big data and cognitive computing

MDPI

Article

An Efficient Multi-Scale Anchor Box Approach to Detect Partial Faces from a Video Sequence

Dweepna Garg [1], Priyanka Jain [2], Ketan Kotecha [3,*], Parth Goel [4] and Vijayakumar Varadarajan [5]

[1] Department of Computer Engineering, Devang Patel Institute of Advance Technology and Research (DEPSTAR), Faculty of Technology and Engineering (FTE), Charotar University of Science and Technology (CHARUSAT), Nadiad-Petlad Road, Highway, Changa 388421, India; dweeps1989@gmail.com
[2] Artificial Intelligence Group, Centre for Development of Advanced Computing, Jasola Vihar, New Delhi 110025, India; priyankaj@cdac.in
[3] Symbiosis Centre for Applied Artificial Intelligence, Symbiosis International (Deemed University), Sena Pati Bapat Road, Pune 411004, India
[4] Department of Computer Science & Engineering, Devang Patel Institute of Advance Technology and Research (DEPSTAR), Faculty of Technology and Engineering (FTE), Charotar University of Science and Technology (CHARUSAT), Nadiad-Petlad Road, Highway, Changa 388421, India; er.parthgoel@gmail.com
[5] School of Computer Science and Engineering, The University of New South Wales, Sydney 1466, Australia; vijayakumar.varadarajan@gmail.com
* Correspondence: head@scaai.siu.edu.in

Abstract: In recent years, face detection has achieved considerable attention in the field of computer vision using traditional machine learning techniques and deep learning techniques. Deep learning is used to build the most recent and powerful face detection algorithms. However, partial face detection still remains to achieve remarkable performance. Partial faces are occluded due to hair, hat, glasses, hands, mobile phones, and side-angle-captured images. Fewer facial features can be identified from such images. In this paper, we present a deep convolutional neural network face detection method using the anchor boxes section strategy. We limited the number of anchor boxes and scales and chose only relevant to the face shape. The proposed model was trained and tested on a popular and challenging face detection benchmark dataset, i.e., Face Detection Dataset and Benchmark (FDDB), and can also detect partially covered faces with better accuracy and precision. Extensive experiments were performed, with evaluation metrics including accuracy, precision, recall, F1 score, inference time, and FPS. The results show that the proposed model is able to detect the face in the image, including occluded features, more precisely than other state-of-the-art approaches, achieving 94.8% accuracy and 98.7% precision on the FDDB dataset at 21 frames per second (FPS).

Keywords: face detection; partial face detection; occluded face detection; deep learning; convolution neural network; FDDB dataset

Citation: Garg, D.; Jain, P.; Kotecha, K.; Goel, P.; Varadarajan, V. An Efficient Multi-Scale Anchor Box Approach to Detect Partial Faces from a Video Sequence. *Big Data Cogn. Comput.* **2022**, *6*, 9. https://doi.org/10.3390/bdcc6010009

Academic Editor: Min Chen

Received: 16 November 2021
Accepted: 7 January 2022
Published: 11 January 2022

1. Introduction

In computer vision, face detection has been a major focus for many years. The main aim of face detection systems is to locate the face in the image, with its bounding box. Face detection has been included in the prior work of some important applications such as face recognition, face analysis, face mask detection, face tracking, and face alignment. Viola-Jones performed the pioneering face detection work by proposing the haar-cascading, feature extraction method [1]. Big data and high-performance computing systems have helped deep learning to achieve remarkable results in many applications, including natural language processing, manufacturing, computer vision, healthcare, and speech recognition. Deep convolutional neural network (DCNN)-based methods have proven to be more effective than conventional methods for object detection. As a result, researchers have started applying DCNN methods for face detection.

Face detection approaches based on deep learning can be divided into two groups: two-stage object detectors and single-stage object detectors. The two-stage object detectors include Faster R-CNN [2], and single-stage object detectors mainly include YOLO [3] and SSD [4]. In two-stage object identification systems, region proposals are generated in the first stage, and then, they are used to recognize the object and to find the coordinates of the bounding box. Though these models are slower than single-stage models, they have proven to show better accuracy rates. On the other hand, single-stage anchor-based techniques [5–7] use regular and dense anchors over a wide range of scales and aspect ratios to identify faces. In these approaches, face detection with a bounding box is performed in the single pipeline of the network.

Single-stage face detection models become more difficult to enhance performance, especially in cases of partial or occluded faces. Detecting faces in real-world images has many difficulties due to the partial features in occluded faces. Partial features of faces in images result from some parts of the face being hidden, e.g., with glasses, masks, hair, scarves, some part of the body, cap, and side angles. Real-life examples of partial faces are presented in Figure 1.

Figure 1. Examples of partial faces.

Recent DCNN models for face detection uses image pyramid concepts to extract fine-grained features of faces [8–10]. First, an image pyramid is developed, and then, it passes to the next layers of the network. The computation cost of this network is high due to the complex training. Thus, detection speed from the video is reduced. These models were successful in achieving better results whenever the face was entirely visible. On the other hand, if the face was partially covered, these models did not show better accuracy, as the features could not be accurately captured and utilized fully in the network.

To overcome this problem, in this study, we utilized a single-stage deep convolution neural network for a face detection pipeline with anchor boxes. The initial stages of the face detection pipeline are convolutional layers with max pooling, to extract features of faces. Then, these features are passed to the anchor box layer. We used eight anchor boxes and two scales instead of three for face detection, to reduce the unnecessary anchor boxes, compared with regular object detection methods. In order to lessen the computational complexity, and to detect partial faces from images or videos, we employed a special anchor box selection strategy.

