High Speed Single-Stage Face Detector using Depthwise Convolution and Receptive Fields

Rahul Yadav[1], Priyanka[2]
ECE Department[1,2]
DCR University of Science and Technology,
Murthal, Sonepat, India, 131039
ORCID ID: 0000-0003-2542-112X[1]

Priyanka Kacker[3]
Institute of Behavioural Science
National Forensic Sciences University,
Gandhinagar, Gujarat, India, 382007

Abstract—**At present face detectors use a large Convolutional Neural Network (CNN) to achieve high detection performance, which is a widely used sub-area of artificial intelligence. These face detectors have a large number of parameters which reduces their detection speed dreadfully on a system with low computational resources. This is a challenging problem to achieve good performance and high detection speed with finite computational power. In this paper, we propose a single-stage end-to-end trained face detector to address this challenging problem. The computational cost is reduced by using depthwise convolution and swiftly reducing the size of an input image. The early layers of the model use CReLU (Concatenated Rectified Linear Unit) activations to preserve the information and generate better representative features of the input. Respective Field (RF) blocks used in the model improve the detection performance. The proposed model is of 1.7 Megabytes size, able to achieve 42 FPS (Frame Per Second) on CPU (i5-8330H) and 179 FPS on GPU (GTX1060). The model is evaluated on various benchmark datasets like WIDER FACE, PASCAL faces and AFW and archive good performance compared to other state of art methods.**

Keywords—Artificial intelligence; computer-vision; Convolutional Neural Network (CNN); face detector

I. Introduction

Face detection is defined as the problem of detecting and localizing faces in a given image. It is a basic and long-standing problem of active research in computer vision. Applications such as face recognition, face tracking and face hallucination, use face detection as a primary and essential pre-processing step. Many practical systems for facial analysis, surveillance and bio-metric, requires fast and accurate face detection.

There are two challenging problems encountered in face detection. The first problem is of classifying faces with a large variety of facial appearances from a complex background. Second of detecting faces of different sizes at different positions in given images. The two problems are related to computational cost and speed of face detection. It is a challenging task to develop a face detector that creates a balance between two problems. Another problem is that is the boundary of an object is blurred by imaging systems also [1], [2].

Face detection methods can be broadly divided into two categories, traditional methods and CNN based methods. The traditional methods, are very fast but does not have good accuracy. These methods use hand-crafted features to train the classifiers. Viola-Jones [3] and Deformable Part Models (DPM) [4] are good examples of traditional methods which have good speed with decent accuracy. The performance of these detectors decreases in an unconstrained environment. This is mainly due to non-robust handcrafted features.

The CNN based methods can achieve high performances at cost of speed. This significant improvement in the accuracy of face detection diverted researchers attention towards CNN based face detectors. CNN models can achieve high performance by using a large number of convolutional layers, which are also responsible for the slow speed of the detector. For example, some recent high performing face detectors like DSFD [5], Pyramidbox [6] and Retinaface [7], use large CNN models like VGG-16 [8] and Resnet-152 [9]. These CNN models consist of a large number of parameters, for example, VGG-16 has 100 million parameters and Resnet-152 has 65 million parameters. CNN methods [5], [6], [7] are slow, hence not suited for many practical systems. Cascade CNN [10], [11] can be used to improve the detection speed. But these detectors suffer two limitations. First, each stage of the cascade is trained and optimized separately which make training difficult and also affect its performance. Second, the speed of the detector directly proportional to the number of faces in an image.

In this paper, a lightweight single-stage end-to-end trained face detector with fast speed and good accuracy is proposed. The proposed method can be divided into two networks, backbone network which extracts feature from input images and detection network which localize the faces. The backbone network uses depthwise separable convolution with large strides to swiftly reduces the dimension of input. Instead of using the max-pooling layer model as given in [12], the proposed model use depthwise separable convolution to reduce the size because it adds extra feature layers and hence provides better feature representation. CReLU [13] activation are used to preserve the information while reducing the size of the input using large strides in the proposed network. The detection network consists a Receptive Field (RF) blocks followed by depthwise convolution layers. A feature map from RF blocks is used for detection.

The main contribution of this paper can be summarized as follows: (1) Propose a new lightweight backbone design to overcome the drawbacks of previous methods. (2) The new lightweight face detection method is proposed by integrating the backbone network with an RF-based detection network for fast and accurate face detection. (3) The experiments performed on multiple benchmark datasets show proposed

