Low Resolution Facial Landmark Detection with Improved Convolutional Neural Network

Qisi Wang
qisi.wang@wisc.edu

Xiaomin Zhang
xzhang682@wisc.edu

Jinman Zhao
jzhao237@wisc.edu

May 6, 2016

1 Abstract

Aiming to detect the landmarks of low resolution images, we propose seven approaches with carefully designed Convolutional Neural Network structures. Some of our approaches contain novel thoughts such as using distribution model, parallel channels and masks. We use 20 x 20 images for detecting. In general, there are 3 advantages of our methods: First, it takes advantage of the characteristic of low resolution, using only 5 landmarks for detection; Second, the network structures we designed are concise and easy-trained; Third, the testing time is fast, which our approach outperforms the state-of-art. We compare all the structures and conclude that the best structure has distribution model and single mask.

2 Introduction

Facial landmark detection refers to the problem of detecting a set of predefined facial fiducial points[13]. It plays a crucial role in computer vision tasks related to faces, such as face and expression recognition, face alignment and face tracking. Despite the large extensive study of facial landmark detection in high resolution images in both controlled or uncontrolled environment, the problem of facial landmark detection in low resolution images is seldom touched.

However, in practical, high resolution of the captured faces are not guaranteed, as in the case of surveillance camera, where wide-angle cameras are normally used and installed to maximize the viewing angle. On the other hand, in such images with severely degraded quality, the performance of the detection system usually declines by a lot. In particular, it has been shown that for the task of face recognition, minimum face image resolution of around 32x32 and 64x64 is required for existing algorithms[12]. Therefore, with the growing installation of surveillance camera and the greater needs for face recognition at a distance, there is an increasing demand for facial landmark detection in low resolution images.

In this project, we explore a various neural network system for effective and efficient detection of facial landmark under the low resolution condition. Structurally, we examine both the base neural network and systems consist of three parallel convolutional neural network each focus on different part of the face and the whole face structure. On the neural network technique side, we adopt mask layer to compensate for lighting condition and focus on potential location of facial landmarks. We also propose different model to evaluate the fidelity of the computed location and design the learning loss accordingly.

More results and supplementary materials can be found in our project website (https://pages.cs.wisc.edu/~qisiw/CS766site/).

3 Background

3.1 Landmark detection

The existing landmark detection frameworks can be classified into two general categories: generative methods and discriminative methods. The generative approaches tries to fit a model of the input face images and optimize overall shape configuration of the landmarks. The discriminative approaches, on the other hand, look for each facial landmark independently and optimize the fitting based on some resemblance metrics at each landmark.

The most widely studied approaches for generative landmark detection methods are Active Appearance Model (AAM)[8] and Active Shape Model(ASM).

The AAM uses a statistical model for shape and texture parameters to generate new instance of facial images. The algorithm then designs some metrics to estimate the difference between the test images and the generated ones and updates the parameters accordingly. The ASM, on the other hand, employs only the shape parameters and uses a local search around each point of the shape to estimate the landmarks.

The discriminative approaches try to learn a classification function and compute a confidence for each position in the images. The image position with the largest confidence is then picked. However, with this kind of approaches, it is sometimes insufficient to detect the correct facial landmark location in the uncontrolled environment since the local feature in the images can be much more complicate and some background features may resemble those of an actual landmark.

3.2 Face in low resolution

Right now there's no study explicitly exploring the problem of detection facial landmark in low resolution images. The problem of face recognition in low resolution images, on the other hand, gains a lot of attention and is discussed extensively in the literature. In some face recognition papers, however, some crude facial landmark detection procedure are used as a preprocessing step. [7], for example, uses the the median location for each poses as a crude estimation of the facial landmark location and use it to guide the subsequent recognition.

One extensively applied technique for processing face images with low resolution is super-resolution (SR). This flavor of algorithm basically exploits the self-similarity of the face images and attempts to recover the missed details to enhance the resolution of the face images.

