Manga Face Detection based on Deep Neural Networks Fusing Global and Local Information

Wei-Ta Chu and Wei-Wei Li

National Chung Cheng University, Taiwan

wtchu@ccu.edu.tw, welcometoway@gmail.com

Abstract

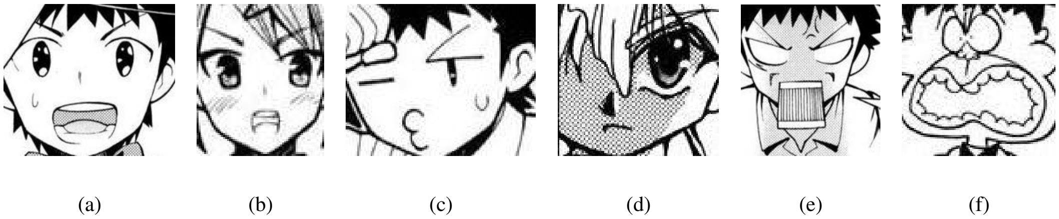
As more and more digitized manga (Japanese comics) books are available, efficient and effective access to manga is urgently needed. Among various elements of manga, character's face plays one of the most important roles in access and retrieval. We propose a deep neural network method to do manga face detection, which is a challenging but relatively unexplored topic. Given a manga page, we first find candidate regions based on the selective search scheme. Three convolutional neural networks are then proposed to detect manga faces of various appearance. We extract information from the entire object region and several local regions, and integrate multi-scale information in an early fusion manner or a late fusion manner. The proposed methods are evaluated based on a large-scale benchmark. Convincing performance compared to the state-of-the-art face detection modules designed for human faces is demonstrated.

Index Terms

Manga face detection, deep neural network, early fusion, late fusion.

I. Introduction

Manga (Japanese comics) is one of the biggest book sales in the world. Although the book market slumped, in Japan the market of compiled manga books keeps creating record-high sales and reaches around 2.4 billion US dollars in year 2014 [1]. As more and more manga books are digitized, efficient access and retrieval of manga is urgently demanded [2]. Among various indexing and retrieval methods, we believe character's face is one of the most important items for accessing manga. Face detection is a fundamental step to many computer vision and multimedia



(a) (b) (c) (d) (e) (f)

Fig. 1. Sample manga faces. The first two faces are normal frontal faces. The middle two have drastically different appearance on eyes. The last two have physically unreasonable expressions.

applications. This topic has been widely studied for natural images [3]. However, much fewer studies have been proposed for manga.

There are at least three differences between faces in natural images and in manga. First, in most manga, only black-and-white and sometimes gray information is available, which is different from color information in natural images. Second, there are extreme variations in faces of different manga. Figure 1(a) and Figure 1(b) show two normal frontal faces, while Figure 1(c) and Figure 1(d) show drastically different visual appearance especially on eyes. Third, manga faces do not entirely possess properties of human faces. The spatial layout, visual appearance, and expression of manga faces may not physically reasonable (Figure 1(e) and Figure 1(f)).

To further showcase the difference between manga faces and human faces, and the necessity of proposing an exclusive method for manga faces, we adopt the MTCNN [4] to extract features respectively from two types of faces and show feature distributions in Figure 2. It was thought that features useful for face detection and face alignment are correlated. Therefore, the idea of MTCNN is proposing a cascaded convolutional network to jointly achieve these two tasks. This network consists of three stages. The first stage is a Proposal Network (P-Net) that estimates bounding boxes potentially containing (human) faces. The second stage is a Refinement Network (R-Net) that rejects a large number of false candidates. The third stage is an Output Network (O-Net) that identifies face regions with more supervision and outputs facial landmarks' positions. We respectively take outputs of the second last layers of the P-Net, R-Net, and O-Net, as the features showing face characteristics.

Figure 2(b) and Figure 2(c) show average feature distributions of 100 randomly selected human faces from the CelebA detaset [5], extracted by P-Net and R-Net, respectively. Figure 2(e) and Figure 2(f) show average distributions of 100 randomly selected manga faces from the Manga109 dataset [2], extracted by P-Net and R-Net, respectively. Comparing Figure 2(b) with Figure 2(e),

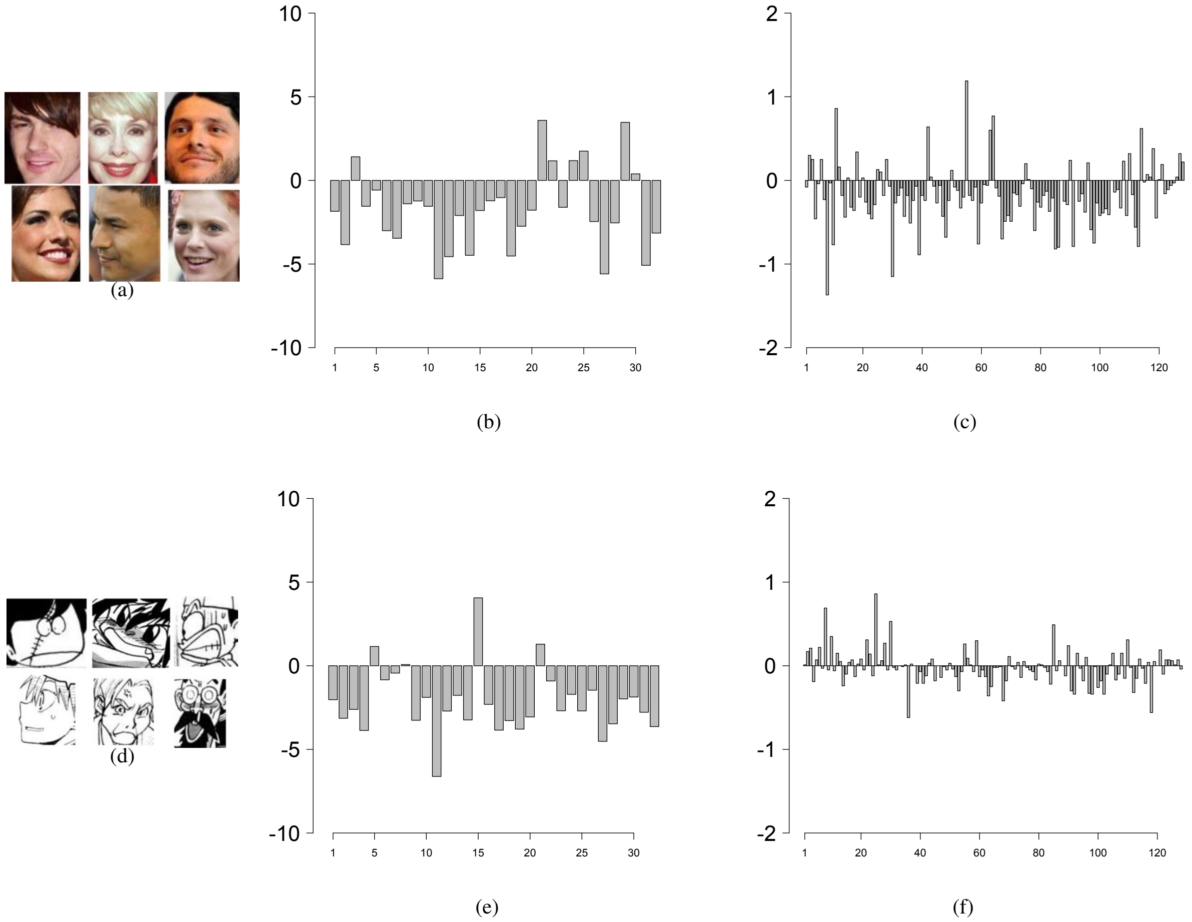


Fig. 2. Distributions of (b) and (c) are average distributions of human faces extracted from P-Net and R-Net, respectively. Distributions of (e) and (f) are average distributions of manga faces extracted from P-Net and R-Net, respectively.

we see that the average feature distributions of human faces and manga faces, from the perspective of the P-Net, are quite different. Similar characteristics can also be seen from Figure 2(c) and Figure 2(f), from the perspective of the R-Net. In fact, features extracted by the O-Net also exhibit a similar trend, but we omit to display them to save space.

