SemOI2 – Building Adaptive And Cost-Effective Recognition Applications With Semantic Augmentation

Achim Reiz[1], Birger Lantow[1] and Kurt Sandkuhl[1]

[1] *Rostock University, 18057 Rostock, Germany*

Abstract

Neural nets are the backbone for many innovative applications when it comes to making sense out of unstructured data. This technology, however, also has some significant downsides. Training these neural nets requires vast amounts of annotated training data and computational power and is, thus, expensive. Adding, altering, or removing concepts all trigger costly retraining of the neural net.

Semantic technologies can curb these systemic disadvantages. They can be connected to the neural net and augment the detection. The ontology can be easily adapted to new situations and use cases without requiring much computational power. This paper presents a prototype application for connecting the general-purpose OpenImages detection with a semantic. Free-to-use training datasets provide a cost-efficient way to use neural nets – often, there exist even pre-trained models for the large machine learning frameworks. The semantic contextualizes the low-level knowledge and enables the realization of specific use cases. We further propose a methodology to transfer the probability of the image detection and the prominence of items in the picture to the ground truth of the ontology by introducing a new measurement *Semantic Confidence* (*SC*).

Keywords

Ontology, CNN, OpenImages, Semantic Augmentation, Image Recognition

1. Introduction

Recent advantages brought tremendous progress in image classification, object detection, and computer vision systems in general. Through the rise in computational power, neural nets are today's state-of-the-art for image recognition tasks and are the backbone of many artificial intelligence (ai) applications.

However, convolutional neural nets (CNN) also have downsides: They need large quantities of annotated training data to produce valid results. Due to the extensive amount of required training data, creating these neural nets is costly in terms of computational and human resources. Every change in the detection requires a retraining of the neural net, which makes the approach inflexible regarding fast-changing requirements.

That being said, the use of neural nets and deep learning for recognition tasks are (today) without a realistic alternative. Nevertheless, a method for strengthening the reuse of neural nets and reducing the need to collect training data and training itself could significantly mitigate the disadvantages of this technology. The combination of semantic technologies with a general-purpose image detection (like OpenImages [1] or COCO [2]) can deliver just that. The general-purpose image detection contains a large set of freely available, annotated training data. Prominent machine learning (ML) libraries like TensorFlow or PyTorch often provide pre-trained models for these prominent datasets. The semantic then sets the detected objects into a new context, adapted to the individual needs of the user.

In A. Martin, K. Hinkelmann, H.-G. Fill, A. Gerber, D. Lenat, R. Stolle, F. van Harmelen (Eds.), Proceedings of the AAAI 2022 Spring Symposium on Machine Learning and Knowledge Engineering for Hybrid Intelligence (AAAI-MAKE 2022), Stanford University, Palo Alto, California, USA, March 21–23, 2022.

EMAIL: achim.reiz@uni-rostock.de (A. 1); birger.lantow@uni-rostock.de (A. 2); kurt.sandkuhl@uni-rostock.de (A. 3)
ORCID: 0000-0003-1446-9670 (A. 1); 0000-0003-0800-7939 (A. 2); 0000-0002-7431-8412 (A. 3)

In this paper, we present a semantic augmentation for the popular open images training set. We implemented a prototype that builds on TensorFlow and connects an ontology that generalizes the results of the image detection. Further, we introduce a measurement *Semantic Confidence (SC)* that transfers the fuzziness of the image detection and the prominence of items in a picture to the ground truth of the semantic. In a previous paper [3], a first iteration of the prototype already proved the general feasibility of the endeavor but had issues regarding a high number of false positives and a lack of performance. This new second iteration tackled a lot of the described problems by reworking the application completely, improving the augmentation results, the response times, and the *SC* calculation.

The rest of the paper is structured as follows: The following section gathers the current state-of-the-art in combining ontology and image recognition. Section three then presents the prototype by describing the architecture of the application and the calculation of the *SC* value, instructing how to use the software, and performs a preliminary evaluation before the research is concluded.