4 Approach

4.1 Facial landmark model

In this project, we focus on the structural design of the training networks. We consider two main models of the networks and the modifications of them. In general, there are five landmarks to be detected. The left eye, right eye, nose, left mouth corner and right mouth corner (Figure 1). Our landmarks are continuous rather than locate on pixels.

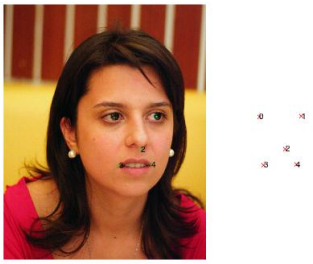


Figure 1: Landmarks

4.1.1 Geometric model

The first model is to detect the five points' coordinates directly. Mathematically our goal is to minimize the mean squared distances between the predictions and true coordinates. So we could use MSE as the objective function.

$$\min \frac{1}{10}\sum_{i=1}^{10}(y_{pred,i} - y_{true,i}))^2$$

where $(y_{pred,2*k-1}, y_{pred,2*k})$, $k = 1, \ldots, 5$ are the coordinates of predicted landmarks, $(y_{true,2*k-1}, y_{true,2*k})$, $k = 1, \ldots, 5$ are the coordinates of true landmarks.

4.1.2 Statistic model

The second model is to represent the location of a facial landmark as a distribution over the face bounding window. The intuition is that neural networks could possibly learn distribution, which is more like a classification, better than regression.

However, the distribution over all image pixels could be intactable because of the complexity and excessive freedom. Thus we propose the distribution instead onto the four corners of the face bounding window.

Assuming $q(i, j)$ is the true distribution onto the corners and $p(i, j)$ is the predicted distribution, where $i, j \in \{0, 1\}$. The categorical cross entropy is used as the objective function[1].

$$\min(H(p) + D_{KL}(p||q)),$$

where $H(p)$ is the entropy of distribution p, and $D_{KL}(p||q)$ is the Kullback-Leibler divergence of q from p(also known as the relative entropy of p with respect to q.)

Since we discretize the distribution in our image,

$$H(p, q) = -\sum_{i,j\in\{0,1\}} p(i, j) \log q(i, j),$$

where (i, j) indicates the four courners of the face bounding window. The original facial landmark coordinate can be recovered from the corner distribution by doing the weighted average of four corners.

$$C(p) = \sum_{i,j\in\{0,1\}} p(i, j)coord(i, j),$$

where $C(p)$ is the corresponding coordinate recovered from distribution p, $coord(i, j)$ is the coordinate for the corners.

4.2 Design of neural networks

We design and try various kinds of structures as following. For easy reference, we use short code to donate all the models we mentioned in this section.

model code	refered model
M1	Section 4.2.1
M2	Section 4.2.2, a variation
M2'	Section 4.2.1
M3	Section 4.2.3
M4	Section 4.2.4
M4'	Section 4.2.5
M1-distr	Section 4.2.6
M4-distr	Section 4.2.7

4.2.1 Base CNN

Figure 2,3,4 show our base CNN structure trying to solve the facial landmark detection. All forthcoming structures are variations or imporvements of the base structures.

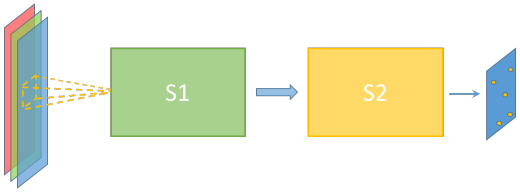


Figure 2: Base CNN

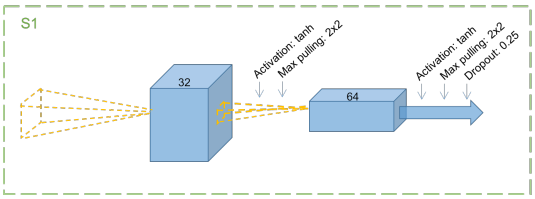


Figure 3: Substructure: S1

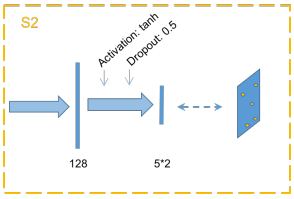


Figure 4: Substructure: S2

We use the normal modeling of facial landmark locations: geometric (coordinate) representation with MSE loss.

