

## Article

# Computer-Vision-Based Statue Detection with Gaussian Smoothing Filter and EfficientDet

Mubarak Auwalu Saleh <sup>1,\*</sup> , Zubaida Said Ameen <sup>1</sup>, Chadi Altrjman <sup>2,3</sup> and Fadi Al-Turjman <sup>1,2</sup> 

<sup>1</sup> Artificial Intelligence Engineering Department, AI and Robotics Institute, Near East University, North Cyprus via Mersin 10, Nicosia 99138, Turkey

<sup>2</sup> Research Center for AI and IoT, Faculty of Engineering, University of Kyrenia, North Cyprus via Mersin 10, Kyrenia 99320, Turkey

<sup>3</sup> Faculty of Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

\* Correspondence: auwalusaleh.mubarak@neu.edu.tr

**Abstract:** Smart tourism is a developing industry, and numerous nations are planning to establish smart cities in which technology is employed to make life easier and link nearly everything. Many researchers have created object detectors; however, there is a demand for lightweight versions that can fit into smartphones and other edge devices. The goal of this research is to demonstrate the notion of employing a mobile application that can detect statues efficiently on mobile applications, and also improve the performance of the models by employing the Gaussian Smoothing Filter (GSF). In this study, three object detection models, EfficientDet—D0, EfficientDet—D2 and EfficientDet—D4, were trained on original and smoothed images; moreover, their performance was compared to find a model efficient detection score that is easy to run on a mobile phone. EfficientDet—D4, trained on smoothed images, achieves a Mean Average Precision (mAP) of 0.811, an mAP-50 of 1 and an mAP-75 of 0.90.

**Keywords:** smart cities; computer vision; object detection; mobile application



**Citation:** Saleh, M.A.; Ameen, Z.S.;

Altrjman, C.; Al-Turjman, F.

Computer-Vision-Based Statue

Detection with Gaussian Smoothing

Filter and EfficientDet. *Sustainability*

2022, 14, 11413. [https://doi.org/](https://doi.org/10.3390/su141811413)

10.3390/su141811413

Academic Editor: Zubair Baig

Received: 22 June 2022

Accepted: 21 July 2022

Published: 12 September 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Smart cities are based on the notion of using various forms of technology to address people's daily needs. It provides residents with a great deal of convenience and value. Tourism has an important function in cities, as well as social places, entertainment, and shopping complexes [1]. Travel blogs, tour guides and maps have been the traditional methods used by tourists when they are on tour. In today's world, those methods are described as outdated. Object detection is a fundamental study area in computer vision and artificial intelligence. Its goal is to find the target of interest in an image, establish the category appropriately, and provide the bounding box of each target. It is a requirement for more advanced computer vision tasks, including target tracking, object detection, pattern recognition, and semantic scene interpretation [2].

The tourist industry is gradually incorporating intelligent tourism applications, and research into these applications has recently begun [3]. Using visual sensors and artificial intelligence, computer vision and machine learning may assist give trustworthy intelligent autonomous solutions to smart tourism in several ways [4–6]. Features and image processing are very important in detection. Filtering is a very important task in image processing; it removes noise, enhances contours and improves texture in images. The Gaussian Smoothing filter has proven to be effective in many areas; the filtering enhances the model's performance in classification [7–9]. The Viola–Jones detector's implementation strategy is similar to that of the standard object detection method. It largely uses artificial design and innovative feature extraction algorithms for recognition and detection [10,11], as well as Support Vector Machines, Decision Trees [12], and other classifiers. The image is frequently preprocessed before detection to improve image quality [13]. To forecast the

object, sliding window processing is frequently used for the image during the detection phase. The best detection performance is attained at that moment.

The sliding window approach, on the other hand, lays heavy demands on the computer's computational capacity because it traverses all potential locations and size ratios. Furthermore, the capacity to express hand-crafted features is limited, adding to a poor overall detection impact. The Region Convolutional Neural Network (RCNN) algorithm [14] was proposed in 2014, and Convolutional Neural Networks(CNN) was used to extract features. CNN is a machine learning method that has pushed the development of object identification tasks to the deep learning level. The gradient descent approach may be used by deep learning to automatically optimize model parameters [15–17]. Various object detection tasks have achieved significant progress.

The motivation of the study is to develop a smart tourism application that can serve as a means of information in modern tourism. Object detection models are affected by noise in images. This motivated us to improve the model's performance by applying the Gaussian Smoothing Filter to enhance the images. The main contributions of this study are as follows:

- Applying the Gaussian Smoothing filter to improve the performance of the trained models.
- A lightweight object detection model that can be deployed to mobile phones and edge devices was proposed to detect statues efficiently.
- A mobile application with the trained model as a backend was developed to detect the statue and give information about the statue.

