applied sciences

MDPI

Article

An Improved Neural Network Cascade for Face Detection in Large Scene Surveillance

Chengbin Peng [1], Wei Bu [1,2,*], Jiangjian Xiao [1], Ka-chun Wong [3] and Minmin Yang [4]

1 Ningbo Institute of Industrial Technology, Chinese Academy of Sciences, Ningbo 315201, China; pengchengbin@nimte.ac.cn (C.P.); xiaojj@nimte.ac.cn (J.X.)
2 School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China
3 Department of Computer Science, City University of Hong Kong, Kowloon Tong, Hong Kong, China; kc.w@cityu.edu.hk
4 School of Electronic and Information Engineering, Ningbo University of Technology, Ningbo 315201, China; laurayangminmin@163.com
* Correspondence: buwei@nimte.ac.cn; Tel.: +86-574-8668-8048

Received: 30 September 2018; Accepted: 8 November 2018; Published: 11 November 2018

Abstract: Face detection for security cameras monitoring large and crowded areas is very important for public safety. However, it is much more difficult than traditional face detection tasks. One reason is, in large areas like squares, stations and stadiums, faces captured by cameras are usually at a low resolution and thus miss many facial details. In this paper, we improve popular cascade algorithms by proposing a novel multi-resolution framework that utilizes parallel convolutional neural network cascades for detecting faces in large scene. This framework utilizes the face and head-with-shoulder information together to deal with the large area surveillance images. Comparing with popular cascade algorithms, our method outperforms them by a large margin.

Keywords: neural network; network cascades; large scene face detection

1. Introduction

Face detection is one of the most classic problems in computer vision. It can be widely used in many areas, such as face recognition [1], people counting [2,3], eye movement tracking [4], etc. Many approaches for face detection have been proposed. Some approaches use channel feature-based methods [5,6] that are able to encode each image channel as rich information in a simple form such as gradient magnitude and oriented gradient histograms. Some approaches consider faces as a combination of facial parts and convert the face detection problem as the part detection problem [7,8].

Some other approaches use a cascade of classifiers for face detection. In many circumstances, a cascade framework is preferable for face detection because it can utilize a set of weak classifiers to improve accuracy, and it can sometimes reject negative samples in the early stage to improve efficiency. For example, the Viola–Jones face detector adopts simple Haar-like features, allowing fast evaluation, and it uses boosted cascade to construct an ensemble of the simple features to achieve an accurate face detector [9]. After that, a cascade structure became a popular and effective framework for practical face detection. Many improvements have been made based on the Viola–Jones face detector during the past few years, and most of them adopt more complex features to improve the detection performance [10–12].

However, if we apply the aforementioned face detection algorithms to images captured from surveillance cameras for large and crowded places, these algorithms could have very poor performance. This is because these face detection algorithms are designed for relatively large faces in high resolution images, which is not possible for large area surveillance due to hardware and bandwidth costs. For large scene surveillance, surveillance cameras are generally installed at relatively high positions in order to obtain a large field of view, and thus far from the faces. Therefore, the captured faces are mostly at a small scale. In addition, the facial details in this situation are not clear and are heavily affected by background and illumination.

Therefore, in this work, we focus on improving neural network cascade for large scene surveillance and tackle it by designing a novel multi-scale cascade that is able to exploit head-shoulder information. It can be useful in many applications. For example, recently, public safety has attracted more and more attention. Many violent incidents and terrorist attacks happened in a number of countries. Among those, a large amount of deaths and injuries mostly occurred in large and crowded places, such as squares, stations, stadiums, theaters, etc. If we could accurately detect faces for individuals in these areas, we would be able to direct high resolution face recognition cameras with telephoto lenses to take pictures of each face of interest exactly. This can help us identify suspicious criminals or terrorists in advance, and consequently improve public safety. Face detection for large scene surveillance can also help in other tasks, for example estimating people densities.

2. Related Work

Many face detection approaches have been proposed over the past decades [13,14]. A recent study shows that the convolutional neural network (CNN)-based [15] deep learning algorithms have achieved great success in many computer vision areas, such as object classification [16], object segmentation [17–19], as well as face detection. This is because the traditional hand-crafted features, like SIFT [20], HoGs [21], LBPs [22–24] and ICF [5,25,26], are not descriptive enough for face detection in practice, but CNNs can automatically learn features to capture complex visual variations by leveraging a large amount of training data.

On the other hand, the cascade framework is able to reduce prediction time by rejecting negative samples in an early stage. The traditional cascade framework adopts relatively simple classifiers [9,27,28]. To improve performance, recently, cascade convolutional neural networks, which use convolutional neural networks as the basic classifiers, have been proposed. Li et al. used a cascade framework with CNNs (CascadeCNN) for face detection [11]. Later, Zhang et al. proposed a multi-task cascaded convolutional network (MTCNN) for face detection by taking facial landmarks into account [12].