Previous works usually obtain the landmarks using only grayscale image and features extracted by SIFT or Hog algorithm. For the input of our network, we use all 3 rgb channels of the image. Also instead of plugging into the compulsory features, we design 2 convolutional neural network to extract the features, add one fully connected network to weight the features and then obtain the landmarks.

Convolutional layer is denoted by $S1$ here. Suppose we have a $H \times N \times N$ which is followed by the convolutional layer and we use a $H \times m \times m$ filter w, our convolutional layer output will be of size $(N-m+1) \times (N-m+1)$. Assuming $o(h', i, j)$ is the output at the l^{th} hidden layer, $in(h, i, j)$ is the output from the previous layer, then

$$o(h', i, j)^l = \sum_{h=h'}^{H+h'} \sum_{k_1=0}^{m-1} \sum_{k_2=0}^{m-1} (w(h, k_1, k_2)^l in(h, i+k_1, j+k_2)^{l-1} + b(h, k_1, k_2)^l)$$

Then the convolutional layer applies its nonlinearity:

$$in(h', i, j)^l = \sigma(o(h', i, j)^l)$$

Fully connected layer is denoted by $S2$ here. It is formulated as :

$$o(j)^l = \sum_{k=0}^{m-1} in(i)^{l-1} w(i, j) + b(j)$$

$$in(j)^l = \tanh(o(j)^l)$$

4.2.2 Add one more fully connected layer

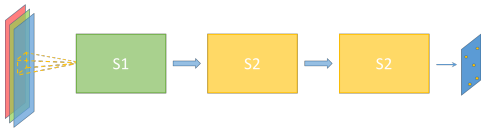


Figure 5: Add one more fully connected layer

We add one more substructure of fully connected network at the end of our base structure (Figure 5) in order to learn a delicate way to synthesis the landmark locations more carefully.

4.2.3 Parallel model

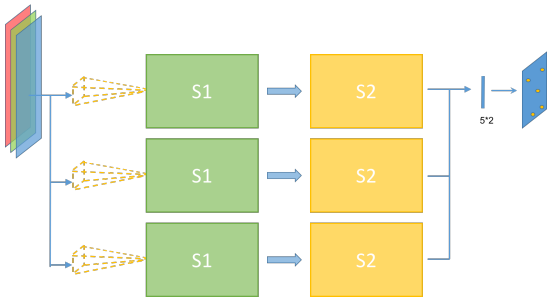
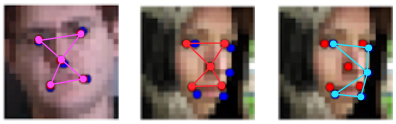


Figure 6: Parallel model

This parallel structure (Figure 6) tries to have multiple base CNN structures work parallely. Hopefully they could be able to focus on different aspect of facial landmark locations and finally give better and more robust results when combined together. One example of this kind of specialization of parallel can be seen in previous work of object recognition task, where two parallel deep CNN autonomously specialized with intensity and color information.

Consider a rotated face, the structure of the 5 landmarks can be deformed a lot. When the face is rotated, the structure of landmarks could be deformed a lot, for example: Fig.7a indicates the frontal face and Fig.7b, 7c indicates the rotated face. The predicted result is Fig.7b. The structure still looks like the frontal face however the truth is Fig.7c. Therefore, we decompose the 5 landmarks into two parts: the upper triangle(eyes and nose) and the lower triangle(nose the mouth corners). And we add these two parts into another two channels. Finally we get a fully connected network to combine these 3 channels together.



(a) a (b) b (c) c

Figure 7: Deformed landmarks' structure

4.2.4 Attention mask

Figure 8 shows a structure with a single mask.