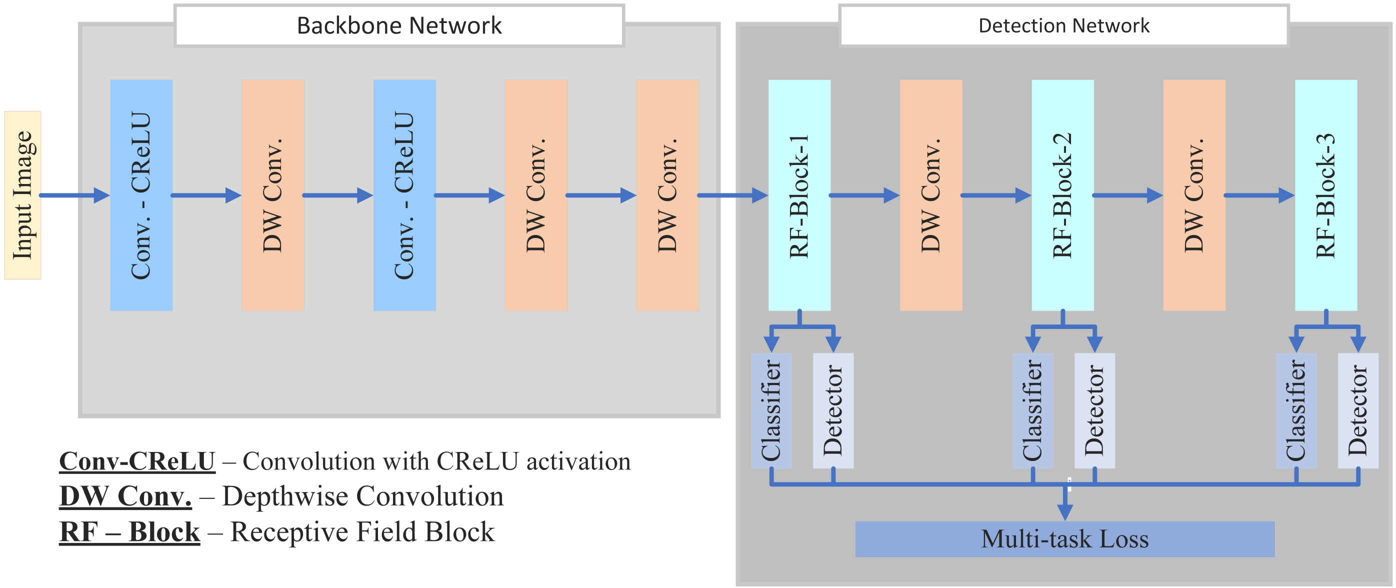


Fig. 1. General Frame Work of the Proposed Model. The Proposed Network can be divided into two parts i.e. Backbone Network and Detection Network.

method performs better than other methods. (4) Experiments performed on CPU and GPU hardware shows that the proposed method is suitable for practical systems. Hereafter the paper is organized as follows, Section 2 contain a brief review of available CNN based face detectors and techniques used in proposed methods. Section 3 is about the proposed method, it explains the framework of the method and its implementation details. Results obtained from experiments are discussed in Section 4, followed by the conclusion in Section 5.

II. Related Works

A. CNN based Face Detectors

Almost all modern days face detectors uses CNN architectures. The CNN based face detectors can be classified into three categories, i.e., cascade face detectors, region-based face detectors and single-stage face detectors.