The main contributions of this paper are summarized as follows:

i. We proposed a deep convolutional neural network for partial face detection using the anchor box selection strategy on the FDDB dataset;
ii. We utilized the class existence probability of anchor proposals to classify the partial features of faces;

iii. We considered eight anchor boxes and two scales to avoid extra computations. Anchor boxes sizes and scales were chosen from facial subparts and shapes;
iv. The proposed method shows the balance performance between precision and detection speed;
v. The proposed method examined the FDDB dataset with partial face examples, and the results were compared with the other state-of-the-art face detection methods.

The paper is organized as follows: Section 2 provides an overview of the related methods. Section 3 explains the methodology. Extensive experimental analysis is presented in Section 4. Finally, Section 5 concludes the paper and shows future directions.

2. Related Work

Partial face detection is one of the problems in object detection, as it requires the location of each part of the face in the image. The face is a biometric human trait that contains vital information about an individual's identification. As a result, precise face detection is the first stage in various applications such as face recognition, face picture retrieval, facial tracking, and video surveillance. The detected face can be seen with the help of a bounding box. The face detection must be robust. Faces must be able to be detected even though they might be using different conditions including different angles, lighting conditions, makeup, age, having glasses on, hats, etc. Face detection is accomplished using two approaches: handmade-based face detectors and deep learning-based face detectors.

2.1. Handcrafted Face Detection Methods

Handcrafted face detection techniques were utilized in a wide range of computer vision applications. Pioneering work in the face detection field is by Viola-Jones. Classical face detection approaches are effective in real-time performance; however, detections are not robust in all conditions. Histograms of oriented gradients (HOGs) [11] and local binary patterns (LBPs) [12] are used as feature extraction methods for face detection and have shown promising outcomes [13,14].

A taxonomy of face identification methods is presented in [15], in which it was divided into two primary classes—namely, feature-based and image-based approaches. Feature-based approaches are applicable when motion and color are present in images or videos and can provide visual cues to focus attention in situations when the multi-resolution window cannot be scanned. By contrast, image-based approaches are the most suitable for images with greyscale. All of these techniques are computationally costly since they depend on the scanning of multi-resolution windows to locate faces of all sizes. The deformable part model (DPM) is also a promising face detection method [16]. This approach utilizes the relationship of deformable facial parts to detect faces. However, face detectors that use classical machine learning have not shown efficient performance in complex situations.

2.2. Deep-Learning-Based Face Detectors

CNN-based face detectors have attained the highest performance in the field of face detection using deep learning techniques. It is possible to attain high accuracy and efficiency concurrently by training a sequence of CNN models using the cascade CNN approach [17,18]. Yu et al. proposed an intersection-over-union (IoU) loss to decrease the gap between the IoUs and annotations, improving location accuracy [19]. Hao et al. focused on the detection of normalized faces by applying a scale proposal stage within a network to zoom in and out of the input image [20]. Yang et al. scored the facial parts according to their spatial structure, in order to detect faces in obstructive and uncontrolled conditions [21]. The decision tree classification approach is used to detect faces in the LDCF+ system [22]. Hu et al. sought to detect small faces with various scales [23]. Shi et al. applied a cascade-style structure to detect rotated faces [24]. Spatial attention modules with specific scales were employed by Song et al. to estimate face locations in images [25]. Many state-of-the-art face detectors are implemented from generic object detection methods. Face R-CNN [26], Face R-FCN [27], and FDNet [28] apply two-stage object detection methods

such as Faster R-CNN and R-FCN [29], with some specific strategies to perform face detection. Tian et al. proposed an improved feature fusion pyramid with segmentation to detect hard faces [30]. Wang et al. addressed the RetinaNet [31] using the attention mechanism in anchor boxes [32]. Zhang et al. combined the features of higher level and lower level to detect faces in various conditions [33]. However, performance suffered due to aggregation. Zhang et al. utilized the SRN detector [34], with attention technique and DensNet-121 as the backbone of the proposed model [35]. Small faces were detected by extending the FPN module with a receptive module in DEFace [36]. Feature hierarchy encoder–decoder network (FHEDN) is a single-stage detection network and applies context prior features of faces [37]. Li et al. employed a pure convolutional neural network without anchors boxes for face detection [38].

3. Methodology

In this section, the proposed methodology is explained. The main aim of the proposed work was to detect partially covered faces of varying sizes. The proposed network was developed with a single-stage, end-to-end network. A batch of randomly occluded and occlusion-free face images from the FDDB dataset was taken as input, and features were generated, after which were utilized to decode the features of the faces.

3.1. Proposed Work

In our proposed work, we utilized 22 convolution layers and 5 max-pooling layers. The proposed work is presented in Figure 2. The proposed work pipeline was divided into two parts—namely, feature extraction layers and object detection. In the feature extraction pipeline, the features of the input image were extracted using the first 16 convolution layers. An input image resolution was kept at 608 × 608 pixels for training. In each convolution layer, 3 × 3 and 1 × 1 kernel sizes were used. Additionally, a 2 × 2 max-pooling layer was applied at the end of each convolution layer to downsample the image and keep important features of faces. After each pooling phase, the number of 3 × 3 and 1 × 1 filters was increased by two.