Given the aforementioned challenges, manga face detection is significantly different from real human face detection. Although real human face detection has been studied for decades, and many elegant methods have been proposed, directly employing them in manga does not work well. Figure 3 shows sample detection results of OpenCV, Microsoft Azure face detection API [6], and our detection result, respectively. As can be seen in Figure 3(a), the OpenCV method (further trained with manga data) falsely detects many regions that are not manga faces. Figure 3(b) shows that the state-of-the-art face detection API works much better in precision, but

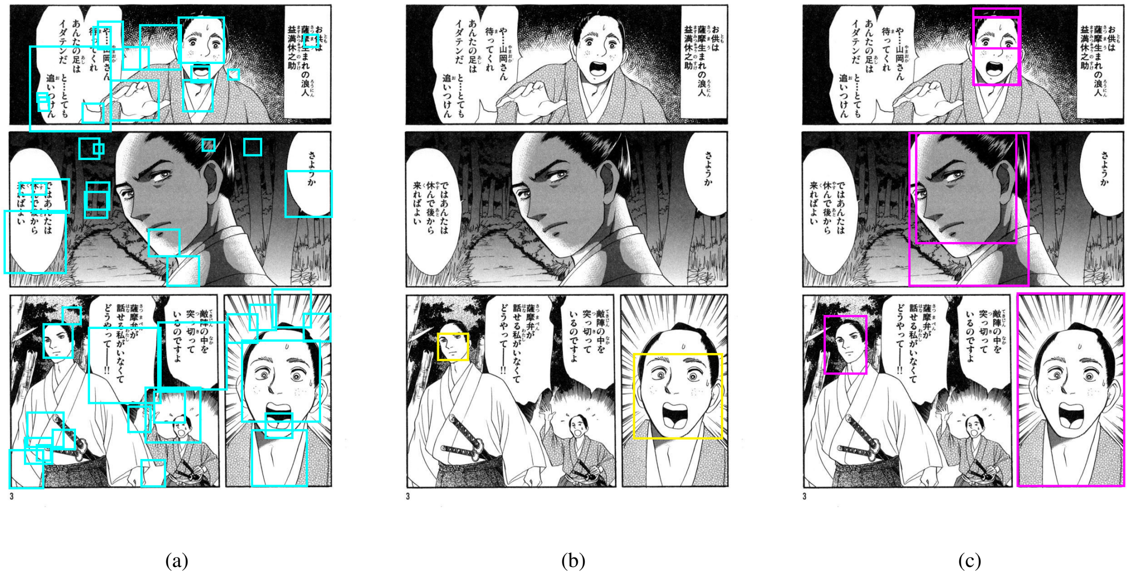

お供は薩摩生まれの浪人益満休之助
や…山岡さん待ってくれ
あんたの足はイダテンだ
と…とても追いつけん
さようか
ではあんたは休んで後から来ればよい
敵陣の中を突っ切っているのですよ
薩摩弁が話せる私がいなくてどうやって——!!
3
(a)
(b)
(c)

Fig. 3. Sample face detection results obtained by (a) OpenCV trained with manga data, (b) Microsoft Azure face API, and (c) our results.

many manga faces cannot be successfully detected. Given that existing face detection systems are not applicable to manga faces, we propose a face detection method specifically designed for manga faces.

One viewpoint to handling the manga face detection problem is viewing it as a spacial object detection problem. Based on various visual features like contour, shape [7], texture [8], and color contrast [9][10][11], tremendous approaches have been proposed to do object matching or object detection. Thanks to the development of deep neural networks, recently very impressive object detection and recognition models have been proposed for natural images/videos [12][13]. However, these models are constructed based on a large collection of natural images. Employing the power of deep networks into manga face detection needs system integration specially designed for manga.

In this work, we integrate the region proposal method [10] designed prior to the deep learning era with the specially designed deep networks to do manga face detection. Figure 4 shows overview of the proposed method. Given a manga page, we first employ the selective search scheme [10] to detect regions probably containing objects. Each region is then examined by the proposed deep neural network, named Manga FaceNet (MFN), to see whether this region is a manga face or not. To accurately identify a region as a face or not, it would be better to

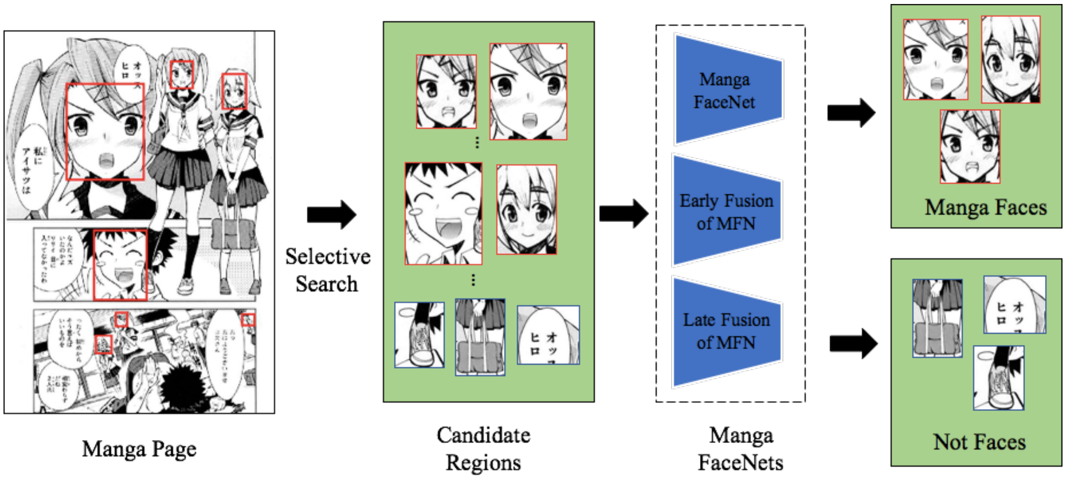


Fig. 4. Illustration of the system flowchart. Given a manga page, candidate object regions are first detected by the selective search scheme. Each candidate region is then examined by the proposed deep networks to determine whether it is a manga face or not.

jointly consider local features and global features in the classifier. With this idea, we design three versions of Manga FaceNet, which respectively correspond to baseline MFN, early fusion of MFN, and late fusion of MFN. In the early fusion version, we concatenate global features with four local features to do classification. In the late fusion version, we take global features to conduct preprocessing, and then take four local features to refine results. We will evaluate the proposed methods based on a large-scale manga dataset, i.e., Manga 109 dataset [2], and will compare with several state-of-the-art methods to demonstrate effectiveness of the proposed methods.