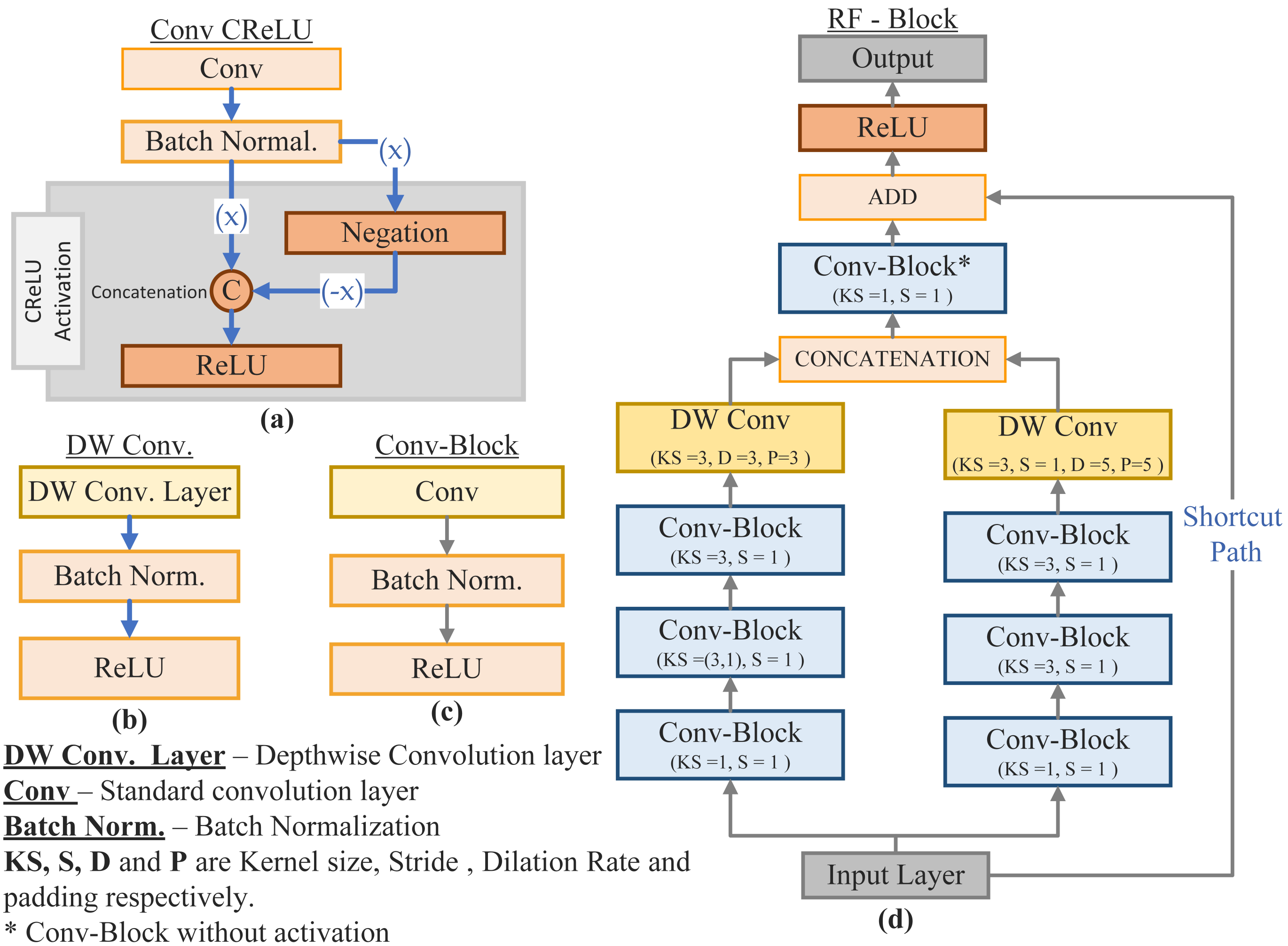
The cascade face detectors divide the detection task into more than one CNN networks. CNN cascade structure introduced in [10], it consists of six CNN networks, three networks for each classification and calibration respectively. Architecture consisted classification network followed by a calibration network. MTCNN [11] reduced the number of networks to three by integrating classification and calibration task into one network. The first network is called P-Net, which proposes a facial region. Later two networks, O-Net and R-net, refines the proposals. The author in [14] divided P-Net into six sub-networks to detect faces at multiple scales. This improves detection performance for tiny faces. The detection performance of cascade face detectors are is improved by adding extra information about facial parts [15], [14]. In cascade framework, the first Network proposes the facial regions and subsequent networks process these regions. This makes the speed of detectors dependent on the number of faces in the images and it is a major limitation of these detectors.

The region-based and single-stage detectors are also known as two-stage and single-stage detectors, respectively. Both the detectors were developed for generic object detection. Later these detectors were modified to be used for face detection. The region-based detectors have two stages, first stage generates object proposal regions from proposal generators. The precise location and class of the object are estimated in the second stages. R-CNN based face detectors [16], [17] use RPN (Region Proposal Networks [18]). The performance of the method is further improved by CMS-RCNN [19] by adding contextual in formations. The region-based detectors use large CNN networks for the second stage. This lead to high detection accuracy but framework processing speed becomes slow.

Single-stage eliminates the region proposal stage and use a single stage to make predictions. These detectors are computationally efficient compared to region-based detectors but suffer detection accuracy. Single-stage face detectors are inspired by generic object detectors like YOLO [20] and SSD [21]. These detectors have attracted more researchers because of there high-speed detection. Different architectures [5], [6], [7] have been proposed recently. Lightweight CNN architecture [12], [22] uses inception module, CReLU activation and also propose densification strategy for anchors to improve recall. LFFD [23] paper proposes an anchor-free lightweight model by using Receptive Fields (RF) as natural anchors for detection. The model parameters were significantly reduced to 0.1 million in [24] by integrating the image pyramid with the CNN network and using weight sharing. But still, there is a large room for improving the processing speed without sacrificing detection accuracy.

B. Receptive Field (RF) and Dilation

Receptive Fields (RF) in CNN are inspired by the human visual system. RF in the visual system is neurons respond to a particular area of the retina. Similarly, in CNN each neuron has an RF field that responds to a particular area of an input [25]. In other words, RF defines the local region of an image to which the neuron will respond. The area RF is determined by the kernel size used in the convolution layer. RF has two important



DW Conv. Layer – Depthwise Convolution layer
Conv – Standard convolution layer
Batch Norm. – Batch Normalization
KS, S, D and **P** are Kernel size, Stride , Dilation Rate and padding respectively.
* Conv-Block without activation

Fig. 2. Detailed Architectural Description of Blocks used in Proposed Methods. (a) Showing the Conv-Blocks with CReLU activation, (b) show Depthwise Separable Convolution used DW Conv block in Backbone Network and RF-Block, (c) Standard Convolution Layer for Conv-blocks used in Backbone Network and RF-Block and (d) detailed Architectural view of RF-Blocks used in Detection Network of Proposed Method.

properties, first, each neuron in CNN has unique activation for a given image region and second, pixels surrounding RF have a large impact on activation. The impact of neighbouring pixels can be represented as Gaussian-Distribution [23], and known as ERF (Effective RF), This RF also helps in detection by adding contextual information to the network.