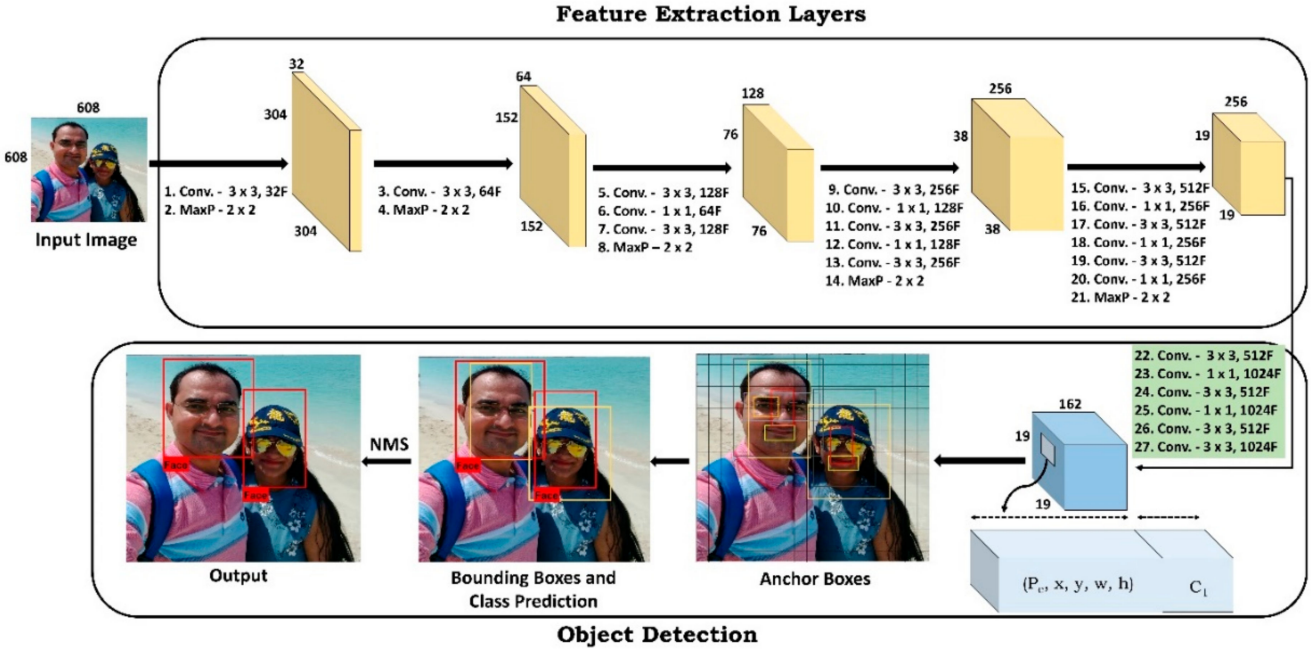

Figure 2. Proposed face detection pipeline.

In order to detect faces, the remaining six convolution layers were used in the second part of the proposed work pipeline. The anchor box's sizes are critical for end-to-end detection methods. When using general, fixed-sized boxes, anchors are made to detect

objects of various sizes. However, these general-sized boxes may not necessarily work for other object detection tasks such as face detection. Bounding boxes of faces are mainly in the size of square or vertical rectangles. Some partially covered faces were challenging to detect and distinguish. Thus, creating an anchor box whose sizes closely match the ground truth is preferable in terms of improving detection accuracy. As a result, a multi-scale anchor box was used in the proposed work. A 19 × 19 grid was employed on the features of the input image. We utilized 8 different sizes of anchor boxes, which are depicted in Figure 3 with an example. The sizes of the anchor boxes were (32, 32), (78, 88), (94, 40), (128, 128), (172, 210), (300, 100), (284, 334), and (512, 512). These anchor boxes were selected based on the input image resolution size and facial parts, as shown in Figure 4. Our model detected the smallest face with the size of 32 × 32 because the smallest size of the anchor box was 32 × 32. Generally, faces are in shapes of a square and vertical rectangle; thus, we used two scales of anchor boxes of 1:1 and 2:1 to reflect square and vertical rectangle shapes. The 1:2 scale was not considered, as the shape of faces cannot be a horizontal rectangle. This is presented in Figure 5. The number of proposals per grid was calculated from Equation (1). In our work, eight anchor boxes, two scales, one the class existence probability (Pc), four bounding box coordinates, and one class were considered, and according to Equation (1), 96 proposals were generated per grid in the final convolution layer. In order to reduce the unnecessary computation of bounding boxes, the concept of a selected multi-scale anchor box was introduced. The sizes and scales of the anchor boxes were varied based on the features and shapes of the faces.

$$\#\text{proposals per grid} = (\#\text{anchor boxes} \times \#\text{scales}) \times (\text{Pc} + \#\text{coordinates} + \#\text{class}) \quad (1)$$

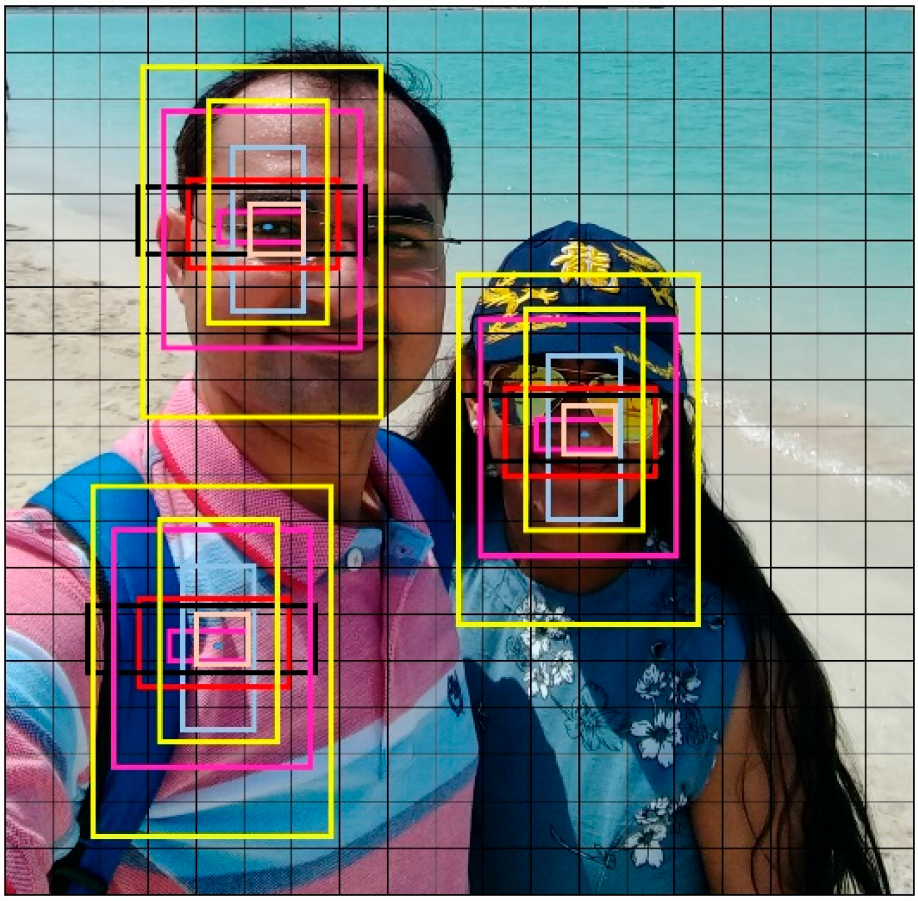

Figure 3. Samples of eight anchor boxes per grid in 19 × 19 grid of input image.

