Pixel-level Class-Agnostic Object Detection using Texture Quantization

Gabriel R. Gonçalves, Jessica Sena, William Robson Schwartz
Smart Sense Lab, Universidade Federal de Minas Gerais
Belo Horizonte, 32500-00
Email: {gabrielrg,jessica,william}@dcc.ufmg.br

Carlos Antonio Caetano
AI/HDL Lab, Samsung R&D Insitute Brazil
Campinas, 13097-104
Email: carlos.ac@samsung.com

***Abstract*—Object detection is a widely explored topic within the computer vision research field mostly because it is necessary for almost every system containing some kind of visual scene understanding or interpretation. Significant advances throughout the last 40 years allowed us to evolve from early techniques based on template matching to modern deep detectors capable of detecting thousands of different classes of objects with reasonable performance. Nonetheless, as approaches kept improving, more challenging topics related to object detection have been proposed. Classic object detectors have to be trained with all classes that might be presented in the testing phase. However, this is a problem in real-world scenarios because it is impossible to know the whole domain of possible objects. Hence, a task has emerged called class-agnostic object detection that essentially detects objects without determining their classes. In this paper, we address this task using a convolutional network and texture graylevel quantization. Our results showed that our model could improve 2.1 percentage points (p.p.) from the best baseline on objects that were not annotated in the training phase.**

I. Introduction

In the past decade, the continuous advances in machine learning techniques led the computer vision research community to new challenging problems in the field. Modern GPUs allowed it to create deep networks composed of millions of parameters that can learn patterns at an ever-increasing level of complexity [1], [2]. For instance, researchers could achieve outstanding results in the biometrics field, introducing new high-level and low-intrusive forms of personal identification such as gait [3] and voice recognition [4].

Another example of a task that had witnessed numerous advances throughout the last years is object detection [5], [6]. These systems essentially consist of models trained to recognize patterns defining the target objects. Therefore, they need to learn these patterns from a previous training data set.

There are many potential applications in the real world for object detection, such as in surveillance systems, robotics vision, semantic segmentation, and other topics related to scene understanding. For these tasks, generally, state-of-the-art methods rely heavily on how good and diverse the training samples are. Thus, despite some efforts of data augmentation [7], [8] and/or generative approaches [9] to increase data variance, most techniques can only be as good as the data they were trained.

In traditional object detection approaches, models need to be fed by thousands of samples from each class to learn how to distinguish them. This poses a problem in real-world scenarios because it is nearly impossible to name each class of objects that might be present somewhere and even more challenging to get meaningful samples from each one. Therefore, a new task emerged in the past years called *class-agnostic object detection* [10], [11]. It consists, essentially, of recognizing different objects in a scene without the need to name each one. This is different from conventional object detection (from now on called class-aware) because all objects are treated as if they are from the same class.

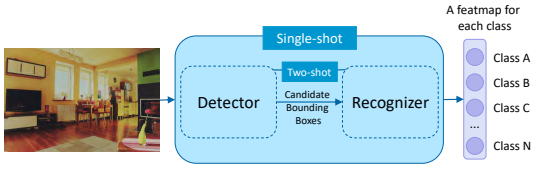

Fig. 1. Default architecture of a class-aware object detector.

Class-aware object detectors generally have a specific output to each known class used to rank these classes and assign the highest score as the correct one. Hence, to train a class-aware object detector using a dataset with C labeled classes, the model has to have an output of size C representing the score for each class. Figure 1 illustrates a simple class-aware detector pipeline. On the other hand, class-agnostic models are trained only to distinguish between objects and non-object. These models can be trained in either box-wise or a pixel-wise manner. In the former case, the model needs to generate candidate regions and decide afterward if those regions are real objects or are to be discarded [11]. In the second case, the model outputs a score for each pixel representing the probability of objectness [12].