In this work, we propose an improved convolutional neural network cascade that is able to adopt head-should information to make face detection possible in large scene surveillance.

3. Cascade Framework with Head-Shoulder Information

3.1. Overall Framework

In this part, we first use a simple example to depict our idea. We can see from Figure 1a that if we only look at the "face region", we can hardly classify whether it is a face or not. However, when we observe the "head with shoulder" region around the "face region", we can recognize it as a face. Similarly, we can see from Figure 1b that we are likely to classify the "face region" as a face if we only look at the "face region", but when we observe the "head with shoulder" region, we know that it is only a crater.

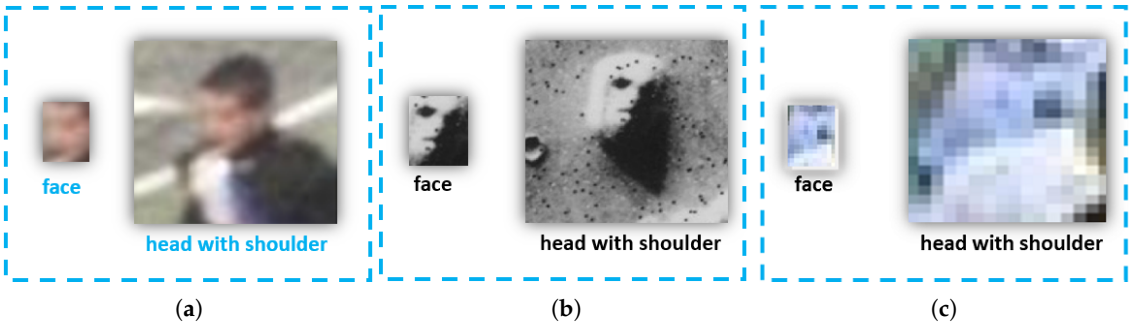


Figure 1. Examples of the "head with shoulder" information helping face detection. (**a**) The existence of the "head with shoulder" region can increase the confidence of detecting a face. (**b**) A region that looks like a face turns out to be a crater when we see its "head with shoulder" region. (**c**) An example of an area mistakenly detected as a face by the multi-task cascaded convolutional network (MTCNN), but rejected by the head-shoulder framework.

Therefore, the overall purpose of our improved framework is to incorporate head with shoulder information into our cascade. The main framework of our face detector is shown in Figure 2. It consists of two parallel CNN cascaded networks. The "small size cascade" is used for detecting the faces with a scale smaller than 20×20 pixels, and a small-scale network is preferable for speed because the number of sliding windows is large for small face detectors. The "big size cascade" is used for detecting faces with a scale lager than 20×20 pixels. The latter cascade takes a longer time to process one sliding window because the network size is larger and more accurate. "NMS" means non-maximum suppression, and the details of each block in Figure 2 are discussed in the following subsections.

Each block in Figure 2 is a module of a convolutional neural network. Blocks named Face-12-small and Face-12-big are face detectors, and their detailed frameworks are illustrated in Figure 3. Block Shoulder-24-1 and Shoulder-24-2 are used to reject false positives in "small size cascade", and block Com-24-1 and Com-24-2 are used to reject in "big size cascade". There detailed structures are shown in Figures 4 and 7, respectively. We use two modules in each cascade for rejection because such a hierarchy can improve efficiency, as early rejections do not need to be fed into later stages. Blocks Reg-12 and Reg-24 are used to regularize bounding boxes for larger face detectors, as they are more sensitive to the positions. Their structure is presented in Figure 8.

Given a test image, we first create two image pyramids as the input of the two CNN cascaded networks. After the evaluation of the two parallel CNN cascaded networks using sliding windows, we merge the two detections results. Because our cascade approach takes head and shoulder information into consideration, we name it "HS-cascade". As it aims to solve surveillance problems, we do not consider face rotations.

3.2. Small Size Cascade

We can see from Figure 2 that the "small size cascade" consists of three convolutional neural networks: "Face-12-small", "Shoulder-24-1", "Shoulder-24-2". We will introduce them one by one in the following sections.

Figure 2. The overall framework of our proposed face detection algorithm, which consists of two parallel CNN cascaded networks for detecting different scales of faces. NMS, non-maximum suppression; Reg, regularize.

