Multi-view Face Detection Using Deep Convolutional Neural Networks

Sachin Sudhakar Farfade
Yahoo
fsachin@yahoo-inc.com

Mohammad Saberian
Yahoo
saberian@yahoo-inc.com

Li-Jia Li
Yahoo
lijiali.vision@gmail.com

ABSTRACT

In this paper we consider the problem of multi-view face detection. While there has been significant research on this problem, current state-of-the-art approaches for this task require annotation of facial landmarks, e.g. TSM [25], or annotation of face poses [28, 22]. They also require training dozens of models to fully capture faces in all orientations, e.g. 22 models in HeadHunter method [22]. In this paper we propose Deep Dense Face Detector (DDFD), a method that does not require pose/landmark annotation and is able to detect faces in a wide range of orientations using a *single* model based on deep convolutional neural networks. The proposed method has minimal complexity; unlike other recent deep learning object detection methods [9], it does not require additional components such as segmentation, bounding-box regression, or SVM classifiers. Furthermore, we analyzed scores of the proposed face detector for faces in different orientations and found that 1) the proposed method is able to detect faces from different angles and can handle occlusion to some extent, 2) there seems to be a correlation between distribution of positive examples in the training set and scores of the proposed face detector. The latter suggests that the proposed method's performance can be further improved by using better sampling strategies and more sophisticated data augmentation techniques. Evaluations on popular face detection benchmark datasets show that our single-model face detector algorithm has similar or better performance compared to the previous methods, which are more complex and require annotations of either different poses or facial landmarks.

Categories and Subject Descriptors

I.4 [**IMAGE PROCESSING AND COMPUTER VISION**]: Applications

General Terms

Application

ICMR'15, June 23–26, 2015, Shanghai, China.

ACM 978-1-4503-3274-3/15/06 ...$15.00.
DOI: http://dx.doi.org/10.1145/2671188.2749408 .

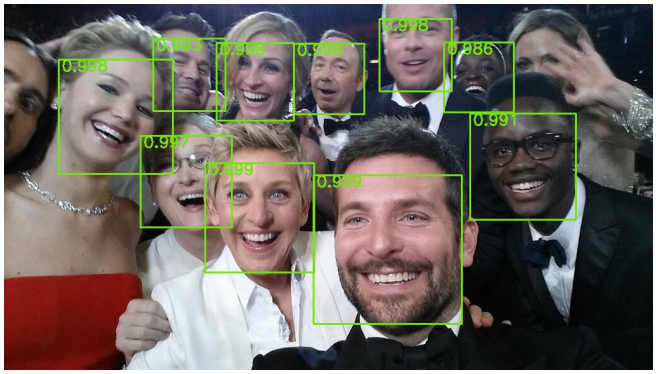

Figure 1: An example of user generated photos on social networks that contains faces in various poses, illuminations and occlusions. The bounding-boxes and corresponding scores show output of our proposed face detector.

Keywords

Face Detection, Convolutional Neural Network, Deep Learning

1. INTRODUCTION

With the wide spread use of smartphones and fast mobile networks, millions of photos are uploaded everyday to the cloud storages such as Dropbox or social networks such as Facebook, Twitter, Instagram, Google+, and Flicker. Organizing and retrieving relevant information from these photos is very challenging and directly impact user experience on those platforms. For example, users commonly look for photos that were taken at a particular location, at a particular time, or with a particular friend. The former two queries are fairly straightforward, as almost all of today's cameras embed time and GPS location into photos. The last query, i.e. contextual query, is more challenging as there is no explicit signal about the identities of people in the photos. The key for this identification is the detection of human faces. This has made low complexity, rapid and accurate face detection an essential component for cloud based photo sharing/storage platforms.

For the past two decades, face detection has always been an active research area in the vision community. The seminal work of Viola and Jones [40] made it possible to rapidly detect up-right faces in real-time with very low computational complexity. Their detector, called detector cascade,

consists of a sequence of simple-to-complex face classifiers and has attracted extensive research efforts. Moreover, detector cascade has been deployed in many commercial products such as smartphones and digital cameras. While cascade detectors can accurately find visible up-right faces, they often fail to detect faces from different angles, e.g. side view or partially occluded faces. This failure can significantly impact the performance of photo organizing software/applications since user generated content often contains faces from different angles or faces that are not fully visible; see for example Figure 1. This has motivated many works on the problem of multi-view face detection over the past two decades. Current solutions can be summarized into three categories:

- Cascade-based: These methods extend the Viola and Jones detector cascade. For example, [41] proposed to train a detector cascade for each view of the face and combined their results at the test time. Recently, [22] combined this method with integral channel features [3] and soft-cascade [1], and showed that by using 22 cascades, it is possible to obtain state-of-the-art performance for multi-view face detection. This approach, however, requires face orientation annotations. Moreover its complexity in training and testing increases linearly with the number of models. To address the computational complexity issue, Viola and Jones [39] proposed to first estimate the face pose using a tree classifier and then run the cascade of corresponding face pose to verify the detection. While improving the detection speed, this method degrades the accuracy because mistakes of the initial tree classifier are irreversible. This method is further improved by [13, 12] where, instead of one detector cascade, several detectors are used after the initial classifier. Finally, [35] and [28] combined detector cascade with multiclass boosting and proposed a method for multiclass/multi-view object detection.

