End-to-End Photo-Sketch Generation via Fully Convolutional Representation Learning

Liliang Zhang
Sun Yat-sen University
zhangll.level0@gmail.com

Liang Lin *
Sun Yat-sen University
linliang@ieee.org

Xian Wu
Sun Yat-sen University
sysuwuxian@gmail.com

Shengyong Ding
Sun Yat-sen University
marcding@163.com

Lei Zhang
The Hong Kong Polytechnic University
cslzhang@comp.polyu.edu.hk

ABSTRACT

Sketch-based face recognition is an interesting task in vision and multimedia research, yet it is quite challenging due to the great difference between face photos and sketches. In this paper, we propose a novel approach for photo-sketch generation, aiming to automatically transform face photos into detail-preserving personal sketches. Unlike the traditional models synthesizing sketches based on a dictionary of exemplars, we develop a fully convolutional network to learn the end-to-end photo-sketch mapping. Our approach takes whole face photos as inputs and directly generates the corresponding sketch images with efficient inference and learning, in which the architecture is stacked by only convolutional kernels of very small sizes. To well capture the person identity during the photo-sketch transformation, we define our optimization objective in the form of joint generative-discriminative minimization. In particular, a discriminative regularization term is incorporated into the photo-sketch generation, enhancing the discriminability of the generated person sketches against other individuals. Extensive experiments on several standard benchmarks suggest that our approach outperforms other state-of-the-arts in both photo-sketch generation and face sketch verification.

Categories and Subject Descriptors

I.2.10 [**ARTIFICIAL INTELLIGENCE**]: Vision and Scene Understanding

Keywords

Sketch-photo generation; face verification; neural nets

*Corresponding author is Liang Lin. This work was supported by the Hi-Tech Research and Development Program of China (no.2013AA013801), Guangdong Natural Science Foundation (no.S2013050014548), and the Hong Kong Scholar program.

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

DOI: http://dx.doi.org/10.1145/2671188.2749321.

1. INTRODUCTION

Sketch is an important artistic drawing style and may be the simplest form since it is only composed of lines. An interesting application is searching image databases using free-hand sketch queries [18, 15].

However, drawing a vivid sketch portrait is time consuming even for a skilled artist. Automatic face sketch generation has been studied for a long time and it has many useful applications for digital entertainment [11].

Another important application based on face sketch is to assist law enforcement. Assumed that we need to automatically retrieval a photo from the database for a query image, which help the police to narrow down the suspect quickly. Unfortunately, the photo of the suspect maybe unavailable in most cases. To deal with such a problem, the best substitute available is an artist drawing based on the recollection of an eyewitness.

In this situation, we only focus on sketches without exaggeration, so that the sketch can realistically reflect the real person. Figure 1 shows two samples of photo-sketch pairs in CUHK Face Sketch Database [22].

(a) (b)

Figure 1: Samples of photo-sketch pairs: (a) photos; (b) sketches drawn by artist

Figure 1 indicates the great difference between photos and sketches. Thus, photo based face verification methods cannot be directly applied in this problem. The key to sketch based face verification is to reduce the modality difference between photos and sketches.

One intuitive idea is to recover the photo image from a sketch. However, it's an ill-pose problem because a sketch may lose many informations during the drawing procedure. An alternative way is to generate a pseudo-sketch from a photo, which has been discussed in [20, 12, 22].

The original generation scheme is trying to find a transformation from photos to sketches. Intuitively, it should be a complex nonlinear mapping. Thus, former works like [20, 12, 22, 24, 25, 21, 19] simplify the generation problem to *synthesis* form. The underlying assumption is that, if two photo images (or patches) are similar, their corresponding sketch images (or patches) should also be similar. Thus, if we find out a way to use other photo images (or patches) to synthesize a photo image (or patch), then we can use the corresponding sketch images (or patches) to synthesize its pseudo-sketch.

However, this simplification may has some limitation, especially the scalability. The time cost in synthesis based method grows linearly with the amounts of training data, because it need to use the samples in training set to synthesize a new one.

On the contrary, our method try to *directly* solve the generation problem with an end-to-end model called fully convolutional network (FCN), which can be regarded as a special kind of convolutional neural networks (CNNs). More precisely, FCN is stacked by *only* convolutional layers. It can solve complex nonlinear problem while producing pixel-wise outputs, which is very suitable for the photo-sketch generation problem. Please refer to Figure 2 to get an intuitive idea.

