СЕКЦІЯ 2
ІНТЕЛЕКТУАЛЬНІ КОМП'ЮТЕРНІ СИСТЕМИ

UDC 004.932 : 004.896

ONE-STAGE OBJECT DETECTION MODELS OVERVIEW

Avramenko S.E., Ph.D. student, Avramenko.St.Y@nmu.one,
Dnipro University of Technology
Zheldak T.A., Ph.D., Associate Professor, zheldak.t.a@nmu.one,
Dnipro University of Technology

Object detection stands as a cornerstone in the realm of computer vision, presenting both profound challenges and immense potential. Beyond mere image classification, it demands the precise localization of objects within an image, a task that holds significant promise and difficulty. Today, its significance resonates deeply with the advancements in the Internet of Things (IoT), finding widespread application in domains such as video surveillance and intelligent transportation. Object detection algorithms can be broadly categorized into two-stage and one-stage approaches, each with its unique methodology and trade-offs. While the two-stage algorithms, typified by R-CNN, excel in accuracy by employing candidate region extraction followed by classification and regression, they often suffer from prolonged processing times. Conversely, the one-stage algorithms, epitomized by YOLOv1 [1] , directly locate and classify objects, offering speed at the expense of some accuracy. As the field evolves, with notable strides made in the YOLO series, this paper delves into the development journey of one-stage object detection algorithms, providing a comprehensive analysis of their module structures and the innovations that have propelled their advancement.

YOLOv1 [1], introduced in 2016 at CVPR, represents a significant milestone as the first one-stage object detection algorithm achieving notable balance between accuracy and speed. Built upon the architecture of GoogLeNet, YOLOv1 innovates by replacing the inception layer with 1x1 or 3x3 convolution operations. Its core concept revolves around treating object detection as a regression problem. The algorithm's simplicity lies in dividing the image into an $s \times s$ grid, where each grid cell predicts the presence of an object whose central point falls within it. Additionally, bounding boxes are predicted, each containing coordinates (x, y, w, h), confidence scores, and category information. Consequently, each bounding box prediction entails $(4 + 1 +$

N) dimensions. YOLOv1's output dimensionality is $(s \times s, b \times (4 + 1 + N))$, with b determining the number of predicted boxes per grid cell. Despite its efficiency in speed, YOLOv1 lacks candidate region extraction, contributing to its rapid detection pace. However, this simplicity poses challenges in detecting multiple objects within a single grid cell and exhibits shortcomings in detecting smaller objects due to differences in loss function processing. The loss function (1) of YOLOv1 consists of three components: confidence loss, class loss, and object loss, computed using the sum-squared error.

$$\begin{gathered}
\lambda_{\text{cord}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} 1_{ij}^{\text{obj}} \left[(x_i - \hat{x_\imath})^2 + (y_i - \hat{y}_i)^2\right] \\
+\lambda_{\text{cord}} \sum_{\mathrm{i}=0}^{\mathrm{s}^2} \sum_{\mathrm{j}=0}^{\mathrm{B}} 1_{\mathrm{ij}}^{\text{obj}} \left[\left(\sqrt{\mathrm{W_i}} - \sqrt{\widehat{\mathrm{w}_\imath}}\right)^2 + \left(\sqrt{\mathrm{h_i}} - \sqrt{\widehat{\mathrm{h}_\imath}}\right)^2\right] \\
+\sum_{\mathrm{i}=0}^{\mathrm{s}^2} \sum_{\mathrm{j}=0}^{\mathrm{B}} 1_{\mathrm{ij}}^{\text{obj}} \left(\mathrm{C_i} - \hat{\mathrm{C}}_\mathrm{i}\right)^2 \qquad (1) \\
+\lambda_{\text{no obj}} \sum_{\mathrm{i}=0}^{\mathrm{s}^2} \sum_{\mathrm{j}=0}^{\mathrm{B}} 1_{\mathrm{ij}}^{\text{noobj}} \left(\mathrm{C_i} - \hat{\mathrm{C}}_\mathrm{i}\right)^2 \\
+\sum_{i=0}^{3} 1_i^{\text{obj}} \sum_{c \text{ celasses}} \left(\mathrm{P_i(c)} - \hat{\mathrm{P}}_\mathrm{i}(\mathrm{c})\right)^2
\end{gathered}$$

Modifications such as the square root of certain terms aim to balance the model's attention between large and small prediction boxes. Moreover, a weight balance factor is introduced to prioritize the regression loss, particularly in early training stages where numerous low-quality boxes are generated. Adjustments in the loss function mitigate learning from low-quality predictions lacking object boxes, enhancing the model's overall performance.