- DPM-based: These methods are based on the deformable part models technique [5] where a face is defined as a collection of its parts. The parts are defined via unsupervised or supervised training, and a classifier, latent SVM, is trained to find those parts and their geometric relationship. These detectors are robust to partial occlusion because they can detect faces even when some of the parts are not present. These methods are, however, computationally intensive because 1) they require solving a latent SVM for each candidate location and 2) multiple DPMs have to be trained and combined to achieve the state-of-the-art performance [22, 25]. Moreover, in some cases DPM-based models require annotation of facial landmarks for training, e.g [25].

- Neural-Network-based: There is a long history of using neural networks for the task of face detection [38, 37, 27, 8, 7, 6, 26, 11, 24, 23]. In particular, [38] trained a two-stage system based on convolutional neural networks. The first network locates rough positions of faces and the second network verifies the detection and makes more accurate localization. In [27], the authors trained multiple face detection networks and combined their output to improve the performance. [8] trained a single multi-layer network for face detection. The trained network is able to partially handle different poses and rotation angles. More recently, [23] proposed to train a neural network jointly for face detection and pose estimation. They showed that this joint learning scheme can significantly improve performance of both detection and pose estimation. Our method follows the works in [8, 23] but constructs a deeper CNN for face detection.

The key challenge in multi-view face detection, as pointed out by Viola and Jones [39], is that learning algorithms such as Boosting or SVM and image features such as HOG or Haar wavelets are not strong enough to capture faces of different poses and thus the resulted classifiers are *hopelessly inaccurate*. However, with recent advances in deep learning and GPU computation, it is possible to utilize the high capacity of deep convolutional neural networks for feature extraction/classification, and train a *single* model for the task of multi-view face detection.

Deep convolutional neural network has recently demonstrated outstanding performance in a variety of vision tasks such as face recognition [34, 30], object classification [19, 31], and object detection [9, 29, 18, 32]. In particular [19] trained an 8-layered network, called AlexNet, and showed that deep convolutional neural networks can significantly outperform other methods for the task of large scale image classification. For the task of object detection, [9] proposed R-CNN method that uses an image segmentation technique, selective search [36], to find candidate image regions and classify those candidates using a version of AlexNet that is fine-tuned for objects in the PASCAL VOC dataset. More recently, [33] improved R-CNN by 1) augmenting the selective search proposals with candidate regions from multibox approach [4], and 2) replacing 8-layered AlexNet with a much deeper CNN model of GoogLeNet [31]. Despite state-of-the-art performance, these methods are computationally suboptimal because they require evaluating a CNN over more than $2,000$ overlapping candidate regions independently. To address this issue, [18] recently proposed to run the CNN model on the full image once and create a feature pyramid. The candidate regions, obtained by selective search, are then mapped into this feature pyramid space. [18] then uses spatial pyramid pooling [20] and SVM on the mapped regions to classify candidate proposals. Beyond region-based methods, deep convolutional neural networks have also been used with sliding window approach, e.g. OverFeat [29] and deformable part models [10] for object detection and [17] for human pose estimation. In general, for object detection these methods still have an inferior performance compared to region-based methods such as R-CNN [9] and [33]. However, in our face detection experiments we found that the region-based methods are often very slow and result in relatively weak performance.

In this paper, we propose a method based on deep learning, called Deep Dense Face Detector (DDFD), that does not require pose/landmark annotation and is able to detect faces in a wide range of orientations using a *single* model. The proposed method has minimal complexity because unlike recent deep learning object detection methods such as [9], it does not require additional components for segmentation, bounding-box regression, or SVM classifiers. Compared to previous convolutional neural-network-based face detectors such as [8], our network is deeper and is trained

on a significantly larger training set. In addition, by analyzing detection confidence scores, we show that there seems to be a correlation between the distribution of positive examples in the training set and the confidence scores of the proposed detector. This suggests that the performance of our method can be further improved by using better sampling strategies and more sophisticated data augmentation techniques. In our experiments, we compare the proposed method to a deep learning based method, R-CNN, and several cascade and DPM-based methods. We show that DDFD can achieve similar or better performance even without using pose annotation or information about facial landmarks.

2. PROPOSED METHOD

In this section, we provide details of the algorithm and training process of our proposed face detector, called Deep Dense Face Detector (DDFD). The key ideas are 1) leverage the high capacity of deep convolutional networks for classification and feature extraction to learn a single classifier for detecting faces from multiple views and 2) minimize the computational complexity by simplifying the architecture of the detector.

We start by fine-tuning AlexNet [19] for face detection. For this we extracted training examples from the AFLW dataset [21], which consists of 21K images with 24K face annotations. To increase the number of positive examples, we randomly sampled sub-windows of the images and used them as positive examples if they had more than a 50% IOU (intersection over union) with the ground truth. For further data augmentation, we also randomly flipped these training examples. This resulted in a total number of 200K positive and and 20 millions negative training examples. These examples were then resized to 227×227 and used to fine-tune a pre-trained AlexNet model [19]. For fine-tuning, we used 50K iterations and batch size of 128 images, where each batch contained 32 positive and 96 negative examples.