The main contributions of this work are three-folds. First, an end-to-end photo-sketch generation model is studied with a novel architecture of fully convolutional networks, which is original to the best of our knowledge. Second, we present a joint generative-discriminative formulation driving the network optimization, such that the generated face sketches can characterize the person detail as well as the discriminability against other individuals. Third, our system demonstrates superior performances compared with other state-of-the-art approaches on several benchmarks.

Rest of the paper is organized as follows: Section 2 discusses the relative work in photo-sketch synthesis and convolutional neural networks. In Section 3, we will interpret our model and its implementation. In Section 4, extensive experiments suggest that our approach outperforms other state-of-the-art methods on several benchmarks. In Section 5, we draw the conclusion of our work and list some ideas we may adopt in the future.

2. RELATIVE WORK

2.1 Photo-Sketch Synthesis and Recognition

In recent ten years, there has been several works on the topic of photo-sketch synthesis and recognition.

Tang and Wang [20] were the first to address the problem with a significant amount of database. They proposed a synthesis method based on eigen transformation (ET). It was under the assumption that the photo-sketch mapping can be approximated as linear, yet this assumption may be too strong especially when the hair region was included. Then, it used reconstruction error based distance for recognition.

Liu et al. [12] proposed a nonlinear face sketch synthesis method called locally linear embedding (LLE), which can be considered as an improved version of [20]. Instead of modelling the whole image, it applied ET on local patches with overlapping. It adopted a kernel-based nonlinear LDA discriminative classifier for sketch recognition.

Another local-based strategy proposed by Xiao et al. [24] was based on Embedded Hidden Markov model (E-HMM). They transformed the sketches to pseudo-photos and applied eigenface algorithm for recognition.

In [22], Wang and Tang proposed a multiscale Markov Random Fields (MRF) for face photo-sketch synthesis, which can be both applied to photo-to-sketch synthesis and sketch-to-photo synthesis. They evaluated a series of classifiers on the pseudo-images. Random Sample LDA (RS-LDA) performed best among them. Zhang et al. [26] then improved the MRF framework by adding shape priors and descriptors robust to lighting variations.

Most of methods above may lose some vital details, which influenced the visual quality and face recognition performance. Thus, Zhang et al. [25] added a refinement step on existing approaches. They applied a support vector regression (SVR) based model to synthesize the high-frequency information. Similarly, Gao et al. [5] proposed a new method called SNS-SRE with two steps, *i.e.* sparse neighbor selection (SNS) to get an initial estimation and sparse-representation-based enhancement (SRE) for further improvement.

2.2 Pixel-Wise Predections via CNNs

Convolutional neural network (CNN) was widely used in computer vision. Its typical structure contains a series of convolutional and pooling layers as feature extractors, and several full connected layers as classifiers to give prediction. It has achieved great success in large scale object classification, localization and detection [9, 7, 16, 6].

One important application of CNN was to produce dense or even pixel-wise predictions. Sermanet et al. [16] developed a CNN-based framework called OverFeat, which integrated recognition, localization and detection. The key component was a pooling layer with offsets, which can imitate sliding window technique and produce dense outputs in the final layer. A similar idea called spatial pyramid pooling (SPP) was proposed by He et al. [7], which was also modified the last pooling layer. Wang et al. [23] proposed a joint architecture for generic object extraction, and Luo et al. [14] applied a deep decompositional network for pedestrian parsing. Both of them shared an idea, which is adopting a full connected layer on the top of the network to produce a dense prediction. Moreover, patch-by-patch scanning technique on the original image has been studied by Ciresan et al. [1] in neuronal membrane segmentation and Farabet et al. [4] for scene labeling.

Recently, Dong et al. [3] applied CNN for image super-resolution and it can produce a pixel-wise output. They discussed its relationship to sparse-coding-based methods, and concluded that they can use three convolutional layers to simulate the representation-mapping-reconstruction procedure in sparse coding.

Our work was mostly inspired by [3]. We conducted our early experiments via their network and then further improved it. We called it fully convolutional network (FCN, a similar idea was also proposed in [13]), for it only contains convolutional layers and the corresponding activation function, but without any other layers like pooling, full connected, local response normalization (LRN), *etc.* We would further discuss it in Section 3.2.