Contributions of this paper are summarized as follows.

- We construct a deep manga face detection framework to resist visual variations. This system can learn representative features of manga faces based on a large number of training data.
- We fuse global and local information of candidate face regions and embed it into the proposed deep networks. Three versions of networks are proposed, and comprehensive experiment results showing performance variations are provided.
- We evaluate performance based on a large-scale benchmark to facilitate fair comparison in the future.

The rest of this paper is organized as follows. Section II gives literature survey on human face detection and manga face detection. Section III provides details of the proposed deep frameworks.

Section IV shows performance of the proposed method as well as comparison with existing methods, and conclusion is given in Section V.

II. Related Works

A. Face Detection in Natural Images

Because face detection has been studied for decades, we limit the literature survey to the most typical and the most recent works in the following. Viola and Jones [14] proposed a classical face detection framework based on the boosting strategy. They extracted Haar-like features to represent a region, and then employed the AdaBoost algorithm to construct a face classifier. Multiple weak classifiers are cascaded to form a strong classifier. This method motivates various face detection studies, and has been embedded into many computer vision libraries like OpenCV. Chen et al. [15] found that facial landmarks usually used for face alignment are useful for face detection. They first adopted the framework proposed by Viola and Jones to detect face region candidates, from them features from facial landmarks are extracted and an SVM classifier is constructed to check whether a candidate is really a face or not. Liao et al. [16] proposed to extract a simple pixel-level feature from images, called the Normalized Pixel Difference (NPD). To get the value of NPD, they computed the ratio of the difference between any two pixel's intensity values to the sum of their values. Based on this feature, they proposed a deep quadratic tree earning method to construct an AdaBoost classifier with single soft-cascade.

Another approach successfully applied to face detection is the exemplar-based method. In the framework proposed by Kumar et al. [17], they extracted local features like dense-SIFT from a huge exemplar database, and then quantized features with a K-means based vocabulary. In the stage of testing, each exemplar takes a vote on the test image at multiple scales. They then collected all the exemplars by the Hough-based voting process, and finally located faces in a given image by using the features' spatial locations.

In [18], a face detector was constructed by integrating deep pyramid features in deformable part models. Given an input image, a seven-level normalized deep feature pyramid was generated. They then adopted the sliding window approach to extract features in the pyramid. A linear SVM taking these features as input was then constructed to classify each considered region as a face or non-face.

Farfade et al. [19] proposed a deep dense face detector. Their method does not require pose/landmark annotation and is able to detect faces in a wide range of orientations. They fine-tuned a pre-trained model, i.e., AlexNet [20], for face detection. They then used this fine-tuned deep network with the sliding window scheme to detect faces. Recently, Sun et al. [21] used faster R-CNN to do face detection. They improved the scheme by combining several important strategies, including feature concatenation, hard negative mining, and multi-scale training. In [22], a convolutional neural network was constructed to estimate the parameters of 3D transformation about rotation and translation. A 3D mean face model was used to generate face proposals and predict face key points.

The aforementioned methods extracted rich features from natural images. However, these features are not suitable to manga because color information is usually missing in manga, and the texture information of manga face is significantly different from real human faces. Motivated by the studies adopting deep neural networks to achieve face detection, in this work we will construct deep neural networks integrating multi-scale information to achieve manga face detection.

B. Face Detection in Manga

Some works were proposed to specially detect manga faces. Sun and Kise [23] extracted Haar-like features, and concatenated a sequence of weak classifiers to construct a face detector. For a limited dataset, frontal manga faces can be detected. However, researchers soon found that the face detection method designed for natural faces is not suitable to comics.

Focusing on colorful comics images, Takayama et al. [24] detected skin color and the jaw contour to find character's face. The symmetric property of face was then adopted to filter out noises. This method may not be generic to faces in different poses. In [25], the deformable part model was used to consider pose and spatial variations. This approach was verified to achieve better reliability compared to previous works. Chu and Chao [26] attempted to resist visual variations by first detecting character's eyes, and then expanded the eye regions to find the face. Not specific for face detection, Sun et al. [27] discovered effective features to do manga face matching.

The aforementioned methods are mostly ad-hoc approaches and are hard to be generalized. In addition, most of them were not evaluated on a large-scale dataset, making the conclusion not convincing enough. Thanks to the recently proposed Manga109 dataset [2], now we have more resource to train the face detector, and can evaluate the proposed method at a larger scale. In our previous work [28], an end-to-end deep network was firstly proposed to do manga face detection. Candidate object regions are detected by the selective search scheme, and then a convolutional neural network jointly considering object classification and spatial displacement is constructed. In this work, we substantially improve our previous work by further considering side-view faces, and further proposed two more neural networks to fuse global information and local information.

III. Manga Face Detection

A. Data Preparation

Recently the power of deep learning has been demonstrated in many domains. Not only for image classification [20] or object detection [12] for natural images, but also for analyzing sketch or line drawings [29] [30]. We therefore propose to construct a deep neural network called Manga FaceNet to do this task. Before training the proposed network, we randomly select 24 manga titles from the Manga109 dataset, and from each title we select the first 60 manga pages as the evaluation dataset. For each page, we manually define the bounding box of each manga face. Overall, we have 3760 frontal faces and 1110 side-view faces in the training set. To train a deep neural network, such limited volume of training data is not enough. Therefore, we propose the following data augmentation strategy.

To increase the number of training faces, we augment the dataset with two methods. First, given each manga page, we employ the selective search scheme [10] to find object regions. For each object region O, we calculate the overlap ratio between it and its spatially closest ground bounding box B: $r = \frac{O \bigcap B}{\max(O,B)}$. The regions with overlap ratios larger than 0.7 are considered as positive examples, while the regions with overlap ratios smaller than 0.3 are considered as negative examples. Figure 5 illustrates this augmentation by showing some samples. The images in the left column are ground truths of four manga faces, the images in the middle column are part of their corresponding positive samples (i.e., overlap ratio larger than 0.7), and the images in the right column are part of negative samples (i.e., overlap ratio smaller than 0.3).

The second method to augment data is horizontally flipping positive samples selected by the first method. This method is commonly used in data augmentation for deep learning, and is able to increase data variations. Overall, these two approaches increase data variations, and largely increase the volume of training faces to 7174 frontal faces and 1596 side-view faces.

The training and testing strategy is as follows. From each manga title, positive examples and negative examples from 50 randomly-selected pages are collected into the training set. All candidate regions found by the selective search scheme from the remaining 10 pages are tested. Precision and recall rates will be reported in the evaluation section.

