http://communications.science.ankara.edu.tr

Commun.Fac.Sci.Univ.Ank.Series A2-A3
Volume 66, Number 1, Pages 118-131 (2024)
DOI: 10.33769/aupse.1378578
ISSN 1303-6009 E-ISSN 2618-6462

Research Article; Received: October 19, 2023; Accepted: January 23, 2024

HEATMAP CREATION WITH YOLO-DEEP SORT SYSTEM CUSTOMIZED FOR IN-STORE CUSTOMER BEHAVIOR ANALYSIS

Murat ŞİMŞEK[1] and Mehmet Kemal TEKBAŞ[2]

[1]Department of Artificial Intelligence Engineering, Ostim Technical University,
Ankara, TÜRKİYE
[2]Department of Electrical and Electronics Engineering, Ankara University,
Ankara, TÜRKİYE

ABSTRACT. Due to the limitations of the hardware system, analysis of retail stores has caused problems such as excessive workload, incomplete analysis, slow analysis speed, difficult data collection, non-real-time data collection, passenger flow statistics, and density analysis. However, heatmaps are a viable solution to these problems and provide adaptable and effective analysis. In this paper, we propose to use the deep sequence tracking algorithm together with the YOLO object recognition algorithm to create heatmap visualizations. We will present key innovations of our customized YOLO-Deep SORT system to solve some fundamental problems in in-store customer behavior analysis. These innovations include our use of footpad targeting to make bounding boxes more precise and less noisy. Finally, we made a comprehensive evaluation and comparison to determine the success rate of our system and found that the success rate was higher than the systems we compared in the literature. The results show that our heatmap visualization enables accurate, timely, and detailed analysis.

1. Introduction

Heatmap is a data visualization technique that uses color-coded representations to present patterns and variations in data [1]. Heatmap analysis provides an effective tool for representing and interpreting data, allowing data analysts and researchers to

Keywords. Heatmap visualization, retail analysis, YOLO, object detection, object tracking, machine learning, deep learning.

✉ murat.simsek@ostimteknik.edu.tr-Corresponding author; iD 0000-0002-8648-3693
✉ kemaltekbas@gmail.com; iD 0009-0008-9877-6021.

gain insight, make informed decisions, and communicate findings more efficiently [2, 3]. It is widely used to analyze and display large data sets, especially in the fields of statistics, business intelligence, science, and image processing [4]. The idea of retail analysis is an indispensable need for the economy [5]. The idea of increasing sales and optimizing in-store business activities by understanding human behavior and the need to optimize business processes using data from in-store traffic and interactions has been a great need in the retail world [6]. With the innovations that came with the increase in machine learning, these ideas began to develop, and better research began to be conducted to solve these problems. According to our studies, the idea of heatmap analysis in retail stores is a suitable solution to our problems [7]. The analysis method we developed reduces the time spent on customer analysis without using excessive cameras and labor to obtain data at a specific point. In this article, we present real-time analytics on retail stores to detect hot spots. In this research, we used the deep sorting algorithm along with the YOLO algorithm to create our heatmap visualization, and these algorithms are suitable for our work. YOLO (You Only Look Once) was chosen for its single-stage detector capabilities [8], allowing it to be significantly faster on object detection tasks. Additionally, YOLO offers the advantage of readily available pre-trained models, making it easy to start a project quickly. The smaller size of the model suite also contributes to its suitability, especially for projects with limited budgets or resource-constrained environments [9]. Moreover, the combination of YOLO with the Deep SORT algorithm allowed us to not only detect objects efficiently but also track them accurately over time [10]. This capability is essential for analyzing customer behavior in retail environments where individuals' movements and interactions must be comprehensively monitored, providing a high success rate even when objects are viewed from different angles [11]. Leveraging these algorithms, we were able to create precise and informative heatmap visualizations that play an important role in understanding and optimizing customer experiences in retail spaces [12]. It is also worth noting that there is still a large gap between theoretical research and real-world applications for reliably detecting people for heatmap visualization. We examined the information in the literature and the areas where heatmaps are most commonly used [13]. After our tests on the model determined to be used in our in-store analyses, we will explain in detail the mathematical and theoretical explanations of our effective and appropriate approaches to the techniques and methods used in the algorithms. Our goal is to make this process faster with higher accuracy and to detect and track multiple objects simultaneously by getting more accurate data from the real-time model.

