Proceedings of the IEEE
International Conference on Information and Automation
Ningbo, China, August 2016

Deep Face Attributes Recognition Using Spatial Transformer Network*

Lianzhi Tan[1,2,*], Zhifeng Li[1,3,*], Qiao Yu[1,4]
[1]*Guangdong Provincial Key Laboratory of Computer Vision and Virtual Reality Technology, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China*
[2] *Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences*
[3] *Key Laboratory of Human-Machine Intelligence-Synergy Systems, Chinese Academy of Sciences*
[4]*Department of Information Engineering, The Chinese University of Hong Kong*
{lz.tan, zhifeng.li, yu.qiao}@siat.ac.cn

***Abstract* –Face alignment is very crucial to the task of face attributes recognition. The performance of face attributes recognition would notably degrade if the fiducial points of the original face images are not precisely detected due to large lighting, pose and occlusion variations. In order to alleviate this problem, we propose a spatial transform based deep CNNs to improve the performance of face attributes recognition. In this approach, we first learn appropriate transformation parameters by a carefully designed spatial transformer network called LoNet to align the original face images, and then recognize the face attributes based on the aligned face images using a deep network called ClNet. To the best of our knowledge, this is the first attempt to use spatial transformer network in face attributes recognition task. Extensive experiments on two large and challenging databases (CelebA and LFWA) clearly demonstrate the effectiveness of the proposed approach over the current state-of-the-art.**

Index Terms – Face Attribute Recognition, Spatial Transformer Network, Face Alignment.

I. INTRODUCTION

Face attributes recognition is an important research topic and achieves increasing attentions due to its wide applications in social communications [1], face recognition [2-5] and image understanding [6, 7]. Despite advances in face attributes recognition[8], it remains a challenging problem due to significant facial variations caused by illumination, pose, and occlusion in the wild.

Recently, deep CNNs greatly improve the recognition performance on face attributes recognition. Zhang et al. [9]trained a pose aligned network in which poselet patches are fed to get an overall pose-normalized representation. Luo et al. [10] proposed a sum-product architecture using detectors to locate face region for further classification. Liu et al. [11] proposed two cascaded CNNs, in which the first locates facial region and the second extracts face representation from the located region. Zhong et al.[12, 13] used an off-the-shelf architecture and explored efficient feature representations of fully-connected layers. However, local method in [9] may fail when detecting unconstrained face images, and face alignment tools used in [11-13] may produce imprecise points. Recent study [14] has proven that face detection with a cascaded CNNs obtains much better performance in detecting the uncontrolled face images in the wild.

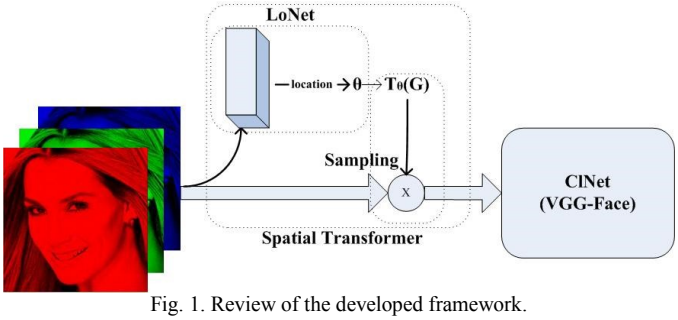

Fig. 1. Review of the developed framework.

In this paper, we build our work on [14] to detect and locate face region. Instead of locating the fiducial points in the face region, we propose an end-to-end framework, in which spatial transformer network is applied (prior to the classification network) to learn face alignment parameters. The spatial transformer network [15] learns spatial transformation for an image or a feature map. The transformation includes scaling, cropping, rotation and non-rigid deformation. By adding a spatial transformer layer between input layer and classification layers, every input sample is transformed before classified. For example, plane rotated images are transformed to frontal face images, and face regions at the edge of images are transformed to the center area. Traditional methods of face alignment are strong alignment, in which the coordinates of facial fiducial points in the aligned images are fixed. The spatial transformer layer belongs to weak alignment since it adjusts parameters according to the classification performance. Therefore, spatial transformer layer is used to avoid strong alignment with imprecisely detected facial fiducial points. An off-the-shelf network, VGG-Face [16], is used after the spatial transformer layer and improves the performance of facial attributes recognition.