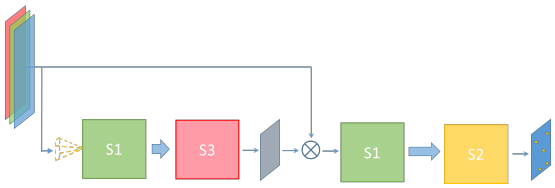


Figure 8: Attention mask

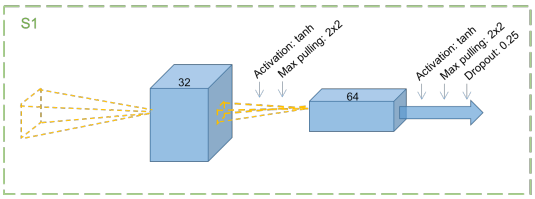


Figure 9: Substructure: S3

The intuition of using mask is to help succsesive layers focus on the important areas for locating landmasks. We are inspired by the attention NN which helped generate natural language descriptions for image scenes [17].

The mask operation is formulated as:

$$o(i,j)^l = M(i,j) * in(i,j)^{l-1},$$

where $M(i,j)$ is either 1 or 0. 1 represents keeping the pixel while 0 represents filtering out the pixel.

4.2.5 Parallel attention mask

Figure 10 shows a structure with masks.

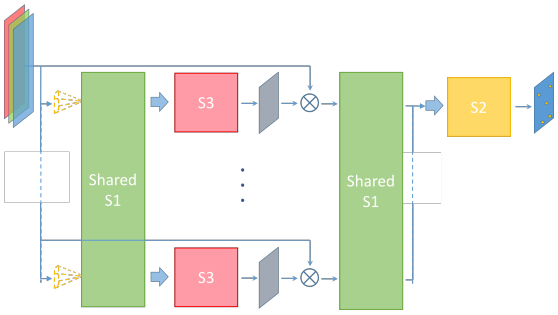


Figure 10: Parellel attention mask

For each landmark, we prepare a mask seperately, in the hope that a specialized mask can be generated for each facial landmark.

4.2.6 Base network for distribution output

Figure 11,12 shows a structure based on the distribution network.

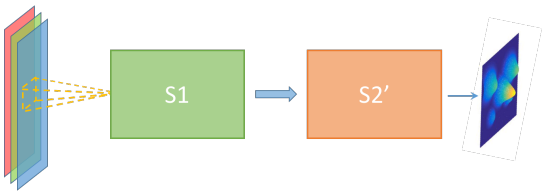


Figure 11: Distribution network

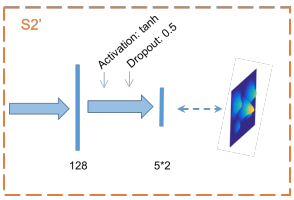


Figure 12: Substructure: S2'

In this and the following structure, we use the distribution model of facial landmarks with categorical loss. The intuition is to construct two convolutional neural network to extract the features and two fully connected networks to combine the features.

The prediction need to be converted back to coordinate representation by taking the weighted average of four corner coordiantes with the distribution density.

4.2.7 Masked network for distribution output

Figure 13 shows a single mask structure with distribution model.

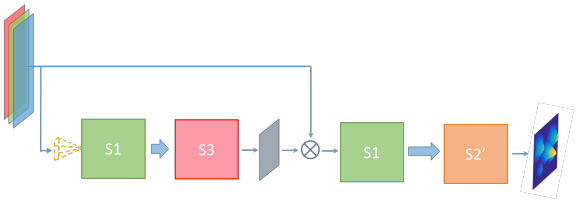


Figure 13: Substructure: M4'

Here we use one mask to filter. This is a variation of the structure M4 incorperated with distribution representation of facial landmarks.

4.3 Implemention details

We explored a bunch of various settings of neuron network parameters, such as number of neurons in the first fully connected layer within sub NN $S1$, stacking relations, different activations. We show here the results only for settings in Table 1.

For activations, we prefer tanh function to sigmoid function. This allows our models to learn faster, because the slope of tanh is larger than sigmoid.

