Argyros, eds. 2019. Computer Vision Systems. Cham: Springer International Publishing, pp. 185–198.

Bratman, M., 1987. Intention, plans, and practical reason. Cambridge: Harvard University Press.

Braubach, L., Lamersdorf, W. and Pokahr, A., 2003. Implementing a BDI-Infrastructure for JADE Agents. EXP, (3), pp. 76–85.

Braubach, L., Pokahr, A. and Lamersdorf, W. Jadex: A short overview. In: Net. Object-Days, pp. 195–207.

Eclipse, 2020. MQTT Client Paho. [online] Available at: <https://www.eclipse.org/paho/> [Accessed 20 May 2020].

EHI Retail Institute, 2019. Robotics4Retail – Automatisierung und Robotisierung in Handelsprozessen. <https://www.robotics4retail.de/fileadmin/Robotics/EHI_Poster_Landkarte_Robo-tics4Retail_2019.pdf> [Accessed 19 November 2019].

FIPA, 2019. FIPA. [online] Available at: <http://www.fipa.org/index.html> [Accessed 5 June 2019].

Gerke, W., 2015. Technische Assistenzsysteme: Vom Industrieroboter zum Roboterassistenten. [e-book]. Berlin: De Gruyter Oldenbourg. <http://tib-hannover.eblib.com/patron/FullRecord.aspx?p=1563389>.

He, K., Girshick, R. and Dollár, P., 2018. Rethinking ImageNet Pre-training. <http://arxiv.org/pdf/1811.08883v1>.

Jansen, P., Broadhead, S., Rodrigues, R., Wright, D., Brey, P., Fox, A., Wang, N., King, O., Chatila, R. and Romano, V., 2018. State-of-the-art Review: WP4 AI & Robotics. <https://www.sienna-project.eu/digitalAssets/787/c_787382-l_1-k_sienna-d4.1-state-of-the-art-review--final-v.04-.pdf> [Accessed 19 May 2020].

Kamarul Bahrin, M. A., Othman, M. F., Nor Azli, N. H. and Talib, M. F., 2016. INDUSTRY 4.0: A REVIEW ON INDUSTRIAL AUTOMATION AND ROBOTIC. Jurnal Teknologi, [e-journal] 78(6-13), pp. 137–143. http://dx.doi.org/10.11113/jt.v78.9285.

Kugler, W. and Gehlich, D., 2013. Einsatz von Agentensystemen in der Intralogistik. In: P. Göhner, ed. 2013. Agentensysteme in der Automatisierungstechnik. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 113–128.

Lee, J. A., Chang, Y. S. and Choe, Y. H., 2018. Assessment and Comparison of Human-Robot Co-work Order Picking Systems Focused on Ergonomic Factors. In: P. Arezes, ed. 2018. Advances in Safety Management and Human Factors. Cham: Springer International Publishing, pp. 516–523.

Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L. and Dollár, P., 2014. Microsoft COCO: Common Objects in Context. <http://arxiv.org/pdf/1405.0312v3>.

Liu, Y.-R., Huang, M.-B. and Huang, H.-P., 2019 - 2019. Automated Grasp Planning and Path Planning for a Robot Hand-Arm System. In: 2019 IEEE/SICE International Symposium on System Integration (SII). 2019 IEEE/SICE International Symposium on System Integration (SII). Paris, France, 14.01.2019 - 16.01.2019: IEEE, pp. 92–97.

Martinez, C., Boca, R., Zhang, B., Chen, H. and Nidamarthi, S., 2015 - 2015. Automated bin picking system for randomly located industrial parts. In: 2015 IEEE International Conference on Technologies for Practical Robot Applications (TePRA). 2015 IEEE International Conference on Technologies for Practical robot Applications (TePRA). Woburn, MA, USA, 11.05.2015 - 12.05.2015: IEEE, pp. 1–6.

Mester, J. and Wahl, F., 2019. Robotik in der Intralogistik - Ein Projekt der Unternehmen Fiege und Magazino. In: I. Göpfert, ed. 2019. Logistik der Zukunft - Logistics for the Future. Wiesbaden: Springer Fachmedien Wiesbaden, pp. 199–211.

Müller, R., Franke, J., Henrich, D., Kuhlenkötter, B., Raatz, A. and Verl, A., 2019. Handbuch Mensch-Roboter-Kollaboration. München: Hanser.