The RF of CNN can be increased by adding convolution layers, depthwise convolution or by using dilated convolutions [26]. Adding convolution layers (increasing depth of networks) increases computational cost. So using dilation convolution and depthwise more effective way to increase RF. The dilation convolution introduced in [27] as astrous convolution. Dilation convolution is very similar to conventional convolution layer except there is a gap in kernel values which is decided by dilation rate. The author in [12], [23] used RF and dilation convolution for face detection.

C. Depthwise Separable CNN

Many states of art CNN architectures [28], [29] uses depthwise separable convolution layer. The depthwise convolution layer is computationally more efficient than a standard convolutional layer. The standard convolution layer performs convolution operation on input volume and combines generated features in one step. The computational cost of standard convolution is $D_k.D_k.M.N.D_F.D_F$ [28]. Where $D_F and D_k$ is the spatial dimension of input feature and kernel size respectively. While M, N are the number of channels in input features and number of convolution filters respectively. To reduce the computational cost, the one-step process is divided into two steps by using factorized convolution also known as depthwise separable convolution.

The first step is depthwise convolution operation performed on each channel of the input feature map separately. two assumptions are made in this step, (1) that the number convolution filter is equal to the number channels of the input feature map and (2) the spatial size of input and output feature maps are the same. If depthwise convolution is performed on input feature map of spatial size $D_k \times D_k \times M$ using filter of $D_F \times D_F$ spatial size. Then $D_k \times D_k \times M \times D_F \times D_F$ multiplication operations are performed in this stage.

Second step is point wise convolution, 1×1 convolution is performed across M channels output of depthwise convolution. This help to gain cross channel information and linearly combine the output. If N filters of 1×1 dimension is used on $D_k \times D_k \times M$ depth wise convolution output. Then $D_k \times D_k \times M \times N$ multiplication operations are performed. Therefore the computational cost of depthwise convolution is $D_k \times D_k \times M \times (D_F \times D_F + N)$. For qualitative comparison consider an image of $100 \times 100 \times 3$ is passed through depthwise convolution layer and standard convolution layer. If N = 10, then standard convolution performs 2.7×10^7 operations while dethwise convolution perform 5.7×10^6 operations which is approximately 4.7 times less than of standard convolution operations.

III. Proposed Method

In this section, the overall framework of the proposed model is introduced. Followed by a detailed description of model training.

A. Overall Framework of Proposed Model

Proposed face detectors can be divided into two networks, i.e. backbone network and detection network as shown in Fig. 1. The backbone network designed to swiftly reduce the dimensions of the input images without losing information during the process. The backbone network consists of a total of five convolution blocks, the first and third blocks are standard convolution layers with CReLU activation and having large strides. The remaining second, fourth and fifth blocks are depthwise separable convolution block. CReLU is used for its reconstruction property which is of information preserving nature, which leads to features reconstruction power of CNN [13]. CReLU activation is applied by concatenating the linear response of the CNN layer and its negation and passing it through ReLU activation as shown in Fig. 2(a). Mathematically it is defined as:

$$\forall x \in \mathbb{R}, \quad \rho_c \triangleq ([x]_+, [-x]_+) \tag{1}$$

Where $\rho_c : \mathbb{R} \rightarrow \mathbb{R}^2$, CeRLU activation and x is linear response of CNN network. From the above equation 1, it can be easily deduced that CReLU activation perverse both negative and positive response. Hence CReLU scheme produces representative features of input data [13]. To reduce the computational complexity depthwise separable convolution block are used. These blocks consist of depthwise convolution followed by batch normalization and ReLU activation as shown in Fig. 2(b). The feature obtained from a backbone network is feed into the detection network for further processing. The detection network is based on the cascade structure of SSD [21]. The model uses features from RF Blocks, which are spatially decreasing but have increasing respective field. Feature maps from different layer form multi-scale feature map to handle faces of variable sizes. RF-block-1, RF-block-2 and RF-block-3 are associated with anchor boxes to detect faces of small, medium and large sizes respectively. Multi-layer, multi-branch RF-blocks uses different kernels and dilation rates. This design has the advantage of classifying faces (with facial variation) from a complex background.