## 2. Related Works

In this section, we highlight the main existing alternatives for the targeted problem. We classify these alternatives into single versus two-stage detection approaches.

### 2.1. Two-Stage Detection

Two steps of the two-stage detection architecture are region proposal and object detection. On the image to be examined, a series of region proposal boxes are first presented. Object detection is then carried out. The RCNN detection system, presented by [14], first generates a region proposal box on the picture using selective search [18]. The CNN is then used to extract features, after which the SVM classifier and bounding-box regression are trained, and the outcome is predicted. Although utilizing CNN for feature extraction improves the detection impact significantly, it comes with several downsides, including a lengthy training process. Fast RCNN has been updated to address the aforementioned concerns [19], and faster RCNN [20] algorithms have been developed. The end-to-end detection procedure is completed faster with RCNN. First, the RPN algorithm was proposed as a replacement for the selective search for regional ideas, resulting in a considerable reduction in the amount of time spent on regional recommendations. Furthermore, by eliminating repeating feature calculations, shared features save time; detection accuracy on the VOC07 dataset [21] is 73.2%, and 42.7% on the COCO dataset. The study referenced in [22] concluded that this not only improves classification risk but also better integrates feature extraction and classifier function, which is critical for pedestrian classification at various sizes. Several studies were carried out on the two-stage detectors and models such as RFCN [23] and the RCNN family [14,19,20,24].

### 2.2. One Stage Detection

In one-stage detection, the region proposal stage is eliminated; unlike in two-stage detection, bounding boxes were assigned at objects' centres randomly on feature maps, and then a single network was used in processing the image. This concept improves detection speed in calculating the bounding boxes and the probability of each zone [25–27]. However, there is a degradation in accuracy due to the limited number of objects, and the models have issues with smaller objects. Speed in detection is very important, especially in autonomous vehicles [28]. YOLOv2 and YOLOv3 were proposed by [29,30], respectively, as a result

of the aforementioned concerns. They were fine-tuned to address the aforementioned difficulties, resulting in a single-stage detector that improved detection accuracy while also striking a reasonable balance between speed and precision.

The researchers [31] have presented a framework for SSD one-stage detection [31], which is different from YOLO. In this technique, SSD, multilayer mapping in the convolutional layer technique, detects multiscale objects via producing feature layers of various sizes; this improves the detection of smaller objects. The RetinaNet detector was proposed by [32] in 2017, which has a new loss function that improves detection by paying attention to classification and imbalance in a data set. The study referenced in [33] proposed ALFnet [33], which detection speed is similar to SDD and detection accuracy is similar to Faster RCNN. To improve the previous loss function, [34] presented DIOU Loss and CIOU Loss in 2019. It considers the overlap area, centre point distance, and aspect ratio, as opposed to the prior object box regression loss. The bounding box with distance loss has a quicker convergence speed and greater convergence accuracy, improving the object detection framework's detection accuracy. The study referenced in [35] suggested YOLOv4. Backbone partly employs the CSPNet structure [36], which takes advantage of the benefits of numerous detection frameworks. Adding the SPP structure [37] and the PAN structure [38] to the neck section allows for feature fusion. The benefits of clustering [39] are also employed to calculate the projected frame size. The PAN structure combines the information collected from multiple layers, and it can enable the network to integrate the features of different scales. Finally, on the coco dataset, YOLOv4 achieves 65FPS detection speed, reaching the optimum balance of current detection frame speed and accuracy.

The YOLO series approaches [27,29,30,35], namely, deep-learning-based SSD, were proposed in recent years. The YOLO techniques may be utilized for a variety of object detection tasks. Because just a small number of objects are anticipated in each anchor, missing detection is common in congested pedestrian scenes, and the algorithm's performance suffers as a result. However, because of the fast detection speed of such algorithms, pedestrian detection technology may be used in the field of intelligent driving. The SSD [31] technique is presented for generic object detection, and it can help with the multiscale detection problem in pedestrian detection.

In the broad object detection field, the RetinaNet [32] detector proposes a novel loss function that can increase detection accuracy. The ALFnet [40] algorithm is mostly used to identify pedestrians. It may be extended to generic object detection to some extent due to the effective enhancement of the task of pedestrian detection. The border regression problem in object identification is investigated by the CIOU Loss [34] method, which significantly increases the detection effect of diverse objects.

### 3. Statue Detection

In this section, we use statue detection as a proof of concept for the proposed approach.

#### 3.1. Dataset and Data Preprocessing

A video of thirty various sculptures surrounding the Near East University was collected for this study; the aim was to capture each statue from a different angle. In total, 300 images were generated from each video. Annotations were carried out on the images using the Roboflow online tool. Images were enhanced by turning them horizontally and vertically, cropping and randomly changing the brightness. The data augmentation will improve the model's robustness and reduce overfitting [41–47]. The dataset was split into 80%, 10% and 10% training, validation and testing, respectively. Table 1 gives details about the data.