2. Model Developments: Mathematical and Theoretical Overview

Before delving into the intricacies of our configuration, it's crucial to establish a solid foundation by elucidating the core principles underpinning key techniques like NMS (Non-Maximum Suppression), Gaussian blur, and BBOX (Bounding Box). A bounding box is a rectangular region that encloses an object of interest detected by the YOLO model during object detection [14]. We want to avoid loss of perspective and adjust where the tracked objects stand without visual confusion, we need to make the bounding box function unique and effective for our purpose.

Gaussian blur is a preprocessing step applied to the input image of the YOLO model. This operation refers to a filtering operation used to soften or blur the input image. We must create an adaptive structure that will work in any situation to enable the analysis of the collected data [15]. It should ensure that the data collected and the heatmap to be superimposed on it create a clear picture of the participation rates in any case. For our heatmap to work properly in crowded and complex areas, we can reduce the noise by smoothing the pixels.

Non-maximum suppression (NMS) is a post-processing technique used to filter out redundant object detections and improve the accuracy of the final output [16]. NMS is applied to the bounding box predictions generated by the YOLO model. When YOLO performs object detection, it divides the input image into a grid and predicts bounding boxes and class probabilities for each grid cell. However, multiple bounding boxes may overlap and detect the same object. NMS helps eliminate duplicate detections by selecting the most confident and accurate bounding boxes while discarding the rest. Due to our needs, we need to make the adjustments.

2.1. **Human-based foot bounding box method.** Instead of using the given Yolo bounding boxes in the literature, we made improvements to our case and started to detect objects from the base point. The function converts these bounding box coordinates to a different representation known as "xywh" format, where x, y represent the center of the bounding box and w, h represent the width and height of the bounding box.

A 2D bounding box, defined as a set B with xb , yb coordinates. Where {(xmin, ymin), (xmin, ymax), (xmax, ymin), (xmax, ymax)} are the four corners of the bounding box for our object [17].

$$B = \{ (x_c , y_c) \in R^2 \mid x_{min} < x_b < x_{max} , y_{min} < y_b < y_{max} \} \quad (1)$$

$$\text{Bounding Box Weight} = | x_{min} - x_{max} | = w \quad (2)$$

$$\text{Bounding Box Height} = | y_{min} - y_{max} | = h \quad (3)$$

$$x_c = (y_{min} + w / 2) \quad (4)$$

$$y_c = (y_{max} - h / 8) \quad (5)$$

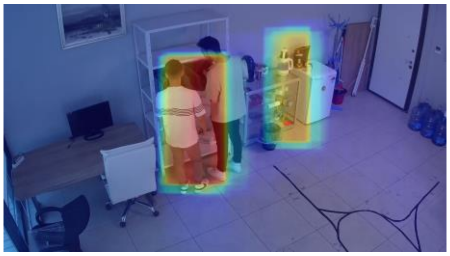

FIGURE 1. Existing bounding box (demo test environment).

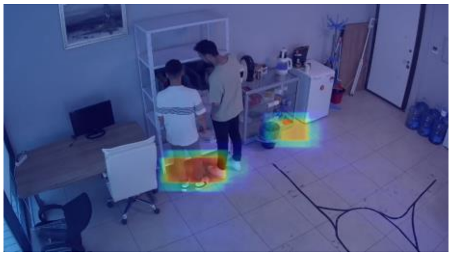

FIGURE 2. Human-based foot bounding box method (demo test environment).

2.2. **Enhancing object detection: NMS hyperparameter optimization.** There are two useful hyperparameters we should mention in this technique reduce the noise in the configuration.

The Intersection over Union (IoU) is a crucial metric used in assessing the performance of object detection and classification algorithms. It quantifies the similarity between a predicted bounding box and a ground truth (target) bounding box. The formula computes the ratio of the area of overlap (intersection) between these two boxes to the area of their union. In the formula, 'Target ∩ Prediction' represents the intersection, which is the shared region between the target and prediction boxes, while 'Target U Prediction' signifies the union, representing the total area covered by both boxes. IoU yields a value between 0 and 1, with higher values indicating better detection accuracy. Typically, a predefined threshold is used, and predictions exceeding this threshold are considered correct detections [18].

