FLDet: A CPU Real-time Joint Face and Landmark Detector

Chubin Zhuang[1,2], Shifeng Zhang[1,2], Xiangyu Zhu[1,2]*, Zhen Lei[1,2], Jinqiao Wang[1,2], Stan Z. Li[1,2]
[1] NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
{chubin.zhuang, shifeng.zhang, xiangyu.zhu, zlei, jqwang, szli}@nlpr.ia.ac.cn

Abstract

Face detection and alignment are considered as two independent tasks and conducted sequentially in most face applications. However, these two tasks are highly related and they can be integrated into a single model. In this paper, we propose a novel single-shot detector for joint face detection and alignment, namely FLDet, with remarkable performance on both speed and accuracy. Specifically, the FLDet consists of three main modules: Rapidly Digested Backbone (RDB), Lightweight Feature Pyramid Network (LFPN) and Multi-task Detection Module (MDM). The RDB quickly shrinks the spatial size of feature maps to guarantee the CPU real-time speed. The LFPN integrates different detection layers in a top-down fashion to enrich the feature of low-level layers with little extra time overhead. The MDM jointly performs face and landmark detection over different layers to handle faces of various scales. Besides, we introduce a new data augmentation strategy to take full usage of the face alignment dataset. As a result, the proposed FLDet can run at 20 *FPS on a single CPU core and* 120 *FPS using a GPU for VGA-resolution images. Notably, the FLDet can be trained end-to-end and its inference time is invariant to the number of faces. We achieve competitive results on both face detection and face alignment benchmark datasets, including AFW, PASCAL FACE, FDDB and AFLW.*

1. Introduction

Face detection and face alignment are two fundamental steps in many subsequent face-related applications, *e.g.,* face recognition [6, 21] and face attribute analysis [8]. Since the milestone work of Viola-Jones [29], face detection and alignment have been intensively studied separately and both of them have achieved tremendous progress in the past few years, especially the CNN based detectors.

Recent CNN based face detectors can be divided into two categories: cascade based methods [15, 35] and anchor based methods [4, 28]. The former can well handle faces with diverse scales, but becomes time-consuming when there are many faces in the image. While the speed of anchor based methods is invariant to the object number, but suffers from unsatisfying efficiency. Therefore, efficient detection of multi-scale faces is still one of the critical issues that remains to be settled, especially for CPU devices.

As for face alignment, CNN based methods have also achieved the state-of-the-art performance, *e.g.*, Coordinate Regression Model [20, 40] and Heatmap Regression Model [1, 31]. However, most of face alignment methods must be initialized by the provided face bounding box in advance, which presents a great demand of joint face and landmark detection.

Since both face detection and alignment aim to find the components of human faces, it is possible to make them share information to improve the efficiency. However, the methods attempting to jointly solve detection and alignment always have degraded performance on either accuracy or efficiency. For example, Chen et al. [3] apply random forest on the differences of pixel values to jointly conduct alignment and detection, but the handcraft features can not achieve satisfactory performance. MTCNN [35] leverages a cascaded architecture with three shallow-to-deep convolution networks to jointly predict face and landmark locations in a coarse-to-fine manner, but the efficiency will deteriorate dramatically as the number of faces increases, which limits its practical application. Besides, the cascaded structure also makes the end-to-end training infeasible and further imposes a great burden on the training phase.

Therefore, joint face detection and alignment is still a challenging issue, especially for the computation restricted devices (*e.g.,* CPU). The concerns may generally come as follows: 1) The large variations of faces in cluttered background require the detector to be robust to face scales; 2) The large search space of possible faces further requires the trade-off between accuracy and efficiency; 3) Even though face detection and alignment are two closely related work, they have different requirement of training set modality, model design and so on. To sum up, it is still a challeng-

*Corresponding author

ing problem to design an efficient detector for joint face and landmark detection on CPU devices to achieve real-time speed as well as maintain high performance.

To solve this problem, we integrate these two tasks into a single-shot model via multi-task learning and develop a CPU real-time speed detector, named FLDet, which has high performance on both face and landmark detection. Different from MTCNN, the FLDet only contains a lightweight yet powerful network and can be trained end-to-end. Specifically, the FLDet consists of the Rapidly Digested Backbone (RDB), Lightweight Feature Pyramid Network (LFPN) and Multi-task Detection Module (MDM). The RDB is designed to quickly digest feature maps so as to guarantee CPU real-time speed. The LFPN integrates these features from different detection layers to enrich the semantic information of low-level layers with little extra time cost. The MDM tiles the anchors with five preset points over different layers and jointly conducts face and landmark detection by multi-task learning. Besides, we introduce a new data augmentation strategy to take full usage of the face alignment dataset. Consequently, for VGA-resolution images, the proposed FLDet can run at 20 FPS on a single CPU core and 120 FPS on a NVIDIA Titan X (Pascal) GPU in inference. We comprehensively evaluate this detector and demonstrate competitive detection performance on several common face detection and face alignment benchmark datasets, including AFW, PASCAL FACE, FDDB and AFLW.