**Table 1.** Dataset split information.

Training	Validation	Testing
6696	774	774

### 3.2. Gaussian Filtering (Smoothing)

Gaussian filters are a class of linear smoothing filters with weights chosen according to the form of the Gaussian function [48]. Equation (1) is a very good filter for eliminating noise taken from the normal distribution of the Gaussian smoothing filter.

$$G(x) = e^{-\frac{x^2}{2\sigma^2}} \quad (1)$$

Equation (1) determines the width of the Gaussian. The two-dimensional discrete Gaussian zero mean function (2) is used as a smoothing filter for image processing.

$$g[i, j] = e^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2)$$

### 3.3. EfficientDet

EfficientDet [49] is a neural network for object detection. It is a TensorFlow object recognition API that supports many model families, including CenterNet. [50], MobileNet [51], ResNet, and Fast R-CNN. The EfficientDet outperform the pre-existing models employed in [32,52]; they are also lightweight models that can be deployed to edge devices and mobile applications.

### 3.4. Model Training

In this study, three EfficientDet architecture models were employed to detect statues. The training was carried out in two stages: firstly, the training was conducted on original augmented images, and secondly, training was conducted on augmented Gaussian Smoothed filtered images. Instead of using CSV annotation, as it is the traditional API, we changed the training process to use PASCAL VOC (XML) annotation to reduce the XML-to-CSV conversion stage. Furthermore, a post-training quantization was performed to reduce the model's size in order to improve CPU and hardware accelerator latency. The post-processing slightly reduces the accuracy of the model.

The three EfficientDet models employed, D0, D2 and D4, were trained for 10 epochs using a batch size of 8. Version D2 was regarded to have a compromise between accuracy and detection speed, making it suitable for deployment to a mobile application. D4 is heavyweight; it has better accuracy but less processing speed on the mobile application. After the models had been trained and exported, the android studio was used to create a mobile app in which the exported model could be used for statue detection. Figure 1 Shows the complete training and deployment process of this study. Google Research created Colaboratory (Colab), which was used for training. Colab is a Linux machine with a user interface based on the Jupyter notebook service that requires no configuration. It offers free access to reasonable computer capabilities, such as the Graphical Processing Unit (GPU) [33]. As a result, popular libraries such as Keras, TensorFlow, PyTorch, and OpenCV are utilized to create deep learning applications. This virtual computer has a 2 core CPU and 12 GB of RAM, which may be expanded to 25 GB for free if necessary. When using the laptop, the GPU will be assigned at random. Nvidia Tesla K80s, T4s, P4s, and P100s are among the GPUs that may be accessed [33].

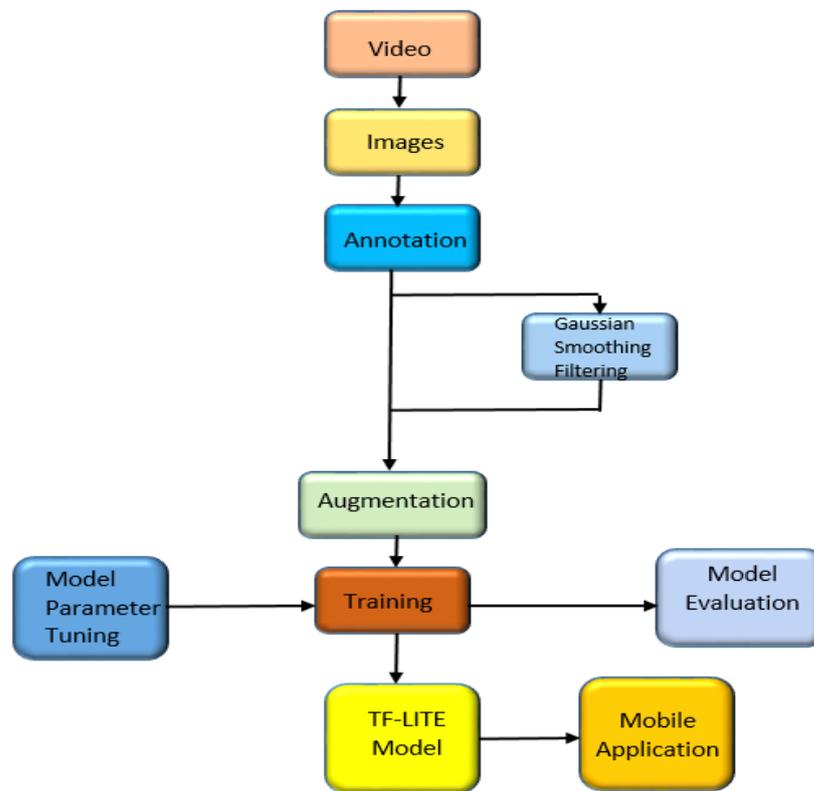


Figure 1. Complete process of the statue detection system.