Using this fine-tuned deep network, it is possible to take either region-based or sliding window approaches to obtain the final face detector. In this work we selected a sliding window approach because it has less complexity and is independent of extra modules such as selective search. Also, as discussed in the experiment section, this approach leads to better results as compared to R-CNN.

Our face classifier, similar to AlexNet [19], consists of 8 layers where the first 5 layers are convolutional and the last 3 layers are fully-connected. We first converted the fully-connected layers into convolutional layers by reshaping layer parameters [14]. This made it possible to efficiently run the CNN on images of any size and obtain a heat-map of the face classifier. An example of a heat-map is shown in Figure 2-right. Each point in the heat-map shows the CNN response, the probability of having a face, for its corresponding 227×227 region in the original image. The detected regions were then processed by non-maximal suppression to accurately localize the faces. Finally, to detect faces of different sizes, we scaled the images up/down and obtained new heat-maps. We tried different scaling schemes and found that rescaling image 3 times per octave gives reasonably good performance. This is interesting as many of the other methods such as [22, 2] requires a significantly larger number of resizing per octave, e.g. 8. Note that, unlike R-CNN [9], which uses SVM classifier to obtain the final score, we removed the SVM module and found that the network output are informative enough for the task of face detection.

Face localization can be further improved by using a bounding-box regression module similar to [29, 9]. In our experiment, however, adding this module degraded the performance. Therefore, compared to the other methods such as R-CNN [9], which uses selective search, SVM and bounding-box regression, or DenseNet [10], which is based on the deformable part models, our proposed method (DDFD) is fairly simple. Despite its simplicity, as shown in the experiments section, DDFD can achieve state-of-the-art performance for face detection.

2.1 Detector Analysis

In this section, we look into the scores of the proposed face detector and observe that there seems to be a correlation between those scores and the distribution of positive examples in the training set. We can later use this hypothesis to obtain better training set or to design better data augmentation procedures and improve performance of DDFD.

We begin by running our detector on a variety of faces with different in-plane and out-of-plane rotations, occlusions and lighting conditions (see for example Figure 1, Figure 2-left and Figure 3). First, note that in all cases our detector is able to detect the faces except for the two highly occluded ones in Figure 1. Second, for almost all of the detected faces, the detector's confidence score is pretty high, close to 1. Also as shown in the heat-map of Figure 2-right, the scores are close to zero for all other regions. This shows that DDFD has very strong discriminative power, and its output can be used directly without any post-processing steps such as SVM, which is used in R-CNN [9]. Third, if we compare the detector scores for faces in Figure 2-left, it is clear that the up-right frontal face in the bottom has a very high score of 0.999 while faces with more in-plane rotation have less score. Note that these scores are output of a sigmoid function, i.e. probability (soft-max) layer in the CNN, and thus small changes in them reflects much larger changes in the output of the previous layer. It is interesting to see that the scores decrease as the in-plane rotation increases. We can see the same trend for out-of-plane rotated faces and occluded faces in Figures 1 and 3. We hypothesize that this trend in the scores is not because detecting rotated face are more difficult but it is because of lack of good training examples to represent such faces in the training process.

To examine this hypothesis, we looked into the face annotations for AFLW dataset [21]. Figure 4 shows the distribution of the annotated faces with regards to their in-plane, pitch (up and down) and yaw (left to right) rotations. As shown in this figure, the number of faces with more than 30 degrees out-of-plane rotation is significantly lower than the faces with less than 30 degree rotation. Similarly, the number of faces with yaw or pitch less than 50 degree is significantly larger than the other ones. Given this skewed training set, it not surprising that the fine-tuned CNN is more confident about up-right faces. This is because the CNN is trained to minimize the risk of the soft-max loss function

$$\mathcal{R} = \sum_{x_i \in \mathcal{B}} \log \left[prob(y_i | x_i) \right], \tag{1}$$

where $\mathcal{B}$ is the example batch that is used in an iteration of stochastic gradient descent and y_i is the label of example x_i. The sampling method for selecting examples in $\mathcal{B}$ can