This work focuses on facial attribute recognition in the wild. We evaluate the performance of the proposed approach

* This work was supported by grants from Natural Science Foundation of Guangdong Province (2014A030313688), Shenzhen Research Program (JSGG20150925164740726, JCYJ20150925163005055, and CXZZ2015093010411552 9), Guangdong Research Program (2014B050505017 and 2015B010129013), National Natural Science Foundation of China (61103164). Corresponding author: Lianzhi Tan, Zhifeng Li.

978-1-5090-4102-2/16/$31.00 ©2016 IEEE

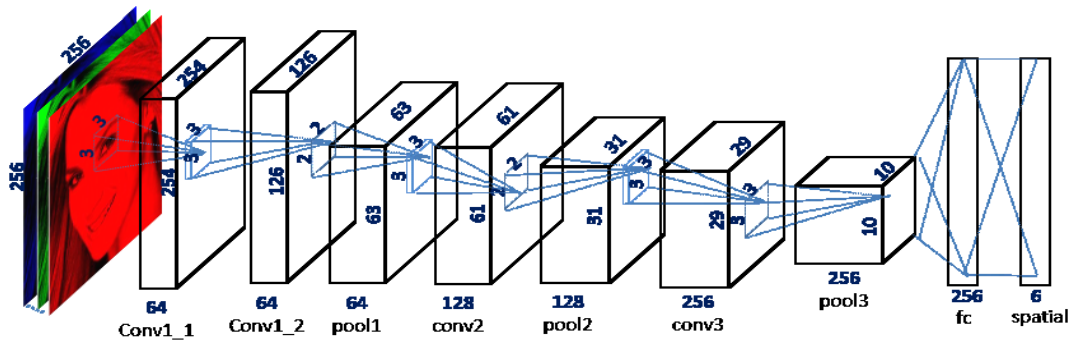

Fig. 2. The proposed framework for locating face region and learning parameters of face image transformation.

on two large and challenging databases (CelebA [17] and LFWA [18]). The major contributions of this work include:

1) To the best of our knowledge, this paper is the first attempt to explore the use of the spatial transformer network in face attributes prediction. It simplifies the pre-processing procedure.
2) We propose a multi-task network to learn face alignment parameters and classify face attributes, which are beneficial to performance improvement.
3) We use an off-the-shelf network for classification and achieve the state-of-the-art performance of facial attributes recognition on two large and challenging databases.

The rest of this paper is organized as follows. In Section II, we introduce the proposed framework, including spatial transformer network, location network and classifi-cation network. The experimental results are presented in Section III. Finally, we conclude this paper in Section IV.

II. Proposed Approach

Spatial transformation is a useful technique in image processing [19, 20]. It has been recently used in face related works [14, 21]. In these works, the spatial transformation technique is commonly used in the preprocessing stage and thus relies heavily on the detected fiducial points. Different from [14, 21], we use the spatial transformer network to learn the parameters of the transformation matrix for face alignment without relying on the detected fiducial points. The proposed approach is elaborated as follows.

A. Spatial Transformer Network

Spatial transformer network is a network which learns the transform parameters from an input image or feature map to a target image or a feature map. For an input image or a feature map, each output pixel is calculated by multiplication of transformation matrix and input. Considering that transformations of face are general, such as scaling, rotation, cropping and translation, we assume the transformation as a kind of 2D affine transformation. The pointwise transformation is as the following:

$$\begin{pmatrix} x_j^i \\ y_j^i \end{pmatrix} = \mathrm{T}_\theta(G_j) = R_\theta \begin{pmatrix} x_j^o \\ y_j^o \\ 1 \end{pmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{pmatrix} x_j^o \\ y_j^o \\ 1 \end{pmatrix} \quad (1)$$