Substructure	layer0	layer1	layer2	layer3		
S1	I(3,20,20)	C(3, 32, tanh)	P(2)	C(3, 64, tanh)	P(2)	Drop(0.25)
S2		F(128, tanh)	Drop(0.5)	F(5*2)		
S2-distr		F(128, tanh)	Drop(0.5)	F(5*4)		
S3		F(128, tanh)	Drop(0.5)	F(20*20)		

System	layer0	layer1	layer2	layer3	layer4	layer5	layer6
M1	I(3,20,20)	S1	S2				
M2	I(3,20,20)	S1	F(128, tanh)	Drop(0.5)	S2		
M2'	I(3,20,20)	S1	S2	S2			
M3	I(3,20,20)	parallel 3*S1	parallel 3*S2	Concat(3*L2)	F(5*2)		
M4	I(3,20,20)	S1	S3	Mask(L1,L2)	S1	S2	
M4'	I(3,20,20)	shared 5*S1	parallel 5*S	3parallel 5*Merge(L1,L2)	shared 5*S1	Concat(5*L4)	S2
M1-distr	I(3,20,20)	S1	S2'				
M4-distr	I(3,20,20)	S1	S3	Mask(L1,L2)	S1	S2'	

Table 1: Detailed structure of proposed models.

Detailed implementation of the NN structures described in this paper. $I(c, m, n)$ stands for input layer of an $m \times n$ image with c channels. $C(w, n, f)$ stands for convolutional layer with n neurons of convolution window size $w \times w$ and with output activation function f. $F(n, f)$ stands for fully connected layer with n neurons and with output activation function f. $P(w)$ stands for max pooling layer with pooling size $w \times w$. $Drop(p)$ means that during training phase, the output of previous layer is randomly dropped with portion p. Note this kind of layers are not effective during testing. $Mask(I, M)$ apply mask M on image I. Mask is a gray image with the same size as I and with pixel value in range $[0, 1]$. $Concat(l_layers)$ merges the outputs of all layers in l_layers by concatenating them into single output. parallel $n \times S$ means that the output of the previous layer are passed separately into n different NN with the same structure S. shared $n \times S$ means that the output of the previous layer are passed separately into n identical NN with the same structure S and also the same weights.

For optimizer, we choose Adadelta. Adadelta takes the consideration of the second derivative which is more closer to the Hessian Matrix information[5, 10, 18]. This could help to go the minimum location accurately.

5 Experiment

5.1 Data processing

We investigate different designs of network structure mentioned above with a training set and testing set gathered from multiple popular datasets in facial landmarks. This section discusses the dataset we examines and the preprocessings procedure we've used to obtain reasonable low-resolution faces as well as corresponding facial landmark data.

5.1.1 Data sets used

We gather data from various face image datasets.The following paragraph summarize the datasets used for our training and testing sets.

Multi-PIE[9] dataset contains more than 750,000 images of 337 people with different facial expression, head pose and lighting condition under lab environment. We take 2 random lighting conditions out of 20 poses from each facial expression for each subject.

Labeled Face Parts in the Wild (LFPW)[6] dataset consists of 1432 faces from images downloaded from the web.

helen[2] data set contains 2330 annotated portrait images gathered from Flickr to cover a broad range of different scenario.

IBUG[14] database, part of the 300-W challenge, consists of 135 imags downloaded from the web with large variations in expression, illumination and pose.

Annotated Faces in-the-Wild(AFW) database contains 250 images with 468 faces randomly sampled from Flickr images.

We get 8003 face images in total, in which we randomly choose 80% (6402) for training and 20% (1601) for testing. The number of faces we take from each image set are shown in Table 2.

5.1.2 Image downsampling

To obtain the targeted low resolution face images with labels, the images from the datasets mentioned above is first preprocessed to reduce the resolution of the images. To generate faces with require low resolution, the location of the face patches is determined by first locating the extreme location of the facial land mark. Then the bounding box of the face is determined with $10\% \pm 2\%$ margin with respect to maximum dimention of the face.

After determination of the bounding box, the resulted face patch can be extracted. To effectively filter out the high frequency component that may cause aliasing in the resulted downsampled image, an Gaussian filter with sigma equals to the downsampling rate is depolyed to blur the image. The blurred image is then downsampled and bilinear interpolation is applied to preserve the location information as much as possible.