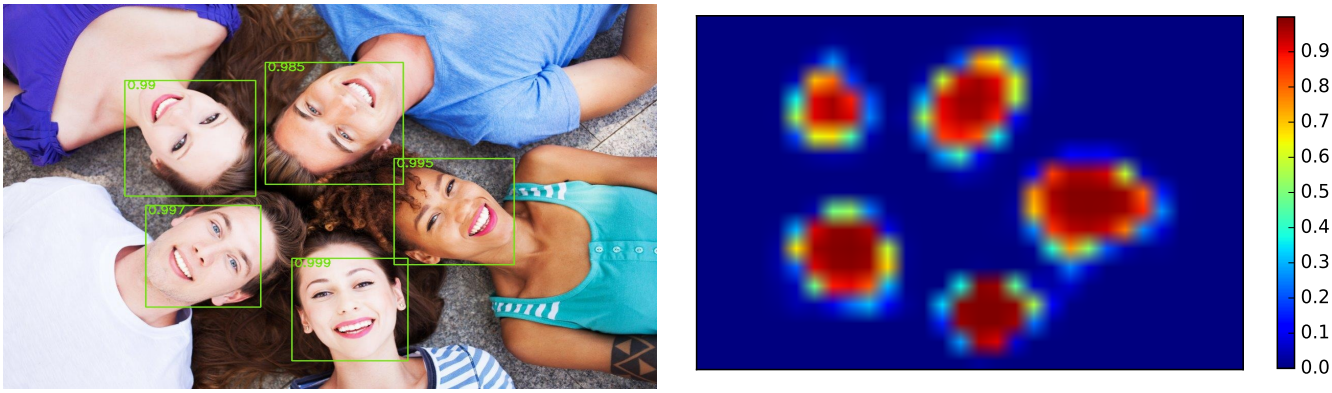

Figure 2: left) an example image with faces in different in-plane rotations. It also shows output of our proposed face detector after NMS along with corresponding confidence score for each detection. right) heat-map for the response of DDFD scores over the image.

significantly hurt performance of the final detector. In an extreme case if $\mathcal{B}$ never contains any example of a certain class, the CNN classifier will never learn the attributes of that class.

In our implementation $|\mathcal{B}| = 128$ and it is collected by randomly sampling the training set. However, since the number of negative examples are 100 times more than the number of positive examples, a uniform sampling will result in only about 2 positive examples per batch. This significantly degrades the chance of the CNN to distinguish faces from non-faces. To address this issue, we enforced one quarter of each batch to be positive examples, where the positive examples are uniformly sampled from the pool of positive training samples. But, as illustrated in Figure 4, this pool is highly skewed in different aspects, e.g. in-plane and out-of-plane rotations. The CNN is therefore getting exposed with more up-right faces; it is thus not surprising that the fine-tuned CNN is more confident about the up-right faces than the rotated ones. This analysis suggests that the key for improving performance of DDFD is to ensure that all categories of the training examples have similar chances to contribute in optimizing the CNN. This can be accomplished by enforcing population-based sampling strategies such as increasing selection probability for categories with low population.

Similarly, as shown in Figure 1, the current face detector still fails to detect faces with heavy occlusions. Similar to the issue with rotated faces, we believe that this problem can also be addressed through modification of the training set. In fact, most of the face images in the AFLW dataset [21] are not occluded, which makes it difficult for a CNN to learn that faces can be occluded. This issue can be addressed by using more sophisticated data augmentation techniques such as occluding parts of positive examples. Note that simply covering parts of positive examples with black/white or noise blocks is not useful as the CNN may learn those artificial patterns.

To summarize, the proposed face detector based on deep CNN is able to detect faces from different angles and handle occlusion to some extent. However, since the training set is skewed, the network is more confident about up-right faces and better results can be achieved by using better sampling strategies and more sophisticated data augmentation techniques.

3. EXPERIMENTS

We implemented the proposed face detector using the Caffe library [16] and used its pre-trained Alexnet [19] model for fine-tuning. For further details on the training process of our proposed face detector please see section 2. After converting fully-connected layers to convolutional layers [14], it is possible to get the network response (heat-map) for the whole input image in one call to Caffe code. The heat-map shows the scores of the CNN for every 227×227 window with a stride of 32 pixels in the original image. We directly used this response for classifying a window as face or background. To detect faces of smaller or larger than 227×227, we scaled the image up or down respectively.

We tested our face detection approach on PASCAL Face [42], AFW [25] and FDDB [15] datasets. For selecting and tuning parameters of the proposed face detector we used the PASCAL Face dataset. PASCAL Face dataset consists of 851 images and 1341 annotated faces, where annotated faces can be as small as 35 pixels. AFW dataset is built using Flickr images. It has 205 images with 473 annotated faces, and its images tend to contain cluttered background with large variations in both face viewpoint and appearance (aging, sunglasses, make-ups, skin color, expression etc.). Similarly, FDDB dataset [15] consists of 5171 annotated faces with 2846 images and contains occluded, out-of-focus, and low resolution faces. For evaluation, we used the toolbox provide by [22] with corrected annotations for PASCAL Face and AFW datasets and the original annotations of FDDB dataset.