#### 4. Results and Discussion

In this study, three object detection models, EfficientDet—D0, EfficientDet—D2 and EfficientDet—D4, were employed to detect statues; the performance of two of the models was compared to find the best performing model. The D0 achieves an mAP of 0.348, mAP-50 of 0.652 and mAP-75 of 0.3. The D2 achieves an mAP of 0.63, mAP-50 of 0.894 and mAP-75 of 0.81, whereas the D4 achieves an mAP of 0.751, an mAP-50 of 1 and an mAP-75 of 0.837. The D4 achieves a higher mAP with over 10%, an mAP-50 with more than 20% and an mAP-75 with over 3%. For the models trained with smoothed images, the D0 achieves an mAP of 0.352, mAP-50 of 0.595 and mAP-75 of 0.41. The D2 trained on smoothed images achieves an mAP of 0.694, mAP-50 of 0.966 and mAP-75 of 0.867. The D4 trained with smoothing images achieves an mAP of 0.811, mAP-50 of 1 and mAP-75 of 0.90.

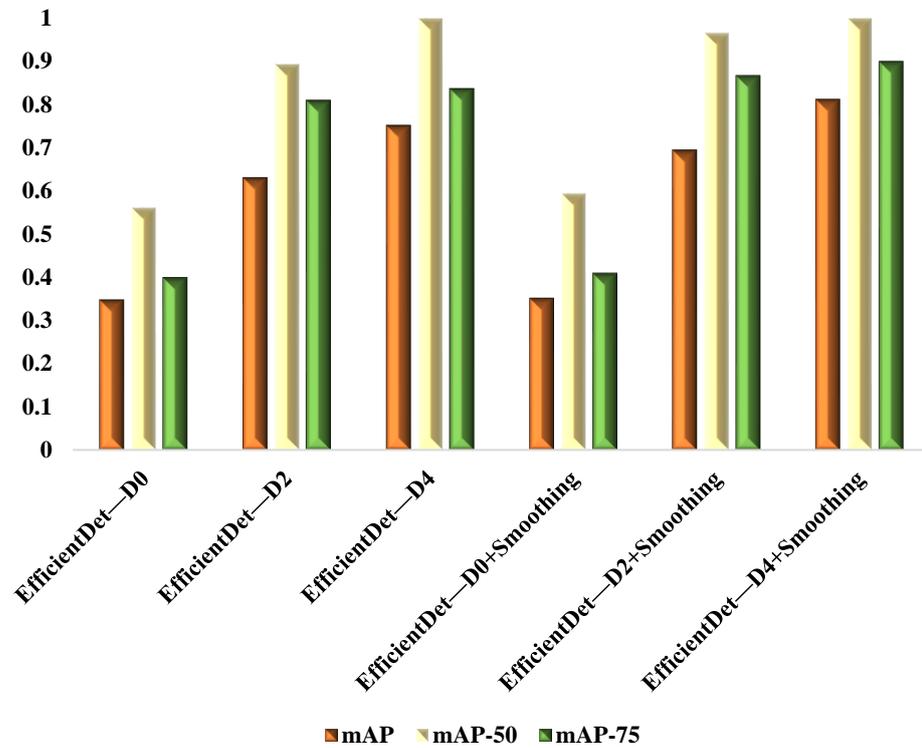
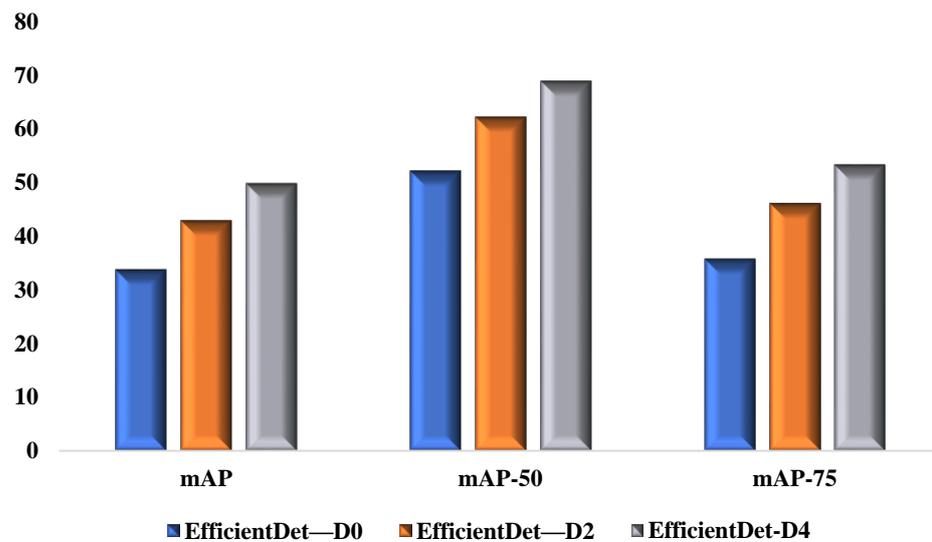
The models trained with smoothed images show improved performance compared with the models trained on the original images. The performance of the models is presented in Table 2. On the COCO datasets, the performance of the EfficientDet family models was compared and presented in Table 3. The EfficientDet—D4 achieves an mAP of 49.9, mAP-50 of 69 and mAP-75 of 53.4. The performance of the proposed model and the performance of the model on the COCO dataset are presented graphically in Figures 2 and 3, respectively.