In this paper, we propose a new pixel-wise class-agnostic detector. Our approach is a neural network composed of a convolutional backbone followed by a classification block. During the training phase, we allowed the model to train only in those positions where pixels were annotated, effectively ignoring those that did not have any object labeled. We also include combinations of two losses as target loss in a multi-task manner.

There are many challenges in training models for class-agnostic object detection. One of the main ones is that

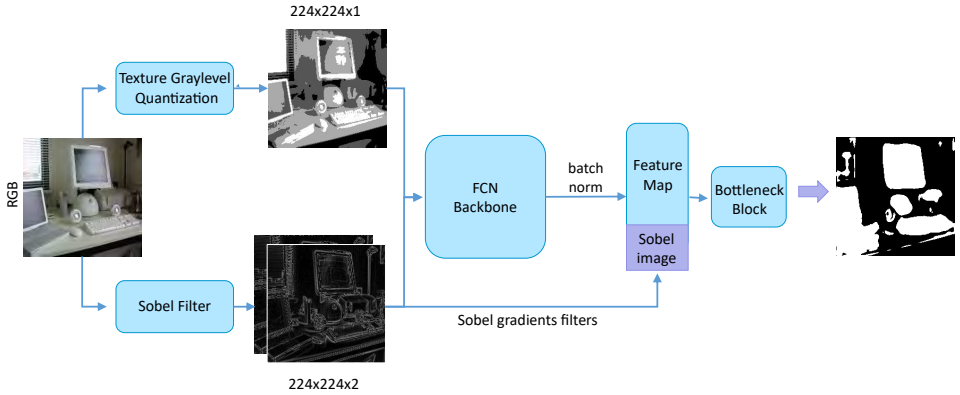

Fig. 2. Architecture of the proposed class-agnostic detector. Inputs are pre-processed before passed towards the resnet50 backbone. Final objectness pixel score is given by the last bottleneck block.

most datasets only annotate objects belonging to a labeled class [13], [14]. This does not exclude the possibility that other objects that do not belong to any labeled class might be present in the image. Therefore, we assume that these datasets have some objects annotated but not all.

We train our network in a fully convolutional manner where the network is fed with images and outputs an objectness map representing the probability that this pixel belongs to an object. Our results show that our approach could generalize very well to unlabeled objects, averaging a recall of 46% in object classes that were not annotated in the training phase.

The main contribution of this paper can be pointed out as a new end-to-end model to perform class-agnostic object detection. This model includes (i) a deep convolutional backbone, (ii) a training protocol, (iii) and a composite loss to supervise it.

II. Related Works

In this section, we detail the most important papers related to topics addressed in this work.

A. Class-aware Object Detection

Class-aware object detectors are the most common image-based detectors present in the literature. They concern those models that need to be trained with samples from each class they have to detect. Class-aware detectors refer an extensive range of different techniques varying from early approaches based on template-matching [15], going through high-level shape-based descriptors like Scale-Invariant Features Transform (SIFT) [16] and Histogram of Oriented Gradients (HOG) [17] to modern deep learning-based detectors like YOLO [18] and CenterNet [19].

In recent years years, object detection was highly influenced by the advent of deep neural networks since representation was one of the main challenges that detectors had to that date, and this is precisely where deep learning can be more helpful.

Region Based Convolutional Neural Networks (RCNN) [20] is one of the first deep learning-based object detectors. It is composed of three steps: (i) generate thousands of region proposals; (ii) scale them to a canonical size to use a regular CNN for feature extraction; (iii) apply an SVM to each vector to decide whether the correspondent region contains an object. Since the detection and classification tasks are done separately, this detector is called a two-stage detector. RCNN was further developed to more robust approaches such as *Faster-RCNN* [21] and *Mask-RCNN* [22]. The latter has the advantage of outputting masks at pixel level instead of rectangular bounding boxes.