3. PHOTO-SKETCH GENERATION

3.1 Formulation

In our model, we use two constraints for sketch generation. The first one is the generated sketches should be as close to the ones drawn by artists as possible. This encourages the deep network to learn the sketching skills from the artists. The second one is the generated sketches should be able to facilitate the law enforcement. That is, given a sketch drawn by the artist, it should be able to identify the subject in the photo database. These two constraints are introduced as generative loss and discriminative regularizer in our objective function, which is similar to [2].

Suppose there are N subjects in the training set with each subject containing one photo P_i and one sketch S_i. We use $f(\mathbf{W}, P_i)$ to denote the sketch generated by a fully convolutional network parameterized by $\mathbf{W}$. We define our loss function as follows where L_{gen} and $L_{discrim}$ represent the generative loss and the discriminative regularizer respectively. Here we use α to control the weight of the discriminative regularizer to the overall objective.

$$L(P, S, \mathbf{W}) = L_{gen}(P, S, \mathbf{W}) + \alpha L_{discrim}(P, S, \mathbf{W}) \quad (1)$$

For the generative loss, we use a straight function which is defined as the pixel-wise difference between the ground-truth sketch and the generated one.

$$L_{gen}(P, S, \mathbf{W}) = \frac{1}{N}\sum_{i=1}^{N}(S_i - f(\mathbf{W}, P_i))^2 \quad (2)$$

For the discriminative regularizer, we encourage the drawn sketch of one particular person should be different from the generated sketch of another person as defined by the following function.

$$L_{discrim}(P, S, \mathbf{W}) = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N}\log\left(1 + \mathrm{e}^{-\frac{\left(S_i - f(W, P_j)\right)^2}{\lambda}}\right) \quad (3)$$

λ is a parameter used to avoid numeric overflow.

Parameter Optimization Using this model, the learning process is to minimize the loss function $L(P, S, \mathbf{W})$. This is achieved by the standard network propagation algorithm in the batch training manner. More precisely, in each iteration, we randomly select a set of photo-sketch pairs and construct the generative loss items and discriminative regularizer items respectively. In order to derive the gradient with respect to the network parameter $\mathbf{W}$, the key step is to calculate the partial derivative of the loss function with respect to the output (generated sketch) of each photo. As one photo may be involved into several cost items, we need to go through each items to accumulate the derivative with respect to the output. Algorithm 1 gives the details of the learning process.

Algorithm 1 Parameter Optimization

Require:
Training photos $\{P_i\}$ and sketches $\{S_i\}$;
Ensure:
Network Parameters $\mathbf{W}$
while $t < T$ **do**
  $t \leftarrow t + 1$;
  Randomly select a subset of photos and sketches $\{P'_i\}, \{S'_j\}$ from the training set;
  for all P'_i **do**
    Do forward propagation to get $f(\mathbf{W}, P'_i)$
  end for
  $\Delta\mathbf{W} = 0$
  for all P'_i **do**
    Calculate the partial derivative with respect to the output: $\frac{\partial L}{\partial f(\mathbf{W}, P'_i)}$
    Run backward propagation to obtain the gradient with respect to the network parameter: $\Delta\mathbf{W}_i$
    Accumulate the gradient: $\mathbf{W} += \Delta\mathbf{W}_i$
  end for
  $\mathbf{W}^t = \mathbf{W}^{t-1} - \lambda_t \Delta\mathbf{W}$
end while

3.2 Fully Convolutional Representation

In this subsection, we will discuss the property of convolutional layers and how they can be cascaded as a pixel-wise fully convolutional representation of an input image.

Composition of Convolutions A typical convolutional layer has K kernels with activation function f can be formulated as

$$y_{ij}^k = f((\mathbf{W}_k * x)_{ij} + b_k) \quad (4)$$

x denotes the input feature maps. $k \in \{1, 2, ..., K\}$ denotes the index of a filter, and $\mathbf{W}_k$ and b_k are the k-th filter weights and bias. y_{ij}^k denotes the element at the coordinate (i, j) on the k-th output feature map.

Equation (4) indicates that the convolutional operation preserves the spatial relationship. What's more, composition of convolutions will not change this property, *i.e.* we can use a stack of convolutional layers to represent a complex nonliear mapping, which can be adopted on an input of arbitrary size, and then produce a corresponding spatial output.