Table 2. Performance of the EfficientDet D0, D2 and D4 employed in the study.

Models	mAP	mAP-50	mAP-75
EfficientDet—D0	0.348	0.562	0.4
EfficientDet—D2	0.63	0.894	0.81
EfficientDet—D4	0.751	1	0.837
EfficientDet—D0 + Smoothing	0.352	0.595	0.41
EfficientDet—D2 + Smoothing	0.694	0.966	0.867
EfficientDet—D4 + Smoothing	0.811	1	0.9

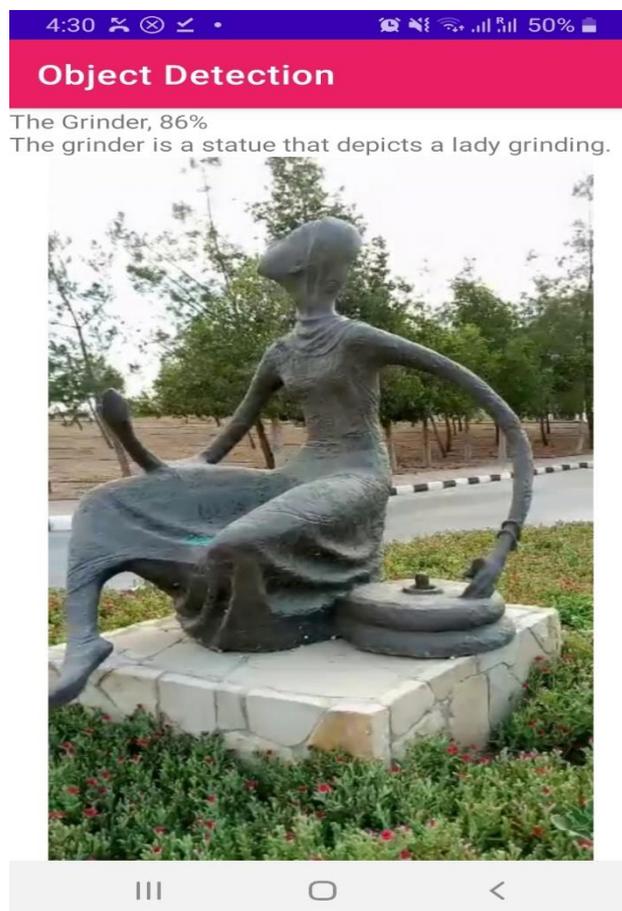
**Table 3.** Performance of the EfficientDet D0, D2 and D4 on the COCO dataset.

Model	mAP	mAP-50	mAP-75
EfficientDet—D0	33.8	52.2	35.8
EfficientDet—D2	43	62.3	46.2
EfficientDet—D4	49.9	69	53.4

**Figure 2.** The performance of the EfficientDet D0, D2 and D4 employed in the study.**Figure 3.** The performance of the EfficientDet D0, D2 and D4 on the COCO dataset.

To implement the mobile-app-based detection system, the trained model which serves as the backend of the system was incorporated into the mobile app. The mobile is set to detect five objects at a time with a minimum detection threshold of 40%. If the statue detection is not 40%, the model will not present the detected object; once the object is

detected, the statue and its information will be presented in another screen called the statue description screen. The detection and information screen can be seen in Figure 4.