RF-blocks consist a bottleneck structure and residual connection as [30], [31]. The first layer of multi-branch design is 1×1 convolution, used to reduces the channel in feature maps. Then to reduce the computational cost 3×1 and 1×3 convolution is used. To increase the non-linearity and effective receptive field, depth-wise separable convolution with different dilation rate are used. Increased non-linearity, generates a more robust feature representation of the input. The increased effective receptive field helps to capture more contextual information for accurate classification. The branches are concatenated and a shortcut path is added to it. Fig. 2(d) shows the detailed architecture of the RF-block. Figure 2b and 2c show the architectural view of convolutional and depth wise separable convolution used in RF-blocks. In the model, each convolution layer is followed by batch normalization and ReLU activation respectively. This is done to reduce overfitting, induce sparsity and to handle the vanishing gradient problem.

B. Implementation Details

The model uses the anchor of 1:1 aspect ratio and densification strategy of [12]. The scale of anchors for RF block-1 are 32,64 and 128, for RF block-2 is 256 and RF block-3 is 512 pixels. The model is trained on WIDER FACE [32] training data set. This dataset consists of 12880 training images with different sizes, occlusion and blurriness levels. The training data is prepared by removing extremely small faces (height or width less than 15 pixels), heavily blur and occlude faces. For data augmentation, different strategies like random cropping, horizontal flipping, scale transformation and colour distortion are used during training. During training, the ground truth anchor boxes are matched to the predicted bounding box if the jaccrad index is more than 0.40. The multi-box loss objective function [21] is used in a training. It is a weighted sum of cross-entropy loss for bounding box confidence and smooth L-1 loss for bounding box coordinate regression. It is defined as:

$$L(c_i, d_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(c_i, c_i^*) + \frac{\lambda}{N_{reg}} \sum_i L_{reg}(d_i, d_i^*) \tag{2}$$

where, $L(c_i, d_i)$ is multi-box loss for given c_i confidence score of i^{th} bounding box with d_i coordinates. $L_{cls}(c_i, c_i^*)$ is cross entropy loss between predicted confidence score c_i^* bounding box and ground truth confidence score c_i. $L_{reg}(d_i, d_i^*$ is smooth L1 loss for predicted and ground bounding box coordinates. λ is hyper parameter used to balance the sum of losses ($\lambda = 2$ is used for training the network). Model is trained using batch size 32 for 280 thousand iterations. SGD optimizer used in training have 0.9 momentum, 5×10^{-4} weight decay. Model is trained using variable learning rates of 10^{-3}, 10^{-4} and 10^{-5} for 160K iterations 10^{-3}, 80K and 40K iterations respectively. The model is implemented using PyTorch framework[1].

IV. Results and Discussion

In this section proposed face detection algorithm is evaluated on the benchmark datasets, followed by speed comparison with available lightweight models.

[1] https://pytorch.org/

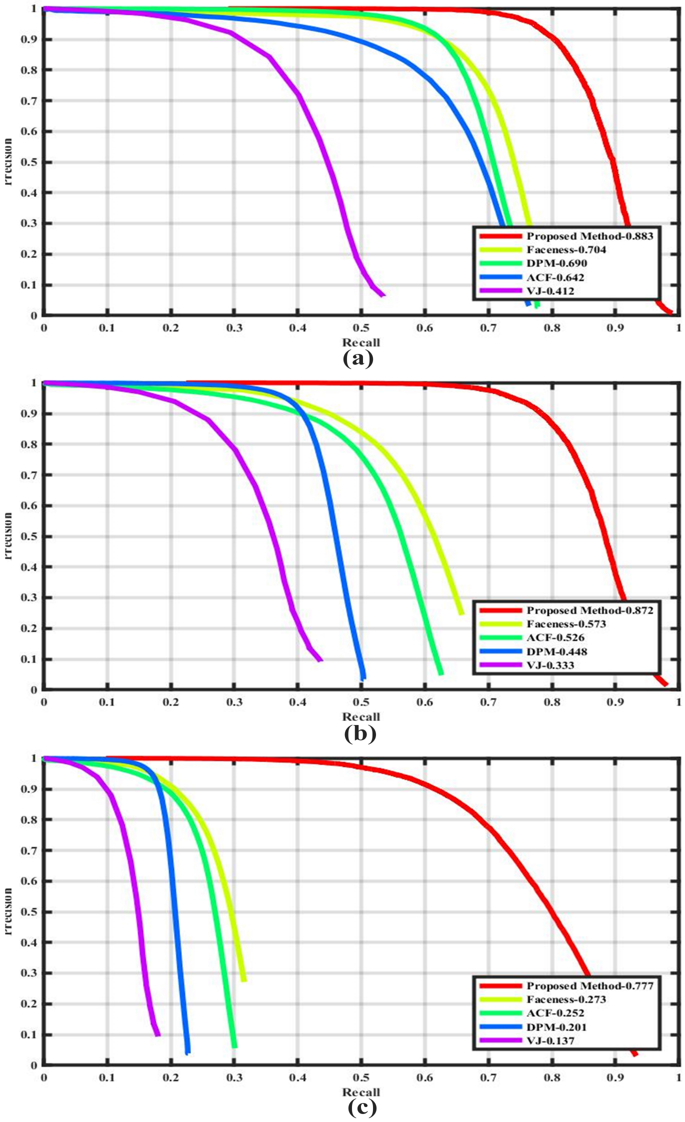


Fig. 3. PR Curve Comparing results of Proposed Methods and other Methods on (a) easy (b) medium and (c) hard validation subsets.