The SSD [2] (Single Shot MultiBox Detector) algorithm, proposed by Liu et al. in 2016, addressed the shortcomings of previous object detection methods like YOLO. It focused on enhancing location accuracy, overall accuracy, and recall rate. Key improvements include feature fusion for better extraction across different scales, direct prediction using CNN instead of YOLO's approach, and incorporating an anchor mechanism from Faster R-CNN to improve recall. However, SSD still struggles with accurately detecting small objects and maintaining balance between positive and negative samples.

In 2018, Lin et al. introduced RetinaNet [3], aiming to address the accuracy issues of regression-based object detection algorithms compared to candidate region-based ones. They argued that the imbalance between positive and negative samples in single-stage algorithms was a key factor. Two-stage algorithms achieved higher accuracy due to a region proposal network (RPN) that filtered out irrelevant background frames, balancing the sample categories. In contrast, single-stage algorithms directly generated candidate regions in each grid, leading to redundant boxes and difficulty in classification during training. To mitigate this, they proposed the Focal loss function (2)

$$\mathrm{FL}(p) = \begin{cases} -\alpha(1-p)^{y}\log(p) & \text{if } y = 1 \\ -(1-\alpha)p^{y}\log(1-p) & \text{otherwise} \end{cases} \tag{2}$$

which reduces the weight of easily distinguishable samples, allowing the network to focus on training challenging ones. The Focal loss function formula is provided, where p represents the probability of a positive sample, and parameters α and γ adjust the loss function. This approach helps address sample imbalance by emphasizing training on difficult samples.

FCOS [4], introduced in 2019 by Chun-Hua Shen et al., revolutionized single-stage object detection by utilizing a pixel-level approach. Unlike traditional methods relying on anchors, FCOS eliminates the need for anchor hyperparameters, thereby reducing computation and enhancing stability. By integrating with FPN, FCOS can detect objects of various sizes and handle challenges like overlapping objects and occlusion. Rather than using anchors, FCOS defines positive and negative samples by mapping points from the feature map to the original image; if a point lies within a ground-truth object, it's labeled positive. FCOS also introduces the concept of center-ness (3) to improve detection quality, ensuring regression boxes are centered around objects. This approach enhances the detection of larger objects significantly.

$$\text{centerness } *= \sqrt{\frac{min(l^{*},\mathrm{r}^{*})}{max(\mathrm{l}^{*},\mathrm{r}^{*})} \times \frac{min(\mathrm{t}^{*},\mathrm{b}^{*})}{max(\mathrm{t}^{*},\mathrm{b}^{*})}} \tag{3}$$

where $(l^{*}, \mathrm{t}^{*}, \mathrm{r}^{*}, \mathrm{b}^{*})$ are the distance from this point to the left, top, and right, and bottom of the object frame.

Conclusion. It was described how single-stage detectors have evolved from anchor-based approaches to more recent anchor-free techniques. Some of the key improvements include using pyramid network structures to extract multi-scale features, designing deeper and wider backbone networks, and employing stronger image augmentation strategies. Single-stage detectors are appealing for real-time applications due to their simple architecture and ability to be deployed on edge devices. However, challenges remain in handling small objects, complex backgrounds, and cross-domain generalization. Current research is focused on enhancing localization accuracy, improving recall, and reducing performance drops across domains. Though progress has been made, single-stage detectors still lag behind two-stage detectors in some scenarios. Further advances in network architecture and training techniques will be needed to close this gap while retaining real-time capabilities.

References

1. Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Internet]. 2016 [cited 2023 Nov 16]. p. 779–88. Available from: https://ieeexplore.ieee.org/document/7780460
2. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: Single Shot MultiBox Detector. In 2016 [cited 2024 Jan 19]. p. 21–37. Available from: http://arxiv.org/abs/1512.02325
3. Lin TY, Goyal P, Girshick R, He K, Dollár P. Focal Loss for Dense Object Detection [Internet]. arXiv; 2018 [cited 2024 Jan 19]. Available from: http://arxiv.org/abs/1708.02002
4. Tian Z, Shen C, Chen H, He T. FCOS: Fully Convolutional One-Stage Object Detection [Internet]. arXiv; 2019 [cited 2024 Jan 19]. Available from: http://arxiv.org/abs/1904.01355