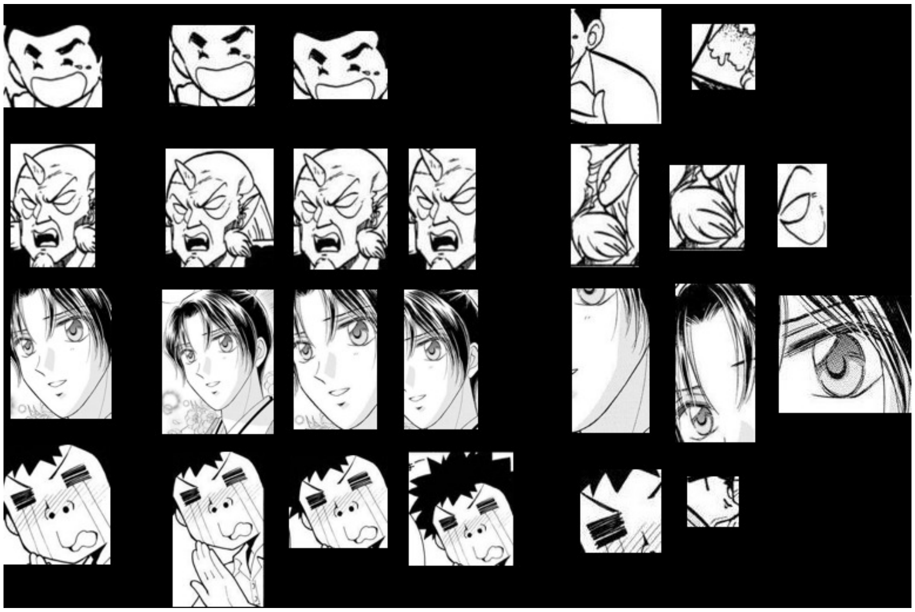


Fig. 5. Left: ground truth. Middle: the regions found by selective search with large overlap with the ground truth. Right: negative samples.

B. Baseline Manga FaceNet

Based on the training data, we construct a face detector based on convolutional neural networks (CNN). Figure 6 shows structure of the proposed Manga FaceNet. Given a candidate region found by selective search, we resize it into 64×64 pixels and input it to a CNN consisting of five convolutional layers to do feature extraction. The output of the fifth convolutional layer is flattened to be a vector, which is then input to two branches. The top branch is a classification network consisting of two fully-connected layers. The activation function of the last fully-connected layer is softmax, and thus it outputs the probability of a given region being a frontal face, a side-view face, or not a face. To train the top branch of Manga FaceNet, the loss function is set as cross entropy between the probability distribution of predicted labels p and the probability distribution of the truth labels q:

$$L_1 = -\sum_{x} p(x) \log q(x), \tag{1}$$

where x is three possible labels, i.e., frontal face, side-view face, and non-face. The reason to discriminate frontal faces from side-view faces in the training set is that side-view faces are significantly different from frontal faces. In our previous work [28], we ignored side-view faces and thus found many side-view faces cannot be detected and classified.

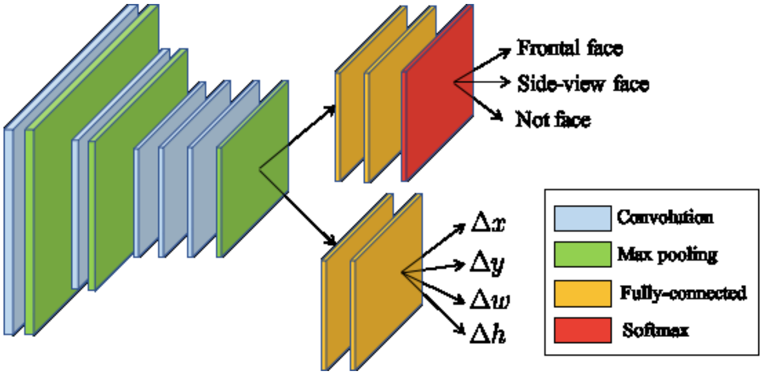


Fig. 6. Structure of the Manga FaceNet. Classification results and spatial displacement are jointly considered in the network.

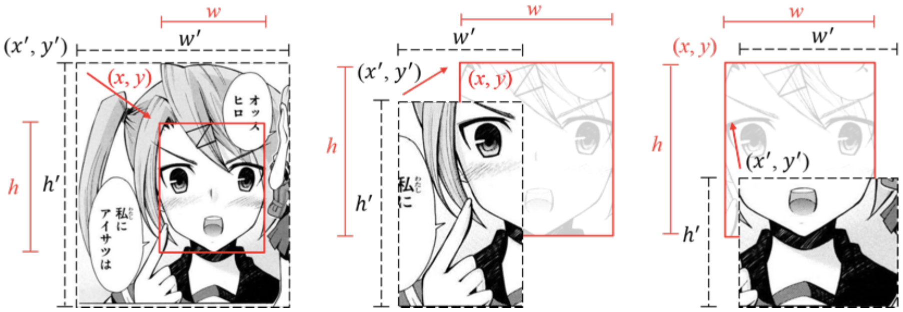


Fig. 7. Sample cases of the relationship between a detected region and the corresponding manga face.

For the bottom branch of Manga FaceNet, we attempt to further consider the spatial displacement of a given training region to its corresponding ground truth, in order to more finely evaluate the goodness of a region being a manga face. The selective search scheme may detect an object region that partially or overly covers a face region. Figure 7 shows three sample cases of the relationship between a detected region and the corresponding manga face. We use the coordinate of the left-top corner as well as width and height to represent a region. In the leftmost figure, the left-top corner of the detected region (dotted line) has to move towards right-down, and the width and the height of the region should be smaller. In the middle figure, the left-top corner of the detected region has to move towards top-right, its width should be increased, and its height should be decreased. In the rightmost figure, the left-top corner of the detected region has to move towards left-top, and its height should be increased.

Given the observations mentioned above, associated with each region, the spatial displacement and aspect ratio to the corresponding ground truth are further considered. More particularly, taking the original manga page as a coordinate system, assume that the left-top corner of a given region is at (x', y'), and its width and height are w' and h', respectively. Let (x, y) denote the coordinate of the left-top corner of its corresponding ground truth, and its width and height are w and h, respectively. Motivated by the settings in [12], the horizontal and vertical spatial displacements are calculated and normalized as $\Delta x = \frac{x'-x}{w}$, and $\Delta y = \frac{y'-y}{h}$, respectively. The width difference Δw and the height difference Δh are calculated as $\Delta w = \log \frac{w'}{w}$, and $\Delta h = \log \frac{h'}{h}$, respectively. Similar to the top branch of Manga FaceNet, we design two fully-connected layers to estimate the spatial displacement and aspect ratio change. Given a candidate region, if the predicted value of spatial displacement and aspect ratio difference are $\Delta \hat{x}$, $\Delta \hat{y}$, $\Delta \hat{w}$, and $\Delta \hat{h}$, respectively, the loss function to train the bottom branch of Manga FaceNet is the mean square error:

$$L_2 = \frac{1}{N} \sum (\Delta x - \Delta \hat{x})^2 + (\Delta y - \Delta \hat{y})^2 + (\Delta w - \Delta \hat{w})^2 + (\Delta h - \Delta \hat{h})^2, \tag{2}$$

where N is the number of training regions.

Overall, the top branch and the bottom branch of the Manga FaceNet are jointly trained by considering the integrated loss

$$L = \lambda_1 L_1 + \lambda_2 L_2, \tag{3}$$

where both weighting parameters λ_1 and λ_2 are currently set as 1.

Table I shows detailed configurations of Manga FaceNet. The input region is processed through five convolutional layers, as shown in the second row of Table I, from left to right. The convolutional parameters are denoted as "conv⟨receptive field size⟩ - ⟨number of channels⟩". The ReLU activation function is used in all convolutional layers. The first, the second, and the fifth convolutional layers are followed by max pooling and dropout with ratio 0.25. Results of convolution are input to two fully-connected layers, where the first one consists of 256 nodes, and the second one consists of 4 nodes and 3 nodes for the bottom branch and the top branch, respectively. Output of the final fully-connected layer of the top branch is fed to a softmax function, yielding the probabilities of frontal face, side-view face, and non-face.