One-stage approaches are known for performing detection and classification in a single network forward. Initially, Redmon et al. [18] proposed the *You Only Look Once* (YOLO) detector. Then, Liu et al. [23] designed the *Single Shot Detector* (SSD) that also divides the image into a grid. More recently, Law et al. [24] proposed a new detector called *CornerNet*. Their technique uses a CNN to predict top-left and bottom-right points in the image as was further expanded in Duan et al. [19] to also enabled center coordinates matching.

Class-aware object detectors achieve far better results than class-agnostic models when detecting classes they were trained for. It happens because class-agnostic models cannot focus on learning how these objects are or how the variety of perspectives they might appear. As opposed to the approaches described in the last section, our proposed detector does not need to see samples from each class it has to detect.

B. Class-agnostic Object Detection

Although one can find early works regarding class-agnostic object detection task [25], many approaches with promising results are relatively new and already take advantage of modern deep learning techniques [11], [26]. In contrast, it is also possible to find handcrafted approaches like the Binarized Normed Gradients (BING) [27], or part-based approaches. [28].

Maaz et al. [26] argue that correlating natural language interpretation with visual features helps the model to generalize novel concepts. They proposed an ambitious approach using multi-modal transformers to locate generic objects using *text-image* alignment semantics.

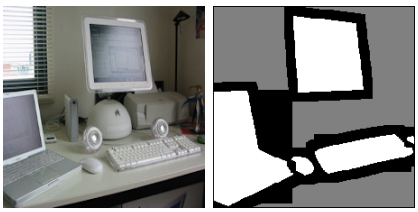

Fig. 3. RGB and heatmap from coco dataset. In the heatmap, the gray pixels represent the non-trained areas. Edge pixels are marked with black and object pixels with white color.

More recently, Jaiswal et al. [11] proposed a neural-based approach that outputs bounding box coordinates. Their significant contribution relies on a discriminator that blocks the network to learn class-specific features. Therefore, the network minimizes the discriminator and cannot learn how to distinguish between the classes. In one of our experiments, this detector was selected to compare our model with a literature baseline. Our model is design to output a feature map representing the objectness score for each pixel in the image. This idea was also explored other works [12], [29].

Finally, semantic segmentation is closely related task to class-agnostic detection [30]. It is more generic as it refers to the problem of assigning class labels to each pixel in the input. Thus, one can note that pixel-wise class-agnostic detectors are like semantic segmentation methods with two classes: object and non-object. Fully-convolutional neural networks (FCNN) [30] and DeepLab [31] models were proposed to address semantic segmentation using fully-convolutional architectures and were chosen in some steps of this work.

III. Proposed Approach

First, we detail the deep neural network used to perform feature extraction. Then, we address the training schema adopted to handle the specific problem of unlabeled class detection. Finally, we present a novel composite loss function used to train the network.

A. Feature Extraction

Deep Convolutional Neural Networks (CNNs) are well-suitable to train detectors because learning abstract representations is one of the main deep learning advantages. They extract features in images because the convolutional layers can correlate adjacent pixels sequentially. Therefore, the feature extraction part of our network is performed using a CNN backbone called Region-Based Fully Convolutional Network [32] based on Resnet50 [33]. As mentioned earlier, it was originally proposed to semantic segmentation tasks.

One challenge to train class-agnostic CNNs is to regularize it to not learning class-specific features. Therefore, to increase generalization, we pre-process the input images to forward only relevant information in the deep network. Instead of entering a normalized RGB image, our input image also contains three channels, but instead of one for each color (R, G, and B), there are two shape-related and one texture-related channel.

The shape-related channels are necessary to represent the edges of image as most objects are recognizable by their forms. To corroborate this hypothesis, Cheng et al. [27] also proposed a straightforward descriptor to class-agnostic models based on shape-based features. Therefore, we employ two directions of the Sobel filter as our image's two first input channels. Moreover, the third channel represents texture information of the image because this can also be relevant to estimate pixel objectness. However, the texture is also very relevant to describe intra-class features and is likely to overfit the training classes [34]. Thus, we decided to quantize the full grayscale image to an image with only few bits and use it as a texture descriptor. Since they have very little information compared to the standard image, we hypothesize it is a better input to represent the difference of texture between object and non-object pixels. The whole method workflow is illustrated in Figure 2.