Noren, A. and Müller, P., 2020. MQTT Broker Mosca. [online] Available at: <https://crycode.de/homepi-mqtt-broker-mosca> [Accessed 20 May 2020].

NVIDIA Corporation, 2020. CUDA-Zone: TRAIN MODELS FASTER. [online] Available at: <https://developer.nvidia.com/cuda-zone>.

Photoneo, 2020. Bin Picking Studio: “Alles in einem” Toolbox für Roboterintegratoren. [online]. Bratislava. Available at: <https://www.photoneo.com/de/bin-picking-studio-alles-in-einem-toolbox-fuer-roboterintegratoren/>.

Redmon, J., 2016. Darknet: Open Source Neural Networks in C. [online] Available at: <http://pjreddie.com/darknet/>.

Redmon, J. and Farhadi, A., 2018. YOLOv3: An Incremental Improvement. <http://arxiv.org/pdf/1804.02767v1>.

Rieder, M. and Verbeet, R., 2019. Robot-human-learning for robotic picking processes.

Rieder, M. and Verbeet, R., 2020. Evaluation and Control of a Collaborative Automated Picking System. In: 2020. Smart and Sustainable Supply Chain and Logistics – Trends, Challenges, Methods and Best Practices.

Salah, K., Chen, X., Neshatian, K. and Pretty, C., 2018 - 2018. A hybrid control multi-agent cooperative system for autonomous bin transport during apple harvest. In: 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA). Wuhan, 31.05.2018 - 02.06.2018: IEEE, pp. 644–649.

Shao, Q. and Hu, J., 2019. Combining RGB and Points to Predict Grasping Region for Robotic Bin-Picking. <http://arxiv.org/pdf/1904.07394v2>.

Thiel, M., Hinckeldeyn, J. and Kreutzfeldt, J., 2018. Deep-Learning-Verfahren zur 3D-Objekterkennung in der Logistik.

Valle, C. A. and Beasley, J. E., 2019. Order allocation, rack allocation and rack sequencing for pickers in a mobile rack environment. <http://arxiv.org/pdf/1903.06702v4>.

VDI, 1994. 3590. Kommissioniersysteme. Berlin: Beuth Verlag. <https://www.vdi.de/richtlinien/details/vdi-3590-blatt-1-kommissioniersysteme-grundlagen-1> [Accessed 19 May 2020].

Verbeet, R. and Baumgärtel, H., 2020. Implementierung von autonomen I4.0-Systemen mit BDI-Agenten. In: M. ten Hompel, B. Vogel-Heuser, and T. Bauernhansl, eds. 2020. Handbuch Industrie 4.0. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 1–36.

Verbeet, R., Rieder, M. and Kies, M., 2019. Realization of a Cooperative Human-Robot-Picking by a Learning Multi-Robot-System Using BDI-Agents. SSRN Electronic Journal. http://dx.doi.org/10.2139/ssrn.3502934.

Wahrmann, D., Hildebrandt, A.-C., Schuetz, C., Wittmann, R. and Rixen, D., 2019. An Autonomous and Flexible Robotic Framework for Logistics Applications. Journal of Intelligent & Robotic Systems, [e-journal] 93(3-4), pp. 419–431. http://dx.doi.org/10.1007/s10846-017-0746-8.

Wang, H., Chen, W. and Wang, J., 2019. Heterogeneous Multi-agent Routing Strategy for Robot-and-Picker-to-Good Order Fulfillment System. In: M. Strand, R. Dillmann, E. Menegatti, and S. Ghidoni, eds. 2019. Intelligent Autonomous Systems 15. Cham: Springer International Publishing, pp. 237–249.

Werner, T., Riedelbauch, D. and Henrich, D., 2017. Design and Evaluation of a Multi-Agent Software Architecture for Risk-Minimized Path Planning in Human-Robot Workcells. In: T. Schüppstuhl, J. Franke, and K. Tracht, eds. 2017. Tagungsband des 2. Kongresses Montage Handhabung Industrieroboter. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 103–112.

Zou, Y., Zhang, D. and Qi, M., 2019. Order Picking System Optimization Based on Picker-Robot Collaboration. In: Unknown. Proceedings of the 2019 5th International Conference on Industrial and Business Engineering - ICIBE 2019. the 2019 5th International Conference. Hong Kong, Hong Kong, 27.09.2019 - 29.09.2019. New York, New York, USA: ACM Press, pp. 1–6.