We started by finding the optimal number of scales for the proposed detector using PASCAL dataset. We upscaled images by factor of 5 to detect faces as small as $227/5 = 45$ pixels. We then down scaled the image with by a factor, f_s, and repeated the process until the minimum image dimension is less than 227 pixels. For the choice of f_s, we chose $f_s \in \{\sqrt{0.5} = 0.7071, \sqrt[3]{0.5} = 0.7937, \sqrt[5]{0.5} = 0.8706, \sqrt[7]{0.5} = 0.9056\}$; Figure 5 shows the effect of this parameter on the precision and recall of our face detector (DDFD). Decreasing f_s allows the detector to scan the image finer and increases the computational time. According to Figure 5, it seems that these choices of f_s has little impact on the performance of the detector. Surprisingly, $f_s = \sqrt[3]{0.5}$ seems to have slightly better performance although it does

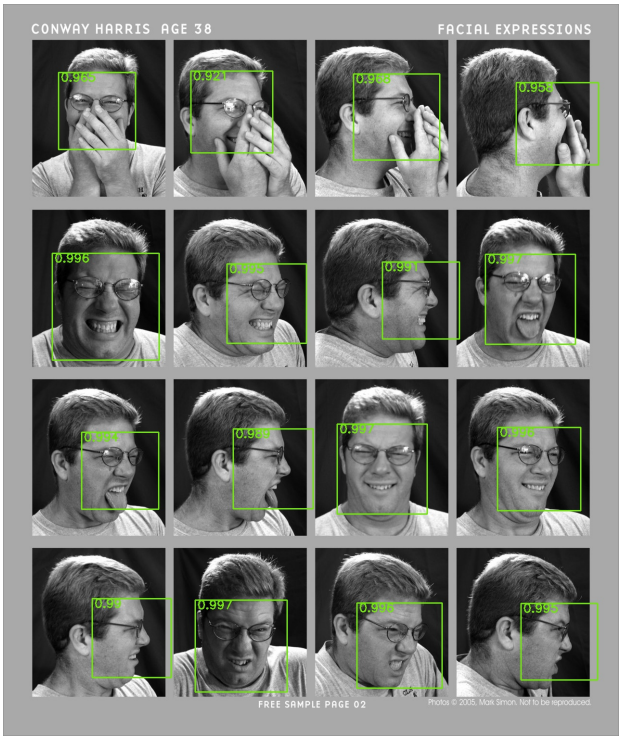

Figure 3: A set of faces with different out-of-plane rotations and occlusions. The figure also shows output of our proposed face detector after NMS along with the corresponding confidence score for each detection.

not scan the image as thorough as $f_s = \sqrt[5]{0.5}$ or $f_s = \sqrt[7]{0.5}$. Based on this experiment we use $f_s = \sqrt[3]{0.5}$ for the rest of this paper.

Another component of our system is the non-maximum suppression module (NMS). For this we evaluated two different strategies:

- **NMS-max**: we find the window of the maximum score and remove all of the bounding-boxes with an IOU (intersection over union) larger than an overlap threshold.
- **NMS-avg**: we first filter out windows with confidence lower than 0.2. We then use groupRectangles function of OpenCV to cluster the detected windows according to an overlap threshold. Within each cluster, we then removed all windows with score less than 90% of the maximum score of that cluster. Next we averaged the locations of the remaining bounding-boxes to get the detection window. Finally, we used the maximum score of the cluster as the score of the proposed detection.

We tested both strategies and Figure 6 shows the performance of each strategy for different overlap thresholds. As shown in this figure, performance of both methods vary significantly with the overlap threshold. An overlap threshold of 0.3 gives the best performance for NMS-max while, for NMS-avg 0.2 performs the best. According to this figure, NMS-avg has better performance compared to NMS-max in terms of average precision.

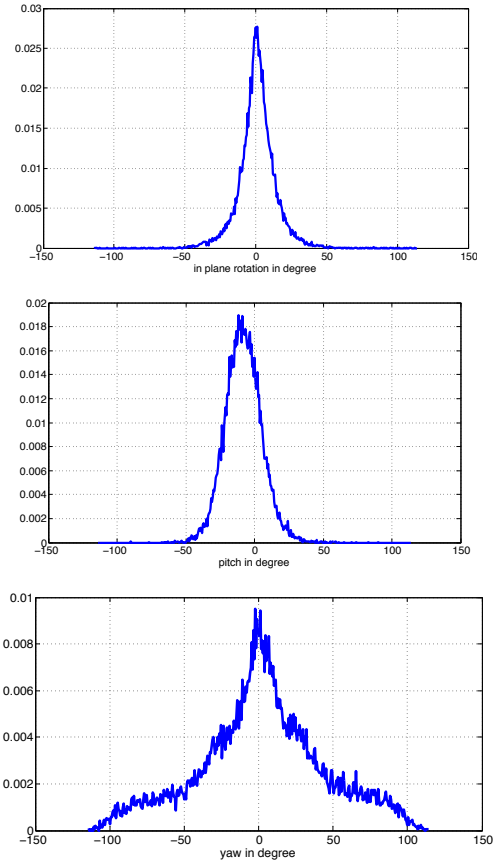

Figure 4: Histogram of faces in AFLW dataset based on their top) in-plane, middle) pitch (up and down) and bottom) yaw(left to right) rotations.

Finally, we examined the effect of a bounding-box regression module for improving detector localization. The idea is to train regressors to predict the difference between the locations of the predicted bounding-box and the ground truth. At the test time these regressors can be used to estimate the location difference and adjust the predicted bounding-boxes accordingly. This idea has been shown to improve localization performance in several methods including [5, 29, 4]. To train our bounding-box regressors, we followed the algorithm of [9] and Figure 7 shows the performance of our detector with and without this module. As shown in this figure, surprisingly, adding a bounding-box regressors degrades the performance for both NMS strategies. Our analysis revealed that this is due to the mismatch between the annotations of the training set and the test set. This mismatch is mostly for side-view faces and is illustrated in Figure 8. In addition to degrading performance of bounding-box regression module, this mismatch also leads to false miss-detections in the evaluation process.