2. Related Work

The idea to connect deep learning recognition tasks with a semantic is not entirely new. Two recently published literature reviews by Ding et al. [4] and Bhandari and Kulikajevas [5] collected advances in the connection of image recognition and semantic. Ding et al.'s paper is structured in single and multiple object recognition tasks. For the former, they collected methods for improving detection accuracy. For the latter, they present papers that (A) use ontologies for the connection of detected low-level features to high-level semantics, using *partOf* relationships or Wordnet, (B) detect behavior in video analysis, and (C) classify environmental areas and maps. Bhandari and Kulikajevas collected works for (A) semantic-based image annotation and labeling label creation, to link the annotated images with each other and additional factual knowledge, (B) for segmentation tasks, to improve the understanding of detected features, and (C) to map detected objects to each other and to high-level concepts using *isa* and *part of* relationships. Further, the review presents selected applications for domain-specific tasks, e.g., applications in robotics, geoinformation systems, and retrieval tasks in sports events.

Besides these two literature reviews, we additional consider relevant the works by Reiz et al. [6] that presents a prototype for the fashion domain to infer sub-items based on the contextualized detected elements and image classifications. They argue for reduced implementation costs and the possibility of detecting objects beyond the accuracy of image detection, e.g., because fashion items often look similar but are worn in different contexts. In [7], Zambrano et al. developed a rule-based video analysis that extracted situations out of CCTV. The videos are analyzed frame by frame and annotated with the type of objects detected, their position and dimension, and their variation regarding the previous frames. The semantic now allows the quick and cost-efficient adaption of new detectable scenarios.

As the literature review shows, the idea of connecting a semantic to enrich the results of the image detection is not entirely new – various authors have already connected the different kinds of ai technologies. However, our approach is unique in the way that (1) we can eliminate the need for training the system entirely by building upon and adapting popular, freely available systems and (2) using a novel metric to connect the level of detection uncertainty and relative importance of an object in the picture to the semantic.

3. Prototype

The following section gives an overview of the developed prototype. At first, the software's architecture is presented, followed by an explanation of how to use the software. At last, we perform a preliminary evaluation of the new approach and derive the next steps.

3.1. Prototype Architecture

SemOI is available under an open source license. The source code is available online in Github[2]. The application is written in python and utilizes the web framework “Django”[3], running inside a docker[4] container. Taking the categorization framework for hybrid systems by [8], it is in its architecture similar to a system that learns an intermediate abstraction for reasoning, taking in model-free image data and transferring it into the categorical open images representation before another transformation is carried out by the semantic.

The detection is based on TensorFlow with pre-trained models from the model sharing website “TensorFlow Hub”[5]. The models are trained using the OpenImages training data set [1]. For the detection, the user has two different neural nets at their disposal: ResNet and MobilNet[6]. The former aims to generate a high accuracy; the latter shows possible integration scenarios for devices with limited computational power [9]. However, it is possible to easily swap the underlying recognition engine due to standardized, freely available models and training data.