B. Model Training

Class-aware object researchers are well served of datasets. The Pascal-VOC [35] and ImageNet [36] were proposed years before the first deep learning methods arise and are still used nowadays. Years later, *Microsoft* proposed the COCO dataset [13] with thousands of images from dozens of different classes and pixel-level annotation. Recently, an even more extensive set called Open Images [14] also brought attention from researchers as it is a collaborative ever-increasing set of images.

Despite this increase in size in the past years, these datasets were proposed to train class-aware detectors. As a result, they only annotate object instances from the classes labeled in the dataset. However, we cannot simply assume everything that is not annotated as non-object to train class-agnostic models. Because of that, we choose to train our model only allowing it to learn what is an object and what are object edges. We implement this by calculating the loss function only when the pixel is annotated as 1 (is an object-pixel) or 0 (is an object-edge-pixel). Training the network only in these pixels forces the model to effectively focus on what defines an object. Moreover, since the input images have minimal representation of texture, it cannot learn great class-aware features.

To ease the comprehension, Figure 3 illustrates the ground truth of an image from the COCO dataset used to train the network. Note that pixels from the object edges are black and object pixels have white color because they are an object pixel. The gray areas are the ones that do not have any object annotated and are ignored in network training. We also used multiple types of data augmentation such as rotation, small translations, gaussian filters, zoom in/out and others.

C. Composite Loss Function

One of the main problems to learn the heatmap of Figure 3 is the imbalance between 0 (object-edge pixel) and 1 (object pixel). Address this problem is essential because if we ignore it, the network would overfit most pixels to 1 as there are a large unbalance between pixel within objects and on its edges.

Therefore, we decided to employ the loss proposed by [37], known as *Focal Loss* proposed exactly to handle the problem of unbalanced clases. he loss function is designed to weight up and down hard and easy samples, respectively.

We would have a perfect output with 1 in all pixels objects and 0 otherwise, in the best-case scenario. Unfortunately, this is not what happens in practice as there are many pixels with borderline scores such as 0.5. To mitigate this problem, we employed another loss to our network called *Dice Loss*. It is an adaptation of the Sørensen-Dice coefficient [38] used to measure similarity between two arrays of samples. Hence, when all pixels are equal, it yields 0 and when they are all different, the function is 1.

We evaluate three methods to combine Focal and Dice losses and create a single composite loss. The first is a naive technique that only sums up both values. The problem of this technique is that if one loss is naturally larger, it might overwhelm others contributions. A solution is to include parameters in the network to represent the losses weights. They become trainable parameters and are optimized jointly with the network parameters. We evaluate two weighted composite losses: a combination proposed by [39] and a weighted linear combination of both losses. Finally, in testing phase, we employ a straightforward connected component labeling technique (CCL) in the heatmap and consider all component to extract bounding boxes [40].

IV. Experiment Evaluation

First, we present the evaluation protocol used in our experiments. Then, we evaluate how to represent the texture information better. Thirdly, we present an ablation study of our proposed approach to demonstrate the real improvement of each contribution. Then we present and experiment to show which loss is the better one for our class-agnostic problem. For this purpose, we evaluate which loss yields the model with best detection rate in objects that were not annotated in the training phase. Therefore, we split the 80 classes of COCO-dataset into 5 folds of 16 classes and perform 5-fold cross-validation. We chose to evaluate only using 5 folds because of the considerable time necessary to train each model. Finally, we show the efficacy of our model against another approach in literature in the *Open Images* dataset. We chose the work by [11] as baseline due to the similarity in the evaluation protocol.