5.1.3 Label generation

Different dataset may have different labeling of facial landmarks, for example Multi-PIE lables 68 landmarks per front face while LFPW labels 29 per face. So we first need to obtain the five interested facial points from

	AFW	Helen	LFPW	Ibug	Multipie	300-W
number of faces	337	2330	1008	120	3629	579

Table 2: Composition of our dataset.

them, namely left eye (LE), right eye (RE), nose (N), left corner of mouth (LM) and right corner of mouth (RM). We simply pick the landmark we are interested if it already exists among the labels, otherwise we perform simple calculation to obtain an reasonable estimation of desired facial points.

5.2 Measurements

In order to compare out method to other state of art, we give up the existed measurement in Keras. The detection error is measured as

$$err = \sqrt{(x - x')^2 + (y - y')^2}/l,$$

where (x, y) and (x', y') are ground truth and the detected position, and l is the width of the bounding box returned by our data processing.

If an error is larger than certain amount x%, we regard it as a x%-failure. We measure our structures on testing images with 5% or 10% failure, which means the distance error is within one or two pixels for 20×20 face bounding windows.

5.3 Results

We utilze neuron network library Keras[3] for programming implementation and use Theano[4] as backend. All the experiments are run with Jupyter IPython notebook based on Python 3.4, on a machine with 3.20GHz Intel Core i5-4460 CPU and NVIDIA GeForce GTX 970 GPU.

We train for 500 epochs with batch size 10 for each structure described in section 4.3.

1. **Accuracy & Error** Fig.14 shows the predicting error for seperate landmark points over all our proposed models. Fig.15 and Fig.16 shows 5% and 10% failure rate for seperate landmark points over all our proposed models.

2. **Example results of landmark detection** Example landmark locations resulted from our base and best NN structure are shown in Fig.19 and fig.20.

3. **Timing** Fig.17 shows the average predicting time over testing dataset for all proposed structures. Note that predicting time per testing image reduces significantly when processed in a batch then one by one. This could be due to the conservation of loading time and/or parallelization.

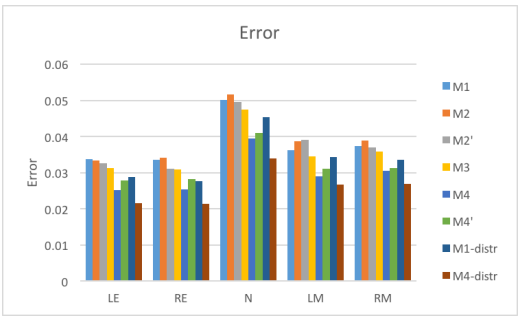


Figure 14: Average detection errors over various structures.

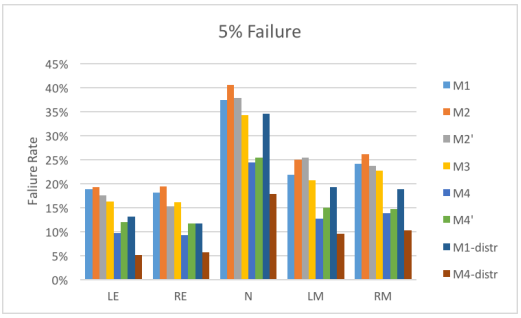


Figure 15: Average 5% detection failure over various structures.

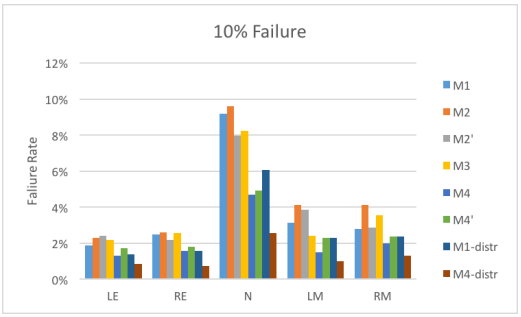


Figure 16: Average 10% detection failure over various structures.