3.1 Comparison with R-CNN

R-CNN [9] is one of the current state-of-the-art methods for object detection. In this section we compare our proposed detector with R-CNN and its variants.

We started by fine-tuning AlexNet for face detection using the process described in section 2. We then trained a SVM

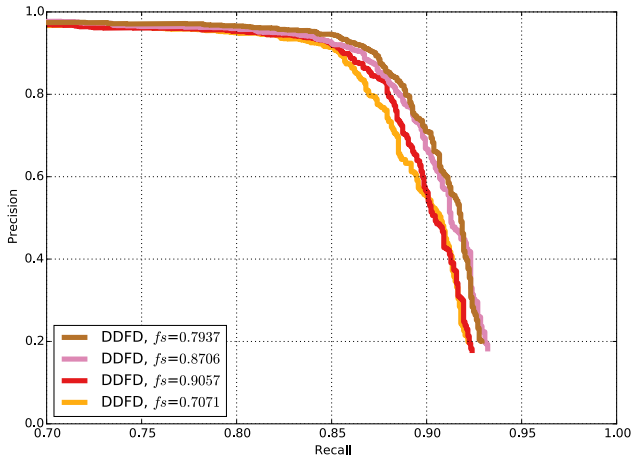

Figure 5: Effect of scaling factor on precision and recall of the detector.

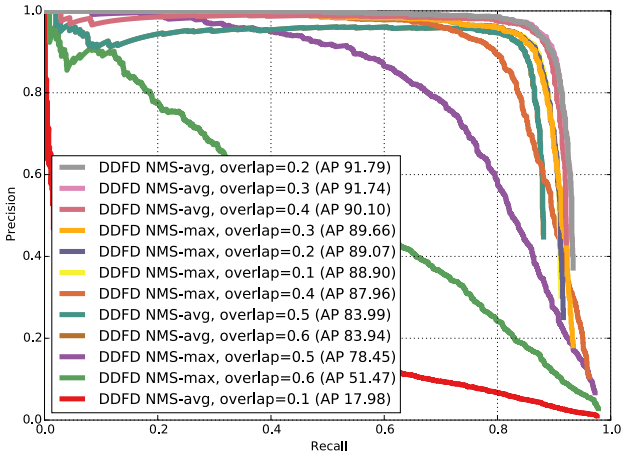

Figure 6: Effect of different NMS strategies and their overlap thresholds.

classifier for face classification using output of the seventh layer ($fc7$ features). We also trained a bounding-box regression unit to further improve the results and used NMS-max for final localization. We repeated this experiment on a version of AlexNet that is fine-tuned for PASCAL VOC 2012 dataset and is provided with R-CNN code. Figure 9 compares the performance of our detector with different NMS strategies along with the performance of R-CNN methods with and without bounding-box regression. As shown in this figure, it is not surprising that performance of the detectors with AlexNet fine-tuned for faces (Face-FT) are better than the ones that are fine-tuned with PASCAL-VOC objects (VOC-FT). In addition, it seems that bounding-box regression can significantly improve R-CNN performance. However, even the best R-CNN classifier has significantly inferior performance compared to our proposed face detector independent of the NMS strategy. We believe the inferior performance of R-CNN are due to 1) the loss of recall since selective search may miss some of face regions and 2) loss in localization since bounding-box regression is not perfect and may not be able to fully align the segmentation bounding-

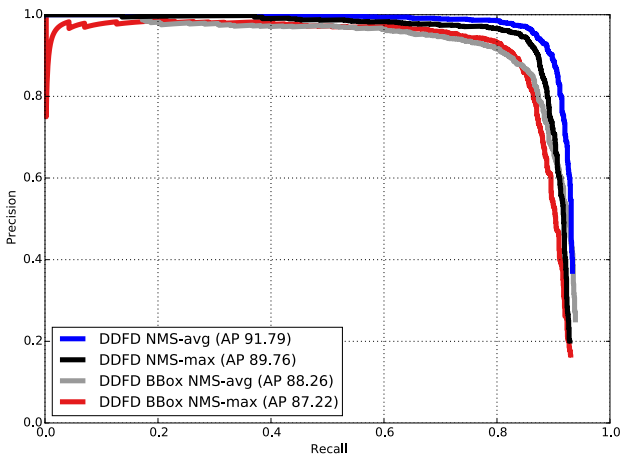

Figure 7: Performance of the proposed face detector with and without bounding-box regression.

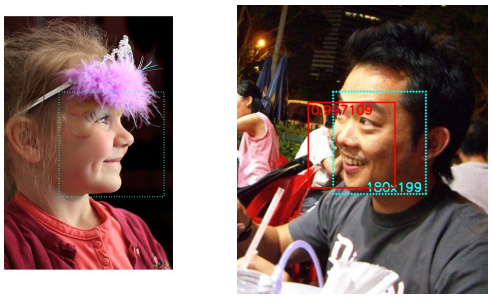

Figure 8: Annotation of a side face in left) training set and right) test set. The red bounding-box is the predicted bounding-box by our proposed detector. This detection is counted as a false positive as its IOU with ground truth is less than 50%.

boxes, provided by selective search [36], with the ground truth.