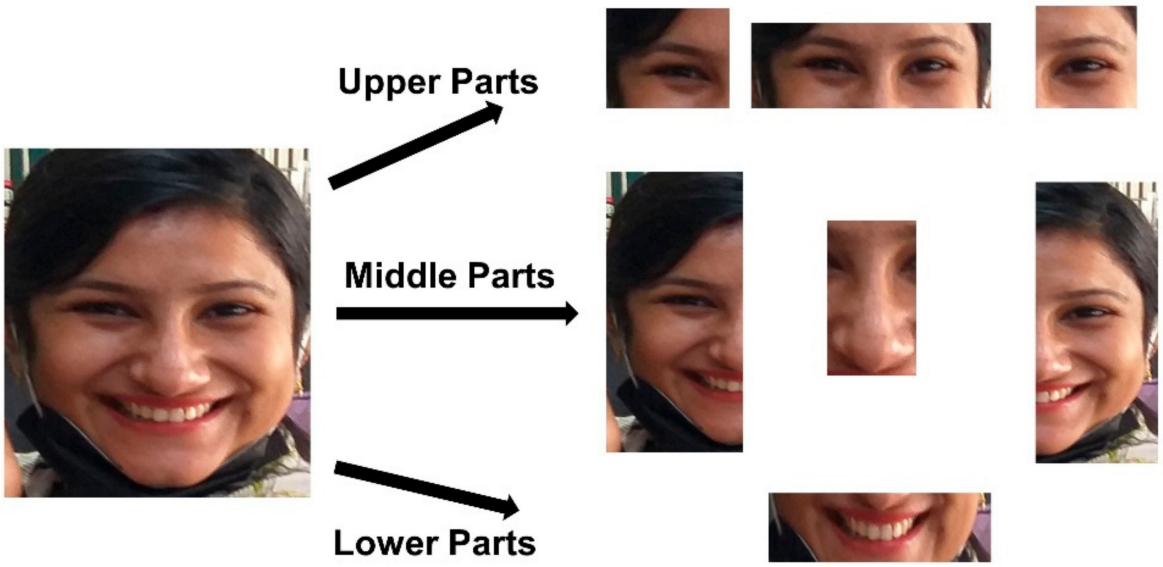

Figure 4. Facial parts having important features to predict the face.

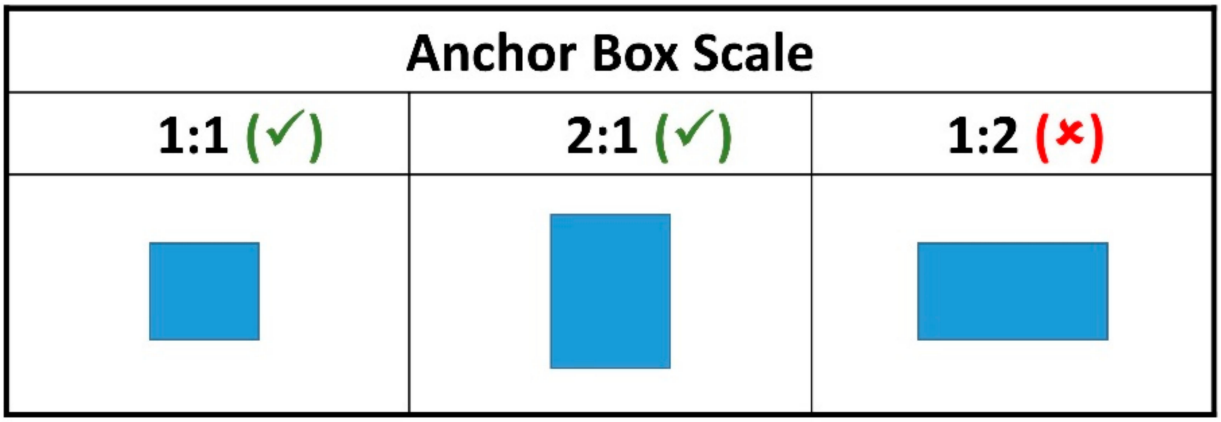

Figure 5. Types of anchor boxes scale.

The bounding box of each face object was predicted using regression on each proposal. An IoU of 0.40 was applied to select the right overlapping bounding box, with the highest probability from a non-max suppression technique.

3.2. Anchor Box Selection Approach

Anchor boxes were utilized for partial face detection and provided the feature maps for the final convolution layer. In order to detect partially covered faces, it is not necessary that all of the anchor boxes contain enough information; nonetheless, the prediction score for each anchor box was calculated, which resulted in increased execution time (training and testing). As a result, real-time detection became time consuming and decreased the frame rate. An effective strategy for selecting anchor boxes was employed to mitigate this problem. Usually, the important feature information is not present in each anchor box; thus, such anchor boxes can be avoided for further processing in the pipeline. The strategy to avoid unnecessary anchor boxes is that the anchor boxes are arranged in descending order per grid. Then, in the absence of relevant information in a large size anchor box, the small-sized anchor boxes were ignored within the same grid. Canny edge detection algorithm [39] was applied to validate the information presented in the given anchor boxes. At the end of the convolution block, this strategy was used, which resulted in a significant reduction in both computational and memory expenses.