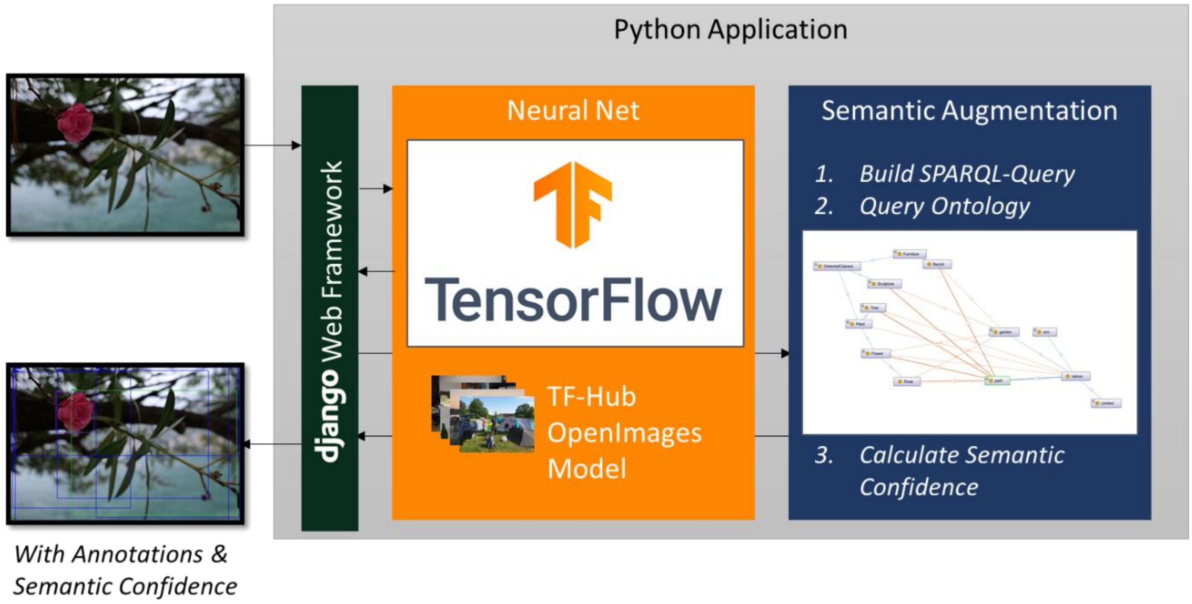


Figure 1: Software Architecture of Prototype

Figure 1 gives an overview on the architecture of the prototype. At first, the out-of-the-box tensorflow model detects the given objects. After a detection is completed, the application calls the semantic augmentation unit with a list of the detected items. Each object detector has an ontological twin in the semantic and can be identified using the detector label or OpenImages classifier ID. The detectors are connected with new, contextualized items (cf. Figure 2). In the semantic augmentation unit, the request is translated into a SPARQL-query that fetches results from the ontology, which returns the context items for further analysis.

After the detected contextualized items are fetched, the semantic confidence SC is calculated. While the SC value of the first SemOI version [3] used only the probability of a detected item and the number of occurrences in the semantic as the foundation for calculating SC, the calculation proposed by this paper also considers the prominence of an item. The prominence $\overrightarrow{oi_{po}}$ is calculated as a percentage value of the area covered by the object’s bounding boxes. The detector’s prominence and probability value (the detection score of the neural net) $\overrightarrow{oi_{pb}}$ are summed up, then multiplied with the inferred element. As the last step, the values of the context elements are aggregated.

[2] https://github.com/Uni-Rostock-Win/SemOI and https://doi.org/10.5281/zenodo.5005618
[3] www.djangoproject.com and www.django-rest-framework.org
[4] https://www.docker.com/
[5] www.tensorflow.org, tfhub.dev
[6] tfhub.dev/google/faster_rcnn/openimages_v4/inception_resnet_v2/1, tfhub.dev/google/openimages_v4/ssd/mobilenet_v2/1'

