Reduction of annotation efforts for multiclass object detection by using a domain awareness data combination strategy

C. Briese 1, B. Hummel2, V. Hoang1, M. Schlüter1, J. Krüger2

1Fraunhofer IPK - Division of Automation Technology, Pascalstraße 8-9, 10687 Berlin, Germany, https://www.ipk.fraunhofer.de/logic
2Technische Universität Berlin - Division of Industrial Automation Technology, Straße des 17. Juni 135, 10623 Berlin, Germany, https://www.iat.tu-berlin.de/
E-mail: clemens.briese@ipk.fraunhofer.de

Abstract. To train convolutional neural networks (CNN) it is common practise to collect a huge amount of data. This is cost intensive and often not applicable. Up to date several studies have investigated the concept of few shoot learning, e.g. 1-3 samples per class. Suboptimal is still the over fitting resulting from the gap between training data and representative test data in the application. Since this is still a field of intensive research, an alternative and common approach is transfer learning with data- and image augmented pictures. However, collecting and labelling data for fine-tuning can still take an enormous amount of time, when it comes to multiclass pictures in industrial applications like assembly kit verification. The kits often contain stock lists with a small interclass and a high intraclass-distance. A specific characteristic of stock lists is that parts are easily adaptable and exchangeable. To bring object detection closer to the industry, we successfully show a dataset driven approach that combines a single class collection of pictures, which we call single class (SC) dataset and adapt with a few samples the specific multiclass use case. In result, we use a model trained on a huge SC dataset that can easily and fast be adapted to specific industrial use cases.

1. INTRODUCTION

In last couple of years Deep Learning made a steady progress. Today’s algorithms perform better than a human.[1][2] Despite the current high performance of neural networks (NN), the problem of the data basis still exists. Especially with regard to multiclass detection, the effort for labeling data is still immense. In order to achieve applicable detection accuracy, hundreds of images with the corresponding number of annotations are required. For the partly critical application in industry and industry-like contexts, the requirements on classification algorithms are much higher than in the consumer area. Accordingly, more data is needed to train algorithms. This problem is one of the key barriers for the use of machine learning, especially deep learning in many companies.

The completeness check of assembly kits is a promising application for the use of object detection algorithms. The goal is to detect and count the required components and sub-assembly parts in the run-up. A major challenge is the detection of very similar objects (e.g. different types of screws). If there is a high variance or change in the composition of these kits, image data must always be extracted and labeled again. This effort rarely justifies a possible conversion of the
technology.
This paper presents an extension of the transfer-learning approach by a step-by-step learning strategy. First the objects are trained and then the complete assembly kit is trained with a small amount of additional image data. This approach provides applicable results and can reduce effort significantly with a high variety of different or changing use cases. The approach is evaluated in an experimental study and analyzed with respect to potential reduction of annotation effort.

2. RELATED WORK
One goal of computer vision and deep learning is to deploy models in a production environment. Different papers show good results in industrial application for quality control task, e.g. surface damages [3].
In recent years, one of the main challenges has been the reducing the time needed to create a dataset and to train the model. Regarding the dataset, there are different approaches like using synthetic data or transfer learning with pretrained models. Pretraining is a widely used approach to learn a ‘universal’ feature representation using large-scale datasets, like ImageNet [4] COCO-Dataset [5] or Pascal VOC [6] [7]. This opinion was disproved, when Mahajan et. al. showed, that the improvements on object detection are relatively small. With a 3000 larger pretraining data set compared to ImageNet, the result was on the scale of +1.5 % average Precision on COCO [8].
Another area of research is the use of data augmentation for the artificial enlargement of data sets. [9]. According Zop et. al. an optimized data augmentation policy can improve the detection accuracy by more than +2.3 % mean average precision (mAP) on the COCO dataset [10].
The effect from huge datasets is limited and in spite of image and data augmentation, the acquisition of many images according to the use case remains necessary for pretraining. Different algorithms recommend typically hundreds or thousands of example representative images per class [11].
This is were a new approach in reducing dataset size comes in, called few shot learning (FSL), e.g. 1-3 samples per class. The idea is that a given set of classes with sufficient training data, ”is used to improve the performance on another set of classes with very few labelled examples“ [12].
While some papers demonstrated promising preliminary results [13], Barz et.al. [12] noted, even though the field makes progress, this is still a field of research because they do not work well. All these presented approaches are helpful and optimize the application of those models, but are rarely tested in the assembly industry. The industry has to deal often with slightly different, but fast changing, highly specialized use cases. Making detection algorithms applicable, the main effort is not computational side, but labelling those multiclass use cases.
Lim et. al. showed that borrowing and augmenting images with similar visual appearances to the target class can significantly improve accuracy especially, when only some examples for the class are available [14].
Additionally Chen et. al. noted that domain adaptation problem between different use cases can be reduced with specialized algorithms, whereby also more training data could possibly alleviate the impact of domain shift [15].
Learning that different classes can produce useful synergies and more data can alleviate the domain shift, collecting a dataset with all products of a stock list, as representative as possible, seems useful.