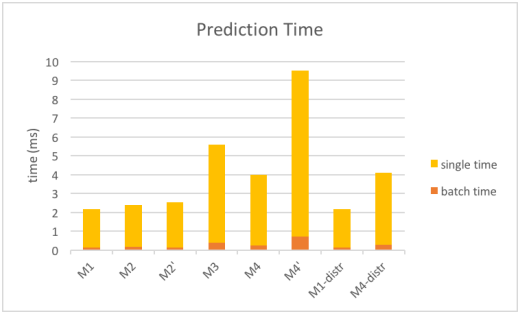


Figure 17: Average prediction time per testing image over various structures.

5.4 Comparison and discussion

Additional fully connected layers before outputing helps slightly with the accuracy. The mask layers helped dramaticaly as the 10% failure rate droped nearly 50% compare to the base structure. The distribution representation of landmark locations again reduced about 15% of error if we compare the evaluation for structure M4-distr and M4, and the failure rate dropped significantly to about 2/3. This means the distribution representation helps a lot when dealing with difficault faces.

Effect of mask. Three of our models M4, M4' and M4-distr engage the idea of attention mask, which helps those models achieve better accuracy. Fig.18 illustrates some output from M4's mask layer. It can be seen for some of the faces (e.g. the face in the middle of the first row), the masks try to emphasize the pixels near the interested facial keypoints. And for some other faces (e.g. the face in right bottom), the mask tries to suppress the background. More generally, we find the masks are trying to both average the lighting and emphasize landmarks. For example, the bottom left face has high mask values on the whole left side because the lighting of the original image comes more from right. Some cases seem not reasonable at first glance (e.g. the middle face in the left column) since the mask value is higher for hair region. However, this will give little effect in the masked face, because the pixel value is already close to zero for the dark hair. The idea of mask and its interaction with other layers may be further explored in order to achieve even better performance.

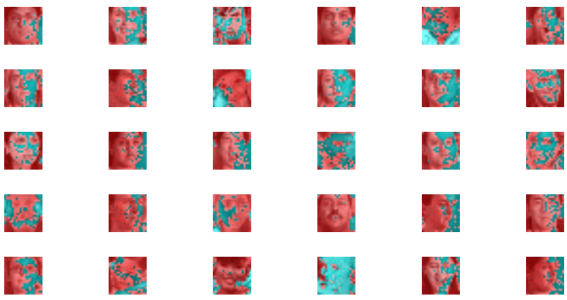


Figure 18: Example output from M4's mask layer. Red indicates higher value in mask, while blue indicates lower level in mask.

Prediction time. Our model is very competitive in the speed it gives prediction, which promises its application in realtime vedio or highly responsive facial landmark detection. It only cost 2 ms in average for our base structure M1, to predict all five landmarks for a low-resolution face, which can be even accelerated to 0.15 ms if processed in a batch. For our best model M4-distr, it also only takes less than 3.8 ms to predict for one face, which is still more than 40 times fast than the reported time in [15], and can be even speeded up to 0.3 ms when processed in a batch.

6 Conclusion

We present several approaches for landmarks detection on 20 x 20 low resolution images. Our single mask distributed network structure (Section 4.2.7) works effectively and gives the best results. For accuracy, although our method loses to [15], it outperforms [11][16]. However, given the much lower resolution of input faces, our methods still demonstrated considerably good facial landmark detection results. For testing time, this method outperforms the state-of-art.

Acknowledgements: We appreciate a lot for the support and suggestions from Prof. Mohit Gupta. We also thank Dr. Brandon M. Smith for his help on dataset and his suggestive discussion.

References

[1] https://en.wikipedia.org/wiki/Cross_entropy.

[2] http://www.ifp.illinois.edu/~vuongle2/helen/.

[3] http://keras.io/.

[4] http://deeplearning.net/software/theano/.

[5] Kevin Bache, Dennis DeCoste, and Padhraic Smyth. Hot swapping for online adaptation of optimization hyperparameters. *arXiv preprint arXiv:1412.6599*, 2014.