$$(\text{IoU}) = (\text{Target} \cap \text{Prediction}) / (\text{Target U Prediction}) \tag{6}$$

Another crucial aspect of object detection and classification is the confidence threshold. Each bounding box prediction generated by the model is accompanied by a confidence score, which signifies the model's level of confidence in that specific detection. During the post-processing step, known as Non-Maximum Suppression (NMS), bounding boxes with confidence scores falling below the predetermined threshold are disregarded. To be considered a valid prediction, the confidence score for a given bounding box must be greater than or equal to the threshold value. In such cases, the model's output, which may include class labels and bounding box coordinates, is accepted as a valid detection. This confidence threshold is an essential tool for filtering out low-confidence predictions, ensuring that only high-confidence detections contribute to the final results, and enhancing the precision of object detection.

There is a need to protect values below the threshold. Especially in crowded environments, it is desirable to consider objects below a certain threshold. This ensures that some items that have low scores but could be important are not eliminated. Thanks to these ideologies, the threshold parameters have been adjusted to ignore the complexity and get a clean analysis of the areas of interest.

Due to our experiments in a crowded place like a retail store, we decided to take the intersection over the union threshold as 0.5 and the confidence threshold as 0.4, which would indicate that there is a 40% chance that the object exists in that box.

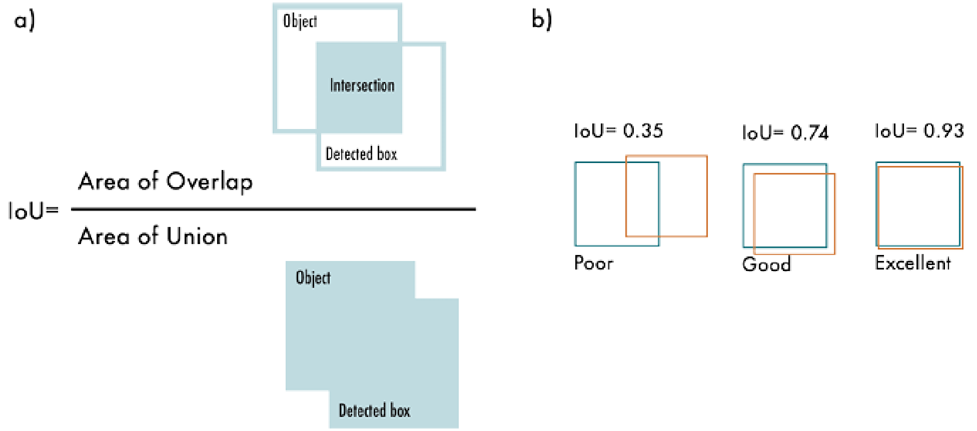

FIGURE 3. Intersection over Union (IoU). a) The IoU is calculated by dividing the intersection of the two boxes by the union of the boxes b) examples of three different IoU values for different box locations [19].

2.3. **Hyperparameter optimization for enhanced Gaussian blur.** Creating Gaussian data is a critical element to increase the sensitivity of the results obtained when creating heatmaps and to provide a more reliable analysis. Components of Gaussian data allow object details and tracking to be more accurate, as well as reduce noise in the data.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{7}$$

Where x is the distance from the origin on the horizontal axis, y is the distance from the origin on the vertical axis, and σ is the standard deviation of the Gaussian distribution. It is important to note that the origin of these axes is at the center (0, 0). When applied in two dimensions, this formula produces a surface whose contours are concentric circles with a Gaussian distribution from the center point [20].

In our configuration (13, 13), specifies the size of the Gaussian kernel. The numbers (13, 13) indicate that the kernel will have a size of 13x13 pixels. 10 is the standard deviation of the Gaussian kernel.

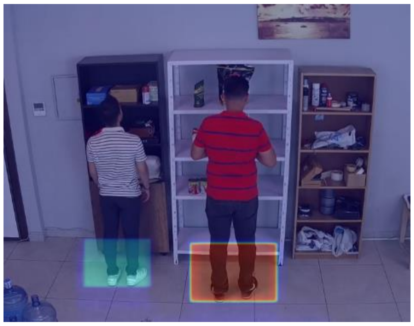

FIGURE 4. Demo test environment.