For clarity, the main contributions of this work can be summarized as follows:

- We design a single-shot framework for joint face and landmark detection with the CPU real-time speed and an end-to-end training fashion.
- We propose a novel landmark anchor with five preset points to jointly predict face and landmark locations.
- We introduce a new data augmentation strategy to promote the performance of face detection and alignment.
- We apply multi-task learning on two completely different datasets, *i.e.*, WIDER FACE and CelebA, and achieve competitive performance on both face detection and alignment tasks.

2. Related Work

Face Detection. Since the pioneering work of Viola-Jones [29], most early face detectors focus on designing robust features and training effective classifiers, but their performance deteriorates severely for the large visual variations of faces. Recent years have witnessed the advance of CNN based detectors. CascadeCNN [15] employs a cascade structure to detect faces in a coarse-to-fine way and following. PCN [27] proposes a cascade-style structure to perform rotation-invariant face detection. Besides, anchor based methods originated from Faster R-CNN [26] and SSD [18] have achieved great progress in recent years. Jiang et al. [13] apply Faster R-CNN in face detection and achieve promising results. The face detection model for finding tiny faces [9] trains separate detectors for different scales. S^3FD [38] presents multiple strategies to improve the performance of small faces. SSH [23] models the context information by large filters on each prediction module. AFD [42] applies the attention mechanism in RefineDet [36] to detect faces. PyramidBox [28] utilizes contextual information with improved SSD network structure.

Face Alignment. In the literature of face alignment, besides classic methods [22] and Cascaded Regression Models [2], recent state-of-the-art performance has also been achieved with Convolutional Neural Networks (CNNs). Zhang et al. [40] frame the problem as a multi-task learning problem to predict landmark and facial attributes at the same time. TSR [20] splits face into several parts to ease the parts variations and regresses the coordinates of different parts respectively. After that, modern detectors with superior performance are mainly based on heatmap regression models. CALE [1] is a two-stage convolutional aggregation model to aggregate score maps predicted by detection stage along with early CNN features for final heatmap regression. JMFA [5] achieves state-of-the-art accuracy by leveraging stacked hourglass network for multi-view face alignment.

Joint Face Detection and Alignment. There are some existing works attempting to jointly solve the problem of face detection and alignment in a single model. Chen [3] et al. apply random forest based on the features of pixel value difference to jointly conduct alignment and detection, but these handcraft features are low-level features and greatly limit its performance. MTCNN [35] leverages a cascaded architecture with three stages of shallow to deep convolution networks to jointly predict face and landmark locations in a coarse-to-fine manner, but the runtime efficiency will deteriorate dramatically as the number of faces getting larger. Besides, the cascaded structure makes the end-to-end training infeasible and further imposes a great burden on the training phase. Notably, the proposed FLDet is a specially designed lightweight yet powerful network in a single-shot fashion, which can run at CPU real-time speed along with remarkable performance on both face detection and alignment with end-to-end training enabled.

3. Approach

This section introduces the details of the FLDet that enable the detector to be accurate and efficient on CPU devices, including network architecture, end-to-end training, data augmentation, as well as implementation details.

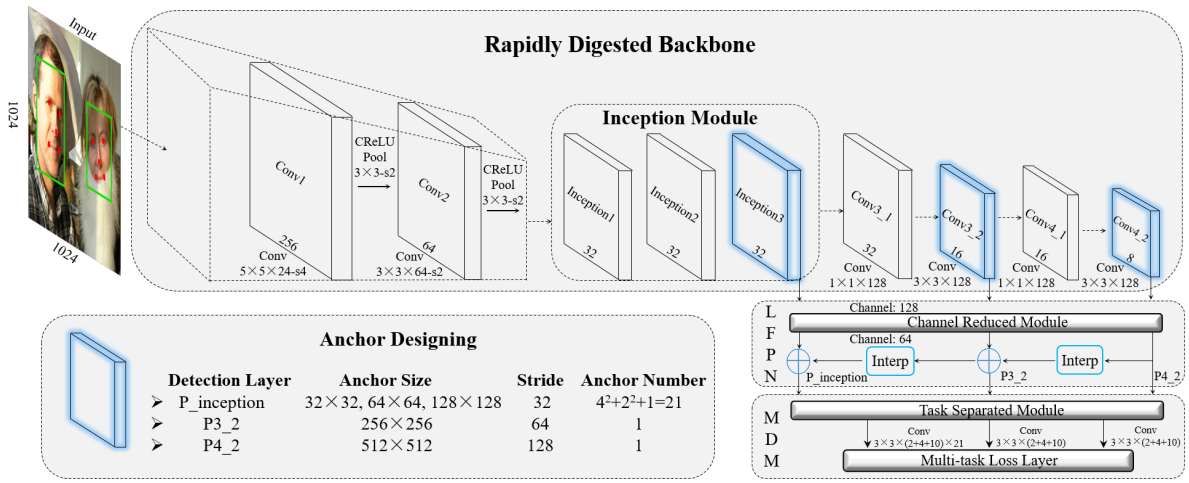


Figure 1. Architecture of FLDet and the detailed information table about our anchor designs. The cubes with blue light represent the layers we select that will be further processed for detection.