Relationship to the Patch-Wise Representation In Equation (4), a pixel on the output feature map only will be influenced by a *patch* on the input feature map, *i.e.* its receptive field. This property is also maintained under composition of convolutions. Thus, fully convolutional representation on the whole image can be regarded as a *batch* of patch-wise representation on patches of the image. Nevertheless, fully convolutional representation is much more efficient, for the patches overlap significantly (we set stride to 1 in all layers).

Border Effect Trade-Off Convolutional operations will shrink the size of the feature map. For example, if we adopt a (3×3) convolution operator on a (3×3) input, the output size will be shrank to (1×1).

As more convolutional layers stacked up, more shrinks will be accumulated in the outputs. A possible solution is to add proper padding before the convolution operation. But it will bring on *border effects*, which has been claimed in [3].

Thus, in the trade-off, we don't use padding in the convolutional layers during both the training and testing time. Each (155×200) photo image will be shrank to (143×188) in the representation (See Figure 2).

3.3 Implementation

Pre-processing As it is described in [22], all the photos and sketches are translated, rotated, and scaled such that the two eye centers of all the face images are at fixed position in the pre-processing step. This simple geometric nor-

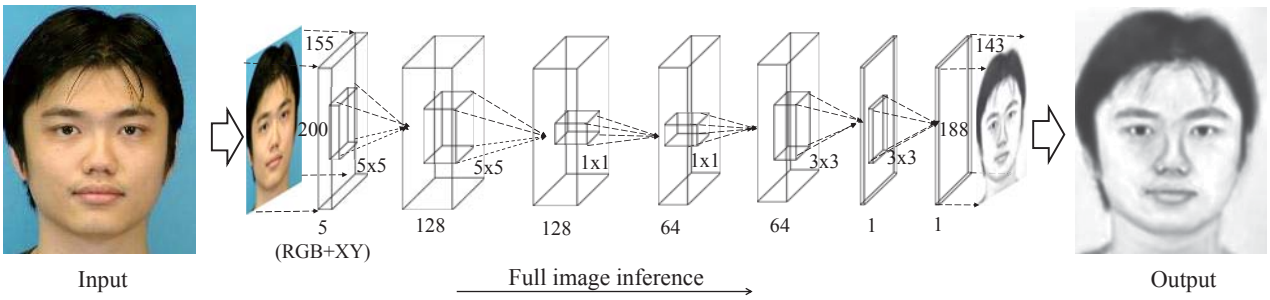

Figure 2: The overview of our model. It takes a full-size photo image as input and directly generates a full-size pseudo-sketch as output. The middle part is the architecture of our fully convolutional network. It contains six convolutional layers, with rectified linear units as activation functions (omitted in the figure).

malization step makes the same face components in different images in roughly alignment.

Another pre-processing step is inspired by the results[1] of [22]. The transformation mentioned above produces a (200×250) image, but it may have some black regions on the border areas. We crop the (155×200) center part of the image in order to exclude this negative influence.

Since we choose to avoid border effects, the sketch images need to be cropped to (143×188) to fit in the network output dimension.

Spatial Patch-wise Learning with Overlapping From the previous works on photo-sketch synthesis[12, 22], patch-wise learning with overlapping is very important to handle the non-linearity between photos and sketches. Intuitively, patches in different positions are diverse from others (*e.g.* eye patches, nose patches and mouth patches). Therefore, learning different patch representations in different spatial positions respectively is a very straightforward idea.

We handle it by adding additional XY channels in the input, *i.e.* the input image data contain five channels, three RGB channels of the photo image, and two channels of the corresponding coordinate (i, j). This tiny modification significantly improves the result, which will be discussed in Section 4.3.

Network Architecture We apply the network proposed in [3] in our early experiments, then we modify its architecture for further enhancement. We borrow the idea in [17], which has a great success in ILSVRC-2014. Their main contribution is to deeper the CNN under computational constraints via very small (3×3) convolutional filters in all layers.

Similarly, we modify the 3 layers network in [3] into a 6 layers network, which is shown in Figure 2. For the (9×9), (1×1), (5×5) convolutional layer in [3], we replace them with two (5×5), (1×1) and (3×3) convolutional layers respectively. Moveover, we double the filter amounts in every layer to improve the network's capacity.

We call it *medium network* due to its size. We have two similar architectures called small network and large network, which will be further discussed in Section 4.3.

Training Details Our results are based on our medium network (See Figure 2). As mentioned above, we first pre-process the photo images into (155×200) and the sketch images into (143×188). The optimization objective has been discussed in Section 3.1. We set λ as 10^9 and set α as 10^4.