Another strategy for partial face detection was applied by using the anchor box class existing probability (Pc). In our work, the anchor boxes comprised only faces, and features of faces are depicted in Figure 4. The face is divided into main three parts—upper parts, middle parts, and lower parts. The upper parts consist of the eyebrows, eyes, and forehead. The middle parts consist of the nose, left and right cheek, along with eyes, and half of the forehead. The lower parts primarily consist of the lips and chin. In any of the anchor boxes in which partial features of the face are present, Pc is higher, compared with other anchor boxes under occluded conditions. During the detection of partial features of faces, the difference between the location of the bounding box and anchor box was large, compared with regular object bounding boxes. Small anchor boxes may ignore certain ground truth boxes if the distance between them is large. Thus, the IoU threshold was reduced from 0.5 to 0.4, in order to alleviate this problem.

4. Experimental Analysis

In this section, the performance of the proposed work is evaluated with extensive experiments on the FDDB dataset.

4.1. Dataset

The Face Detection Dataset and Benchmark (FDDB)g [40] comprises annotated faces and is a subset of the Faces in the Wild dataset, including greyscale and color images. The FDDB images were obtained from Yahoo! News. This dataset comprises a collection of 2845 images with 5171 face annotations, with different resolutions. Images from this dataset contain various challenges, notably, faces with side angles, multiple expressions, scale, illumination, and occlusion. Samples of the FDDB dataset are depicted in Figure 6. In the FDDB dataset, faces were annotated in elliptical areas, which were converted into rectangles or square areas before training the model.

Figure 6. Samples of the FDDB dataset with variations in pose, expression, scale, illumination, and occlusion.

4.2. Evaluation Matrices

Accuracy, precision, recall, and F1 score were utilized to evaluate the performance of the proposed work. These evaluation matrices were calculated using true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs). Accuracy, precision, recall, and F1 score were computed using Equations (2)–(5), respectively. TPs indicate correct face detection. FNs represent incorrect face detection, such as misidentifying a background as the face, whereas FPs show that incorrect face detection is the background of the face.

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{FN} + \text{TN}) \quad (2)$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (3)$$

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \quad (4)$$

$$\text{F1-Score} = (2 \times \text{precision} \times \text{recall})/(\text{precision} + \text{recall}) \quad (5)$$

4.3. Experimental Setup

Experiments were performed on the proposed model using a machine with 32 GB RAM, 2 TB hard disk, Intel core i7 8th generation processor, NVIDIA Titan Xp 12 GB GPU,

and 64-bit Windows operating system. Experiments were performed using python 3.7 programming language with OpenCV, Keras, and TensorFlow libraries.

4.4. Discussion

In our experiments, we considered an 80:20 training and testing dataset split. The model was trained for 50 epochs, with 128 batch sizes, 0.0001 initial learning rate, and 0.9 momenta.

Figure 7 shows the performance of the proposed work with state-of-the-art face detection algorithms [41,42]. We compared our method with other methods in terms of average precision (AP), which targets the detection of the faces from the FDDB dataset. As a result, the proposed method outperformed other methods, with 98.7% accuracy. From this finding, we could confirm that the proposed work is suitable and more accurate to detect partially covered faces.

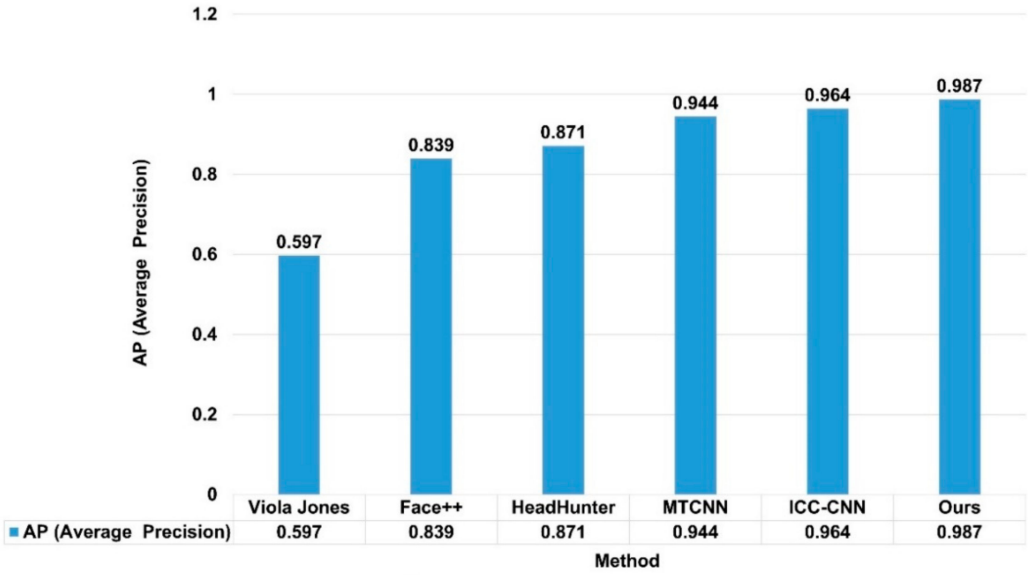

Figure 7. Result analysis of proposed work with other state-of-the-art face detectors in terms of AP.