A. *Experimental Setup*

The proposed method is implemented using Pytorch version 1.6.0 on i5-8330H@2.30GHz processor system with 16 Gigabytes RAM and NVIDIA GTX 1060 GPU (Graphical Processing Unit).

B. *Evalution on Benchmark Dataset*

The proposed algorithm is evaluated on three benchmark face detection dataset, WIDER FACE [32], Pascal Face [33] and AFW [34] [33]. The proposed method is compared with other state of art lightweight detector using Average Precision (AP) percentage metric and PR (Precision-Recall) curves.

1) WIDER FACE Dataset: The dataset contains total 32203 images of faces different pose, scale, facial expressions and illumination. The dataset contain training and validation set. Validation set have three subsets validation data based on difficulties level of face detection, these are easy, medium and hard. The proposed method is trained on training set and validated results on all three validation subsets. Proposed method is validated against baseline methods [3], [4], [35], [36] and other methods [11], [12], [22], [37], [38], [39], [24]. Table I shows the results of performance comparison proposed methods with other methods. The proposed method shows the better result on easy and medium validation set, comparable result on hard dataset. This could be due to the fact that the network was trained on face which have height or width greater than 15 pixels and heavily occlude and blur faces were removed from the training set. Fig. 3 shows the PR curve of the proposed method compared against base line methods.

2) AFW and PASCAL Face Dataset: AFW dataset is Flickr images collection of 205 images with 473 face annotation. Table 2 shows the performance comparison of proposed method with standard methods using mAP% metrics. The

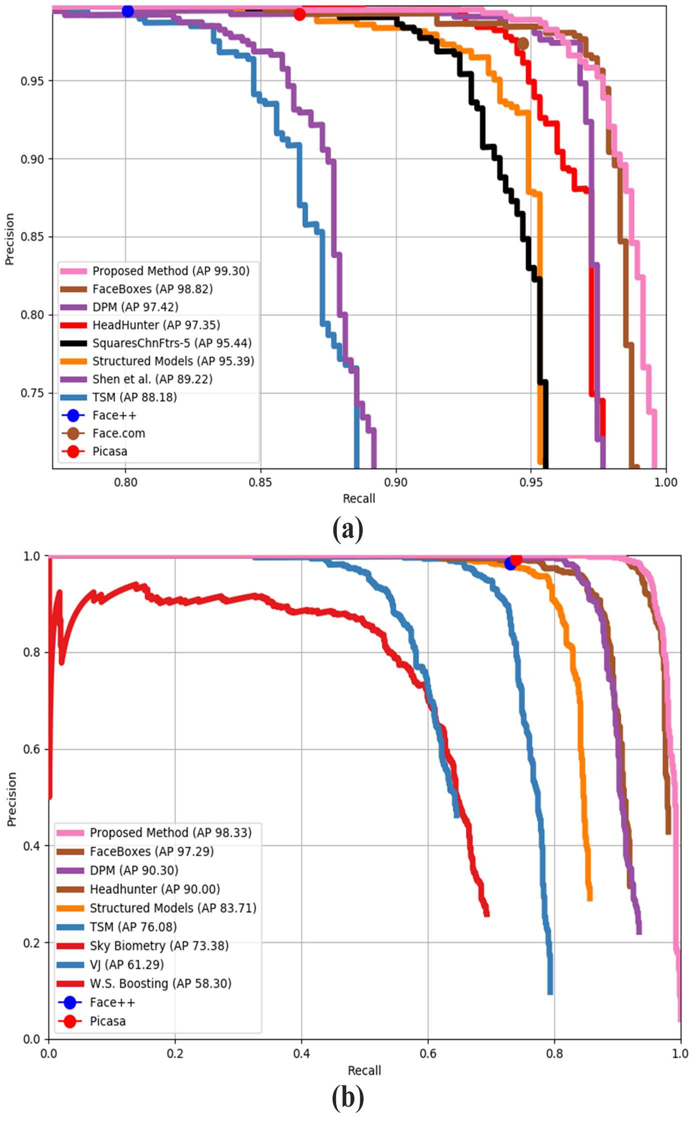


Fig. 4. PR Curve Comparing results of Proposed Methods and other Methods on (a) AFW and (b) Pascal dataset.