3.2 Comparisons with state-of-the-art

In this section we compare the performance of our proposed detector with other state-of-the-art face detectors using publicly available datasets of PASCAL faces [42], AFW [25] and FDDB [15]. In particular, we compare our method with 1) DPM-based methods such as structural model [42] and TSM [25] and 2) cascade-based method such as head hunter [22]. Figures 10 and 11 illustrate this comparison. Note that these comparison are not completely fair as most of the other methods such as DPM or HeadHunter use extra information of view point annotation during the training. As shown in these figures our single model face detector was able to achieve similar or better results compared to the other state-of-the-art methods, without using pose annotation or information about facial landmarks.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a face detection method based on deep learning, called Deep Dense Face Detector (DDFD). The proposed method does not require pose/landmark annotation and is able to detect faces in a wide range of ori-

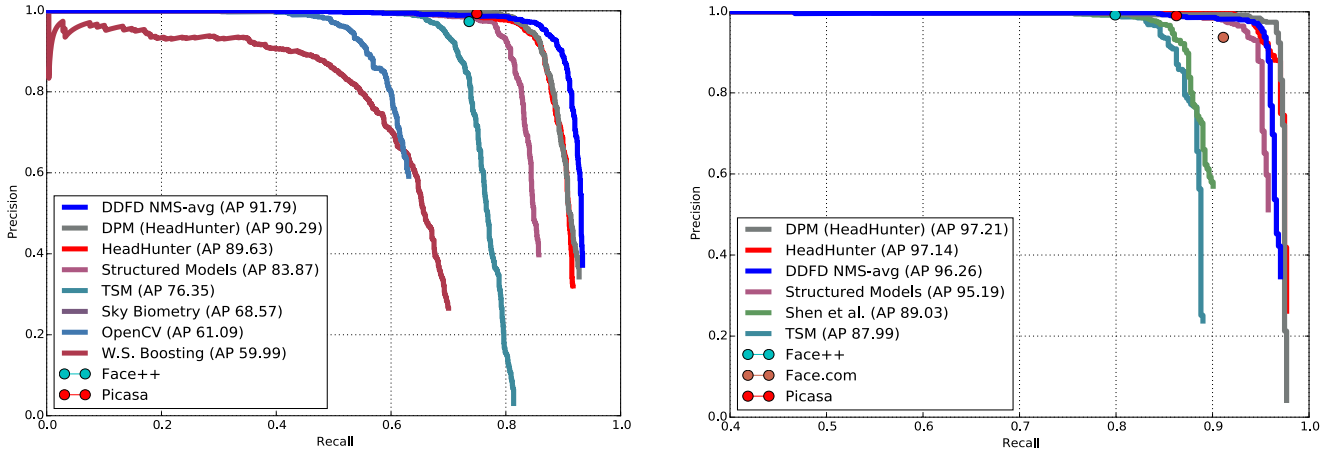

Figure 10: Comparison of different face detectors on left) PASCAL faces and right) AFW dataset.

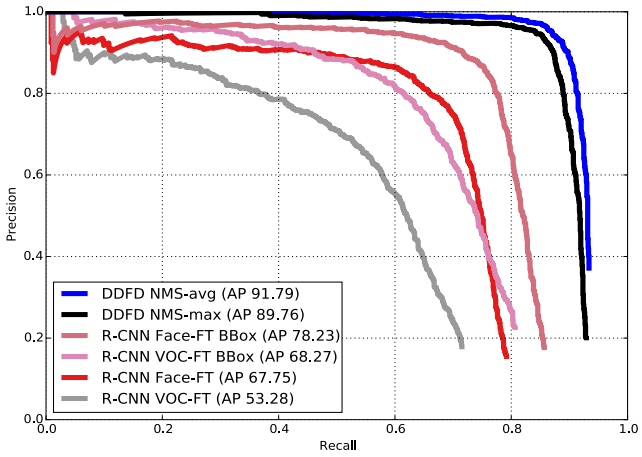

Figure 9: Comparison of our face detector, DDFD, with different R-CNN face detectors.