3.2.1. Face-12-Small

Face-12-small refers to the first CNN in the "small size cascade". Its structure is shown in Figure 3. Face-12-small is a shallow binary classification CNN for quickly scanning the test image and rejecting the non-face regions. Given a test image of size $W \times H$, first we build it into an image pyramid (the scale factor between each pyramid level is f) to cover faces at different scales. If the minimum face size that we need to detect is $a \times a$, we resize images at each level of the image pyramid by a coefficient of $\frac{12}{a}$, so that the smallest patch becomes 12×12. We feed the resized images into Face-12-small. Then, the Face-12-small net will scan the input image with two-pixel spacing for 12×12 detection windows and reject the non-face windows. After that, we employ non-maximum suppression (NMS) to merge highly overlapped candidate windows.

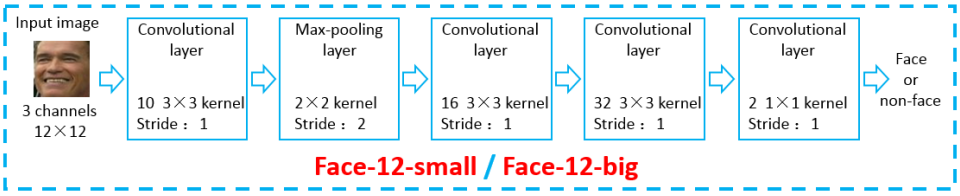


Figure 3. The CNN structure of Face-12-small and Face-12-big.

3.2.2. Shoulder-24-1

Shoulder-24-1 refers to the second CNN in the "small size cascade". Its structure is shown in Figure 4. Shoulder-24-1 is a binary classification CNN for further rejecting the non-face windows. When images are collected from squares, stations or stadiums, the facial details are not clear, so we take the "head with shoulder" into consideration. Detection windows of the Face-12-small are zoomed on the input image according to a predefined geometrical relationship (shown in Figure 5) to get the "head with shoulder" regions. Then, we crop out the "head with shoulder" regions and resize them into 24×24 images as the input of Shoulder-24-1. Shoulder-24-1 will further reject the non-face windows. After that, again, we employ NMS to merge highly overlapped candidate windows.

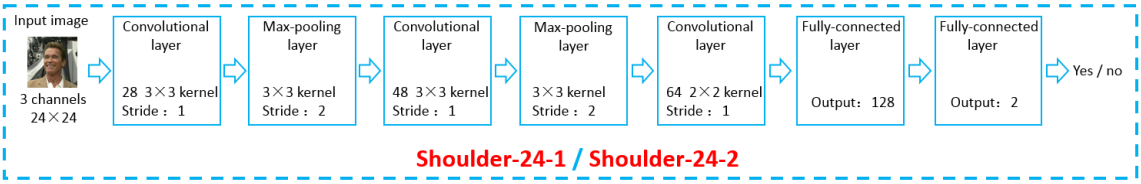


Figure 4. The CNN structure of Shoulder-24-1 and Shoulder-24-2.

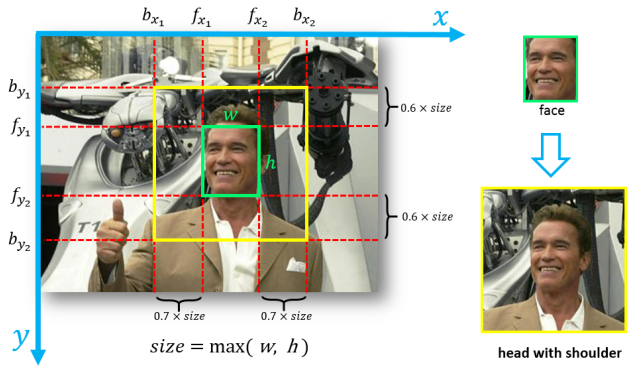


Figure 5. The geometrical relationship between the face region and the "head with shoulder" region.

3.2.3. Shoulder-24-2

Shoulder-24-2 refers to the last CNN in the "small size cascade". Its structure is shown in Figure 4. The same as Shoulder-24-1, Shoulder-24-2 is a binary classification CNN for further rejecting the non-face windows.

3.3. Big Size Cascade

We can see from Figure 2 that the "big size cascade" consists of five convolutional neural networks: "Face-12-big", "Reg-12", "Com-24-1", "Reg-24", "Com-24-2". We will introduce them one by one in the following subsections.

3.3.1. Face-12-Big

Face-12-big refers to the first CNN in the "big size cascade". Its structure is shown in Figure 3. Face-12-big is a shallow binary classification CNN for quickly scanning the test image and rejecting the non-face regions. Given a test image of size $W \times H$, first we build it into an image pyramid (the scale factor between each pyramid level is f) to cover faces at different scales. Since the minimum face size that "big size cascade" needs to detect is 20×20, we resize images at each level of the image pyramid again, similarly as for Face-12-small. Then, the Face-12-big net will densely scan the resized image with two-pixel spacing for 12×12 detection windows and reject the non-face windows. After that, we employ non-maximum suppression (NMS) to merge highly overlapped candidate windows.