A. Evaluation Protocol

Our resulting end-to-end model contains around $32M$ learnable parameters. Although this is not many if compared with other methods in literature [41], it still needs a considerable amount of computational power to learn the weights. In addition, as a result of our protocol to train only on pixels where there are objects annotated, the convergence becomes even more slower so we decided to use a large batch size and multiple GPUs to improve the variance within batches.

All learning parameters were calibrated empirically and are presented in Table I. Moreover, the models averaged 55 minutes to finalize each epoch, which means that all epochs averaged 25 hours to complete.

TABLE I
Parameters used to train the proposed detector.

Parameter	Value
# of epochs	30
learning rate (lr)	$1e-5$
lr decay	$2e-1$
lr decay step	9
optimizer	*adam*
focal loss alpha	0.25
focal loss gamma	2

Many available datasets work at a bounding box level [35], [36], so they only need four coordinates per object as an annotation. Therefore, we selected a dataset with pixel-level object annotation to train our model, *Common Objects in Context (COCO)*[1]. This is a large-scale dataset created to be employed in a variety of tasks such as object detection, semantic segmentation, and image captioning [13]. *COCO* contains more than $200,000$ images from 80 different object categories (treated as one by our class-agnostic approach). Moreover, the dataset totalizes more than 1.5 million object instances.

We also used the *Open-Images*[2], a large-scale crowd-sourced dataset with around 9 million images, 16 million bounding boxes from 600 different object classes in our experiments [14].

All approaches were evaluated using the same metric already employed in other works in literature [11], [42], average recall (AR). The average recall is defined as the average of true positive boxes divided by the total number of annotated boxes.

B. Texture Graylevel Quantization

We perform tests to evaluate the number of bits that are needed to describe an object with minimal bias. For this purpose, we vary the number of bits to represent the texture and evaluate how the model behaves in the *Open Images* dataset. Figure 4 shows a graph of the average recall varying the number of bits to represent the texture. According to the figure, the best results were achieved using 4 bits. Therefore, we use the same value in the following experiments.

C. Ablation Study

This section describes an ablation study of the minor adjustments we proposed in the training phase. We selected the best proposed model according to the results from this section and then removed traits one by one. The complete model comprises Sobel+texture image, full convolutional backbone followed by the bottleneck block (w/block), Sobel image concatenation, and the composite loss. Therefore, we evaluate the impact of the removal of these characteristics.

[1]https://www.cocodataset.org/

[2]https://opensource.google/projects/open-images-dataset

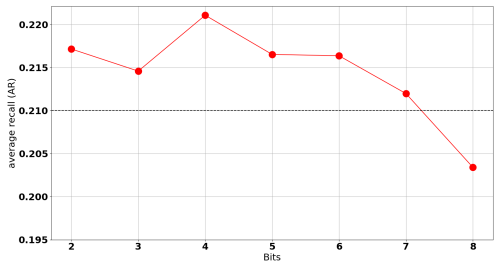

Fig. 4. Average recall as a function of the # of bits to represent the image.

TABLE II
ABLATION STUDY OF THE PROPOSED APPROACH. NOTE THAT THE BEST RESULT WAS ACHIEVED WHEN ALL CHARACTERISTICS WERE EMPLOYED.

approach	average recall (AR) open-images (unseen)
with RGB image	19.7%
w/o block	19.1%
w/o sobel	21.0%
w/o graylevel quant.	20.3%
with all	22.1%

According to results shown in Table II, the detection rate drops by 2.4 p. p. when using RGB input. This was how we first approach the problem and, therefore, when we realize we need to handle this problem of texture regularization. The block placed after the backbone is also essential as its removal degrades the result by 2 p.p. Finally, we can see that all characteristics proposed in Section III are helpful and valid to achieve better results.

D. Cross-Validation in COCO Dataset

In the third experiment, we intend to select the best loss version to optimize our model when applied to the class-agnostic problem. Hence, we perform 5-fold cross-validation in the COCO dataset and train the models with five different losses.