3.1. Network Architecture

Our FLDet is a lightweight yet powerful network that consists of the Rapidly Digested Backbone (RDB), Lightweight Feature Pyramid Network (LFPN) and Multi-task Detection Module (MDM). These modules are elaborated as follows.

3.1.1 Rapidly Digested Backbone

Most CNN based methods are compromised on the runtime efficiency due to the expensive computation of the convolution operation when the size of input, kernel and output are large, especially for CPU devices. To this end, our RDB is specially designed to fast shrink the input spatial size by suitable kernel size with reduced output channels, so as to minimize the time overhead as well as maintain enough semantic features.

As illustrated in Fig. 1, we follow the backbone design of FaceBoxes [37] and further adjust it to be even thinner. Specifically, we utilize smaller kernel size of 5×5 and 3×3 for $Conv1$ and $Conv2$ and replace the output channels of $Conv3\_2$ and $Conv4\_2$ both with 128. Therefore, our model first shrinks the input image with spatial size quickly reduced by 32 times. Then the Inception Module is applied to enrich the receptive fields, since we jointly till anchors of different scales on the same detection layer, *e.g.*, layer $P\_inception$. Finally $Conv3$ and $Conv4$ are designed to further reduce the spatial size of feature maps so as to handle faces of larger scales.

3.1.2 Lightweight Feature Pyramid Network

After obtaining the feature maps from different detection layers, we apply Feature Pyramid Network (FPN) [17] to integrate the high-level features into low-level layers so as to enrich the semantic information of lower layers, since the features extracted from these shallow layers are not deep and robust enough to detect the small scale faces.

However, the common implementation of FPN is quite time-consuming due to the low efficiency of deconvolution and crop operation on CPU devices, especially for feature maps with large input channels. To this end, as shown in Fig. 1, the deconvolution is replaced with interpolation operation, which directly enlarges the feature maps by bilinear interpolation. Besides, we also utilize a simple convolution layer with kernel size of 1×1 to reduce the channels of these feature maps from 128 to 64 before applying integration so as to further promote the runtime efficiency. We sequentially perform these operations on layer $Inception3$, $Conv3\_2$ and $Conv4\_2$ and finally obtain $P\_inception$, $P3\_2$ and $P4\_2$ as our detection layers. Consequently, when integrated with the aforementioned RDB, our model achieves a great trade-off between accuracy and efficiency.

3.1.3 Multi-task Detection Module

Joint face detection and alignment can be generally divided into three subtasks, *i.e.*, face classification, bounding box regression and landmark regression. Although there exists correlation between them, the differences still remain since the focuses of these tasks are different. Specifically, face classification judges whether the anchor belongs to a face or not. Bounding box regression aims at predicting the area of face. While landmark regression pays more attention to the location of every landmark.

An intuitive way is to directly implement the three predictions on the same feature maps, which makes the features suboptimal to all subtasks. To prevent these tasks from in-

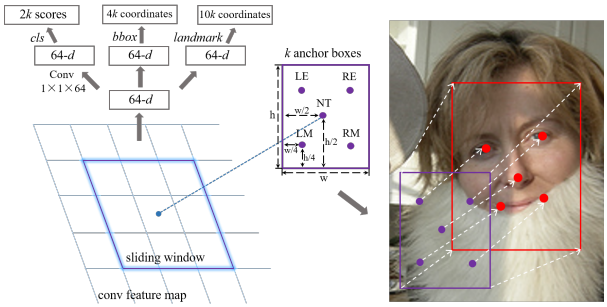


Figure 2. **Left:** Task separated module. **Middle:** Landmark anchor with five preset points. **Right:** Example of anchor regression. The purple rectangle with five preset points represents the default anchor and the red rectangle with five landmarks is the matched ground-truth will be further regressed to.

terfering with each other, as illustrated in the left image of Fig. 2, we apply a simple convolution layer with kernel size of 1×1 to separately map the features extracted from LFPN to three different 64-dimensional subspaces, and then make predictions on these divided feature spaces, respectively. As a result, we have the features for multi-task learning decoupled, and further promote the performance of both face detection and face alignment at the expense of little extra time.