3.3.2. Reg-12

Reg-12 refers to the CNN after Face-12-big for bounding box calibration. Its structure is shown in Figure 6. Remaining detection windows from Face-12-big are cropped out and resized into 12×12. They are then processed by Reg-12. Given a detection window from Face-12-big, its coordinate in the test image is (x_1, y_1, x_2, y_2). Then, Reg-12 will output four calibration parameters $(\hat{x}_1, \hat{y}_1, \hat{x}_2, \hat{y}_2)$. After calibration, the coordinate of the detection window will be (x'_1, x'_2, y'_1, y'_2):

$$x'_1 = (x_2 - x_1)\hat{x}_1 + x_1, \tag{1}$$
$$x'_2 = (x_2 - x_1)\hat{x}_2 + x_2, \tag{2}$$
$$y'_1 = (y_2 - y_1)\hat{y}_1 + y_1, \tag{3}$$
$$y'_2 = (y_2 - y_1)\hat{y}_2 + y_2. \tag{4}$$

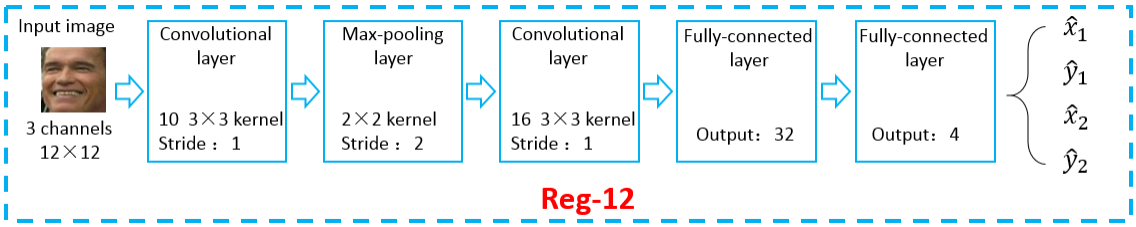


Figure 6. The CNN structure of Reg-12.

3.3.3. Com-24-1

Com-24-1 refers to the CNN after Reg-12, and it is a binary classification CNN for further rejecting the non-face windows. Its structure is shown in Figure 7. Unlike the intermediate binary classification CNN of "Cascade-CNN" or "MTCNN", it has two inputs and one output. Given a detection window from Reg-12, first, we crop it out and resize it into 24×24 as the input image_1, then we zoom the detection window on the test image according to a setting geometrical relationship (shown in Figure 5) to get the "head with shoulder" regions, then crop the "head with shoulder" region out and resize

it into 24 × 24 as the input image_2. In this way, Com-24-1 takes both the face information and the surrounding context information into consideration, which makes it much more accurate for evaluating whether a detection window is a face or not.

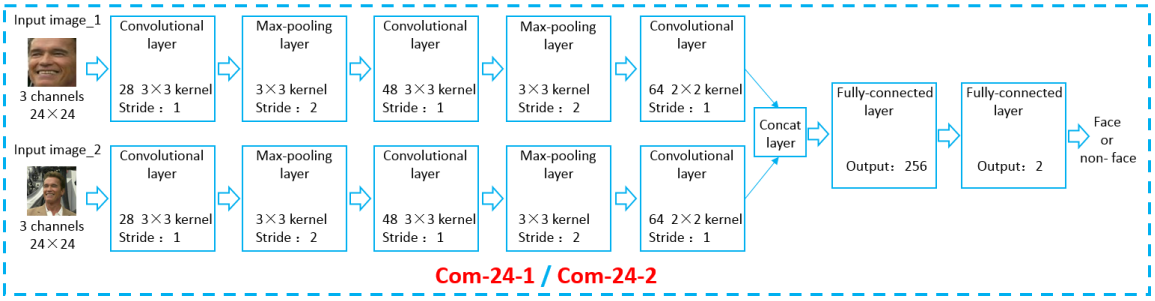


Figure 7. The CNN structure of Com-24-1 and Com-24-2.

3.3.4. Reg-24

Reg-24 refers to the CNN after Com-24-1 for further bounding box calibration. Its structure is shown in Figure 8. Remaining detection windows from Com-24-1 are cropped out and resized into 24 × 24. They are then processed by Reg-24.