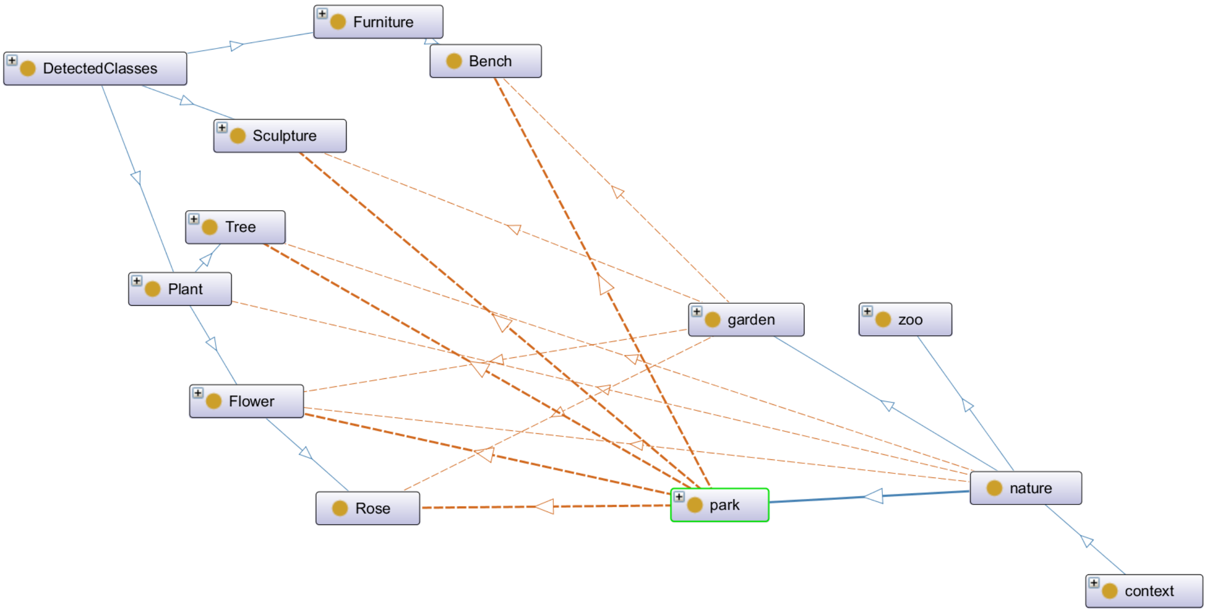


Figure 2: The Augmentation of OpenImages Using an Ontology

$$SC = \left(\overrightarrow{oi_{po}} + \overrightarrow{oi_{pb}}\right) * S = \vec{oi} * S = \begin{pmatrix} oi_1 \\ \vdots \\ oi_i \end{pmatrix} \begin{pmatrix} s_{11} & \cdots & s_{1k} \\ \vdots & \ddots & \vdots \\ s_{i1} & \cdots & s_{ik} \end{pmatrix} \tag{1}$$

Table 1 shows an example that fits the objects shown in Figure 2: A *Rose*, a *Plant*, and multiple instances of *Tree*s are detected with the probabilities $\overrightarrow{oi_{pb}}$ and prominences $\overrightarrow{oi_{po}}$. At first, $\overrightarrow{oi_{po}}$ and $\overrightarrow{oi_{pb}}$ for each of the detected items are aggregated to $\vec{oi}$, then multiplied with the connected context classes S (Step 1). The results are a list of weighted context items. As a next step, the values of the context items are aggregated (Step 2) and normalized to a percentage value that always considers the highest SC value as 100% (Step 3).

Table 1
Example for the Calculation of Semantic Confidence (Step 1)

Detected Item	Probability $\overrightarrow{oi_{pb}}$	Prominence $\overrightarrow{oi_{po}}$	Result $\vec{oi} * S$
Rose	0,936	0,0441	0,980 (park + garden)
Plant	0,126	0,588	0,714 nature
Tree	0,899	0,933	1,832 (nature + park)
Tree	0,733	0,615	1,348 (nature + park)
Tree	0,340	0,568	0,908 (nature + park)
Tree	0,148	0,46	0,608 (nature + park)
Tree	0,148	0,553	0,701 (nature + park)c

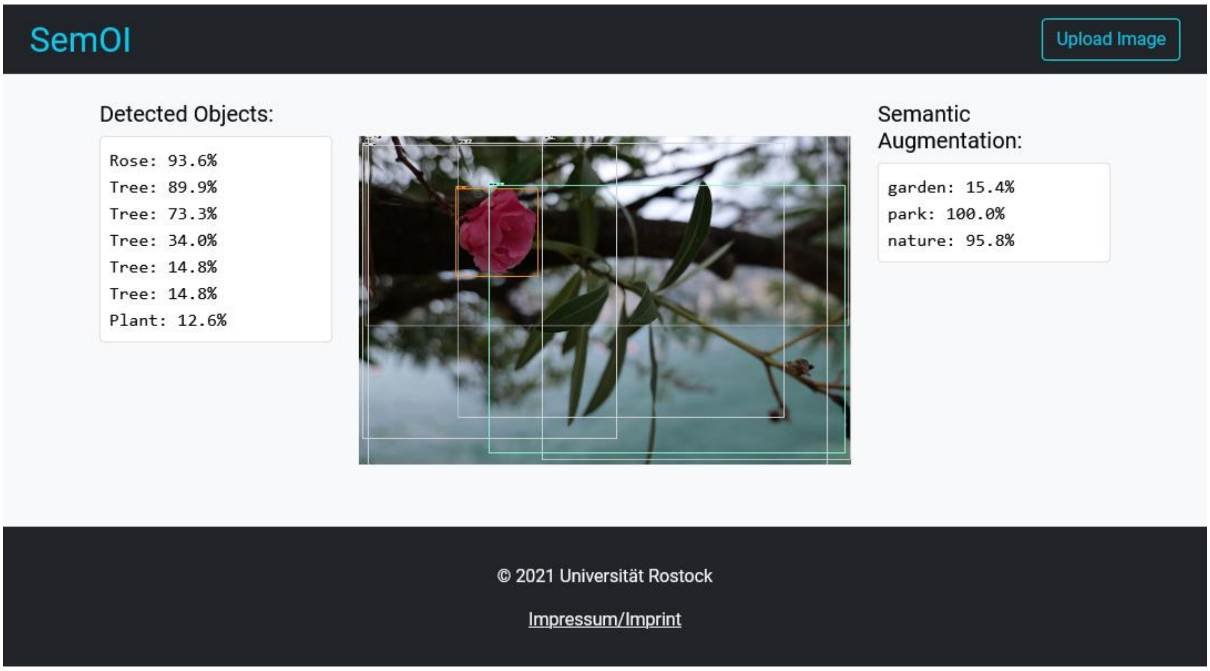
$$SC = 6{,}108 \text{ nature} + 6{,}374 \text{ park} + 0{,}9801 \text{ garden} \qquad \text{(Step 2)}$$