where R_θ is the transformation matrix. $G_j = (x_j^o \quad y_j^o)$ represent the target coordinates of the regular grid in the output feature map. (x_j^i, y_j^i) are the source coordinates in the input image or feature map. Coordinates are normalized to [-1 1] for calculation. The process of transformation can be regarded as sampling from the input feature map U to produce the sampled output feature map V. The process can be written as:

$$V_j^c = \sum_n^H \sum_m^W U_{nm}^c k(x_j^i - m; \Phi_x) k(y_j^i - n; \Phi_y)$$
$$\forall j \in [1 \dots H'W'] \forall c \in [1 \dots C] \quad (2)$$

where Φ_x, Φ_y are the parameters of the sampling kernel $k(\cdot)$ which defines image transformation. V_j^c is the output value for pixel j at location (x_j^o, y_j^o) in channel c. For example, kernel of bilinear sampling can be written as:

$$k(x_j^i - m; \Phi_x) = \max(0, 1 - |x_j^i - m|) \quad (3)$$

In order to allow backpropagation, the gradient with respect to U and G can be written as the partial derivatives of V to U and G (including x and y),

$$\frac{\partial V_j^c}{\partial U_{nm}^c} = \sum_n^H \sum_m^W \max(0, 1 - |x_j^i - m|) \max(0, 1 - |y_j^i - n|) \quad (4)$$

$$\frac{\partial V_j^c}{\partial y_j^i} = \sum_n^H \sum_m^W U_{nm}^c \max(0, 1 - |x_j^i - m|) \begin{cases} 0 & if |n - y_j^i| \geq 1 \\ 1 & if \ n \geq y_j^i \\ -1 & if \ n \leq y_j^i \end{cases} \quad (5)$$

and partial derivative of x can be calculated by similar formulas.

This work intends to learn the transformation parameters needed in face alignment. Figure 1 showing that the spatial transformer network is a combination of the location network, and sampling kernel.

B. Location Network

The location network is used to locate the face region and to estimate the transformation parameters. The sampling kernel is used to calculate the transformation operation of transformation matrix and the input image or feature map. The framework of LoNet is shown in Figure 2. The spatial transformer layer is used to predict parameters of the transformation matrix. In LoNet, we set the filter size to 3x3 and there are four convolutional layers and two fully-connected layers. Stride of "conv1_2", "pool1", "pool2" are set to 2, and stride is set to 3 in "pool3". Each convolutional layer and the first fully-connected layer follows a ReLU layer.

Dropout ratio in the "fc" layer is set to 0.8 to avoid over-fitting.

C. Classification Network

After the spatial transformer network we use a classification network called VGG-Face [16] for attributes classification. The VGG-Face network is trained using millions of face images which allows the network to learn features more related to face related works. The VGG-Face network comprises eleven blocks, and the first eight blocks are convolutional layers and the last three blocks are fully-connected layers. Different from the input size of VGG-Face, our input size of images is 256x256. We add a pooling layer before the first fully-connected layer. Pooling kernel size is set to 2 and stride is set to 1. Our attribute classifier uses the empirical *cross-entropy loss*. Loss function is:

$$L = -\frac{1}{N}\sum_{n=1}^{N} y_n \log\left(\frac{e^{x_{nk}}}{\sum_{k'} e^{x_{nk'}}}\right) \quad k' \in \{1,\ldots,K\} \tag{6}$$

where K denotes the number of classes, $\frac{e^{x_{nk}}}{\sum_{k'} e^{x_{nk'}}}$ is the output of *softmax* function which represents probabilities of the sample classified to each class.

III. Experiment and Result

Caffe [22], a public-domain deep learning framework, is used in this experiment. Convolutional layers, fully-connected layers, pooling layers, activation layers and loss layers are all supported in Caffe.

A. Datasets

The CelebA Dataset [17]: The CelebA dataset has over 200 thousand images and over eight millions face attribute labels. Each image in CelebA is annotated with forty binary attributes, labeled with -1 or 1. Five key points are given, but not comparable with detected points by [14]. We use 162770 images for training and 19962 images for testing as suggested in [11].