**Figure 4.** Object detection and the information screen.

## 5. Conclusions

One area of rapidly growing research is smart education. Several object detection models have been developed to efficiently detect objects with high accuracy. In this study, EfficientDet—D0, EfficientDet—D2 and EfficientDet—D4 were proposed to detect statues for outdoor tourism and the trained model was used to develop a light mobile application. The EfficientDet—D4 trained on smoothed images achieves a mean average precision (mAP) of 0.811, mAP-50 of 1 and mAP-75 of 0.90. The model's performance demonstrates that it is capable of carrying out the desired job. The drawbacks of object detection models are datasets and high computational resources. More networks will be examined in a future study to improve detection performance and speed the detection. The study's drawbacks thus far are the need for extra training in the event of more statues.

**Author Contributions:** Conceptualization, M.A.S. and F.A.-T.; methodology, M.A.S.; software, M.A.S. and Z.S.A.; validation, M.A.S., C.A. and Z.S.A.; data curation, M.A.S. and Z.S.A.; writing—original draft preparation, M.A.S. and Z.S.A.; writing—review and editing, M.A.S. and Z.S.A.; visualization, C.A.; supervision, F.A.-T.; project administration, F.A.-T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Acknowledgments:** We would like to acknowledge the International Research Centre for AI and IoT and the AI and Robotics Institute, Near East University for the provided support in this research.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Fritz, F.; Susperregui, A.; Linaza, M.T. Enhancing Cultural Tourism experiences with Augmented Reality Technologies. In Proceedings of the 6th International Symposium on Virtual Reality, Archaeology and Cultural Heritage (VAST), Pisa, Italy, 8–11 November 2005.
2. Deng, J.; Xuan, X.; Wang, W.; Li, Z.; Yao, H.; Wang, Z. A review of research on object detection based on deep learning. *J. Phys. Conf. Ser.* **2020**, *1684*, 012028. [[CrossRef](#)]
3. Zaifri, M.; Azough, A.; El Alaoui, S.O. Experimentation of visual augmented reality for visiting the historical monuments of the medina of Fez. In Proceedings of the 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 2–4 April 2018; IEEE: Manhattan, NY, USA, 2018; pp. 1–4. [[CrossRef](#)]
4. Majidi, B.; Bab-Hadiashar, A. Aerial tracking of elongated objects in rural environments. *Mach. Vis. Appl.* **2009**, *20*, 23–34. [[CrossRef](#)]
5. Shamisa, A.; Majidi, B.; Patra, J.C. Sliding-window-based real-time model order reduction for stability prediction in smart grid. *IEEE Trans. Power Syst.* **2019**, *34*, 326–337. [[CrossRef](#)]
6. Mansouri, A.; Majidi, B.; Shamisa, A. Metaheuristic neural networks for anomaly recognition in industrial sensor networks with packet latency and jitter for smart infrastructures. *Int. J. Comput. Appl.* **2021**, *43*, 257–266. [[CrossRef](#)]
7. Gedraite, E.S.; Hadad, M. Investigation on the effect of a Gaussian Blur in image filtering and segmentation. *Proc. Elmar Int. Symp. Electron. Mar.* **2011**, 393–396. Available online: <https://www.semanticscholar.org/paper/Investigation-on-the-effect-of-a-Gaussian-Blur-in-Gedraite-Hadad/6c1144d8705840e075739393a10235fcc4cd0f4b#citing-papers> (accessed on 21 June 2022).
8. Magnier, B.; Montesinos, P.; Diep, D. Ridges and valleys detection in images using difference of rotating half smoothing filters. In *International Conference on Advanced Concepts for Intelligent Vision Systems*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6915, pp. 261–272. [[CrossRef](#)]
9. Liu, J.; Xu, C.; Zhao, Y. Improvement of Facial Expression Recognition Based on Filtering and Certainty Check. In Proceedings of the 2021 International Conference on Electronic Information Engineering and Computer Science (EIECS), Changchun, China, 23–26 September 2021; IEEE: Manhattan, NY, USA, 2021; pp. 209–213. [[CrossRef](#)]
10. Ke, Q.; Zhang, J.; Song, H.; Wan, Y. Big data analytics enabled by feature extraction based on partial independence. *Neurocomputing* **2018**, *288*, 3–10. [[CrossRef](#)]
11. Zhou, B.; Duan, X.; Ye, D.; Wei, W.; Woźniak, M.; Połap, D.; Damaševičius, R. Multi-level features extraction for discontinuous target tracking in remote sensing image monitoring. *Sensors* **2019**, *19*, 4855. [[CrossRef](#)]
12. Wei, W.; Poap, D.; Li, X.; Woźniak, M.; Liu, J. Study on Remote Sensing Image Vegetation Classification Method Based on Decision Tree Classifier. In Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI), Bangalore, India, 18–21 November 2018; IEEE: Manhattan, NY, USA, 2019; pp. 2292–2297. [[CrossRef](#)]
13. Zhang, L.; Shen, P.; Peng, X.; Zhu, G.; Song, J.; Wei, W.; Song, H. Simultaneous enhancement and noise reduction of a single low-light image. *IET Image Process.* **2016**, *10*, 840–847. [[CrossRef](#)]
14. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; IEEE: Manhattan, NY, USA, 2014; pp. 580–587. [[CrossRef](#)]
15. Wei, W.; Zhou, B.; Maskeliūnas, R.; Damaševičius, R.; Połap, D.; Woźniak, M. Iterative Design and Implementation of Rapid Gradient Descent Method. In Proceedings of the International Conference on Artificial Intelligence and Soft Computing, Zakopane, Poland, 29 April–3 May 2012; Springer: Cham, Switzerland, 2019; Volume 11508, pp. 530–539. [[CrossRef](#)]
16. Ameen, Z.S.; Saleh Mubarak, A.; Altrjman, C.; Alturjman, S.; Abdulkadir, R.A. C-SVR Crispr: Prediction of CRISPR/Cas12 guideRNA activity using deep learning models. In Proceedings of the 2021 International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT), Nicosia, Turkey, 27–28 December 2021; IEEE: Manhattan, NY, USA, 2021; Volume 60, pp. 9–12.
17. Ameen, Z.S.; Saleh Mubarak, A.; Altrjman, C.; Alturjman, S.; Abdulkadir, R.A. Explainable Residual Network for Tuberculosis Classification in the IoT Era. In Proceedings of the International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT), Nicosia, Turkey, 27–28 December 2021; IEEE: Manhattan, NY, USA, 2021; pp. 9–12. [[CrossRef](#)]
18. Van De Sande, K.E.A.; Uijlings, J.R.R.; Gevers, T.; Smeulders, A.W.M. Segmentation as selective search for object recognition. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; IEEE: Manhattan, NY, USA, 2011; pp. 1879–1886. [[CrossRef](#)]
19. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; IEEE: Manhattan, NY, USA, 2015; pp. 1440–1448. [[CrossRef](#)]
20. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)]