The first two losses are the ones described in Section III, *Focal* and *Dice* applied standalone. The second two are the composite losses described in the same section. One is the combination proposed by [39] and the other is a weighted linear combination of both losses using weights activated by softmax. The last is the simple sum of two losses, which researchers more commonly use.

We calculate the AR in seen and unseen classes representing annotated classes and not annotated in the training phase, respectively. The harmonic mean between the two scores was also reported.

As one can see in Table III, only the combination proposed by Kendal et al. [39] was not able to improve the results when compared to single ones. On the other hand, the softmax composite loss was the best one, improving the results by 0.6, 2.1, and 0.5 percentage points if looking to seen, unseen classes, and harmonic mean, when compared to second-best loss (dice one). Moreover, between the two single ones, dice loss was able to achieve slightly better results. The single losses as well as the one approach to weight losses were not able to improve the results.

TABLE III
PROPOSED LOSSES TO TRAIN OUR CLASS-AGNOSTIC DETECTOR. RESULTS ARE REPORT IN AVERAGE BETWEEN FOLDS AND THE STANDARD DEVIATION BETWEEN PARENTHESIS. COMPOSITE LOSSES ARE DENOTED BY *.

Loss	average recall (AR) seen	average recall (AR) unseen	harmonic mean
focal loss	32.6% (2.4)	44.1% (4.0)	37.4% (1.6)
dice loss	35.0% (1.9)	44.5% (4.2)	39.7% (1.8)
Kendal et al. [39]*	32.9% (2.2)	43.2% (4.8)	37.7% (2.4)
softmax loss*	35.6% (4.3)	46.5% (4.3)	40.2% (3.7)
sum of losses*	30.8% (3.5)	44.3% (3.9)	36.2% (2.8)

TABLE IV
COMPARED BETWEEN THE JAISWAL APPROACH AND OURS IN NON-OVERLAPPING CLASSES OF *Open Images*.

approach	average recall (AR)
Jaiswal et al. [11] with FRCNN	19.2%
Jaiswal et al. [11] with SSD	21.0%
proposed approach	22.1%

A paired p-test was employed to confirm that these improvements were statistically significant. We compared the composite softmax approach against dice loss as they were the best two results. We obtained two *p-values* meaning comparison in terms of unseen and seen classes. Values were 0.346 and 0.035, which means that, unfortunately, they were statistically identical when regards to seen classes. Nonetheless, as the p-value of unseen classes was inferior to the threshold of 0.05, we can reject the null hypothesis and consider that softmax loss is better than the single dice, for that matter.

According to the results, we can conclude that a composite loss is a good choice to approach the problem and, therefore, is employed in other experiments of this section.

E. Baseline Comparison in Open Images

In this experiment, we compared our best model against a baseline in the literature. We choose the approach proposed by Jaiswal et al. [11] with both backbones (FRCNN and SSD). To have a standard evaluation environment, we evaluate our model in non-overlapping classes of *Open Images* dataset. It means that we remove all classes that were present in the *Coco* dataset.

According to Table IV, we outperform the best baseline in 1.1 percentage point. Although the results might seem low at first look, this is very hard task as the *Open-Images* dataset is very challenging. Finally, we believe our approach was able to achieve a higher results due the improvement we employed in training phase. Specifically, removing texture information allowed the model to determine what defines an object better than our baseline.

V. Conclusions

In this work, we proposed a new class-agnostic object detector composed by a convolutional backbone followed by a bottleneck block that outputs a binary objectness map. We then employ *Otsu* binarization and a connected component labeling technique to collect object bounding boxes. We proposed a training schema that only trains what an object and an object edges are. Moreover, the model was trained with a composition of two losses, focal and dice loss trained multi-taskly.