We adopt mini-batch of size 100, and the learning process updates network parameters for 60 epochs. The learning algorithm is RMSprop (for Root Mean Square Propagation) [31], with the learning rate 0.001. We employ a strategy similar to the "image-centric" sampling strategy [32] to the train the network. A mini-batch contains positive samples and negative samples both sampled from the same manga title, i.e., more like "manga-title-centric" sampling.

TABLE I

Detailed configuration of Manga FaceNet.

input (64×64 gray images)				
conv3-32 maxpooling dropout(0.25)	conv3-64 maxpooling dropout(0.25)	conv3-128	conv3-128	conv3-128 maxpooling dropout(0.25)
fully-connected (256 nodes) (both branches) dropout(0.5)(both branches)				
fully-connected (4 nodes) (bottom branch) fully-connected (3 nodes) – softmax (top branch)				

C. Fused Manga FaceNet

Anwer et al. [33] investigated two deep architectures, namely early and late fusion, to combine the texture and color information. They verified that the information from the texture network and the color network can be combined by joining them either at the convolutional level or at the fully connected layers. In our work, we design two versions of fused Manga FaceNet (fMFN) to combine information from the global face and local face parts. In early fusion, we concatenate information derived from multiple regions to do classification. In late fusion, we fuse classification results respectively obtained based on the global face model and four local face models.

Figure 8 illustrates the idea of early fusion. Given a candidate region found by selective search, we equally divide it into two parts horizontally and vertically, i.e., a candidate region is divided into four local regions. The four local regions are the left-top, right-top, left-bottom, and right-bottom parts of a face. The original candidate region and the four local regions are then resized into 64×64 pixels. In the early fusion of fMFN, we jointly consider global and local information.

We separately input these regions into the VGG-16 model [34] that consists of thirteen convolutional layers to extract features. Table II shows detailed configuration of the VGG-16 model, which is a powerful CNN model trained based on the large-scale ImageNet data. The VGG-16 model has been demonstrated to yield very promising performance for image recognition. We thus use VGG-16 as a pre-trained model to help us extract features. The output of the final fully-connected layer is viewed as the feature representation. Representations of four local parts and the global face are concatenated as the final representation, which is then input

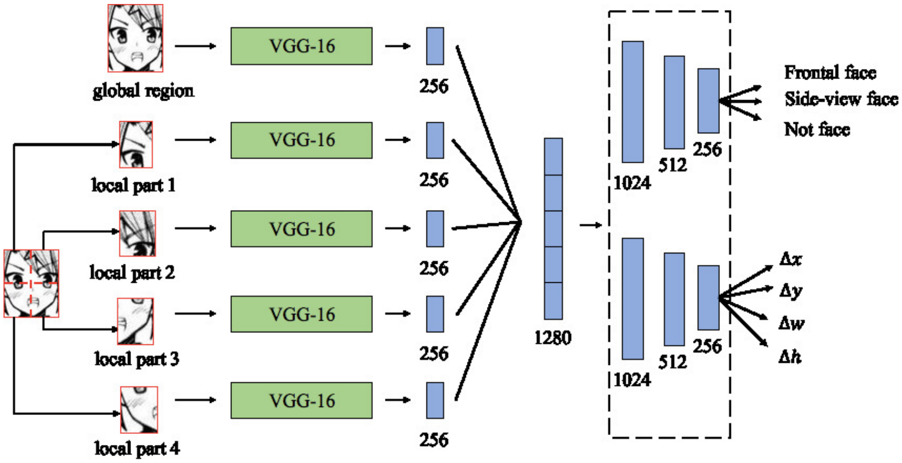


Fig. 8. Illustration of the idea of the early fusion version of fMFN.

TABLE II

DETAILED VGG-16 CONFIGURATION [34].

conv3-64 conv3-64 maxpooling	conv3-128 conv3-128 maxpooling	conv3-256 conv3-256 conv3-256 maxpooling	conv3-512 conv3-512 conv3-512 maxpooling	conv3-512 conv3-512 conv3-512 maxpooling
fully-connected (4096 nodes)				
fully-connected (4096 nodes)				
fully-connected (1000 nodes) softmax				

to two branches of fully-connected layers, similar to the structure shown in Figure 4. In early fusion of fMFN, we concatenate the five features and use it as the input to three fully-connected layers, inside them 1024, 512, and 256 nodes are adopted, respectively.

Figure 9 illustrates the idea of late fusion. Similarly, a candidate region detected by selective search is divided into four local regions. The global region is input to the VGG-16 model [34], followed by a fully-connected layer containing 256 nodes, to extract features and conduct classification. The regions classified as frontal faces or side-view faces by the first CNN model are further refined by checking their corresponding local regions. Four local regions are input to four CNN models, followed by a fully-connected layer containing 256 nodes, to extract features and conduct second-round classification. If more than two local regions are also classified as frontal faces or side-view faces, the original global region is finally verified as a manga face.

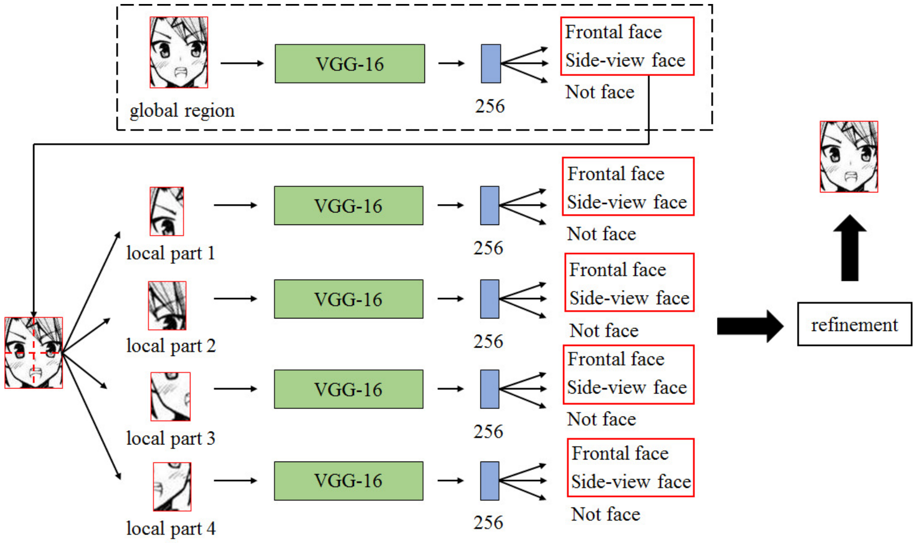


Fig. 9. Illustration of the idea of the late fusion version of fMFN.

In both the early fusion and the late fusion of fMFN, we fine-tune the VGG-16 model based on our training data to extract features. To train the top branch of the early fusion and the late fusion of fMFN, the loss function is set as:

$$L_1 = -\pi \times \exp(-\sigma) \times \sum_x p(x) \log q(x), \tag{4}$$

where $\sigma = \frac{A_s \bigcap A_g}{\max(A_s, A_g)}$. A_s is a considered region (either a global region or a local region), and A_g is the corresponding ground truth. We consider intersection of overlap between a considered region and its corresponding ground truth. In training, if the region's σ is smaller, larger penalty will be given if it is falsely predicted. The value π is set as 0.1 when the sample is a frontal face or a non-face, and π is set as 0.8 when sample is a side-view face. Because the number of training data of frontal faces is much larger than that of side-view faces, side-view faces are panelized more when they are falsely predicted.