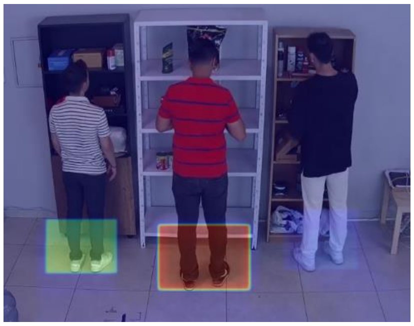

FIGURE 5. Demo test environment.

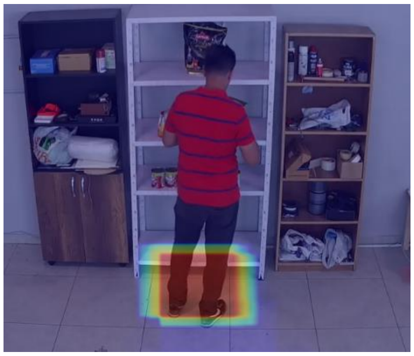

FIGURE 6. Demo test environment.

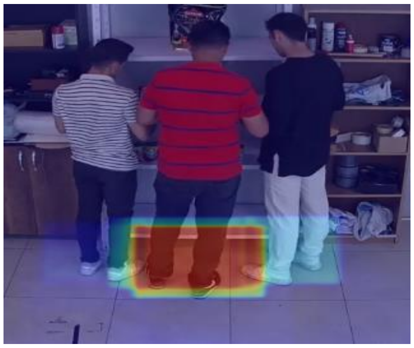

FIGURE 7. Demo test environment.

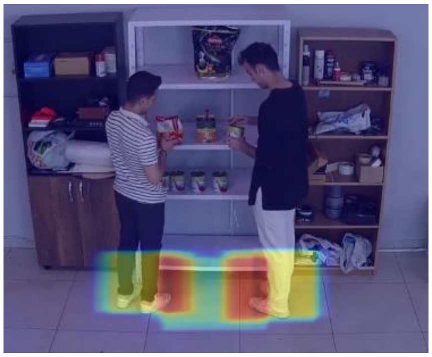

FIGURE 8. Demo test environment.

The processes outlined above have been developed in our demo test environment through the optimization of hyperparameters and the application of enhanced techniques. These advancements reflect a comprehensive approach to refining the methodologies, ensuring both the efficiency and efficacy of the results.

3. Experimental Results

In this section, we delve into the experimental results, highlighting the comprehensive examination of our study through rigorous empirical investigations. The experimental endeavors undertaken in this research are pivotal in unraveling the intricacies of the proposed methodology. We meticulously employed a set of well-defined performance parameters to assess the efficacy and robustness of our approach. These carefully selected metrics not only serve as benchmarks for evaluating the experimental outcomes but also contribute to the elucidation of the underlying dynamics of the phenomena under investigation. Through a systematic and methodical analysis of these performance parameters, we aim to provide a nuanced understanding of the outcomes derived from our experimental trials, thereby contributing to the advancement of knowledge in our field of inquiry.

$$\text{Precision} = TP / (TP + FP) \tag{8}$$

TP (true positive): the number of samples that actually belong to that class and are correctly predicted as that class by the model.

FP (false positive): The number of samples that do not belong to this class but are incorrectly predicted by the model as belonging to this class.

Precision: It is the percentage of correctly predicted positive samples out of the total positive predicted samples. In other words, it shows how accurate the model is when predicting whether an object belongs to a certain class.

$$\text{Recall} = TP / (TP + FN) \tag{9}$$

FN (false negatives): The number of samples that actually belong to that class but are incorrectly predicted by the model as another class.

Recall: It is the total sample replacement rate of correctly predicted positive samples belonging to that classified class. In other words, it shows the performance of the model in detecting a particular class.

$$\text{F1-score} = 2\ (\text{Precision x Recall}) / (\text{Precision} + \text{Recall}) \tag{10}$$

Harmonics of precision and Recall values are determined. So, there is a balanced distribution between precision and recall. While it is desired to increase both values, sometimes increasing one causes the other to decrease. F1-score combines these two metrics to provide an overall performance specification [21].

$$AP = (x+a)^n = \sum_{k=0}^{k=n-1} \binom{n}{k} x^k a^{n-k} \tag{11}$$

AP: average precision. Represents the area under the precision-recall curve for a class.

Recall(k): k. It is the sensitivity (recall) value calculated for the threshold value.