We use Caffe [8] for implementation. The filter weights are initialized by drawing randomly from a Gaussian distribution with zero mean and standard deviation 0.01, and the bias are initialized by zero. We set the learning rate as 10^{-11}, then it takes several hours to converge on a NVIDIA Tesla K40 GPU.

4. EXPERIMENTS

We evaluate our model on CHUK student dataset [22]. It includes 188 faces. 88 faces are selected for training and remaining 100 faces are selected for testing. For each face, there is a sketch drawn by the artist and a photo taken in a frontal pose, under normal lighting condition, and with a neural expression.

We mainly do three aspects of experiments, which will be shown in the next three subsections. Section 4.1 will discuss our photo-sketch generation results and comparison with the synthesis based method in [22]. Section 4.2 will show that our generated pseudo-sketches significantly reduce the modality between photos and sketches. Thus they can be adopted to a sketch-based face verification system. Section 4.3 will involve empirical study about our model. Several factors such as network depth and filter numbers will be discussed on the evaluation protocol both qualitatively and quantitatively.

4.1 Face-Sketch Generation

In Figure 3, we show some examples of our generated results and compare them to [22]. For fair comparison, we list all 14 pseudo-sketches which can be found on their website.

Figure 3 indicates that our results contain more vital details. For example, the man in column (e) of the first row has two little locks of hair on his forehead. But this detail is missing in the pseudo-sketch in [22] (in column (g)). On the contrary, our generated result (in column(f)) retains this detail information which may be very important to distinguish this man from others. Another example is the woman in column (a) of the second row. She has a distinctive feature comparing to other persons in the database, for she has two big eyes. Our method can also capture this detail in the pseudo-sketch (in column(b)).

It suggests that synthesis based methods may fail in some cases. The main reason is that they are under an assumption that the synthesized sketch image (or patch) can be recon-

[1]http://www.ee.cuhk.edu.hk/~xgwang/sketch_multiscale.html

Figure 3: Compared to synthesis based method [22], our generative approach could retain more details in the original photos, *e.g.* the two locks of hair in forehead of the man in column (e) of row 1; the two big eyes of the woman in column (a) of row 2. (a)&(e) photos; (b)&(f) our generative pseudo-sketches; (c)&(g) synthesized pseudo-sketches in [22]; (d)&(h) sketches drawn by the artist.

structed by the sketch images (or patches) in the training set. However, this assumption may be too strong, for some persons have their facial distinctions in fact.

Our generative method overcomes this difficulty. We try to *directly* learn the transformation from photo space to sketch space. Even though it's a complex nonlinear mapping, FCN has the capacity to handle this challenge.

Compared to traditional synthesized method, our novel generative approach has two folds of advantages. Firstly, it could retain more detail information from the photos. Secondly, our inference time is independent on the amount of training data. The time cost in synthesis based method grows linearly with the data amounts, while the runtime of our approach is only affected by the size of the input photo.

4.2 Sketch-Based Face Verification

In this subsection, we will show that our generated pseudo-sketch significantly reduce the modality between photos and sketches. We follow the same testing procedure in [20, 12, 22], which can be concluded in two steps: (a) convert the photos in testing set into corresponding pseudo-sketches; (b) define a feature or transformation to measure the distance between the query sketch and the pseudo-sketches.

In our implementation, we use our model to generate the pseudo-sketches in procedure (a), and use our generative loss (see Equation (2)) as the distance measurement. Since the sizes of our pseudo-sketches are (188×143), we also crop the query sketch into (188×143).

Following the same protocol described in [20], we compare our approach with previous methods on cumulative match score (CMS) in Table 1. CMS measures the percentage of "the correct answer is in the top n matches", where n is called the rank.

As shown in Table 1, we form a baseline experiment, which is to convert the photo into gray scale as somehow "pseudo-sketch" to clarify the modality difference between photo space and sketch space. In this baseline, the accuracy for the first rank only equals to 41%, which is far away from satisfying. In early method like ET described in [20], has got 71% for the top one match. Latter trials in [22], [26], [25] have greatly improved the verification accuracy on top one candidate to 96%, 99% and even 100%.

From the last row in Table 1, our approach also gets 100% correct answer on its first guess. It's worth to notice that our approach is quite different from the former methods, for it applies generation instead of synthesis.