The loss function of the bottom branch of the early fusion and late fusion of fMFN is the same as eqn. (2). Finally, we jointly train the top and the bottom branches based on the loss function integrating L_1 and L_2. The integration factors are the same as that defined in eqn. (3), i.e., λ_1 and λ_2 are set as 1.

To train the two fMFNs, we adopt mini-batch of size 25, and the learning process updates network parameters for 60 epochs. The learning algorithm is stochastic gradient descent, with the learning rate 0.0001. The “manga-title-centric” sampling strategy is also used in training the fused Manga FaceNets.

IV. Evaluation

A. Dataset and Evaluation Settings

We evaluate the proposed system based on a large-scale manga benchmark, i.e., the Manga109 dataset [2]. We randomly select 24 titles from the 109 titles, and from each title we select 60 manga pages and manually define ground truths of manga faces. After data augmentation, there are 7,174 frontal faces and 1,596 side-view faces in total. To make the number of positive samples and the number of negative samples balanced, we randomly selected 7,830 negative samples from the selected manga pages.

According to the framework shown in Figure 4, when testing we first detect object regions by the selective search scheme, and then estimate the probability of each region being a face. If the probability of frontal face or side-view face is larger than non-face, we say the region is a manga face.

One may wonder that if the selective search scheme is able to detect regions covering manga faces. To verify this, we calculate the ratio of manga faces that can be included in the regions found by selective search. According to our experiment, this ratio is around 92%. In the following experiments, we will use precision and recall values to measure performance of manga face detection. Because of the designed procedure shown in Figure 4, the value 0.92 is therefore the upper bound of the recall value we can obtain.

B. Performance of Manga Face Detection

Table III shows performance comparison between different manga face detection methods. As we expect, the AdaBoost method (embedded in the OpenCV library) originally designed for detecting real human faces does not work well (the first row). To decrease the mismatch problem between real human faces and manga faces, we retrain the OpenCV model based on our manga data. This is more like the method proposed in [23], though the training data are not the same. From the second row of Table III, we see that the recall rate can be largely improved, but the precision rate largely decreases. Overall, only 0.11 F-measure can be obtained by the AdaBoost method.

Intuitively the shape of eyes is similar to round or elliptical. Eyes are the most important elements showing different artists' drawing styles or showing different characters. Therefore, Chu and Chao [26] extracted HOG features and constructed an SVM classifier to detect eye regions. They then extended the detected eye regions to find a minimum bounding box covering a character's face. The third row of Table III shows that the eye-based method still yields limited performance, because accurately detecting manga eyes itself is not a trivial problem.

We also try the state-of-the-art face detection API on the cloud. The fourth row shows performance of face detection yielded by Microsoft Cloud API [6]. Twenty-seven predefined landmark points on a face are detected by this API. However, these points cannot be reliably detected on manga faces, and the results are far from satisfaction. Two of the state-of-the-art face detection methods, i.e., MTCNN [4] and Facenet [35], are also evaluated. As can be seen in the fifth and the sixth rows, MTCNN and Facenet achieve higher precision just like Microsoft API, but they still miss many manga faces and the recall rates are low.

The seventh row shows performance of the proposed Manga FaceNet. We see much better balance between precision and recall can be obtained. When both branches are considered in training, better performance with F-measure equal to 0.60 can be obtained. Comparing Manga FaceNet with MS Cloud API and Facenet, we clearly see the value of a dedicated manga face detection method, which yields much better overall performance.

To further improve MFN, we propose two fused MFNs. The last two rows show that performance much better than the baseline MFN approach can be obtained. This verifies that combining global features and local features is effective. Overall, the late fusion version works slightly better than the early fusion version.

Figure 10 further shows precision-recall curves of the baseline MFN, early fusion of fMFN, and late fusion of fMFN, respectively. We see that the fused MFNs clearly work better than the baseline MFN. The overall performance obtained by the early fMFN and by the late fMFN is similar (Table III). However, the early fMFN achieves higher precision when the recall rate is relatively lower, and the late fMFN achieves higher precision when the recall rate is relatively higher. We thus can choose one of them to detect manga faces, depending on which factor (precision or recall) is emphasized more.

TABLE III

PERFORMANCE COMPARISON BETWEEN METHODS.

Methods	Precision	Recall	F-measure
OpenCV (pre-trained)	0.42	0.03	0.06
OpenCV (trained with manga) [23]	0.06	0.54	0.11
Eye-based method [26]	0.06	0.12	0.08
MS Cloud API [26]	0.98	0.03	0.05
MTCNN [4]	0.50	0.06	0.10
Facenet [35]	0.90	0.10	0.18
Manga FaceNet	0.51	0.73	0.60
Early fMFN	0.62	0.73	**0.67**
Late fMFN	0.64	0.75	**0.69**

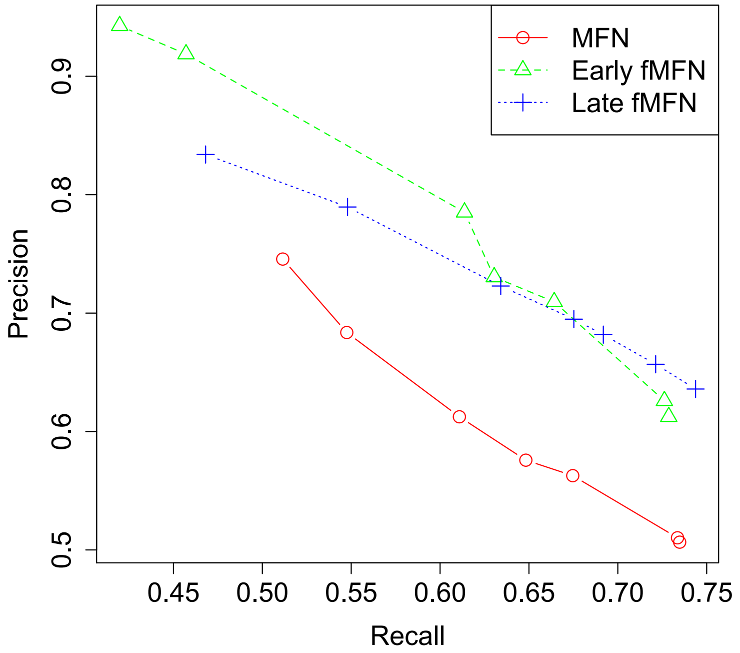


Fig. 10. PR curves of the three proposed Manga FaceNets.

TABLE IV

PERFORMANCE VARIATIONS GIVEN BY DIFFERENT FINE-TUNING STRATEGIES.

Methods	Precision	Recall	F-measure
Early fMFN (w/o fine-tuning)	0.57	0.69	0.62
Early fMFN (only the 5th con. blocks are fine-tuned)	0.58	0.76	0.66
Early fMFN (with fine-tuning)	0.62	0.73	**0.67**

TABLE V

PERFORMANCE OF THE BASELINE MFN WITH DIFFERENT VOLUMES OF TRAINING DATA.