Table 1 shows the result and performance analysis of the proposed work with the other state-of-the-art object detection methods on various image resolutions. We compared our method with other single-stage and two-stage object detection approaches mentioned in the above table, with accuracy, precision, recall, F1 score, and inference time evaluation matrices. Our method outperformed in accuracy, precision, and F1 score in all resolutions. However, recall of the proposed method in resolutions of 416 × 416 and 480 × 480 is 91.6% and 94.8%, which are less, compared with Faster R-CNN (91.8%) and YOLOv4 (95.3%), respectively. It can also be seen that there is not much difference in recall values. Results from the table show that the proposed methods maintain the accuracy, precision, and recall value along with the F1 score. Out of three resolutions, our model achieved the highest performance in 608 × 608 resolution. We analyzed the inference time on the test images of the FDDB dataset and executed Nvidia Titan Xp GPU. Results of inference time show that YOLOv1 attained the minimum inference time in ms. However, our model also performed well and reported the second-lowest inference time. YOLOv1 is a single-stage object detection method and has only a 7 × 7 grid size and two anchor boxes per grid. In contrast, our work has a 19 × 19 grid size and eight anchor boxes. However, we introduced an anchor selection strategy to reduce the computation time, and this is evident from the results of inference time, compared with other detection algorithms except for YOLOv1. Faster R-CNN took the highest inference time because it is a two-stage object detection algorithm.

Table 1. Experiment analysis of the proposed work with other object detectors on the FDDB dataset.

Resolution	Method	Accuracy	Precision	Recall	F1-Score	Inference Time (ms)
416 × 416	Yolov1	71.4	80.2	83.4	81.77	29.41
	Yolov2	73.5	84.3	82.5	83.39	32.47
	Yolov3	74.8	86.7	85.1	85.89	41.67
	Faster RCNN	79.45	90.6	91.8	91.20	55.56
	Yolov4	79.6	91.2	88.4	89.78	34.48
	Ours	80.4	94.8	91.6	93.17	31.25
480 × 480	Yolov1	76.8	84.8	85.4	85.10	35.71
	Yolov2	78.9	88.7	86.2	87.43	40.49
	Yolov3	79.8	90.3	88.7	89.49	47.62
	Faster RCNN	84.96	94.3	92.2	93.24	83.33
	Yolov4	85.3	94.9	95.3	95.10	43.48
	Ours	86.7	97.2	94.8	95.99	37.74
608 × 608	Yolov1	80.1	88.9	86.5	87.68	40.00
	Yolov2	86.3	90.7	89.8	90.25	50.00
	Yolov3	89.4	93.2	91.3	92.24	58.82
	Faster RCNN	92.4	97.2	92.3	94.69	111.11
	Yolov4	93.7	96.1	95.2	95.65	52.63
	Ours	94.8	98.7	97.8	98.25	47.62

Table 2 shows the performance analysis of our method by taking different IoU thresholds. We achieved 98.7% AP at 0.4 IoU. This IoU was selected to detect partial features of faces. Comparatively, space occupied by faces in images is small in real-life captured images. When partial features of faces are detected, the distance between the bounding box and anchor box locations is substantial in comparison with conventional object bounding boxes. If the distance between two small anchor boxes is large, they may ignore certain ground truth boxes. Thus the performance of the proposed work outperformed by accurately detecting fully visible faces as well as partially visible faces. Detection results can be seen in Figure 8.

Table 2. Proposed work performance (AP) at different IoU thresholds on the FDDB dataset.

IoU	0.4	0.5	0.6
Ours	98.7	98.1	95.4

Figure 8 presents the detection results of the proposed method on the FDDB dataset in the first row (a); our samples in the second row (b); samples of the MAFA dataset [43] in the last row (c). The images presented in Figure 1 were fed into the model, and the detection results are shown in the second row of Figure 8. These sample images of the second row and samples from the MAFA dataset were tested to show the robustness of the proposed method. These samples were provided neither during the training nor testing phases of the model, but rather from other distributions. Detection results of Figure 8 also illustrate that the model is able to detect the face in various conditions including different poses, occlusion due to mask, scarf, hands, mobile phone, and blurred faces.

We assessed the real-time computational speed of the proposed network trained on the FDDB dataset on an NVIDIA Titan Xp with 12 GB GPU. We analyzed the running speed of our method with other state-of-the-art object detection methods, and the results

are presented in Figure 9. We utilized a real-time video from a web camera as an input to evaluate the real-time performance and compared the proposed model with other detection approaches. It can be seen that our method obtained the second-highest frame per second (FPS) on all resolutions, due to the anchor boxes selection strategy. YOLOv1 achieved the highest FPS due to less computation, and Faster R-CNN attained the lowest FPS because of the two stages of the network.

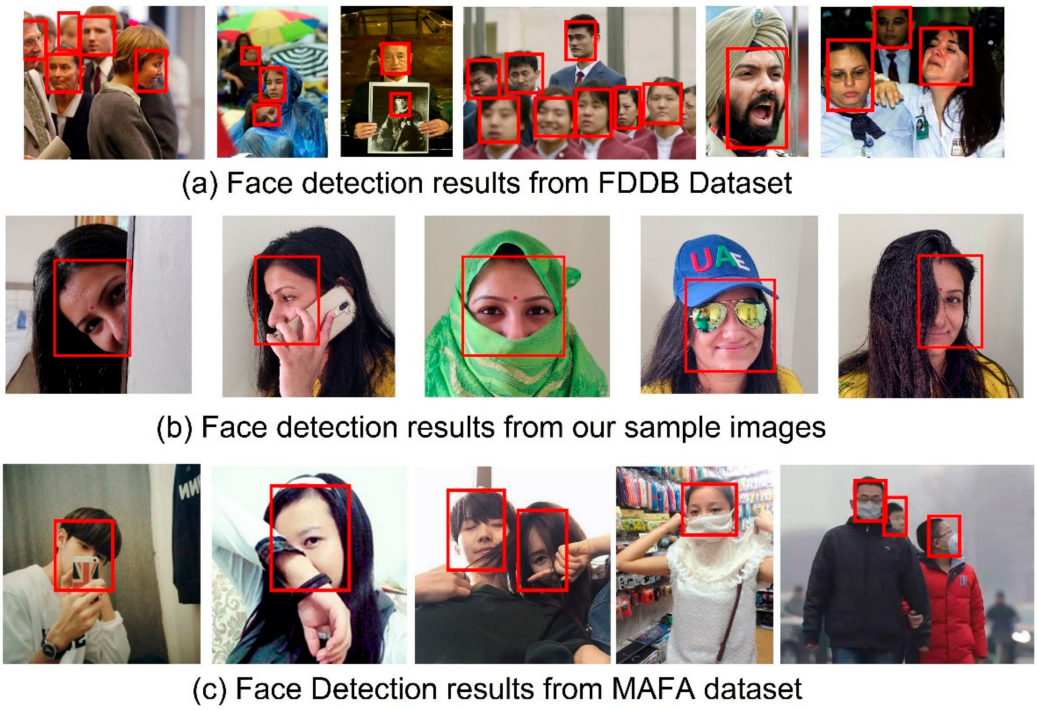

Figure 8. Face detection results of the proposed method.