3.2. End-to-end Training

3.2.1 Five-point Anchor

As depicted in Fig. 1, our default anchors are associated with multi-scale feature maps, *i.e.*, $P\_inception$, $P3\_2$ and $P4\_2$. These layers, as a multi-scale design along the dimension of network depth, discretize anchors over multiple layers with different resolutions to naturally handle faces of various sizes. Specifically, layer $P\_inception$ works for small faces, layer $P3\_2$ and $P4\_2$ are responsible for medium and large faces, respectively.

Besides, to better match default anchors and ground-truth bounding boxes in WIDER FACE and CelebA training set, we impose 1.25:1 aspect ratio for the default anchors, which is almost the average aspect ratio of ground-truth bounding boxes. The scale of anchor for the $P\_inception$ layer is 32, 64 and 128 pixels, for the other two layers are 256 and 512 pixels, respectively. Since the stride of layer $P\_inception$ is not suitable for the anchor size of 32 and 64, making these two small scale anchors sparsely distributed in the image space. Therefore, we further apply the Anchor Densification Strategy [37] to densify the 32×32 anchor 4 times and the 64×64 anchor 2 times to guarantee that the tiled anchors can fully cover the areas of small faces.

Different from the anchors commonly used in face detection, we propose a novel landmark anchor with five preset points to jointly make predictions of face and landmark locations. As shown in Fig. 2, these preset points roughly locate in the five corners of anchor boxes and each is responsible for one landmark regression task. Specifically, point *LE* and *RE* are served as the initial locations of left and right eyes, point *NT* is responsible for the regression of nose tip, and point *LM* and *RM* are respectively the initial locations of left and right corners of the mouth. By this simple design, we naturally change this joint complicated detection task into a common location regression task.

During the training phase, we first match each face to the anchor with the best jaccard overlap, and then match anchors to any faces with jaccard overlap higher than 0.35. For bounding box regression, we adopt the parameterizations of the 4 coordinates following [26], and further generalize it to landmark regression as follows:

$$\begin{aligned} t_x^i &= (x^i - x_a^i)/w_a^i,\ t_y^i = (y^i - y_a^i)/h_a^i, \\ t_X^i &= (X^i - x_a^i)/w_a^i,\ t_Y^i = (Y^i - y_a^i)/h_a^i, \end{aligned} \tag{1}$$

where i is the index of landmark, $i = 0, 1, ..., 4$. x, y, w and h denote the landmark coordinates and the width and height of box. Variables x, x_a and X are for the predicted box, anchor box and ground-truth box, respectively (likewise for y, w and h). This can be thought of as joint bounding box and landmark regression from an anchor box to a nearby ground-truth box.

3.2.2 Multi-task Loss Function

We jointly perform model training on WIDER FACE [32] and CelebA [19] datasets for three subtasks: face/non-face classification, bounding box regression and facial landmark regression. Since landmark annotations are labelled only in CelebA dataset, we just carry out face detection training for WIDER FACE images and set the loss of landmark regression to 0. We assign a binary class label to each matched anchor and regress its location and size to the target bounding box. In this case, this multi-task loss function can be defined as:

$$\begin{aligned} L(p, x, y) = &\frac{\lambda_1}{N_{cls}} \sum_{i=1} L_{cls}(p_i, p_i^*) + \\ &\frac{\lambda_2}{N_{box}} \sum_{i=1} p_i^* L_{box}(x_i, x_i^*) + \frac{\lambda_3}{N_{lan}} \sum_{i=1} p_i^* q_i L_{lan}(y_i, y_i^*), \end{aligned} \tag{2}$$

where i is the index of an anchor and p_i is the predicted probability that anchor i is a face, the ground-truth label p_i^* is 1 if the anchor is positive, 0 otherwise. x_i and y_i are vectors respectively representing the 4 and 10 parameterized coordinates of the predicted face and landmark locations, while x_i^* and y_i^* are the corresponding ground-truth box parameters associated with a positive anchor. The classification loss $L_{cls}(p_i, p_i^*)$ is softmax loss over two classes (face *vs.* background), the bounding box regression loss $L_{box}(x_i, x_i^*)$ and landmark regression loss $L_{lan}(y_i, y_i^*)$ are both the smooth L1 loss, $p_i^* L_{box}$ means the bounding box

regression loss is activated only for positive anchors, and $p_i^* q_i L_{lan}$ indicates that landmark regression loss will be computed only for positive anchors with landmark annotations, where q_i is 1 for CelebA dataset and 0 for WIDER FACE. These three terms are normalized by N_{cls}, N_{box} and N_{lan}, and further weighted by balancing parameter λ_1, λ_2 and λ_3. In our implementation, the cls term is normalized by the number of positive and negative anchors, the box term is normalized by the number of positive anchors, and the lan term is normalized by the number of positive anchors along with landmark annotations.