The LFWA Dataset [18]: The LFWA dataset has 13233 images of 5749 identities. We use the same train and test data as those in [11]. Totally there are 2749 identities including 6263 images for training and 3000 identities including 6880 images for testing. The LFWA is annotated with five key points and forty attributes labeled with 0 or 1. We use method in [14] for detection and the images for training SVM in [11] are not used.

B. Data Processing

We first detect face regions using a cascaded CNNs. A cascaded CNNs [14] is used to detect faces in images, and face regions are located and cropped to 256 x 256 according to the mean size of bounding boxes on the Celeb database.

C. Training

To initialize our network with better learned parameters, we use parameters in transformation matrix from an un-aligned input to the aligned input as supervisory signals to train the LoNet. We train the network to learn from the transformation parameters which are calculated by closely precise face fiducial points. Figure 3b, 3a show the face images with and without

Fig. 3. The left image(3a) is a detected face image in CelebA. The center image(3b) is aligned using method in [14], the right image(3c) is transformed given by the affine transformation from the 3a to 3b.

alignment. We only use the affine transformation to align face images. Figure 3c shows the face image after transformation, given the transformation parameters from a not aligned image to an aligned image. We train the LoNet with initial parameters [1 0 0 0 1 0], which corresponding to the unit matrix in the transformation matrix. The input image is not transformed initially. The learning rate decreases from 0.01, and we stop training after 100 iterations. We also train the ClNet with and without alignment on CelebA and LFWA, results are shown in III.D.

After training the LoNet and ClNet separately, we freeze the convolutional layers. Fine-tuning learning rate is initialized to 0.001 and decreases after 4000， 8000， 10000 iterations. The max iterations number is set to 12000. We use both loss layers used in LoNet and ClNet. Therefore, the proposed network not only learns face alignment parameters but also classifies face attributes. Loss weight is set to 0.9 for ClNet's loss and 0.1 for LoNet's loss.

D. Comparative Experiments

We compare our method against several state-of-the-art methods, including (1) FaceTracer [18], (2) PANDA-l [9], (3) LNet+ANet [11], and (4) Off-the-shelf [23, 24]. The comparative results on both the CelebA and LFWA databases are reported in Table 1. It is encouraging to see that our approach consistently achieves the state-of-the-art performance on both the databases, outperforming the existing methods in recognizing most of the face attributes. We also report the average recognition performance of the forty face attributes on CelebA and LFWA in Table 2. Our approach improves the average accuracy of face attributes recognition by approximately 4% on CelebA and approximately 2% on LFWA. Next, we compare our result using the spatial transformer network against the result without training the ClNet separately. First, we analyze the result of training the ClNet separately. The results show that performances of some attributes (such as, smiling) are higher when the input image is aligned while performances of others

Table 2 Average performance of different methods on CelebA and LFWA

Methods	CelebA	LFWA
Our method	**91.1**	**86.0**
FaceTracer[18]	80.9	73.7
PANDA-l[9]	85.1	80.6
LNet+ANet[11]	87.0	83.6
Off-the-shelf[12]	86.3	84.5

		5oClockShadow	Arched Eyebrows	Attractive	BagsUnderEyes	Bald	Bangs	Big Lips	Big Nose	Black Hair	Blond Hair	Blurry	Brown Hair	BushyEye	Chubby	DoubleChin	Eyeglasses	Goatee	Gray Hair	Makeup	Cheekbones
Celeb	FaceTracer[18]	85	76	78	76	89	88	64	74	70	80	81	60	80	86	88	98	93	90	85	84
	PANDA-l[9]	88	78	81	79	96	92	67	75	85	93	86	77	86	86	88	98	93	94	90	86
	LNet+ANet[11]	91	79	81	79	98	95	68	78	88	95	84	80	90	91	92	99	95	97	90	87
	Off-the-shelf[12]	89	83	82	79	96	94	70	79	87	93	87	79	87	88	89	99	94	95	91	87
	Our method	**95**	**84**	**83**	**85**	**99**	**96**	**71**	**85**	**89**	**96**	**96**	**88**	**93**	**96**	**96**	**99**	**98**	**98**	**91**	**88**
LFW A	FaceTracer[18]	70	67	71	65	77	72	68	73	76	88	73	62	67	67	70	90	69	78	88	77
	PANDA-l[9]	84	79	81	80	84	84	73	79	87	94	74	74	79	69	75	89	75	81	93	86
	LNet+ANet[11]	84	82	83	83	88	88	75	81	90	97	74	77	82	73	78	95	78	84	95	88
	Off-the-shelf[12]	77	83	79	83	91	91	78	83	91	97	88	76	83	75	80	91	83	87	95	88
	Our method	78	**83**	**80**	**84**	**92**	**91**	**79**	**84**	**91**	**97**	87	**79**	**86**	**76**	**83**	**93**	**84**	**88**	**96**	**89**