21. Everingham, M.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [[CrossRef](#)]
22. Cai, Z.; Saberian, M.; Vasconcelos, N. Learning Complexity-Aware Cascades for Pedestrian Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 2195–2211. [[CrossRef](#)] [[PubMed](#)]
23. Dai, J. R-FCN: Object Detection via Region-based Fully Convolutional Networks. *Adv. Neural Inf. Process. Syst.* **2016**, 29–38. Available online: <https://proceedings.neurips.cc/paper/2016/hash/577ef1154f3240ad5b9b413aa7346a1e-Abstract.html> (accessed on 21 June 2022).
24. Divvala, S.K.; Hoiem, D.; Hays, J.H.; Efros, A.A.; Hebert, M. An empirical study of context in object detection. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; IEEE: Manhattan, NY, USA, 2009; pp. 1271–1278. [[CrossRef](#)]
25. Mubarak, A.S.; Sa'id Ameen, Z.; Tonga, P.; Al-Turjman, F. Smart Tourism: A Proof of Concept For Cyprus Museum of Modern Arts In The IoT Era. In Proceedings of the 2021 International Conference on Artificial Intelligence of Things (ICAIoT), Nicosia, Turkey, 3–4 September 2021; IEEE: Manhattan, NY, USA, 2021; pp. 49–53. [[CrossRef](#)]
26. Mubarak, A.S.; Ameen, Z.S.; Tonga, P.; Altrjman, C.; Al-Turjman, F. A Framework for Pothole Detection via the AI-Blockchain Integration. In Proceedings of the 2021 International Conference on Forthcoming Networks and Sustainability in AIoT Era (FoNeS-AIoT), Erbil, Iraq, 28 September 2022; IEEE: Manhattan, NY, USA, 2022; pp. 398–406.
27. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE: Manhattan, NY, USA, 2016; pp. 779–788. [[CrossRef](#)]
28. Wu, B.; Wan, A.; Iandola, F.; Jin, P.H.; Keutzer, K. SqueezeDet: Unified, Small, Low Power Fully Convolutional Neural Networks for Real-Time Object Detection for Autonomous Driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; IEEE: Manhattan, NY, USA, 2017; pp. 446–454. [[CrossRef](#)]
29. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger Joseph. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; IEEE: Manhattan, NY, USA, 2017; pp. 187–213.
30. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
31. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.; Berg, A.C. SSD: Single Shot MultiBox Detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
32. Lin, T.; Ai, F.; Doll, P. Focal Loss for Dense Object Detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; IEEE: Manhattan, NY, USA, 2017; pp. 2980–2988.
33. Bisong, E. Google Colaboratory. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform*; Apress: Berkeley, CA, USA, 2019; pp. 59–64.
34. Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7 February 2020; Association for the Advancement of Artificial Intelligence (AAAI): Menlo Park, CA, USA, 2020; Volume 34, pp. 12993–13000. [[CrossRef](#)]
35. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
36. Wang, C.-Y.; Liao, H.-Y.M.; Wu, Y.-H.; Chen, P.-Y.; Hsieh, J.-W.; Yeh, I.-H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580. [[CrossRef](#)]
37. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Cham, Switzerland, 2014; Volume 8691, pp. 346–361. [[CrossRef](#)]
38. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. PANet: Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE: Manhattan, NY, USA, 2018; pp. 8759–8768.
39. Zhang, D.S.; Li, S.Z.; Wei, W. Visual clustering methods with feature displayed function for self-organizing. In Proceedings of the 2010 The 2nd International Conference on Industrial Mechatronics and Automation, Wuhan, China, 30–31 May 2010; IEEE: Manhattan, NY, USA, 2010; Volume 2, pp. 452–455. [[CrossRef](#)]
40. Liu, W.; Liao, S.; Hu, W.; Liang, X.; Chen, X. Learning Efficient Single-Stage Pedestrian Detectors by Asymptotic Localization Fitting. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; Springer: Cham, Switzerland, 2018; Volume 11218, pp. 643–659. [[CrossRef](#)]
41. Mubarak, A.; Said, Z.; Aliyu, R.; Al Turjman, F.; Serte, S.; Ozsoz, M. Deep learning-based feature extraction coupled with multi-class SVM for COVID-19 detection in the IoT era. *Int. J. Nanotechnol.* **2021**, *1*, 1. [[CrossRef](#)]
42. Mubarak, A.S.; Serte, S.; Al-Turjman, F.; Ameen, Z.S.; Ozsoz, M. Local binary pattern and deep learning feature extraction fusion for COVID-19 detection on computed tomography images. *Expert Syst.* **2021**, *39*, e12842. [[CrossRef](#)]