3.3. Data Augmentation

To our best knowledge, there is no suitable and publicly available dataset for joint face and landmark detection. The common training set for face detection like WIDER FACE [32] suffers from the missing of landmark annotations, while the landmark training set like CelebA [19] only contains a large amount of easy faces, which is harmful to the training of face detection. Consequently, the difference on image modalities between face detection and alignment training sets imposes a great burden on the joint face and landmark detection. To this end, we first present the image pyramid strategy to mitigate the modality gaps between WIDER FACE and CelebA datasets, then introduce the pose based data balance strategy to help enhance the performance from data perspective.

3.3.1 Image Pyramid

Since we jointly perform training on WIDER FACE and CelebA datasets for face detection and face alignment tasks, the different image modalities of these two datasets are actually inevitable problems remain to be settled. Specifically, the images in WIDER FACE contains different numbers, poses and scales of faces in complicated backgrounds, while the CelebA is composed of large amount of frontal faces with only one face presents per image. Direct training on this joint dataset will severely deteriorate the performance of face detection because of the easy faces in CelebA, as well as impose a challenge on landmark detection of small scale faces since the scarcity of small faces in CelebA.

As shown in Fig. 3, we rescale the images in CelebA by the ratio of $1/2^n$ $(n = 0, ..., N)$, and then stitch these randomly flipped subimages into a new image pyramid to construct a WIDER FACE fashion landmark dataset with several face annotations present in one image. The shrink factor N is determined by the shortest side of ground-truth bounding boxes as computed in the following equation:

$$N = min(2, floor(min(face_w, face_h)/50)) \quad (3)$$

where $face_w$ and $face_h$ represent width and height of the face annotation in CelebA image. In this way, it can be ensured that the rescaled face annotations are still larger than 25 pixels, since too small faces are harmful to the training of landmark localization.

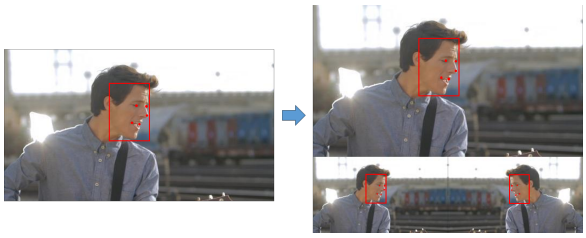


Figure 3. An example of image pyramid. The image in left column is the input image with face size of 86 pixels and the right one is the reconstructed image with shrink factor $N = 1$.

3.3.2 Pose Based Data Balance

As indicated in [7], one of the challenges of face alignment in large poses is the data imbalance, since most of faces in the landmark datasets are frontal. The detector trained on such a dataset is easy to overfit to the frontal pose and can not well adapt to faces with various poses.

To mitigate this challenging issue, we study the pose distribution of the faces in CelebA images, and roughly divide these images into four subsets based on the pose angle (*i.e.*, $None$, $Small$, $Medium$, $Large$). As summarized in Tab. 1, we randomly strick out the images of $None$ subset with possibility of 0.4 to reduce the number of frontal faces, and further duplicate the images of $Medium$ and $Large$ subset by 2 and 5 times separately to keep the number of total images consistent. After that, we adjust the face images with different poses to a more balanced proportion.

Table 1. Data balance on CelebA.

Pose	None	Small	Medium	Large	All
Before	137,710	34,110	24,889	5,890	202,599
After	88,687	34,110	49,778	29,450	202,025

3.4. Other Implementations

Training dataset. Our FLDet is trained end-to-end on the joint dataset of WIDER FACE and CelebA. Since the data imbalance of these two datasets, we duplicate the WIDER FACE dataset by 10 times and integrate these balanced images into a new dataset, in which there are $128,800$ images from WIDER FACE training set and $202,025$ images from the processed CelebA. To increase the robustness of training data, each training image is sequentially processed by color distortion, random cropping, horizontal flipping and scale transformation, and finally get a 1024×1024 square subimages from the original image.

Hard negative mining. After anchor matching step, the positive and negative training samples are extremely imbalanced, because most of the anchors are negative, making

the training process slow and unstable. Thus we sort these samples by the loss values and choose the top ones to make sure that the ratio between negative and positive samples is almost 7:1.

Optimazition. We randomly initialize the parameters of the prediction layers with "$xavier$" method and the other layers with "$msra$" method. Besides, we also apply batch normalization operation [10] to all convolution layers except layers used for prediction and set "$relu$" as the activation function. We fine-tune the model using $Adam$ with 0.9 momentum, 0.0004 weight decay and batch size 64. The maximum number of iteration is $240k$ and we use 10^{-3} learning rate for the first $160k$ iterations, and continue training for $40k$ iterations with 10^{-4} and 10^{-5}, respectively. Our method is implemented in the Caffe [12] library.