Recall(k+1): k+1. It is the sensitivity (recall) value calculated for the threshold value.

Precision(k): k. It is the precision value calculated for the threshold value.

[Recall(k)−Recall(k+1)]: This expression represents the difference between two consecutive sensitivity values. This represents a horizontal slice on the precision-recall curve [22].

The formula creates a precision-recall curve with sensitivity and sensitivity values calculated against different threshold values. The AP value is equal to the size of the area under this curve. This formula sums the horizontal slices for each of the threshold values specified to calculate this area [23]. In this article, we discussed heatmap visualization to better understand and analyze customer behavior and movements in retail stores. In this context, while examining the techniques we developed by customizing the YOLOv7 model, we also compared this model with itself and its upper version, YOLOv8, available in the literature. Before we get into performance comparisons, it's important to make a more fundamental comparison between YOLOv7 and YOLOv8. YOLOv7 and YOLOv8 are among the advanced object detection models of the YOLO series. It has an architecture based on YOLOv7, YOLOv4, YOLO-R and Scaled YOLOv4 and increases network efficiency by using Extended Efficient Layer Aggregation Network (E-ELAN) [24]. This model improves the accuracy and results of the model by considering various factors such as the number of layers, input image size, and channel width with compound scaling methods. On the other hand, YOLOv8 offers a more complex structure with the combination of FPN and PAN modules and detects objects of different sizes and shapes more effectively using advanced techniques such as Soft-

NMS [25]. In terms of performance, both models offer high accuracy, but YOLOv8 achieves a slightly higher mAP score on the COCO dataset and shows a slight reduction in the number of parameters despite more FLOPs, offering more complex calculations. These features highlight the key differences between the simplicity and efficiency of YOLOv7 and the advanced architecture and accuracy of YOLOv8 [26]. We chose the YOLOv7 model to perform fast and efficient analysis, especially in in-store systems. The size of the model and its more compact structure make it an ideal option for such applications. In addition, thanks to its compatibility, it provides the advantage of easily integrating old systems and applications into the new model. As a result of the bounding box techniques and tests we use, our approaches to hyperparameters increase the desired performances and show that our system is more effective in this area. Performance metrics on our system are calculated using an NVIDIA GeForce GTX 1650 TI, an Intel (R) CoreTM i5-10300H CPU at 2.50 GHz, and 8 GB of RAM.

TABLE 1. Evaluation of the system with image input size of 640.

PERFORMANCE METRİC	**YOLOv7_tiny**	**YOLOv8_s**	**Our system**
Precision	0.9000	0.9722	0.9250
Recall	0.8372	0.8140	0.8605
F1 Score	0.8675	0.8861	0.8916
AP	0.7535	0.7913	0.7959

As a result of our different approaches from the literature, we can see that our system stands out as an option that offers balanced performance and gives the highest result in terms of F1 score. Although the Yolov8 model has a higher ability to detect objects accurately, our system can help achieve better results in crowded and noisy environments by balancing its abilities to accurately detect and capture objects.

4. CONCLUSION

Our approach creates a heatmap visualization driven by a YOLO object detection model. Before developing this approach, we reviewed similar approaches available in the literature and adjusted our parameters and techniques to further improve the analysis of detected objects and tracked images. The main challenge we faced was analyzing hot spots in crowded or small rooms. We have identified a potential problem in these cases where the customer is closer to the security cameras due to the illusion of perspective blurring a key point rather than just showing the customer's point of interest. Our experiments continued with the aim of more effectively detecting and tracking objects in our retail analysis. Through our approach, we determined that displaying the bounding box to customers at the base of their feet was an ideal solution. This solution allowed people to better observe the points in front of the shelves and reduced complexity. The images [Figure 9–Figure 11] used in the data collection phase of this article were sourced from Pixabay. Pixabay is a platform offering copyright-free images and videos, allowing for the free distribution of downloaded photos and graphics. The particular images are licensed under the Creative Commons CC0 license, indicating that they are freely available for public and non-commercial use.

FIGURE 9. Heatmap visualization with Customized Yolo-Deep SORT System.