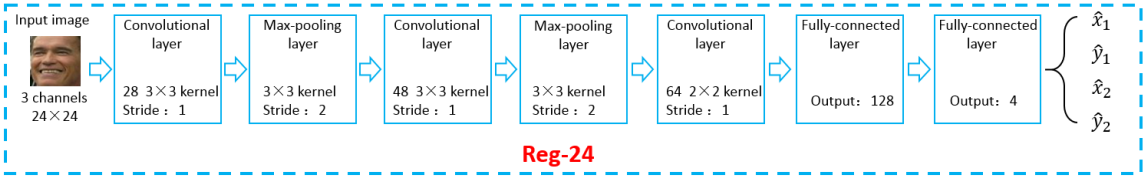


Figure 8. The CNN structure of Reg-24.

3.3.5. Com-24-2

Com-24-2 refers to the last CNN in the "big size cascade". It is a binary classification CNN for further rejecting the non-face windows. Its structure is shown in Figure 7, the same as Com-24-1. It also has two inputs for the face information and the surrounding context information, which makes the evaluation much more accurate.

3.4. Training

We use back-propagation to train the model. For the ease of data labeling, we use a subset of the WIDER FACE dataset [29] to train our model. The subset that we use contains mostly low resolution faces by excluding those larger than 40 × 40 pixels.

3.4.1. Train Face-12-Small

For training Face-12-small, we collect two kinds of training samples: (i) negatives: regions whose intersection-over-union (IoU) ratio is less than 0.3 with respect to any ground truth faces; (ii) positives: IoU > 0.65 to a ground truth face, both of whose height and width are less than 20 pixels. Finally, we collect 724,650 negative samples and 181,150 positive samples, and then, we resize all the samples into 12 × 12 for training Face-12-small. In each iteration, we randomly select the same number of negative samples as positive samples to tackle the imbalanced data problem.

3.4.2. Train Shoulder-24-1

For training Shoulder-24-1, we zoom the training samples (namely, face regions) for Face-12-small on the original images according to a setting geometrical relationship (shown in Figure 5), and then, we resize all the samples into 24 × 24 for training Shoulder-24-1.

3.4.3. Train Shoulder-24-2

(i) Negatives: Firstly, we apply a two-stage cascade consisting of the Face-12-small and Shoulder-24-1 on the WIDER FACE dataset to choose threshold t_1 for Face-12-small and threshold t_2 of Shoulder-24-1 at a 99% recall rate. Then, we use the two-stage cascade to evaluate the original training images, and we choose the detection windows with a confidence score larger than t_2 and IoU < 0.3 to any ground truth faces. (ii) Positives: We use the same positive training samples as for training Shoulder-24-1.

3.4.4. Train Face-12-Big

Similar to training Face-12-small, we collect two kinds of training samples: (i) negatives: regions whose IoU ratio are less than 0.3 to any ground truth faces; (ii) positives: IoU > 0.65 to a ground truth face whose height or width is larger than 20, but less than 30 pixels. Finally, we collect 676,249 negative samples and 193,214 positive samples, and then, we resize all the samples into 12×12 for training Face-12-big.

3.4.5. Train Reg-12 and Reg-24

We choose the rectangle regions with IoU between 0.4 and 0.65 to a ground truth face (height or width is larger than 20 pixels) as the training samples to train Reg-12 and Reg-24, and we call it "part face". For instance, given a ground truth face, its coordinate in the original image is (x'_1, x'_2, y'_1, y'_2), and we choose a "part face". Its coordinate in the original image is (x_1, y_1, x_2, y_2). Then, we can use four factors $(\hat{x}_1, \hat{y}_1, \hat{x}_2, \hat{y}_2)$ to represent the offset of the ground truth face to the "part face":

$$\hat{x}_1 = (x'_1 - x_1)/(x_2 - x_1) \tag{5}$$

$$\hat{y}_1 = (y'_1 - y_1)/(y_2 - y_1) \tag{6}$$

$$\hat{x}_2 = (x'_2 - x_2)/(x_2 - x_1) \tag{7}$$

$$\hat{y}_2 = (y'_2 - y_2)/(y_2 - y_1) \tag{8}$$

We use the four factors $(\hat{x}_1, \hat{y}_1, \hat{x}_2, \hat{y}_2)$ as the training labels and resize the "part face" into 12×12 and 24×24 for training Reg-12 and Reg-24 separately.

3.4.6. Train Com-24-1

For training Com-24-1, firstly, we resize the training samples for Face-12-big into 24×24 to train an intermediate net "Face-24", and its structure is shown in Figure 9. Obviously, Face-24 can evaluate whether a detection window is a face or not. Then, we zoom the training samples for Face-12-big on the original images according to a setting geometrical relationship (shown in Figure 5) to train another intermediate net "Shoulder-24" (shown in Figure 9), and this net can evaluate whether a detection window is "head with shoulder" or not. Then, we copy the parameters of the convolutional layers in Face-24 and Shoulder-24 to the corresponding convolutional layers in Com-24-1. For training the rest parameters of Com-24-1, as the Com-24-1 has two inputs, we input the training samples of Face-12-big and their corresponding "head with shoulder" regions in pairs into Com-24-1. In this way, Com-24-1 can more precisely evaluate whether a detection window is a face or not.