4. Experiments

In this section, we first analyze our model in an ablative way, then evaluate it on the common face detection and face alignment benchmarks, and finally introduce the runtime efficiency.

4.1. Ablative Study

We evaluate our model on the FDDB and CelebA datasets by extensive experiments, the experiments are carried out on the same settings, except for specified changes to the components.

Ablative Settings. To have a better analysis of FLDet, we remove each component one after another to examine how each proposed component affects the final performance. Firstly, we replace the Rapidly Digested Backbone with the backbone proposed in FaceBoxes [37] and further cut off the feature fusion module. Then, we degrade the five-point anchors to a naive way, which directly predicts landmark locations like MTCNN [35]. Finally, the data augmentation applied in CelebA dataset is ablated.

Table 2. Ablative results on FDDB and CelebA datasets. mAP means true positive rate at $1,000$ false positives, MSE is the mean square error of predicted and ground-truth landmark locations.

Component	FLDet			
RDB				√
Five-point Anchor			√	√
Data Augmentation		√	√	√
FDDB (mAP)	93.6	94.2	94.6	**95.2**
CelebA (MSE)	5.8	5.3	5.1	**4.7**
CPU Speed (ms)	53.75	53.75	53.75	**51.02**

Combination of RDB and lightweight FPN is great. From the results listed in Tab. 2, we can see that integrated with specially designed lightweight yet powerful Feature Pyramid Network, our model presents a better performance with higher speed compared with the backbone applied in FaceBoxes [37].

Five-point anchor is better. The comparison between the second and third columns in Tab. 2 indicates that the five-point anchor not only effectively improves the landmark prediction, but also helps preserve the independency of face detection and alignment tasks, which is important to get the global optimal of each task.

Augmentation of CelebA dataset is crucial. Compared with directly applying CelebA dataset to train models, our FLDet receive 0.6% mAP increase and 0.5% MSE decrease owning to the augmentation operation, which helps to mitigate the gaps of image modality between WIDER FACE and CelebA datasets and further enhances the performance of both face detection and alignment.

4.2. Evaluation on Benchmark

We evaluate the proposed FLDet on the common face detection benchmarks including the Annotated Faces in the Wild (AFW), PASCAL Face and Face Detection Data Set and Benchmark (FDDB), as well as the face alignment benchmark, *i.e.*, the Annotated Facial Landmarks in the Wild (AFLW).

AFW dataset [41]. It has 205 images with 473 faces. We evaluate FLDet against some well-known works as well as commercial face detectors. As illustrated in Fig. 4, our FLDet presents superior performance.

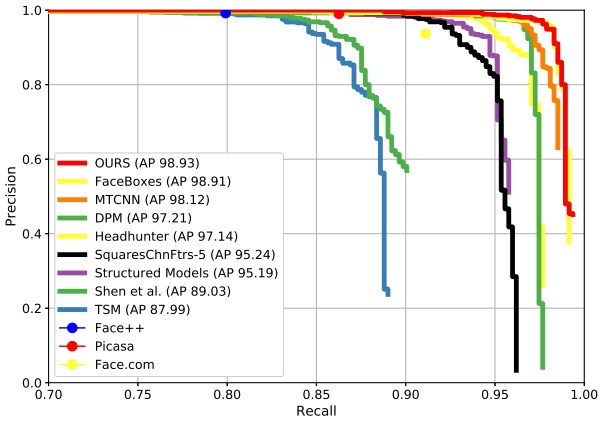


Figure 4. Precision-recall curves on the AFW dataset.

PASCAL FACE [30]. It is collected from the test set of PASCAL person layout dataset, consisting of $1,335$ faces with large face appearance and pose variations from 851 images. Fig. 5 shows the precision-recall curves on this dataset. Our method significantly outperforms all other methods (*e.g.*, FaceBoxes [37], MTCNN [35]) and commercial face detectors (*e.g.*, Face++, SkyBiometry and Picasa).

FDDB dataset [11]. It consists of $5,171$ faces in $2,845$ images. Since FDDB uses ellipse face annotations, while our