Methods	Precision	Recall	F-measure
20 pages	0.45	0.71	0.55
30 pages	0.46	0.73	0.56
40 pages	0.47	0.76	0.58
50 pages	0.51	0.73	0.60

C. Performance Variations Given by Fine-tuning

In fMFNs, we use the VGG-16 model as the pre-trained model to help us extract features. Here we compare performance with and without fine-tuning parameters of the convolutional layers in VGG-16. Based on early fusion of fMFN, the first row of Table IV shows the performance obtained without fine-tuning. We see it is slightly better than the baseline MFN (comparing with Table III). If we fine-tune the 5th convolutional block of VGG-16, the performance improves (second row of Table IV). If all convolutional blocks are fine-tuned, the best performance can be obtained. This experiment verifies effectiveness of fine-tuning.

D. Influence of the Volume of Training Data

We also study how the volume of training data influence performance. We intentionally choose 20, 30, 40, and 50 pages from the 24 manga titles in the Manga109 dataset as training data, respectively. Based on these settings, the numbers of training data are 7000, 9600, 13400, and 16600 faces, respectively. Table V shows that, when more training data are available, better performance can be obtained. This encourages us to manually or semi-manually label more training data in the future.

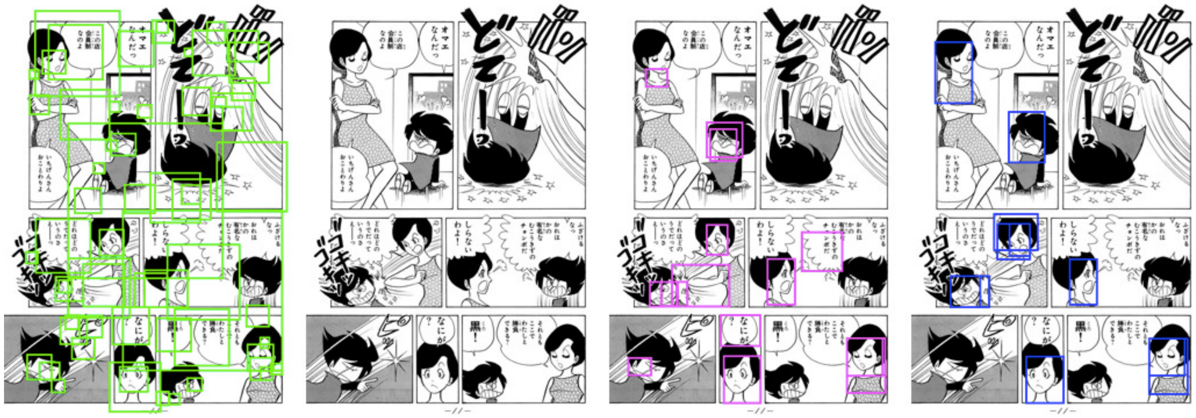


Fig. 11. Sample detection results. From left to right: (a) OpenCV trained with manga; (b) Microsoft Azure face API; (c) Manga FaceNet; (d) Late fMFN.

E. Sample Results

In addition to Figure 3, we show two more sets of sample results obtained by different methods in Figure 11 and Figure 12. In each figure, the subfigures from left to the right are results obtained by OpenCV trained with manga, by the Microsoft Azure face API, by the baseline Manga FaceNet, and by the late fusion version of fMFN, respectively. As can be seen, there are many false alarms detected by the OpenCV module. Although the recall rate of OpenCV is satisfactory, precision of OpenCV is quite bad (the second row of Table III). On the other hand, the regions detected by the Microsoft Azure face API are very likely manga faces. However, many manga faces cannot be successfully detected because this API was designed for human face detection. The proposed Manga FaceNet and late fusion of fMFN can achieve detection results with more balanced precision and recall rates. Comparing the baseline MFN with the late fMFN, we found that sometimes the speech balloons are erroneously detected as manga faces by the baseline MFN, while the late fMFN improves this case slightly.

Although the proposed models yields encouraging detection performance, they are still far from perfect. The top row of Figure 13 shows manga faces correctly detected by the late fused Manga FaceNet. We see that faces in different poses and expressions can be detected. The middle row shows false alarms. We observe that speech balloons or regions with large white area may be falsely detected as faces. In the future, we may develop a module that specifically detects text regions to eliminate false alarms. More negative training data may be also helpful in reducing false alarms. The bottom row shows miss cases. We see that faces wearing glasses yield

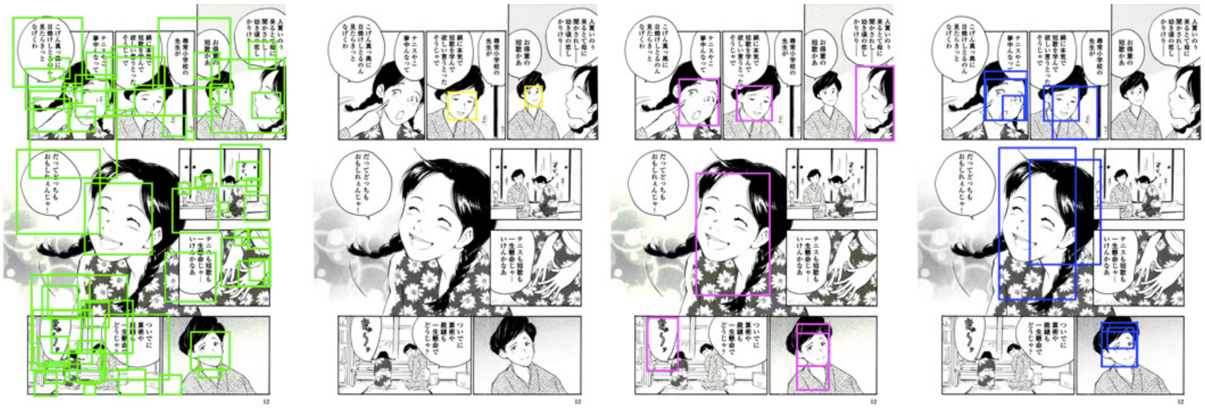


Fig. 12. Sample detection results. From left to right: (a) OpenCV trained with manga; (b) Microsoft Azure face API; (c) Manga FaceNet; (d) Late fMFN.

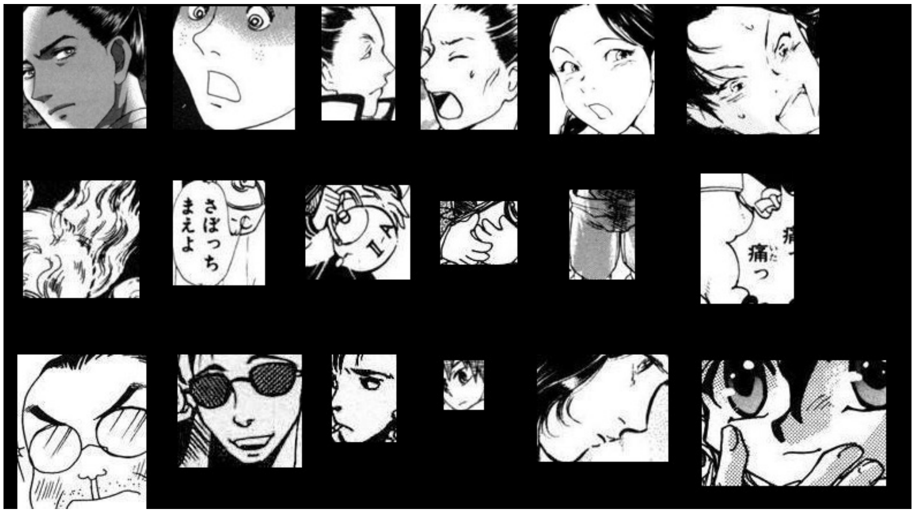


Fig. 13. Sample face detection results. Top row: detected faces; middle row: false alarms; bottom row: miss.

significant challenges. The third and the fourth images of the bottom row show that tiny faces are difficult to detect. Finally, the proposed models are still not robust to the faces in abnormal poses (like the second last image).