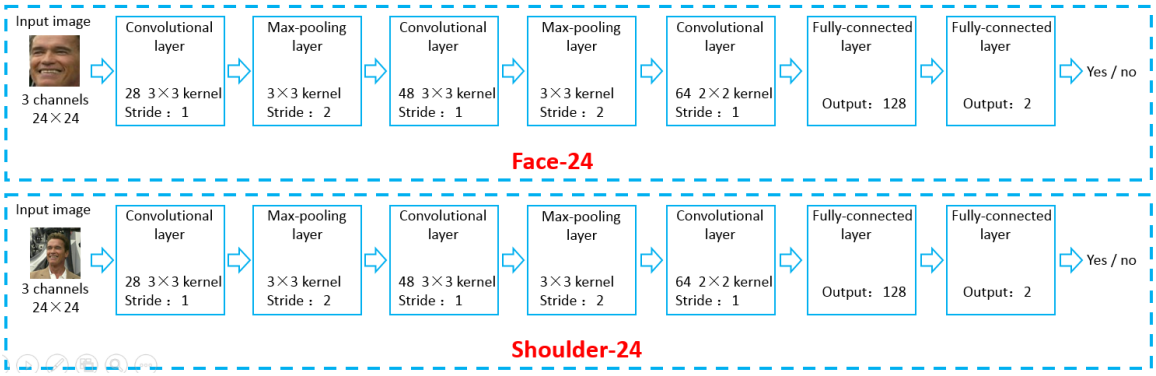


Figure 9. The CNN structure of the two intermediate nets Face-24 and Shoulder-24.

3.4.7. Train Com-24-2

(i) Negatives: Firstly, we apply a two-stage cascade consisting of Face-12-big, Reg-12, Com-24-1 and Reg-24 on the WIDER FACE dataset to choose threshold t_1' of Face-12-big and threshold t_2' of Com-24-1 at a 97% recall rate. Then, we use the two-stage cascade to evaluate the original training images, and we choose the detection windows with a confidence score larger than t_2' and IoU < 0.3 with respect to any ground truth faces. (ii) Positives: We use the same positive training samples as for training Com-24-1. Similar to training Com-24-1, we still need to train two intermediate nets to get the parameters of the convolution layers and then train the remaining parameters.

4. Experiment

4.1. Testing Dataset

Since we design our face detection algorithm for large scene conditions and there is a lack of such datasets, we propose a large area surveillance dataset, which includes the one hundred most crowded images from the crowd counting dataset, the "Shanghaitech dataset" [30]. We manually label the faces for these images. We employ the same evaluation criterion as PASCAL VOC [31] to evaluate the predicted detection windows: if the detection window's IoU ratio is larger than 0.5 to one of the ground truth faces, we label it as a correct one.

The WIDER FACE test image set is not directly suitable for our test because it contains many large faces, and the ground truth bounding boxes for the test images are not released, so that we cannot evaluate the accuracy of the detection algorithms exclusively for faces in large scenes. Thus, rather than choosing the whole test set, we choose a few images that contain many small faces to test.

4.2. Testing Result

We compare our face detection method (HS-cascade) against two other cascade methods, CascadeCNN and MTCNN. We implement our method with Caffe. We set the minimum detection window size to be 5×5 for all three algorithms, and we use the default setting of the two compared algorithms for the hyperparameters, which are the best settings for face detection by the authors. As other approaches, when testing, different sizes of detection windows are normalized to fit the input size of each algorithm. Figure 10 shows that our method (red) outperforms the two compared approaches by a large margin. Some examples are demonstrated in Figure 11. Table 1 illustrates the number of true positive and false positive detections when we tune over different threshold values. When the threshold value is smaller, more faces are accepted, as well as more false alarm. Yet, overall, our approach performs relatively well over different thresholds.

Table 1. The pair of true positive and false positive predictions over different thresholds. HS, head-shoulder.

HS-CNN	TP	980	939	885	711	657	496	389
	FP	244	206	168	79	57	31	16
CascadeCNN	TP	402	389	362	348	295	268	228
	FP	216	151	96	71	36	20	14
MTCNN	TP	214	187	174	161	120	93	80
	FP	131	101	78	56	28	17	14

We also compared our algorithm on WIDER FACE testing images in Figure 12 and other images in Figures 13–15. From these examples, we can see that although the performances are similar when faces are near the camera, our approach (HS-cascade) can significantly improve face detection cascades for large and crowd areas.