	Rank 1	Rank 3	Rank 5	Rank 10
Baseline	41	56	59	70
ET [20]	71	81	88	96
MRF [22]	96	-	-	100
MRF+ [26]	99	-	-	100
SVR [25]	100	-	-	-
Ours	**100**	**100**	**100**	**100**

Table 1: Cumulative Match Scores (CMS) comparison on full training set (88 photo-sketch pairs). Our method achieves 100% accuracy on the first guess.

Moreover, our generative method is very robust and has excellent generalization ability. In Figure 4, it suggests that our method can using a portion of training data to get a 100% accuracy for the first rank. For example, we only need 5 training samples to get to 95% accuracy, and 27 training samples is enough to get a 100% score for top one candidate.

We also design a verification on our optimization objective on Figure 4. It suggests that the model trained with the discriminative regularizer consistently outperforms the model trained without discriminative regularizer.

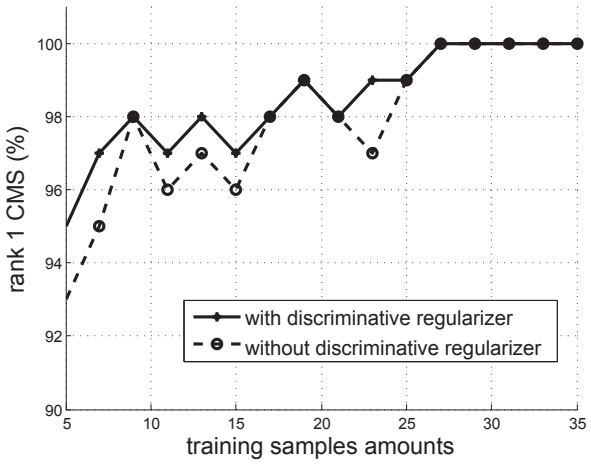

Figure 4: The model trained with discriminative regularizer consistently outperforms the model trained without discriminative regularizer. Both model gets 100% accuracy on rank 1 candidate while involves less than half training samples. *i.e.* 27 samples in 88 photo-sketch pairs.

4.3 Empirical Study

In this subsection, further analyses will be conducted on the factors which affect our photo-sketch generation performance.

Different Network Architectures Our first concern is about the depth of the network and the numbers of filters. We use the network proposed in [3] (we call it SR network) in our early experiments. However, the result is not quite satisfying, for this network seems too small to learn the complex nonlinear mapping from photos to sketches. Thus, we improve the network by: (a) adding more layers to the network; and (b) adding more numbers of filters to each layer.

We conduct our experiments on three different network architectures. Due to their scales, we call them small, medium and large network respectively.

The medium network architecture can be found in Figure 2. The difference between it and SR network please refer to Section 3.3. Moreover, the only difference between our small, medium and large network is their filter numbers. The medium network has two times of filters compared to the small one. And the large network doubles the filter numbers of the medium network.

For comparison, we define a measurement call $MPRL$, which is short for *multiscale pixel-wise reconstruction loss*. It evaluates the pixel-wise accuracy on different scales. To explain $MPRL$, we firstly introduce *pixel-wise reconstruction loss* (PRL) on one photo-sketch pair.

For a photo-sketch pair, we denote the sketch as the ground truth (GT) and generated pseudo-sketch as the prediction

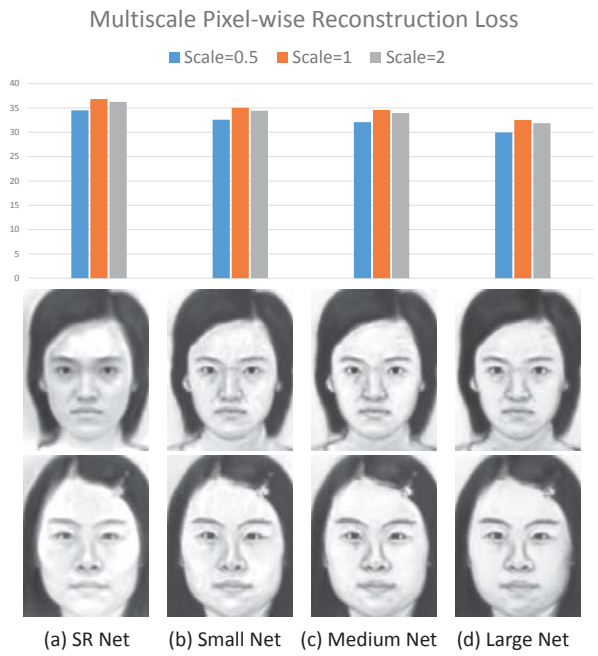

Figure 5: Comparison on the $MPRL$ measurement and generative pseudo-sketches via different network architectures. The model achieves better performance while to deeper the network or to add more filters. (a-d) pseudo-sketches generated via network in [3], our small network, our medium network and our large network.