Our experiments show that our model was able to detect an average of 46.3% of classes that were not trained to detect using the *COCO* dataset. It also outperforms a class-agnostic baseline in the literature in an experiment using *Open Images* dataset by 1.1 percentage points. We believe class-agnostic object detection is currently a hot topic within the computer vision community, therefore, significant advances are expected to come in next years.

More diverse and well-annotated datasets can also improve the robustness of our training process, therefore, we intend to see the feasibility of employing the *Open Images* dataset in our training phase, which is challenging due to the absence of many pixel-level annotated objects.

VI. Acknowledgment

The authors would like to thank the National Council for Scientific and Technological Development – CNPq (Grant 309953/2019-7) and the Minas Gerais Research Foundation – FAPEMIG (Grant PPM-00540-17). This work was supported in part by the Coordination for the Improvement of Higher Education Personnel (CAPES) (*Programa de Cooperação Acadêmica em Segurança Pública e Ciências Forenses # 88881.516265/2020-01*).

References

[1] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, 2017.

[2] Z. Wang and X. Wang, "A deep stochastic weight assignment network and its application to chess playing," *Journal of Parallel and Distributed Computing*, 2018.

[3] A. Sepas-Moghaddam and A. Etemad, "Deep gait recognition: A survey," *arXiv:2102.09546*, 2021.

[4] S. Duraibi, "Voice biometric identity authentication model for iot devices," *IJSPTM*, 2020.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *PAMI*, 2016.

[6] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *CVPR*, 2020.

[7] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," in *AAAI*, 2020.

[8] J. Wang, L. Perez *et al.*, "The effectiveness of data augmentation in image classification using deep learning," *Journal of Convolutional Neural Networks Visual Recognition*, 2017.

[9] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv:1406.2661*, 2014.

[10] S. Jiang, S. Liang, C. Chen, Y. Zhu, and X. Li, "Class agnostic image common object detection," *IEEE TIP*, 2019.

[11] A. Jaiswal, Y. Wu, P. Natarajan, and P. Natarajan, "Class-agnostic object detection," in *WACV*, 2021.

[12] K. J Joseph, R. Chunilal Patel, A. Srivastava, U. Gupta, and V. N Balasubramanian, "Mason: A model agnostic objectness framework," in *ECCV*, 2018.

[13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[14] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov *et al.*, "The open images dataset v4," *IJCV*, 2020.

[15] Y. Yakimovsky, "Boundary and object detection in real world images," *JACM*, 1976.

[16] D. G. Lowe, "Object recognition from local scale-invariant features," in *ICCV*, 1999.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016.

[19] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *ICCV*, 2019.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *CVPR*, 2014.

[21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *arXiv:1506.01497*, 2015.

[22] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[23] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *ECCV*, 2016.

[24] H. Law and J. Deng, "Cornernet: Detecting objects as paired keypoints," in *ECCV*, 2018.

[25] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on Computers*, 1973.

[26] M. Maaz, H. Rasheed, S. Khan, F. S. Khan, R. M. Anwer, and M.-H. Yang, "Class-agnostic object detection with multi-modal transformer," *arXiv:2111.11430*, 2021.

[27] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *CVPR*, 2014.

[28] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *PAMI*, 2009.

[29] B. Xiong, S. D. Jain, and K. Grauman, "Pixel objectness: learning to segment generic objects automatically in images and videos," *PAMI*, 2018.

[30] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[31] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *PAMI*, 2017.

[32] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," *arXiv:1605.06409*, 2016.

[33] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures," *arXiv:1603.08029*, 2016.

[34] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," *arXiv:1811.12231*, 2018.

[35] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, 2010.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009.

[37] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *ICCV*, 2017.

[38] T. J. Sørensen, *A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons*. Munksgaard Copenhagen, 1948.

[39] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *CVPR*, 2018.

[40] M. B. Dillencourt, H. Samet, and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representations," *JACM*, 1992.

[41] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.

[42] C. L. Zitnick and P. Dollar, "Edge boxes: Locating object proposals from edges," in *ECCV*, 2014.