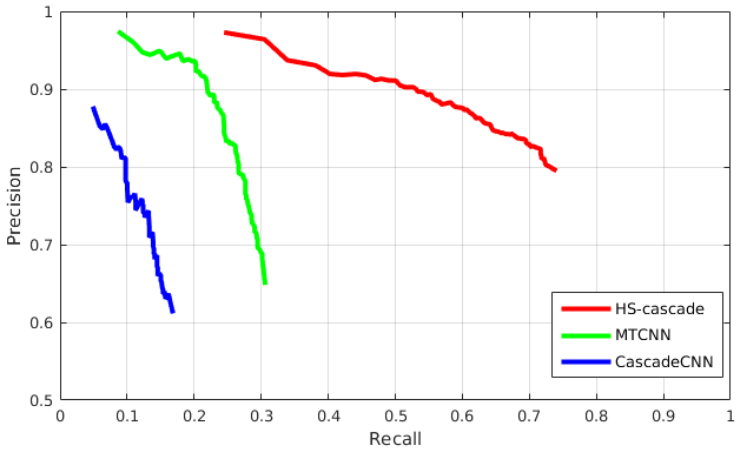


Figure 10. Evaluation on the "Shanghaitech dataset".

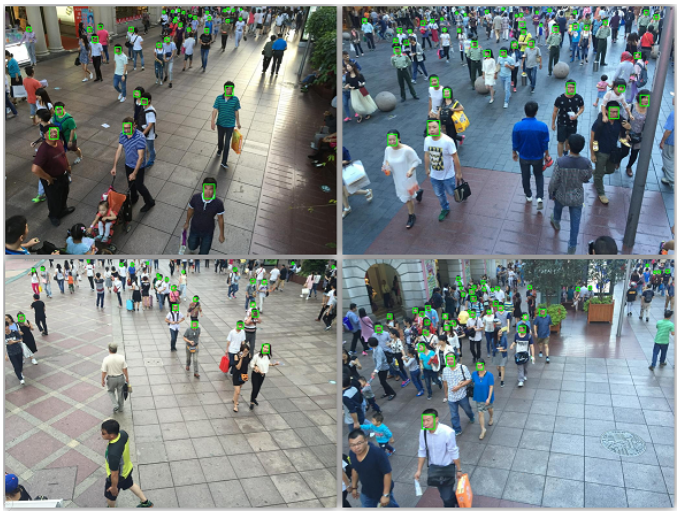
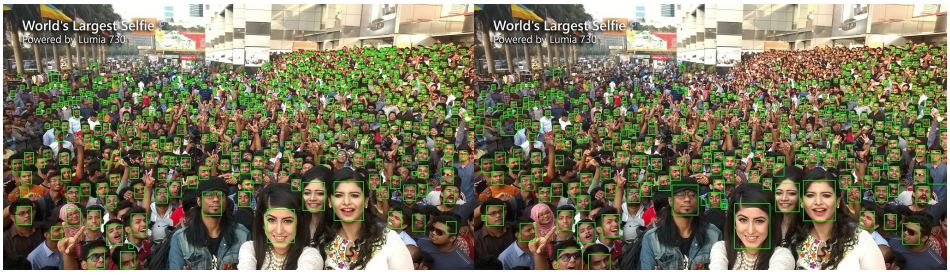


Figure 11. Examples of detected faces by HS-cascade.

(**a**) HS-cascade (**b**) MTCNN

(**c**) HS-cascade (**d**) MTCNN

Figure 12. Comparison between HS-cascade and MTCNN on the WIDER FACE dataset.



(**a**) HS-cascade (**b**) MTCNN

Figure 13. Comparison between HS-cascade and MTCNN (Example 1).

(**a**) HS-cascade (**b**) MTCNN

Figure 14. Comparison between HS-cascade and MTCNN (Example 2).

(**a**) HS-cascade (**b**) MTCNN

Figure 15. Comparison between HS-cascade and MTCNN (Example 3).

5. Conclusions

In this paper, we propose a novel framework for face detection in large areas, like squares, stations and stadiums. It could be very useful for directing high resolution face recognition cameras (for example, bullet cameras) to take photos of faces of interest. It could also be used for crowd density estimation, pedestrian registration, etc. Our method consists of two parallel carefully-designed CNN cascades for separately detecting small and lager faces in one image. Different from the previous cascade-based face detection methods, we combine the facial information and the "head with shoulder" information into the cascade framework for dealing with the missing facial features in surveillance images of crowed places. Experimental results demonstrate the capability of our algorithm for large area surveillance.

Author Contributions: Investigation, C.P. and W.B. Methodology, C.P. Resources, J.X. Software, W.B. and M.Y. Writing, original draft, C.P. and W.B. Writing, review and editing, J.X. and K.-c.W.