TABLE I. PERFORMANCE RESULT OF PROPOSED METHOD ON THE VALIDATION UBSETS OF WIDER FACE. THE REPORTED VALUES ARE MAP%

Methods	Easy	Medium	Hard
VJ*[3]	41.2	33.3	13.7
DPM* [4]	69.0	44.8	20.1
ACF* [35]	64.2	52.6	25.2
Faceness* [36]	70.4	57.3	27.3
MTCNN [11]	85.1	82.0	60.7
Faceboxes [12]	79.1	79.4	71.5
Faceboxes-2 [22]	87.9	85.7	77.1
ICC-CNN [37]	85.1	82.9	77.2
FDCNN [38]	73.3	67.8	51.0
Fastfaces [38]	83.3	79.6	60.3
Luo *et. al.* [40]	87.1	87.3	78.0
Proposed method	88.30	87.2	77.7

proposed method shows the better performance then other methods. [4], [12], [33], [41], [42]. Fig. 4(a) shows PR curve for proposed method,standard methods and commercial face detectors(Face.com, Face++ and Picasa).

Pascal Face dataset is formed from pascal person layout dataset. It contain 851 images with 1335 face annotations. The comparison of proposed method with standard dataset [3], [4], [12], [33], [42], [34] is given in Table II. Fig. 4(b) shows the PR curve of proposed method, standard method and commercial methods. Proposed method showed better results on dataset.

TABLE II. PERFORMANCE COMPARISON PROPOSED METHOD WITH OTHER METHODS ON AFW AND PASCAL DATASET. THE REPORTED VALUES ARE MAP%

Methods	AFW	PASCAL face
VJ [3]	-	61.29
DPM [4]	97.42	90.30
Headhunter [4]	97.35	90.0
SquareChnFtrs [4]	95.44	-
StrusctredModel [33]	95.39	83.71
TSM [42]	88.18	76.08
Shen *et al.* [41]	89.22	-
WSBoosting [34]	-	58.30
Faceboxes[12]	97.29	98.82
Proposed method	99.30	98.33

C. Running Efficiency

To check the practicality of a proposed method, it is tested on CPU and GPU hardware. Results obtained are then compared against the state of art methods reported running efficiencies in original paper.

TABLE III. RUNNING EFFICIENCY COMPARISON OF THE PROPOSED METHOD WITH STATE OF ART METHODS. THE SPEED OF THE METHOD ON CPU AND GPU ARE REPORTED IN FPS (FRAME PER SECOND)

Method	CPU	GPU
Faceness [36]	–	20 (Titan Black)
MTCNN [11]	16 (i7-4770K)	99(TitanX)
Faceboxes [12]	20 (E5-2660v3)	120(TitanX)
Faceboxes-2 [22]	28 (E5-2660v3)	245(TitanX)
ICC-CNN [37]	12 (i7-4770K)	40 (Titan)
FDCNN [38]	–	31 (GTX 1080)
ACF [35]	20 (i7-3770)	–
Luo *et. al.* [40]	50 (i7-6850 K)	180 (RTX 2080Ti)
Proposed method	42 (i5-8330H)	179 (GTX 1060)

The qualitative results are summarized in Table III. The results compared on image size 640X480. The detailed description of system on which test was performed is mentioned in section above. The proposed methods performance is very satisfactory, but it has comparatively slow then [40] because hardware (both CPU and GPU) on which the experiments performed are computationally inferior to other hardware on which the state of arts methods are tested.

V. CONCLUSION

This paper introduces a fast and high performing face detector. The high processing speed is achieved by using a lightweight backbone network. The feature extractor rapidly reduces the size of input without losing information during this process. The information is retained by the CReLU activation function. The performance of the face detector is achieved by efficiently utilizing the feature maps obtained from the feature extractor. The detector having RF blocks imitating the human visual system. As the results suggest the proposed method works well on the images with images having faces of the height of more than 15 pixels. The model limitation to detect tiny faces and heavily occluded faces. The proposed model can further be compressed using CNN optimization techniques such as pruning. The experiments performed using proposed face detectors on benchmark datasets has shown good results and have high processing speeds on both CPU and GPU devices.

ACKNOWLEDGMENT

The authors wish to acknowledge the National Project Implementation Unit (NPIU), a unit of the Ministry of Human Resource Development, Government of India, for the financial assistant-ship through the TEQIP-III Project at Deenbandhu Chhotu Ram University of Science and Technology, Murthal, Haryana. The authors would like to thank the editors and anonymous reviewers for providing insightful suggestions and comments to improve the quality of the research paper.

REFERENCES

[1] P. Kaur, I. Lamba, and A. Gosain, "A robust method for image segmentation of noisy digital images," in *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*. IEEE, 2011, pp. 1656–1663.

[2] P. Kaur, A. Soni, and A. Gosain, "Image segmentation of noisy digital images using extended fuzzy c–means clustering algorithm," *International journal of computer applications in technology*, vol. 47, no. 2-3, pp. 198–205, 2013.