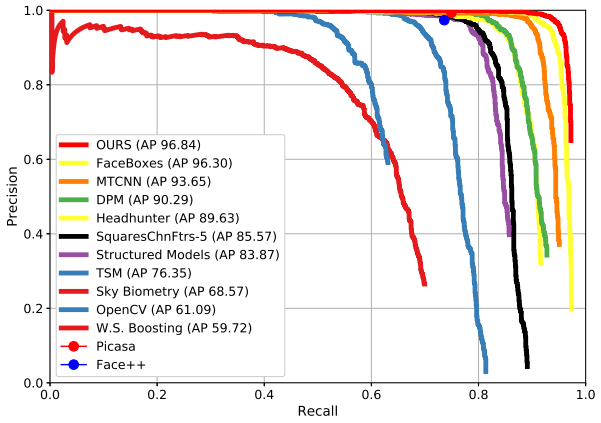
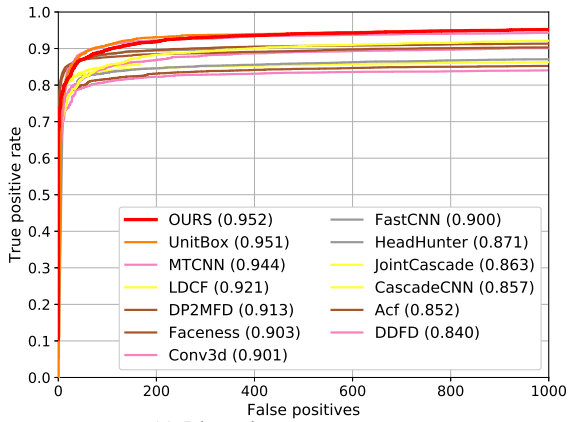
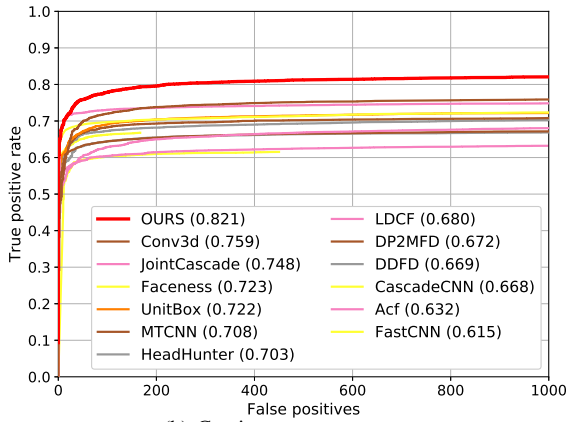


Figure 5. Precision-recall curves on the PASCAL Face dataset.

model outputs rectangle bounding boxes. For a more fair comparison, we train an elliptical regressor to transform our predicted bounding boxes to bounding ellipses. As illustrated in Fig. 6, our model achieves the best performance and outperforms all other methods [16, 24, 25, 33, 35].



(a) Discontinuous score curves



(b) Continuous score curves

Figure 6. Evaluation on the FDDB dataset.

AFLW dataset [14]. It contains 25,993 faces with up to 21 landmarks per image. The mean error is measured by the distances between the estimated landmarks and the ground-truth landmarks, and normalized with respect to the inter-ocular distance. Since some landmark annotations are missing in this dataset, we filter out these images and only conduct evaluation on the images with complete landmark annotations. Tab. 3 shows that our method presents superior performance on landmark localization.

Table 3. Evaluation on AFLW for face alignment. Performance is evaluated by MSE.

Method	left eye	right eye	nose tip	left mouth	right mouth	total
TSPM [41]	14.6	9.6	21.8	19.5	14.3	15.9
CDM [34]	10.1	10.9	14.9	14.6	15.0	13.1
ESR [2]	12.1	11.5	12.4	12.2	15.0	12.4
TCDCN [39]	7.9	8.1	8.7	7.6	7.5	8.0
MTCNN [35]	4.9	**4.8**	4.9	5.0	5.0	4.9
FLDet	**4.9**	4.9	**4.7**	**4.5**	**4.6**	**4.7**

4.3. Runtime Efficiency

During the inference phase, we first filter out output boxes by a confidence threshold of 0.05 and keep the top 400 boxes before applying NMS, then we perform NMS with jaccard overlap of 0.3 and keep the top 200 boxes. The inference time is measured by Titan X (Pascal) and Intel Xeon E5-2660v3@2.60GHz. Consequently, our FLDet can run at **20 FPS** on the CPU and can be further accelerated to **120 FPS** using a single GPU and has only **3.3 MB** in size.

5. Conclusion

In this paper, we propose a novel single-shot framework for joint face detection and alignment with superior performance on both speed and accuracy. The proposed FLDet is a specially designed lightweight yet powerful network with three main components: RDB, LFPN and MDM, among which the RDB enables FLDet to achieve real-time speed, the LFPN is used to integrate high-level features into low-level layers at the cost of little extra time overhead, and the MDM is designed to complete multi-task prediction on different detection layers. Besides, we also apply some data augmentation methods to tackle with the modality gaps between WIDER FACE and CelebA datasets so as to help promote the total performance. Consequently, our FLDet can run at 20 FPS on CPU for VGA-resolution images and can be further accelerated to 120 FPS on GPU devices with superior performance retained.

Acknowledgments

This work was supported by the Chinese National Natural Science Foundation Projects #61876178, #61806196, #61872367, #61572501.