(P). Both of their sizes are $(W \times H)$. Thus, PRL can be formulated as

$$PRL = \frac{1}{W \times H} \sqrt{\sum_{x=1}^{W} \sum_{y=1}^{H} (GT(x,y) - P(x,y))^2} \quad (5)$$

$MPRL$ is the multiscale version of PRL. In practice, for each photo-sketch pair, we rescale both the GT and the P to three scales {0.5, 1, 2} and evaluate their PRL to form the $MPRL$. Then, for the whole test set, we evaluate all the pairs and average their $MPRL$ as the final measurement.

Figure 5 summarizes the results both in qualitative and quantitative. The $MPRL$ of SR network is {34.5, 36.8, 36.2}. Our small network reduces it to {32.6, 35.0, 34.4}. Then, our medium and large network get better performance as {32.1, 34.6, 34.0} and {30.0, 32.5, 31.9}.

The pseudo-sketch examples generated by each network are on the bottom part of Figure 5. It suggests that the results of SR network can capture the overall structures, but they are far away from satisfying in details. Then, Our small network's generation is quite better and has clearer contours. Moreover, the pseudo-sketches generated by medium network are more vivid than the small ones. The results of large network are as good as medium ones, but the large network is much more time consuming.

Table 2 shows that our small network is almost as fast as the SR network. The medium network's cost is about 2 times of the small one, and the large network's runtime is about three times of the medium one.

To balance the effectiveness and efficiency, we prefer our medium network as the final choice.

	SR Net	Small Net	Medium Net	Large Net
Time	8.5ms	8.6ms	17.1ms	50.8ms

Table 2: Runtime of single (155×200) image on NVIDIA Tesla K40 GPU

Difference between trained with and without XY channels Now we consider another factor that influences the model's performance. As mentioned above, we add XY channels to the RGB color channels. We assume these additional spatial messages could help the network to distinguish different spatial patches and learning different representation on them.

Similarly, we use $MPRL$ for quantitative measurement and use pseudo-sketch comparison for qualitative analysis. In Figure 6, the left part is for results generated by our medium network but trained without XY channels, and the right part is for results generated by our medium network trained normally. Figure 6 suggests that latter one outperforms former one both in numbers and perception.

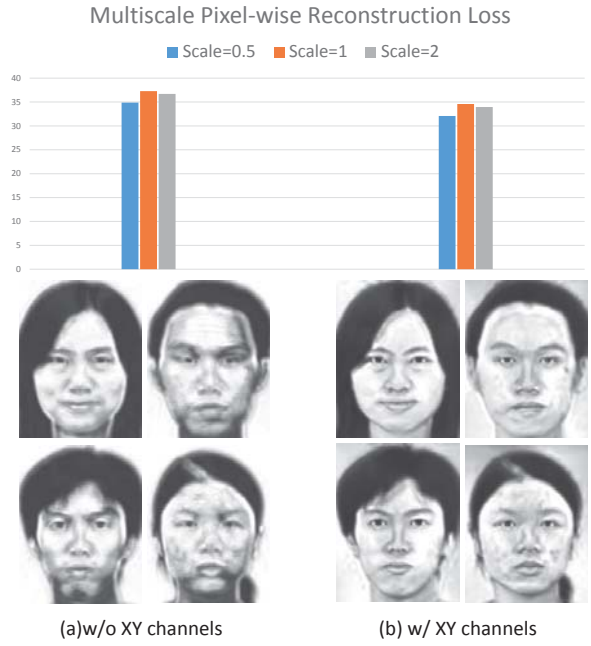

Figure 6: Comparison on the $MPRL$ measurement and generative pseudo-sketches via network trained without and with XY channels. Adding XY channels can make the model more robust. (a) pseudo-sketches generated via network trained without XY channels; (b) pseudo-sketches generated via network trained with XY channels.