$$SC_{normalized} = 100\% \text{ park} + 95.8\ \% \text{ park} + 15.8\% \text{ garden} \qquad \text{(Step 3)}$$

3.3. Using the Prototype

The prototype is available online at semoi2.informatik.uni-rostock.de. At first, one needs to upload a picture. As soon as the upload is completed and the image is presented in the preview window, the application asks the user to choose the underlying model – “Faster” triggers the *MobileNet*, “Accurate” utilizes the *ResNet*. Even though on powerful machines (like the server that the application runs on), the calculation times are not expected to differ widely, this distinction allows the simulation of different application scenarios.

A click on “Analyze” triggers the recognition and augmentation tasks. After the analysis is completed, two boxes appear. The left box “Detected Objects” consists of the results from the image recognition; the box on the right “Semantic Augmentation” contains the augmentation results given by the semantic.



Detected Objects:

```
Rose: 93.6%
Tree: 89.9%
Tree: 73.3%
Tree: 34.0%
Tree: 14.8%
Tree: 14.8%
Plant: 12.6%
```

Semantic Augmentation:

```
garden: 15.4%
park: 100.0%
nature: 95.8%
```

© 2021 Universität Rostock

Impressum/Imprint

Figure 3: Analyzed Picture with Results. On the Left are the Results From the Image Recognition, on the Right are the Results From the Semantic Augmentation.

3.4. Evaluation

While the software is not yet adapted to a productive environment, the evaluation shall outline the current performance and identify possible weak points. For our analysis, we manually decided for every picture if it fits to the inferred classes and whether some of them are missing. That being said, the evaluation does not consider the results or efficacy of the image recognition that is the input for the semantic. A possible error in the recognition, thus, propagates to the semantic.

The ontology is currently not targeted at a specific business use case but generalizes the results and infers contexts. Without these use cases, a precise evaluation scenario is missing. Taking the example of Figure 3, the semantic inferred the situation *garden*, *nature*, and *park*. One could argue that additional elements in the ontology fit the picture as well, in this case, e.g., *holiday*. Thus, applying binary *relevant / not relevant* categories is, at times, fuzzy. In these arguable cases, we evaluated in favor of the application.

The values for our analysis are presented in the table below. In total, we analyzed ten pictures. The application reached a precision value of 73% on average of all figures (67% average of the detected items) and a recall of 88% (89%). The results of the evaluation are available online[7].

Table 2
Evaluation Results

Figure	Not Relevant	Relevant	Missing	Precision	Recall
1		*3*		*100%*	*100%*
2		*6*		*86%*	*100%*
3	*1*	*2*	*1*	*33%*	*67%*
4	*4*	*5*	*1*	*56%*	*83%*
5	*4*	*1*		*33%*	*100%*
6	*2*	*1*	*1*	*100%*	*50%*
7		*2*		*100%*	*100%*
8	*1*	*5*		*83%*	*100%*
9	*1*	*3*	*1*	*75%*	*75%*
10	*2*	*3*		*60%*	*100%*
Total	**15**	**31**	**4**	**73%** (67%)	**88%** (89%)

On the one hand, the values indicate that we need to work further on the mapping accuracy and more carefully design the ontology to prevent irrelevant items from being inferred. On the other hand, however, the significance of the evaluation is limited without a real-world application scenario. Such a use case with a specifically developed ontology would further validate the capability of the system.