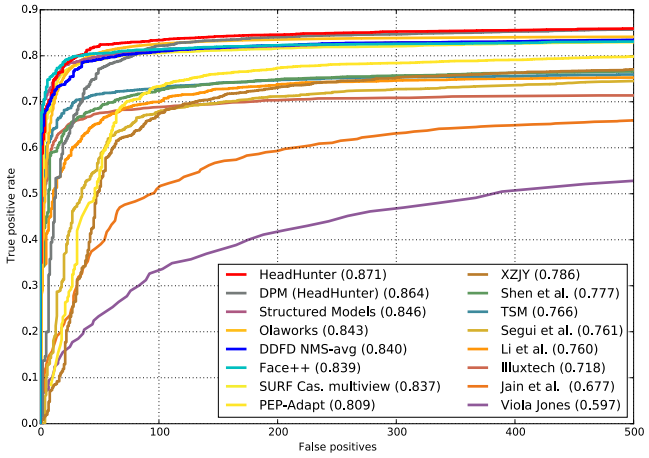

Figure 11: Comparison of different face detectors on FDDB dataset.

entations using a *single* model. In addition, DDFD is independent of common modules in recent deep learning object detection methods such as bounding-box regression, SVM, or image segmentation. We compared the proposed method with R-CNN and other face detection methods that are developed specifically for multi-view face detection e.g. cascade-based and DPM-based. We showed that our detector is able to achieve similar or better results even without using pose annotation or information about facial landmarks. Finally, we analyzed performance of our proposed face detector on a variety of face images and found that there seems to be a correlation between distribution of positive examples in the training set and scores of the proposed detector. In future we are planning to use better sampling strategies and more sophisticated data augmentation techniques to further improve performance of the proposed method for detecting occluded and rotated faces.

5. REFERENCES

[1] L. Bourdev and J. Brandt. Robust object detection via soft cascade. In *Proceedings of CVPR*, 2005.

[2] P. Dollár, R. Appel, S. Belongie, and P. Perona. Fast feature pyramids for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[3] P. Dollar, Z. Tu, P. Perona, and S. Belongie. Integral channel features. In *Proceedings of the British Machine Vision Conference*, 2009.

[4] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. 2014.

[5] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proceedings of CVPR*, 2008.

[6] C. Garcia and M. Delakis. A Neural Architecture for Fast and Robust Face Detection. In *Proceedings of IEEE-IAPR International Conference on Pattern Recognition*, Aug. 2002.

[7] C. Garcia and M. Delakis. Training Convolutional Filters for Robust Face Detection. In *Proceedings of IEEE International Workshop of Neural Networks for Signal Processing*, Sept. 2003.

[8] C. Garcia and M. Delakis. Convolutional face finder: a neural architecture for fast and robust face detection.

IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of CVPR*, 2014.

[10] R. B. Girshick, F. N. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *CoRR*, 2014.

[11] P. E. Hadjidoukas, V. V. Dimakopoulos, M. Delakis, and C. Garcia. A high-performance face detection system using openmp. *Concurrency and Computation: Practice and Experience*, 2009.

[12] C. Huang, H. Ai, Y. Li, and S. Lao. Vector boosting for rotation invariant multi-view face detection. In *Proceedings of ICCV*, 2005.

[13] C. Huang, H. Ai, Y. Li, and S. Lao. High-performance rotation invariant multiview face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[14] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*, 2014.

[15] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[16] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[17] Y. L. Jonathan J. Tompson, Arjun Jain and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Proceedings of NIPS*, 2014.

[18] S. R. Kaiming He, Xiangyu Zhang and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Proceedings of ECCV*, 2014.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of NIPS*, 2012.

[20] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of CVPR*, 2006.

[21] P. M. R. Martin Koestinger, Paul Wohlhart and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *Proceedings of IEEE International Workshop on Benchmarking Facial Image Analysis Technologies*, 2011.

[22] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *Proceedings of ECCV*, 2014.

[23] M. Osadchy, Y. L. Cun, and M. L. Miller. Synergistic face detection and pose estimation with energy-based model. In *Proceedings of NIPS*, 2005.

[24] R. Osadchy, M. Miller, and Y. LeCun. Synergistic face detection and pose estimation with energy-based model. In *Proceedings of NIPS*, 2004.

[25] D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings of CVPR*, 2012.

[26] S. Roux, F. Mamalet, and C. Garcia. Embedded convolutional face finder. In *Proceedings of IEEE International Conference on Multimedia and Expo*, 2006.

[27] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. In *Proceedings of CVPR*, 1996.

[28] M. Saberian and N. Vasconcelos. Multi-resolution cascades for multiclass object detection. In *Proceedings of NIPS*. 2014.

[29] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *Proceedings of International Conference on Learning Representations*, 2014.

[30] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *Proceedings of NIPS*. 2014.

[31] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *CoRR*, 2014.

[32] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. *CoRR*, 2014.

[33] C. Szegedy, S. Reed, D. Erhan, and D. Anguelov. Scalable, high-quality object detection. *CoRR*, 2014.

[34] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of CVPR*, 2014.

[35] A. Torralba, K. Murphy, and W. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[36] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 2013.

[37] R. Vaillant, C. Monrocq, and Y. LeCun. An original approach for the localisation of objects in images. In *Proceedings of International Conference on Artificial Neural Networks*, 1993.

[38] R. Vaillant, C. Monrocq, and Y. LeCun. Original approach for the localisation of objects in images. *IEE Proc on Vision, Image, and Signal Processing*, 1994.

[39] M. Viola and P. Viola. Fast multi-view face detection. In *Proceedings of CVPR*, 2003.

[40] P. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 2004.

[41] B. Wu, H. Ai, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.

[42] J. Yan, X. Zhang, Z. Lei, and S. Li. Face detection by structural models.