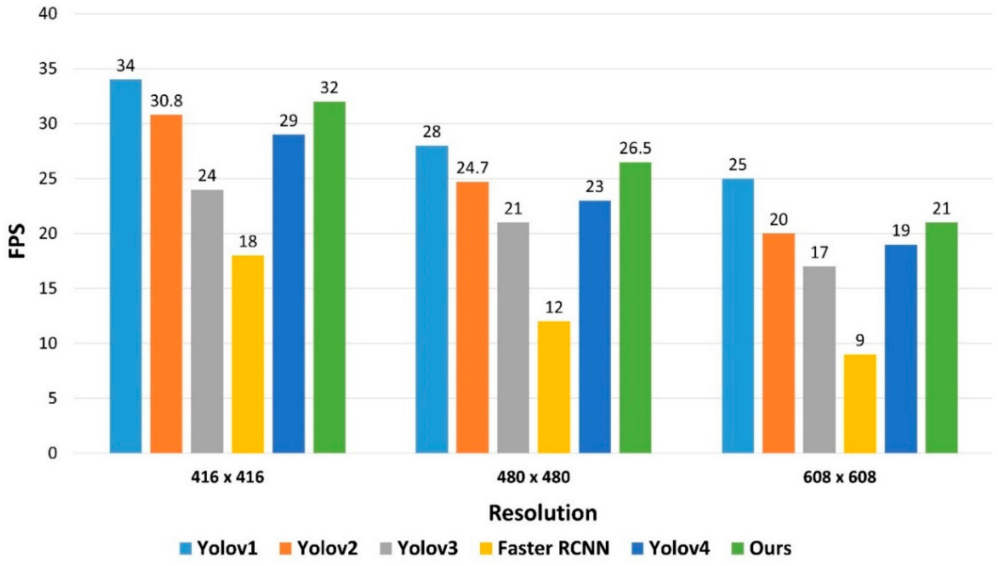

Figure 9. Comparative analysis of FPS of proposed method with existing DCNN detection approaches.

5. Conclusions

In this paper, a single-stage, deep convolution neural network-based face detection method was addressed for detecting partial faces with different occlusions. We applied an anchor box selection strategy to reduce the computation time. Furthermore, we also

utilized class existence probability to identify parts of faces from small-sized anchor boxes. Additionally, we investigated the effect of implementation factors such as IoU on detection performance. Finally, we compared the average precision (AP) of the proposed work with other face detection algorithms using the FDDB dataset. Our network was also assessed with existing object detection models considering the FDDB dataset with different resolutions. The experimental findings show that our model exhibited impressive results, with 98.7% precision, and had a better inference speed. The proposed model was also evaluated on video and attained 21 FPS, 26.5 FPS, and 32 FPS on 608 × 608, 480 × 480, and 416 × 416 resolutions, respectively.

Our proposed method shows a balance between accuracy and speed. In future work, this model can be utilized as a base to detect faces for various applications in real life for which face detection is the first phase such as security monitoring, face recognition, face mask detection, forensic investigations, attendance monitoring, and face tracking.

Author Contributions: Conceptualization, D.G., P.J., P.G. and K.K.; methodology, D.G., P.J. and P.G.; validation, K.K. and V.V.; formal analysis, V.V.; investigation, D.G., P.J. and K.K.; resources, K.K. and V.V.; data curation, D.G. and P.G.; writing—original draft preparation, D.G., P.J., P.G. and K.K.; writing—review and editing, D.G., P.G. and K.K.; visualization, D.G. and P.J.; supervision, K.K. and V.V.; funding acquisition, V.V. All authors have read and agreed to the published version of the manuscript.

Funding: Fund is provided by Symbiosis International University, Pune, India.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: CKAN is a complete out-of-the-box software solution that makes data accessible and usable—by providing tools to streamline publishing, sharing, finding and using data (including storage of data and provision of robust data APIs). CKAN is aimed at data publishers (national and regional governments, companies and organizations) wanting to make their data open and available. CKAN is used by governments and user groups worldwide and powers a variety of official and community data portals including portals for local, national and international government, such as the UK's data.gov.uk and the European Union's publicdata.eu, the Brazilian dados.gov.br, Dutch and Netherland government portals, as well as city and municipal sites in the US, UK, Argentina, Finland and elsewhere.