4. Conclusion

Training neural nets for image recognition tasks is a tedious and costly task. While there are free-to-use models for general-purpose detection jobs available, these models will most likely not fit directly to a specific use case. The connection with a semantic, however, can provide this adaption.

In the paper, we presented a new version of SemOI, a prototype for the connection of OpenImages with a semantic augmentation. The driving idea behind this publication is the simplification of setting up AI applications using semantic adaptions to pre-trained neural nets. These quick adaptions are enabled through adjustments of the ontology. Possible changes are manifold: One could add or alter context items, change, delete and add relations between the contexts and the detected items, or switch the underlying recognition model to reuse another public accessible or private neural net. The semantic can be easily adapted to a specific use case, enabling the rapid development of new recognition applications. We believe that this connection of pre-trained models with ontologies has the potential to mitigate a lot of the downsides that come with the deployment of image recognition technologies.

The current ontology is crafted manually. We first built a script for translating the open images taxonomy into rdf, then connected the given detectors to context items. Future applications might also, depending on the given use case, consider the application automated ontology creation techniques based on machine learning or text retrieval algorithms.

However, the new methodology also has limitations. The target application needs to be concerned with at least part of the detectors/classifiers in the pre-defined model. The new approach does not work if no pre-trained data is available, e.g., in highly specific detections like material properties in a factory.

The presented prototype proves the general feasibility of the idea. Further research is concerned with implementing more sophisticated detection scenarios. We also plan to further work on the semantic confidence (*SC*) value, to incorporate a weighting of the relations, e.g., based on their specificity, utilizing the research of ontology matching algorithms. At last, implementing the software into a real-world application would arguably deliver the most learnings regarding the future challenges we need to tackle. Here, we invite the community to fork and adapt the prototype and share their experiences.

[7] https://doi.org/10.5281/zenodo.5837124

Acknowledgments

We thank the students Henrik Bongertmann, Carl Pommerencke, and Daniel Hahn for their support in realizing the Prototype.

References

[1] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, S. Kamali, M. Malloci, J. Pont-Tuset, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, K. Murphy, OpenImages: A public dataset for large-scale multi-label and multi-class image classification, Dataset available from https://storage.googleapis.com/openimages/web/index.html (2017).

[2] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C.L. Zitnick, P. Dollár, Microsoft COCO: Common Objects in Context, 2014.

[3] A. Reiz, K. Sandkuhl, B. Lantow, SemOI: A Semantical Augmentation for the Open Images Detection, in: Proceedings of the FRUCT'28, Moscow, 2021, pp. 610–613.

[4] Z. Ding, L. Yao, B. Liu, J. Wu, Review of the Application of Ontology in the Field of Image Object Recognition, in: Review of the Application of Ontology in the Field of Image Object Recognition, North Rockhampton, QLD, Australia, ACM Press, New York, New York, USA, 2019, pp. 142–146.

[5] S. Bhandari, A. Kulikajevas, Ontology based image recognition: A review, in: Proceedings of the International Conference on Information Technologies, Kaunas, Lithuania, 2018.

[6] A. Reiz, M. Albadawi, K. Sandkuhl, M. Vahl, D. Sidin, Towards More Robust Fashion Recognition by Combining of Deep-Learning-Based Detection with Semantic Reasoning, in: Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021), CEUR-WS, Stanford University, Palo Alto, California, USA, 2021.

[7] A. Zambrano, C. Toro, C. Sanín, E. Szczerbicki, M. Nieto, R. Sotaquira, Video Semantic Analysis Framework based on Run-time Production Rules - Towards Cognitive Vision, Journal of Universal Computer Science 21 (2015) 856–870. https://doi.org/10.3217/jucs-021-06-0856.

[8] F. van Harmelen, A. ten Teije, A Boxology of Design Patterns forHybrid Learningand Reasoning Systems, JWE 18 (2019) 97–124. https://doi.org/10.13052/jwe1540-9589.18133.

[9] X. Wu, D. Sahoo, S.C. Hoi, Recent advances in deep learning for object detection, Neurocomputing 396 (2020) 39–64. https://doi.org/10.1016/j.neucom.2020.01.085.