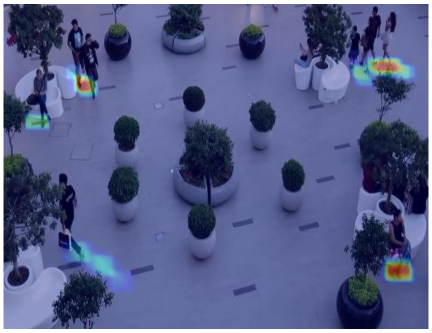

FIGURE 10. Heatmap visualization with Customized Yolo-Deep SORT System.

FIGURE 11. Data collecting.

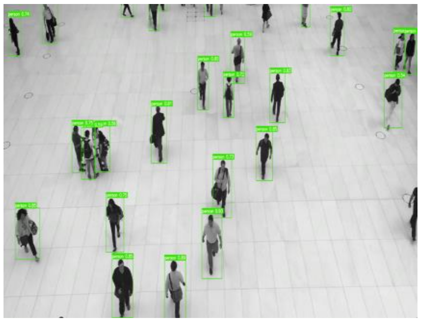

FIGURE 12. Use and development of detection algorithm.

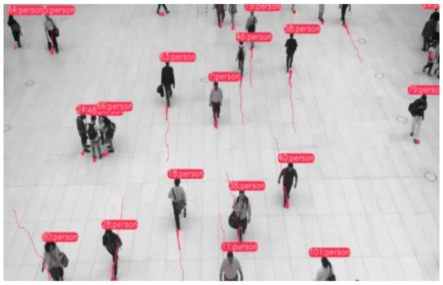

FIGURE 13. Use and development of tracking algorithm.

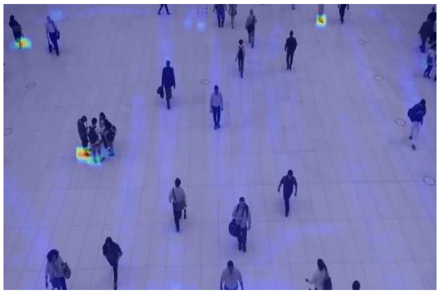

FIGURE 14. Heatmap visualization with Customized Yolo-Deep SORT System.

Author Contribution Statements Authors are equally contributed to the paper. All authors read and approved the final copy of the manuscript.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement This research did not receive any specific grant from funding agencies in public or not-for-profit sectors.

References

[1] Liu, M., Lee, J., Kang, J., Liu, S., What we can learn from the data: a multiple-case study examining behavior patterns by students with different characteristics in using a serious game, *Tech. Knowl. Learn.*, 21, (2016), 33-57, https://dx.doi.org/10.1007/s10758-015-9263-7.

[2] Fernandez, N., Gundersen, G., Rahman, A., Grimes, M., Rikova, K., Hornbeck, P., Ma'ayan, A., Clustergrammer, A web-based heatmap visualization and analysis tool for high-dimensional biological data, *Sci. Data*, 4, (2017), 170151, https://dx.doi.org/10.1038/sdata.2017.151.

[3] Gu, Z., Complex heatmap visualization, *iMeta*, 1 (3), (2022), https://doi.org/10.1002/imt2.43.

[4] Deng, W., Wang, Y., Liu, Z., Cheng, H., Xue, Y., Hemi: a toolkit for illustrating heatmaps, *PLoS ONE*, 9 (11), (2014), https://doi.org/10.1371/journal.pone.0111988.

[5] Mondal, S., Das, S., Musunuru, K., Dash, M., Study on the factors affecting customer purchase activity in retail stores by confirmatory factor analysis, *ESPACIOS*, 38 (61), (2018), 30.

[6] Girgensohn, A., Shipman, F., Wilcox, L. D., Determining activity patterns in retail spaces through video analysis, *Proc. ACM Conf. Multimedia*, (2008), 889-892, https://doi.org/10.1145/1459359.1459514.

[7] Oliveira, K., RetailNet: A Deep Learning Approach for People Counting and Hot Spots Detection in Retail Stores, Rio de Janeiro, Brazil, 2019.

[8] Onıga, F., Bacea, D., Single stage architecture for improved accuracy real-time object detection on mobile devices, *Img. Vis. Comput.*, 130 (9), (2023), 104613, https://doi.org/10.1016/j.imavis.2022.104613.

[9] Diwan, T., Anirudh, G., Tembhurne, J. V., Object detection using YOLO: challenges, architectural successors, datasets and applications, *Multimed. Tools Appl.*, 82 (6), (2023), 9243-9275, https://doi.org/10.1007/s11042-022-13644-y.