[3] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[4] M. Mathias, R. Benenson, M. Pedersoli, and L. V. Gool, "Face detection without bells and whistles," in *European conference on computer vision*, 2014, pp. 720–735.

[5] J. Li, Y. Wang, C. Wang, Y. Tai, J. Qian, J. Yang, C. Wang, J. Li, and F. Huang, "DSFD: dual shot face detector," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5060–5069.

[6] X. Tang, D. K. Du, Z. He, and J. Liu, "Pyramidbox: A context-assisted single shot face detector," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 797–813.

[7] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, "RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5203–5212.

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

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 5325–5334.

[11] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[12] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "Faceboxes: A CPU real-time face detector with high accuracy," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*, 2017, pp. 1–9.

[13] W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *international conference on machine learning*, 2016, pp. 2217–2225.

[14] D. Zeng, F. Zhao, S. Ge, and W. Shen, "Fast cascade face detection with pyramid network," *Pattern Recognition Letters*, vol. 119, pp. 180–186, 2019.

[15] K. Zhang, Z. Zhang, H. Wang, Z. Li, Y. Qiao, and W. Liu, "Detecting faces using inside cascaded contextual cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3171–3179.

[16] H. Jiang and E. Learned-Miller, "Face detection with the faster r-cnn," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. IEEE, 2017, pp. 650–657.

[17] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: An improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[19] C. Zhu, Y. Zheng, K. Luu, and M. Savvides, "Cms-rcnn: contextual multi-scale region-based cnn for unconstrained face detection," in *Deep learning for biometrics*. Springer, 2017, pp. 57–79.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21–37.

[22] S. Zhang, X. Wang, Z. Lei, and S. Z. Li, "Faceboxes: A CPU real-time and accurate unconstrained face detector," *Neurocomputing*, vol. 364, pp. 297–309, 2019.

[23] Y. He, D. Xu, L. Wu, M. Jian, S. Xiang, and C. Pan, "Lffd: A light and fast face detector for edge devices," *arXiv preprint arXiv:1904.10633*, 2019.

[24] J. Luo, J. Liu, J. Lin, and Z. Wang, "A lightweight face detector by integrating the convolutional neural network with the image pyramid," *Pattern Recognition Letters*, 2020.

[25] G. Kobayashi and H. Shouno, "Interpretation of resnet by visualization of preferred stimulus in receptive fields," *ArXiv*, vol. abs/2006.01645, 2020.

[26] N. Adaloglou, "Understanding the receptive field of deep convolutional networks," *https://theaisummer.com/*, 2020 (accessed: 25.05.2020). [Online]. Available: https://theaisummer.com/receptive-field/

[27] L.-C. C. Author, G. Papandreou, I. Kokkinos, K. Murphy, and A. LYuille, "Deeplab: Semantic image segmentation with deep convolutional nets, astrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[28] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018. [Online]. Available: http://dx.doi.org/10.1109/CVPR.2018.00474

[29] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *arXiv preprint arXiv:1905.11946*, 2019.

[30] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," *arXiv preprint arXiv:1602.07261*, 2016.

[31] S. Liu and D. Huang, "Receptive field block net for accurate and fast object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 385–400.

[32] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face: A face detection benchmark," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5525–5533.

[33] J. Yan, X. Zhang, Z. Lei, and S. Z. Li, "Face detection by structural models," *Image and Vision Computing*, vol. 32, no. 10, pp. 790–799, 2014.

[34] Z. Kalal, J. Matas, and K. Mikolajczyk, "Weighted Sampling for Large-Scale Boosting." in *BMVC*, 2008, pp. 1–10.

[35] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," in *IEEE international joint conference on biometrics*, 2014, pp. 1–8.

[36] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3676–3684.

[37] K. Zhang, Z. Zhang, H. Wang, Z. Li, Y. Qiao, and W. Liu, "Detecting faces using inside cascaded contextual cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3171–3179.

[38] D. Triantafyllidou, P. Nousi, and A. Tefas, "Fast deep convolutional face detection in the wild exploiting hard sample mining," *Big data research*, vol. 11, pp. 65–76, 2018.

[39] H. Zhang, X. Wang, J. Zhu, and C.-C. J. Kuo, "Fast face detection on mobile devices by leveraging global and local facial characteristics," *Signal Processing: Image Communication*, vol. 78, pp. 1–8, 2019.

[40] J. Luo, J. Liu, J. Lin, and Z. Wang, "A lightweight face detector by integrating the convolutional neural network with the image pyramid," *Pattern Recognition Letters*, 2020.

[41] X. Shen, Z. Lin, J. Brandt, and Y. Wu, "Detecting and aligning faces by image retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3460–3467.

[42] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *2012 IEEE conference on computer vision and pattern recognition*, 2012, pp. 2879–2886.