References

[1] A. Bulat and G. Tzimiropoulos. Convolutional aggregation of local evidence for large pose face alignment. In *BMVC*, 2016. 1, 2

[2] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. *IJCV*, 2014. 2, 7

[3] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun. Joint cascade face detection and alignment. In *ECCV*, 2014. 1, 2

[4] C. Chi, S. Zhang, J. Xing, Z. Lei, S. Z. Li, and X. Zou. Selective refinement network for high performance face detection. In *AAAI*, 2019. 1

[5] J. Deng, G. Trigeorgis, Y. Zhou, and S. Zafeiriou. Joint multi-view face alignment in the wild. *arXiv*, 2017. 2

[6] C. Ding and D. Tao. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *TMAPI*, 2018. 1

[7] Z. Feng, J. Kittler, M. Awais, P. Huber, and X. Wu. Wing loss for robust facial landmark localisation with convolutional neural networks. *arXiv*, 2017. 5

[8] H. Han, A. K. Jain, S. Shan, and X. Chen. Heterogeneous face attribute estimation: a deep multi-task learning approach. *TPAMI*, 2017. 1

[9] P. Hu and D. Ramanan. Finding tiny faces. In *CVPR*, 2017. 2

[10] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv*, 2015. 6

[11] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical report, Technical Report, 2010. 6

[12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACMMM*, 2014. 6

[13] H. Jiang and E. Learned-Miller. Face detection with the faster r-cnn. In *FG*, 2017. 2

[14] M. Koestinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *ICCV workshop*, 2011. 7

[15] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua. A convolutional neural network cascade for face detection. In *CVPR*, 2015. 1, 2

[16] Y. Li, B. Sun, T. Wu, and Y. Wang. Face detection with end-to-end integration of a convnet and a 3d model. In *ECCV*, 2016. 7

[17] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 3

[18] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: single shot multibox detector. In *ECCV*, 2016. 2

[19] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015. 4, 5

[20] J. Lv, X. Shao, J. Xing, C. Cheng, X. Zhou, et al. A deep regression architecture with two-stage re-initialization for high performance facial landmark detection. In *CVPR*, 2017. 1, 2

[21] I. Masi, F. Chang, J. Choi, S. Harel, J. Kim, K. Kim, J. Leksut, S. Rawls, Y. Wu, T. Hassner, et al. Learning pose-aware models for pose-invariant face recognition in the wild. *TPAMI*, 2018. 1

[22] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *ECCV*, 2008. 2

[23] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis. SSH: single stage headless face detector. In *ICCV*, 2017. 2

[24] E. Ohn Bar and M. M. Trivedi. To boost or not to boost? on the limits of boosted trees for object detection. In *ICPR*, 2016. 7

[25] R. Ranjan, V. M. Patel, and R. Chellappa. A deep pyramid deformable part model for face detection. In *ICB*, 2015. 7

[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 2, 4

[27] X. Shi, S. Shan, M. Kan, S. Wu, and X. Chen. Real-time rotation-invariant face detection with progressive calibration networks. In *CVPR*, 2018. 2

[28] X. Tang, D. K. Du, Z. He, and J. Liu. Pyramidbox: A context-assisted single shot face detector. In *ECCV*, 2018. 1, 2

[29] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 2004. 1, 2

[30] J. Yan, X. Zhang, Z. Lei, and S. Z. Li. Face detection by structural models. *IVC*, 2014. 6

[31] J. Yang, Q. Liu, and K. Zhang. Stacked hourglass network for robust facial landmark localisation. In *CVPR workshop*, 2017. 1

[32] S. Yang, P. Luo, C. L. Chen, and X. Tang. Wider face: A face detection benchmark. In *CVPR*, 2016. 4, 5

[33] J. Yu, Y. Jiang, Z. Wang, Z. Cao, and T. Huang. Unitbox: An advanced object detection network. In *ACMMM*, 2016. 7

[34] X. Yu, J. Huang, S. Zhang, W. Yan, and D. N. Metaxas. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *ICCV*, 2014. 7

[35] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *SPL*, 2016. 1, 2, 6, 7

[36] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li. Single-shot refinement neural network for object detection. In *CVPR*, 2018. 2

[37] 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 *IJCB*, 2017. 3, 4, 6

[38] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li. S^3FD: single shot scale-invariant face detector. In *ICCV*, 2017. 2

[39] Z. Zhang, P. Luo, C. L. Chen, and X. Tang. Facial landmark detection by deep multi-task learning. In *European Conference on Computer Vision*, 2014. 7

[40] Z. Zhang, P. Luo, C. C. Loy, and X. Tang. Learning deep representation for face alignment with auxiliary attributes. *TPAMI*, 2016. 1, 2

[41] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, 2012. 6, 7

[42] C. Zhuang, S. Zhang, X. Zhu, Z. Lei, and S. Z. Li. Single shot attention-based face detector. In *CCBR*, 2018. 2