[10] Lakshmi Rishika, A., Aishwarya, Ch., Sahithi, A., Premchender, M., Real-time vehicle detection and tracking using yolo-based deep sort model: a computer vision application for traffic surveillance, *Turkish J. Comp. Math. Edu*., 14 (1), (2023), 255-264, https://doi.org/10.17762/turcomat.v14i1.13530.

[11] Aich, S., Stavness, I., Improving Object Counting with Heatmap Regulation, (2018), https://doi.org/10.48550/arXiv.1803.05494.

[12] Ilikci, B., Chen, L., Cho, H., Liu, O., Heat-map based emotion and face recognition from thermal images, *Comput. Commun. IoT Appl.*, (2019), 449-453.

[13] Bulat, A., Tzimiropoulos, G., Human Pose Estimation via Convolutional Part Heatmap Regression, Amsterdam, Netherlands, (2016).

[14] Pharr, M., Humphreys, G., Bounding box, *Physically Based Rendering*, 3, (2017).

[15] Huang, Z., Li, W., Xia, X.-G., Tao, R., A general Gaussian heatmap label assignment for arbitrary-oriented object detection, *IEEE Transc. Img. Process*., (2022), https://doi.org/10.1109/TIP.2022.3148874.

[16] Salim, M. P., Ong, J. J., IS, E., Surhatono, D., Object detection for child Learning media, *Inter. Conf. Sci. Tech*. (ICST), 8, Yogyakarta, Indonesia, (2022), 1-6.

[17] He, Y., Zhu, C., Wang, J., Savvides, M., Zhang, X., Bounding box regression with uncertainty for accurate object detection, *Proc. IEEE/CVF Conf. Comp.Vision Pattern Recog*., (2019), 2888-2897, https://doi.org/10.48550/arXiv.1809.08545.

[18] Hosang, J., Benenson, R., Schiele, B., Learning non-maximum suppression, *Proc. IEEE Conf. Comp. Vision Pattern Recog*. (CVPR), (2017), 4507-4515, https://doi.org/10.48550/arXiv.1705.02950.

[19] Córdova-Esparza, M., Terven, J., A comprehensive review of yolo: from yolov1 to yolov8 and beyond, *Mach. Learn. Knowl. Extr*. 5, (2023), 1680-1716 https://doi.org/10.3390/make5040083.

[20] Chandel, R., Gupta, G., Image filtering algorithms and techniques: a review, *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, 3 (10), (2013).

[21] Hicks, S. A., Strümke, I., Thambawita, V., Hammou, M., Riegler, M. A., Halvorsen, P., Parasa, S., On evaluation metrics for medical applications of artificial intelligence, *Sci. Rep*., 12 (1), (2022), 5979, https://doi.org/10.1038/s41598-022-09954-8.

[22] Ajayi O. G. , Ashi J., Guda B., Performance evaluation of YOLO v5 model for automatic crop and weed classification on UAV images, *Smart Agricult. Tech.*, 5, (2023), 100231.

[23] Atik, M. E., Duran, Z, Ozgunluk, R., Comparison of YOLO versions for object detection from aerial images, *Int. J. Environ. Geoinform.*, 9 (2), (2022), 87-93, https://doi.org/10.30897/ijegeo.1010741.

[24] Karadağ, B., Arı, A., Akıllı mobil cihazlarda YOLOv7 modeli ile nesne tespiti, *Politeknik J.*, 26 (3), (2023), 1207-1214, https://doi.org/10.2339/politeknik.1296541.

[25] Özel, M. A., Baysal, S. S., Şahin, M., Derin öğrenme algoritması (YOLO) ile dinamik test süresince süspansiyon parçalarında çatlak tespiti, *Eur. J. Sci. Technol.*, (26), (2021), 1-5, https://doi.org/10.31590/ejosat.952798.

[26] Bayram, A. F., Nabiyev, V., Derin öğrenme tabanlı saklanan kamufle tankların tespiti: son teknoloji YOLO ağlarının karşılaştırmalı analizi, *Gümüşhane Univ. J. Nat. Appl. Sci.*, 13 (4), (2023), 1082-1093, https://doi.org/10.17714/gumusfenbil.1271208.