[6] Peter N Belhumeur, David W Jacobs, David J Kriegman, and Narendra Kumar. Localizing parts of faces using a consensus of exemplars. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2930–2940, 2013.

[7] Santosh Biswas, Geeta Aggarwal, Patrick J Flynn, and Kevin W Bowyer. Pose-robust recognition of low-resolution face images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):3037–3049, 2013.

[8] Timothy F Cootes, Gareth J Edwards, and Christopher J Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):681–685, 2001.

[9] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010.

[10] Ching-Pei Lee and Chuan-bi Lin. A study on l2-loss (squared hinge-loss) multiclass svm. *Neural computation*, 25(5):1302–1323, 2013.

[11] Lin Liang, Rong Xiao, Fang Wen, and Jian Sun. Face alignment via component-based discriminative search. In *Computer Vision–ECCV 2008*, pages 72–85. Springer, 2008.

[12] Yui Man Lui, David Bolme, Bruce A Draper, J Ross Beveridge, Geoff Givens, and P Jonathon Phillips. A meta-analysis of face recognition covariates. In *IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems, Washington, DC*, 2009.

[13] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 397–403, 2013.

[14] Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 397–403, 2013.

[15] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3476–3483, 2013.

[16] Michel Valstar, Brais Martinez, Xavier Binefa, and Maja Pantic. Facial point detection using boosted regression and graph models. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2729–2736. IEEE, 2010.

[17] Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015.

[18] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

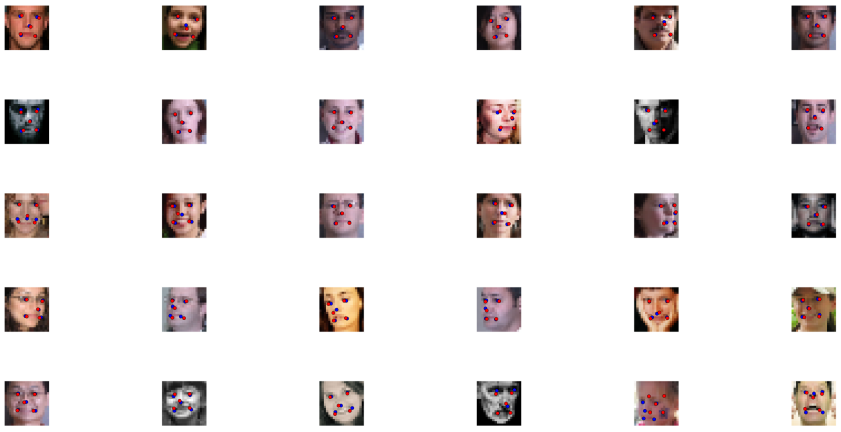


Figure 19: Example landmark location result from our base NN structure M1.

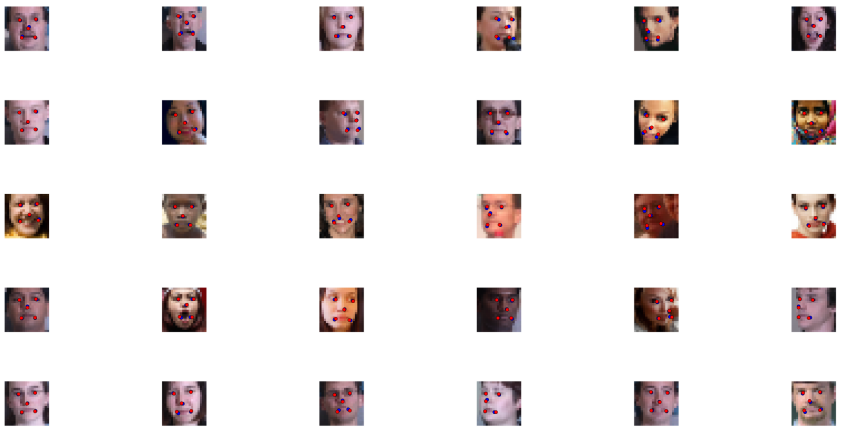


Figure 20: Example landmark location result from our best NN structure M4-distr.