5. CONCLUSION AND FUTURE WORK

In this paper, we propose an end-to-end fully convolutional network in order to directly model the complex nonlinear mapping between face photos and sketches. From the experiments, we find out that fully convolutional network

is a powerful tool which can handle this difficult problem while providing a pixel-wise prediction both effectively and efficiently.

However, this solution still has some limitations. Firstly, the synthesized pseudo-sketches in [22] have shaper edges and clearer contours than our generated ones. Secondly, the database involved in our experiments only contains Asiatic face images, which may limit the generalization ability of our model to other racial groups.

In future work, we will further improve our loss function and try various databases in our experiments, and we may explore about the relation between our work and those involved with non-photorealistic rendering [10].

Acknowledgements We gratefully acknowledge NVIDIA for GPU donations.

References

[1] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in neural information processing systems*, pages 2843–2851, 2012.

[2] S. Ding, L. Lin, G. Wang, and H. Chao. Deep Feature Learning with Relative Distance Comparison for Person Re-identification. *Pattern Recognition*, 2015. doi: 10.1016/j.patcog.2015.04.005.

[3] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In *Computer Vision–ECCV 2014*, pages 184–199. Springer, 2014.

[4] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1915–1929, 2013.

[5] X. Gao, N. Wang, D. Tao, and X. Li. Face sketch–photo synthesis and retrieval using sparse representation. *Circuits and Systems for Video Technology, IEEE Transactions on*, 22(8):1213–1226, 2012.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv preprint arXiv:1311.2524*, 2013.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *Computer Vision–ECCV 2014*, pages 346–361. Springer, 2014.

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

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[10] L. Lin, K. Zeng, Y. Wang, Y.-Q. Xu, and S.-C. Zhu. Video stylization: Painterly rendering and optimization with content extraction. *Circuits and Systems for Video Technology, IEEE Transactions on*, 23(4):577–590, 2013.

[11] J. Liu, Y. Chen, and W. Gao. Mapping learning in eigenspace for harmonious caricature generation. In *Proceedings of the 14th annual ACM international conference on Multimedia*, pages 683–686. ACM, 2006.

[12] Q. Liu, X. Tang, H. Jin, H. Lu, and S. Ma. A nonlinear approach for face sketch synthesis and recognition. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 1005–1010. IEEE, 2005.

[13] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. *CVPR (to appear)*, Nov. 2015.

[14] P. Luo, X. Wang, and X. Tang. Pedestrian parsing via deep decompositional network. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 2648–2655. IEEE, 2013.

[15] S. Ren, C. Jin, C. Sun, and Y. Zhang. Sketch-based image retrieval via adaptive weighting. In *Proceedings of International Conference on Multimedia Retrieval*, page 427. ACM, 2014.

[16] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations (ICLR 2014)*. CBLS, April 2014.

[17] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[18] J. R. Smith and S.-F. Chang. Visualseek: a fully automated content-based image query system. In *Proceedings of the fourth ACM international conference on Multimedia*, pages 87–98. ACM, 1997.

[19] Y. Song, L. Bao, Q. Yang, and M.-H. Yang. Real-time exemplar-based face sketch synthesis. In *Proceedings of European Conference on Computer Vision*, pages 800–813, 2014.

[20] X. Tang and X. Wang. Face sketch recognition. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(1):50–57, 2004.

[21] S. Wang, L. Zhang, L. Y., and Q. Pan. Semi-coupled dictionary learning with applications in image super-resolution and photo-sketch synthesis. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012.

[22] X. Wang and X. Tang. Face photo-sketch synthesis and recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(11):1955–1967, 2009.

[23] X. Wang, L. Zhang, L. Lin, Z. Liang, and W. Zuo. Deep joint task learning for generic object extraction. In *Advances in Neural Information Processing Systems*, pages 523–531, 2014.

[24] B. Xiao, X. Gao, D. Tao, and X. Li. A new approach for face recognition by sketches in photos. *Signal Processing*, 89(8):1576–1588, 2009.

[25] J. Zhang, N. Wang, X. Gao, D. Tao, and X. Li. Face sketch-photo synthesis based on support vector regression. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 1125–1128, Sept 2011. doi: 10.1109/ICIP.2011.6115625.

[26] W. Zhang, X. Wang, and X. Tang. Lighting and pose robust face sketch synthesis. In *Computer Vision–ECCV 2010*, pages 420–433. Springer, 2010.