43. Haque, A.B.; Rahman, M. Augmented COVID-19 X-ray Images Dataset (Mendely) Analysis using Convolutional Neural Network and Transfer Learning. *2020*, *19*. Available online: [https://www.researchgate.net/publication/340514197\\_Augmented\\_COVID-19\\_X-ray\\_Images\\_Dataset\\_Mendely\\_Analysis\\_using\\_Convolutional\\_Neural\\_Network\\_and\\_Transfer\\_Learning?channel=doi&linkId=5e8e0a9592851c2f5288a56e&showFulltext=true](https://www.researchgate.net/publication/340514197_Augmented_COVID-19_X-ray_Images_Dataset_Mendely_Analysis_using_Convolutional_Neural_Network_and_Transfer_Learning?channel=doi&linkId=5e8e0a9592851c2f5288a56e&showFulltext=true) (accessed on 21 June 2022). [[CrossRef](#)]
44. Apostolopoulos, I.D.; Mpesiana, T.A. Covid-19: Automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. *Phys. Eng. Sci. Med.* **2020**, *43*, 635–640. [[CrossRef](#)]
45. Hussein, S.; Kandel, P.; Bolan, C.W.; Wallace, M.B.; Bagci, U. Lung and Pancreatic Tumor Characterization in the Deep Learning Era: Novel Supervised and Unsupervised Learning Approaches. *IEEE Trans. Med. Imaging* **2019**, *38*, 1777–1787. [[CrossRef](#)]
46. Loey, M.; Smarandache, F.; Khalifa, M.N.E. Within the Lack of Chest COVID-19 X-ray Dataset: A Novel Detection Model Based on GAN and Deep Transfer Learning. *Symmetry* **2020**, *12*, 651. [[CrossRef](#)]
47. Mahmud, T.; Rahman, M.A.; Fattah, S.A. CovXNet: A multi-dilation convolutional neural network for automatic COVID-19 and other pneumonia detection from chest X-ray images with transferable multi-receptive feature optimization. *Comput. Biol. Med.* **2020**, *122*, 103869. [[CrossRef](#)] [[PubMed](#)]
48. Wang, M.; Zheng, S.; Li, X.; Qin, X. A new image denoising method based on Gaussian filter. In Proceedings of the 2014 International Conference on Information Science, Electronics and Electrical Engineering, Sapporo, Japan, 26–28 April 2014; IEEE: Manhattan, NY, USA, 2014; Volume 1, pp. 163–167. [[CrossRef](#)]
49. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; IEEE: Manhattan, NY, USA, 2020; pp. 10778–10787. [[CrossRef](#)]
50. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; IEEE: Manhattan, NY, USA, 2019; pp. 6568–6577.
51. Howard, A.G.; Wang, W. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861.
52. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; IEEE: Manhattan, NY, USA, 2017; pp. 2961–2969. [[CrossRef](#)]