V. Conclusion

We have presented a deep-based face detection method specially designed for Manga. Given a manga page, we first find candidate regions by the selective search scheme, and then determine each region as a manga face or not by the three proposed versions of Manga FaceNets. In addition to classification, we also jointly consider spatial displacement and aspect ratio in the three proposed networks. Being able to model high variations on visual appearance and expression, the three proposed methods significantly outperform the methods designed for real human faces and conventional manga face detection methods.

Currently we train the proposed network based on manga faces. For detecting real human faces, several deep neural networks have been constructed based on large-scale face images. In the future, we will put more focus on transferring the knowledge embedded in such models to our Manga FaceNets, in order to get performance improvement. Furthermore, we will consider the scheme proposed in [36] to detect tiny manga faces.

References

[1] A. N. Network. Japanese manga book market rises to record 282 billion yen. [Online]. Available: http://www.animenewsnetwork.com/news/2015-01-23/japanese-manga-book-market-rises-to-record-282-billion-yen/.83614

[2] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based manga retrieval using manga109 dataset," *Multimedia Tools and Applications*, 2016.

[3] T. Zhang, J. Li, W. Jia, J. Sun, and H. Yang, "Fast and robust occluded face detection in atm surveillance," *Pattern Recognition*, 2017.

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

[5] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision*, 2015.

[6] *Microsoft Azure Face Detection API*, Microsoft, August 2017, https://azure.microsoft.com/en-us/services/cognitive-services/face/.

[7] H. Wei, C. Yang, and Q. Yu, "Contour segment grouping for object detection," *Journal of Visual Communication and Image Representation*, vol. 48, pp. 292–309, 2017.

[8] J. Shen, X. Zuo, J. Li, W. Yang, and H. Ling, "A novel pixel neighborhood differential statistic feature for pedestrian and face detection," *Pattern Recognition*, vol. 63, pp. 127–138, 2017.

[9] X. Yan, Y. Wang, Q. Song, and K. Dai, "Salient object detection via boosting object-level distinctiveness and saliency refinement," *Journal of Visual Communication and Image Representation*, vol. 48, pp. 224–237, 2017.

[10] J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[11] Y. Ban, S.-K. Kim, S. Kim, K.-A. Toh, and S. Lee, "Face detection based on skin color likelihood," *Pattern Recognition*, vol. 47, no. 4, pp. 1573–1585, 2014.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proceedings of Neural Information Processing Systems*, 2015.

[13] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *Proceedings of IEEE International Conference on Computer Vision and Patten Recognition*, 2017.

[14] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of IEEE International Conference on Computer Vision and Patten Recognition*, 2001, pp. 511–518.

[15] D. Chen, S. Ren, Y. Wei, X. Cao, and J. Sun, “Joint cascade face detection and alignment,” in *Proceedings of European Conference on Computer Vision*, 2014, pp. 109–122.

[16] S. Liao, A. K. Jain, and S. Z. Li, “A fast and accurate unconstrained face detector,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 211–223, 2016.

[17] V. Kumar, A. Namboodiri, and C. Jawahar, “Visual phrases for exemplar face detection,” in *Proceedings of IEEE International Conference on Computer Vision*, 2015, pp. 1994–2002.

[18] R. Ranjan, V. Patel, and R. Chellappa, “A deep pyramid deformable part model for face detection,” in *Proceedings of IEEE International Conference on Biometrics Theory, Applications and Systems*, 2015.

[19] S. Farfade, M. Saberian, and L.-J. Li, “Multi-view face detection using deep convolutional neural networks,” in *Proceedings of ACM International Conference on Multimedia Retrieval*, 2015, pp. 643–650.

[20] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Proceedings of Neural Information Processing Systems*, 2012.

[21] X. Sun, P. Wu, and S. C. Hoi, “Face detection using deep learning: An improved faster rcnn approach,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2017.

[22] Y. Li, B. Sun, T. Wu, Y. Wang, and W. Gao, “Face detection with end-to-end integration of a convnet and a 3d model,” in *Proceedings of European Conference on Computer Vision*, 2016.

[23] W. Sun and K. Kise, “Similar partial copy detection of line drawings using a cascade classifier and feature matching,” in *Proceedings of International Workshop on Computational Forensics*, 2010, pp. 121–132.

[24] K. Takayama, H. Johan, and T. Nishita, “Face detection and face recognition of cartoon characters using feature extraction,” in *Proceedings of IIEEJ Image Electronics and Visual Computing Workshop*, 2012.

[25] H. Yanagisawa, D. Ishii, and H. Watanabe, “Face detection for comic images with deformable part model,” in *Proceedings of IIEEJ Image Electronics and Visual Computing Workshop*, 2014.

[26] W.-T. Chu and Y.-C. Chao, “Line-based drawing style description for manga classification,” in *Proceedings of ACM International Conference on Multimedia*, 2014, pp. 781–784.

[27] W. Sun, J.-C. Burie, J.-C. Ogier, and K. Kise, “Specific comic character detection using local feature matching,” in *Proceedings of International Conference on Document Analysis and Recognition*, 2013.

[28] W.-T. Chu and W.-W. Li, “Manga facenet: Face detection in manga based on deep neural network,” in *Proceedings of ACM International Conference on Multimedia Retrieval*, 2017, pp. 412–415.

[29] F. Wang, L. Kang, and Y. Li, “Sketch-based 3d shape retrieval using convolutional neural networks,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1875–1883.

[30] F. Zhu, J. Xie, and Y. Fang, “Learning cross-domain neural networks for sketch-based 3d shape retrieval,” in *Proceedings of AAAI Conference on Artificial Intelligence*, 2016, pp. 3683–3689.

[31] G. Hinton, N. Srivastava, and K. Swersky. rmsprop: Divide the gradient by a running average of its recent magnitude. [Online]. Available: http://www.cs.toronto.edu/∼tijmen/csc321/slides/lecture_slides_lec6.pdf

[32] R. Girshick, “Fast r-cnn,” in *Proceedings of International Conference on Computer Vision*, 2015, pp. 1440–1448.

[33] R. Anwer, F. S. Khan, J. van de Weijer, and J. Laaksonen, “Tex-nets: Binary patternns encoded convolutional neural networks for texture recognition,” in *Proceedings of ACM International Conference on Multimedia Retrieval*, 2017, pp. 125–132.

[34] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proceedings of IEEE International Conference on Learning Representations*, 2015.

[35] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2015.

[36] P. Hu and D. Ramanan, “Finding tiny faces,” in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2017.