3. PROPOSED METHOD
An extended transfer-learning approach for object detection is developed. It consists of three training steps:
Diagram 1: Proposed process of training a multiclass detection algorithm

With the new intermediate step, the detection algorithm learns to extract and classify the features of possible objects of an assembly kit. For this, image data of the possible objects (single class images) must be generated and annotated once. Simultaneous labeling of several identical objects in an image is much faster than labeling different objects. The last step closes the resulting domain gap to multiclass detection using "small dataset learning". Due to the small amount of image data required for the respective application, the effort for annotation is reduced in the long-term. Since the annotation of single-class images is much simpler and faster compared to multiclass annotation, most of the effort is shifted to this area. It also can be easier supported by the use of data-driven algorithms.

As neural-network a YOLOv5-Framework is being used. It was firstly published in May 2020 by Jocher et al. [16] which is an improved PyTorch implementation of YOLOv3 by Redmon et al. [17] YOLOv5 offers four different models of which the largest one, YOLOv5x consisting of 9 million parameters and 284 layers, has been used.

4. EXPERIMENTS

As an experimental study, data of different complexities of assembly kits (MC data) and their individual parts (SC data) are recorded. With these data different strategies are applied to the training of a Yolov5x-NN. The images are acquired with the help of a static image setup with one consumer webcam and static lighting. The SC dataset consists out of 24 objects with 1000 labels each split with a training/validation from 2/1 ratio. The investigated complexities are:

(i) least complex: four different objects, two visually being similar
(ii) more complex: eight different objects, two times three visually being similar
(iii) most complex: twelve different objects, two times five being visually similar

Additionally, two training scenarios with different amounts of data are tested.

In the first training scenario each complexity-setup of the assembly training kit dataset consists of 50 images. The data is split up into 30 training-, 10 validation- and 10 test images. For the second training scenario only 20 images were used. Only the best approaches from the first scenario are used in the second.

Five strategies are compared, the new method presented in this paper (see chapter 3) and the possible variations (see tab. 1).

As evaluation metric two versions of mean average precision (mAP) is used. The mAP metric describes how many of the predictions correspond to reality, i.e. how accurate the predictions are. Mostly the value is considered at different IoU (intersection over unit), the value describes how well the object was localized (Eq: 1). One Metric is the mAP at a IoU-value of 50 % (mAP@.5) and the second is the mAP at is at different points of the IoU from 50% to 95% in 5% steps (mAP@.5:.95). So the second metric is the more stringent one.

\[
    mAP = \sum_{classes} \frac{True \ positives}{True \ positives + False \ positives} \cdot IoU = \frac{area \ overlap}{area \ of \ union} \tag{1}
\]

Google Colab was employed, with a Tesla T4 as GPU. For training the batch size was set to 16 and the epochs to 100. The training took about one hour.
Table 1: mAP@.5 and mAP@.5:.95 values for the different trials. SC-Data being the abbreviation for single class data

| Initial | Pre-Trained | Scratch |
|---------|-------------|---------|
|         | SC-Data     | None    | SC-Data |
|         | Kit         | No Kit  | Kit     |
| mAP@0.5| .5          | .5      | .5      |
| @ 50 samples | 0.983 | 0.812  | 0.671   |
| @ 20 samples | 0.938 | 0.742  | -       |
| MoreComplex |         |        |          |
| @ 50 samples | 0.954 | 0.682  | 0.497   |
| @ 20 samples | 0.875 | 0.613  | -       |
| MostComplex |         |        |          |
| @ 50 samples | 0.783 | 0.547  | 0.310   |
| @ 20 samples | 0.591 | 0.571  | -       |