Funding: This work is supported by the National Natural Science Foundation of China (No. 61802372), the Qianjiang Talent Plan (No. QJD1702031) and the National Natural Science Foundation of Ningbo, China (No. 2018A610050).

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses or interpretation of data; in the writing of the manuscript; nor in the decision to publish the results.

References

1. Klare, B.F.; Klein, B.; Taborsky, E.; Blanton, A.; Cheney, J.; Allen, K.; Grother, P.; Mah, A.; Jain, A.K. Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus Benchmark A. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1931–1939.
2. Zhao, X.; Delleandrea, E.; Chen, L. A people counting system based on face detection and tracking in a video. In Proceedings of the Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, Genova, Italy, 2–4 September 2009; pp. 67–72.

3. Cho, S.I.; Kang, S.J. Real-time People Counting System for Customer Movement Analysis. *IEEE Access* **2018**, *6*, 55264–55272. [CrossRef]
4. Kang, S.J. Multi-user identification-based eye-tracking algorithm using position estimation. *Sensors* **2016**, *17*, 41. [CrossRef] [PubMed]
5. Yang, B.; Yan, J.; Lei, Z.; Li, S.Z. Aggregate channel features for multi-view face detection. In Proceedings of the IEEE International Joint Conference on Biometrics (IJCB), Clearwater, FL, USA, 29 September–2 October 2014; pp. 1–8.
6. Yang, B.; Yan, J.; Lei, Z.; Li, S.Z. Convolutional channel features. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 82–90.
7. Zhu, C.; Zheng, Y.; Luu, K.; Savvides, M. Cms-rcnn: Contextual multi-scale region-based cnn for unconstrained face detection. In *Deep Learning for Biometrics*; Springer: New York, NY, USA, 2017; pp. 57–79.
8. Ranjan, R.; Patel, V.M.; Chellappa, R. A deep pyramid deformable part model for face detection. *arXiv* **2015**, arXiv:1508.04389.
9. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, 8–14 December 2001; Volume 1.
10. Zhang, C.; Zhang, Z. *A Survey of Recent Advances in Face Detection*; Technical Report; Microsoft Research: Redmond, WA, USA, June 2010.
11. Li, H.; Lin, Z.; Shen, X.; Brandt, J.; Hua, G. A convolutional neural network cascade for face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5325–5334.
12. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
13. Hjelmås, E.; Low, B.K. Face detection: A survey. *Comput. Vis. Image Underst.* **2001**, *83*, 236–274. [CrossRef]
14. Zafeiriou, S.; Zhang, C.; Zhang, Z. A survey on face detection in the wild: Past, present and future. *Comput. Vis. Image Underst.* **2015**, *138*, 1–24. [CrossRef]
15. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time series. In *The Handbook of Brain Theory and Neural Networks*; MIT Press: Cambridge, MA, USA, 1995; Volume 3361.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
17. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef] [PubMed]
18. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs. *arXiv* **2016**, arXiv:1606.00915.
19. Zheng, S.; Jayasumana, S.; Romera-Paredes, B.; Vineet, V.; Su, Z.; Du, D.; Huang, C.; Torr, P.H. Conditional random fields as recurrent neural networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1529–1537.
20. Lowe, D.G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [CrossRef]
21. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
22. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [CrossRef]
23. Ahonen, T.; Hadid, A.; Pietikäinen, M. Face recognition with local binary patterns. In Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic, 11–14 May 2004; pp. 469–481.
24. Jun, B.; Choi, I.; Kim, D. Local transform features and hybridization for accurate face and human detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1423–1436. [CrossRef] [PubMed]
25. Dollár, P.; Tu, Z.; Perona, P.; Belongie, S. Integral channel features. In Proceedings of the British Machine Vision Conference, London, UK, 7–10 September 2009; pp. 91.1–91.11.
26. Mathias, M.; Benenson, R.; Pedersoli, M.; Van Gool, L. Face detection without bells and whistles. In Proceedings of the 13th European Conference, Zurich, Switzerland, 6–12 September 2014; pp. 720–735.

27. Huang, C.; Ai, H.; Wu, B.; Lao, S. Boosting nested cascade detector for multi-view face detection. In Proceedings of the 17th International Conference on Pattern Recognition, Cambridge, UK, 26 August 2004; Volume 2, pp. 415–418.
28. Wu, J.; Brubaker, S.C.; Mullin, M.D.; Rehg, J.M. Fast asymmetric learning for cascade face detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 369–382. [PubMed]
29. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. Wider face: A face detection benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5525–5533.
30. Zhang, Y.; Zhou, D.; Chen, S.; Gao, S.; Ma, Y. Single-image crowd counting via multi-column convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 589–597.
31. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]