Acknowledgments: We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

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

References

1. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
2. Faster, R. Towards real-time object detection with region proposal networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2015; pp. 91–99.
3. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
4. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016*; Springer: Cham, Switzerland, 2016; pp. 21–37.
5. Najibi, M.; Samangouei, P.; Chellappa, R.; Davis, L.S. Ssh: Single stage headless face detector. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4875–4884.
6. Tang, X.; Du, D.K.; He, Z.; Liu, J. Pyramidbox: A context-assisted single shot face detector. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 797–813.
7. Li, J.; Wang, Y.; Wang, C.; Tai, Y.; Qian, J.; Yang, J.; Wang, C.; Li, J.; Huang, F. DSFD: Dual shot face detector. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5060–5069.
8. Zeng, D.; Liu, H.; Zhao, F.; Ge, S.; Shen, W.; Zhang, Z. Proposal pyramid networks for fast face detection. *Inf. Sci.* **2019**, *495*, 136–149. [CrossRef]

9. Luo, J.; Liu, J.; Lin, J.; Wang, Z. A lightweight face detector by integrating the convolutional neural network with the image pyramid. *Pattern Recognit. Lett.* **2020**, *133*, 180–187. [CrossRef]
10. Hou, S.; Fang, D.; Pan, Y.; Li, Y.; Yin, G. Hybrid Pyramid Convolutional Network for Multiscale Face Detection. *Comput. Intell. Neurosci.* **2021**, *2021*, 9963322. [CrossRef] [PubMed]
11. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
12. Ahonen, T.; Hadid, A.; Pietikainen, M. Face description with local binary patterns: Application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 2037–2041. [CrossRef] [PubMed]
13. Adouani, A.; Henia, W.M.B.; Lachiri, Z. Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences. In Proceedings of the 2019 16th International Multi-Conference on Systems, Signals & Devices (SSD), Istanbul, Turkey, 21–24 March 2019; pp. 266–271.
14. Roy, A.; Marcel, S. Haar local binary pattern feature for fast illumination invariant face detection. In Proceedings of the British Machine Vision Conference 2009, London, UK, 7–10 September 2009. number CONF.
15. Hjelmås, E.; Low, B.K. Face detection: A survey. *Comput. Vis. Image Underst.* **2001**, *83*, 236–274. [CrossRef]
16. Mathias, M.; Benenson, R.; Pedersoli, M.; Van Gool, L. Face detection without bells and whistles. In *Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014*; Springer: Cham, Switzerland, 2014; pp. 720–735.
17. Li, H.; Lin, Z.; Shen, X.; Brandt, J.; Hua, G. A convolutional neural network cascade for face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5325–5334.
18. Qin, H.; Yan, J.; Li, X.; Hu, X. Joint training of cascaded CNN for face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3456–3465.
19. Yu, J.; Jiang, Y.; Wang, Z.; Cao, Z.; Huang, T. Unitbox: An advanced object detection network. In Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15–19 October 2016; pp. 516–520.
20. Hao, Z.; Liu, Y.; Qin, H.; Yan, J.; Li, X.; Hu, X. Scale-aware face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6186–6195.
21. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. From facial parts responses to face detection: A deep learning approach. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 3676–3684.
22. Ohn-Bar, E.; Trivedi, M.M. To boost or not to boost? on the limits of boosted trees for object detection. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 3350–3355.
23. Hu, P.; Ramanan, D. Finding tiny faces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 951–959.
24. Shi, X.; Shan, S.; Kan, M.; Wu, S.; Chen, X. Real-time rotation-invariant face detection with progressive calibration networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 2295–2303.
25. Song, G.; Liu, Y.; Jiang, M.; Wang, Y.; Yan, J.; Leng, B. Beyond trade-off: Accelerate fcn-based face detector with higher accuracy. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7756–7764.
26. Wang, H.; Li, Z.; Ji, X.; Wang, Y. Face r-cnn. *arXiv* **2017**, arXiv:1706.01061 2017.
27. Wang, Y.; Ji, X.; Zhou, Z.; Wang, H.; Li, Z. Detecting faces using region-based fully convolutional networks. *arXiv* **2017**, arXiv:1709.05256 2017.
28. Zhang, C.; Xu, X.; Tu, D. Face detection using improved faster rcnn. *arXiv* **2018**, arXiv:1802.02142 2018.
29. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in Neural Information Processing Systems*; Curran Associates Inc.: New York, NY, USA, 2016; pp. 379–387.
30. Tian, W.; Wang, Z.; Shen, H.; Deng, W.; Meng, Y.; Chen, B.; Zhang, X.; Zhao, Y.; Huang, X. Learning better features for face detection with feature fusion and segmentation supervision. *arXiv* **2018**, arXiv:1811.08557 2018.
31. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
32. Wang, J.; Yuan, Y.; Yu, G. Face attention network: An effective face detector for the occluded faces. *arXiv* **2017**, arXiv:1711.07246.
33. Zhang, J.; Wu, X.; Hoi, S.C.; Zhu, J. Feature agglomeration networks for single stage face detection. *Neurocomputing* **2020**, *380*, 180–189. [CrossRef]
34. Chi, C.; Zhang, S.; Xing, J.; Lei, Z.; Li, S.Z.; Zou, X. Selective refinement network for high performance face detection. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 8231–8238.
35. Zhang, Y.; Xu, X.; Liu, X. Robust and high performance face detector. *arXiv* **2019**, arXiv:1901.02350 2019.
36. Hoang, T.M.; Nam, G.P.; Cho, J.; Kim, I.J. Deface: Deep efficient face network for small scale variations. *IEEE Access* **2020**, *8*, 142423–142433. [CrossRef]
37. Zhou, Z.; He, Z.; Jia, Y.; Du, J.; Wang, L.; Chen, Z. Context prior-based with residual learning for face detection: A deep convolutional encoder–decoder network. *Signal Processing Image Commun.* **2020**, *88*, 115948. [CrossRef]
38. Li, X.; Lai, S.; Qian, X. DBCFace: Towards Pure Convolutional Neural Network Face Detection. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, 1. [CrossRef]

39. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6569–6578.
40. Jain, V.; Learned-Miller, E. *Fddb: A Benchmark for Face Detection in Unconstrained Settings*; Technical Report, UMass Amherst Technical Report; University of Massachusetts: Amherst, MA, USA, 2010.
41. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
42. Zhang, K.; Zhang, Z.; Wang, H.; Li, Z.; Qiao, Y.; Liu, W. Detecting faces using inside cascaded contextual cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3171–3179.
43. Ge, S.; Li, J.; Ye, Q.; Luo, Z. Detecting masked faces in the wild with lle-cnns. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2682–2690.