5. EVALUATION
The test results of the Experiments are shown in Table 1. The first observation is that all experiments, with the new pretrained step on SC-Data, have reached the highest values for both mAP variants. In the highest complexity, the mAPs are around 80% for the simple metric variant and over 50% for the very stringent variant. The more complex the kit, the better the performance of the models trained on the SC-Data. The mAP values differ at highest complexity by almost the double. It is interesting that the experiments with a randomly initialized NN performed slightly better than the pretrained initial variant for the less complex kits and the simpler mAP-metric. This shows that the influence of a pretrained initial neural network is only really noticeable above a certain variance of objects.

The time needed to label an assembly kit image, averaged over all three complexities, is about 95 s. Correspondingly with 20 images ≈ 32 min. Added to this is the one-time effort for the single-class data. It took about four seconds per SC-Label with about 1000 labels per object and 24 objects, which makes ≈ 800 min ≈ 13.33 h initial effort. Assuming that without the SC-training step about 100 labelled images of the kit are needed, this results in an effort of ≈ 158 min ≈ 2.6h per kit. Even with the conservative assumption of the four seconds labeling-time per object in the SC dataset, the associated initial effort is amortized from a kit number of 7 (see Tab. 2). This can be greatly reduced if the annotation of the single-class dataset is supported by intelligent algorithms like Background-subtraction or (semi-)automatic labeling.

6. CONCLUSION
This paper has shown that adding a training step with single-class data can improve the performance of detection algorithms for the visual check of assembly kits. We achieved mAP
values twice as high compared to the training without the intermediate. With the approach presented here, the number of images required per assembly kit was reduced to about 20. This significantly reduces the effort for data collection. Despite a conservative estimate of the initial effort, the approach amortizes from about the sixth kit to be annotated.

REFERENCES
[1] John Markoff. A learning advance in artificial intelligence rivals human abilities. The New York Times, 2015.
[2] Licheng et. al. Jiao. A survey of deep learning-based object detection. IEEE Access 7, page 128837–128868, 2019.
[3] Qianwen et. al. Lv. Few-shot learning combine attention mechanism-based defect detection in bar surface. ISIJ International, 59:1089–1097, 06 2019.
[4] J. Deng and W. et. al. Imagenet: A large-scale hierarchical image database. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 248–255, 2009.
[5] Tsung-Yi Lin et. al. Microsoft coco: Common objects in context. 2015.
[6] M. et. al. Everingham. The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision, 111(1):98–136, jan 2015.
[7] Kaiming He et. al. Rethinking imagenet pre-training. CoRR, abs/1811.08883, 2018.
[8] Dhruv et. al. Mahajan. Exploring the limits of weakly supervised pretraining. In Proceedings of the European Conference on Computer Vision (ECCV), September 2018.
[9] Luis Perez et. al. The effectiveness of data augmentation in image classification using deep learning. CoRR, abs/1712.04621, 2017.
[10] Barret Zoph et. al. Learning data augmentation strategies for object detection. CoRR, abs/1906.11172, 2019.
[11] C. H. Lampert et. al. Learning to detect unseen object classes by between-class attribute transfer. In 2009 IEEE Conference on Computer Vision and Pattern Recognition, pages 951–958, 2009.
[12] Björn Barz et. al. Deep learning on small datasets without pre-training using cosine loss. CoRR, abs/1901.09054, 2019.
[13] Eli Schwartz et. al. Repmet: Representative-based metric learning for classification and one-shot object detection. CoRR, abs/1806.04728, 2018.
[14] Joseph Lim. Transfer learning by borrowing examples for multiclass object detection. 04 2013.
[15] Yuhua Chen et. al. Domain adaptive faster R-CNN for object detection in the wild. CoRR, abs/1803.03243, 2018.
[16] Glenn Jocher et. al. ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements. October 2020.
[17] Joseph Redmon et. al. Yolov3: An incremental improvement. CoRR, abs/1804.02767, 2018.