		Male	Mouth open	Mustache	Narrow Eyes	No Beard	Oval Face	Pale Skin	Pointy Nose	Receding	Rosy Cheeks	Sideburns	Smiling	Straight Hair	wavy hair	Wearing Earrings	Wearing Hat	Wearing Lipstick	wear necklace	wear necktie	Young
Celeb	FaceTracer[18]	91	87	91	82	90	64	83	68	76	84	94	89	63	73	73	89	89	68	86	80
	PANDA-l[9]	97	93	93	84	93	65	91	71	85	87	93	92	69	77	78	96	93	67	91	84
	LNet+ANet[11]	98	92	95	81	95	66	91	72	89	90	96	92	73	80	82	99	93	71	93	87
	Off-the-shelf[12]	99	92	93	78	94	67	85	73	87	88	95	92	73	79	82	96	93	73	93	86
	Our method	**99**	**94**	**97**	**88**	**96**	**75**	**97**	**78**	**93**	**95**	**98**	**93**	**84**	**84**	**90**	**99**	**94**	**86**	**96**	**89**
LFWA	FaceTracer[18]	84	77	83	73	69	66	70	74	63	70	71	78	67	62	88	75	87	81	71	80
	PANDA-l[9]	92	78	87	73	75	72	84	76	84	73	76	89	73	75	92	82	93	67	91	84
	LNet+ANet[11]	94	82	92	81	79	74	84	80	85	78	77	91	76	76	94	88	95	88	79	86
	Off-the-shelf[12]	94	81	94	81	80	75	73	83	86	82	82	90	77	77	95	90	95	90	81	86
	Our method	**94**	**83**	**94**	**84**	**82**	**78**	**84**	**85**	**86**	78	**82**	**92**	**81**	**81**	**95**	**90**	**95**	**90**	81	**86**

Table 1. Performance comparison of attributes prediction

attributes (such as straight hair, bald) are not. This is because rotated male images may loss features around the head region after face alignment. Fortunately, this can be avoided by spatial transformer network which adjusting the degree of aligning to get the best performance of attributes classification. On CelebA, average performance of forty face attributes on images with alignment is 91.2%, and without alignment 91.1%. Similarly, average performance of forty face attributes on images with alignment is 85.7%, and without alignment 85.9% on LFWA. Although the difference of average performance is very small, the difference of specific attribute performance is apparent. On CelebA, performance on smiling is 1% higher when the input image is aligned while the performance of straight hair is nearly 1% higher when the input image is not aligned. In our experiments, the accuracy rate is rounded up or down to integers. Figure 4 shows our method, written as "without alignment + stn" achieves the state-of-art performance compared to training the ClNet only. On LFWA, performance on blurry is 1% higher when the input image is aligned while the performance of big lips is nearly 1% higher when the input image is not aligned. Performance of images by our method on LFWA is shown in Table 1. Average performance of images on LFWA is shown in Table 2 (right). Figure 5 shows our method, written as "without alignment + stn" achieves the state-of-art performance compared to training the ClNet only too.

Specifically, we compare our method with images with alignment using the same training and testing images. As shown in figure 4, performance of aligned images on CelebA

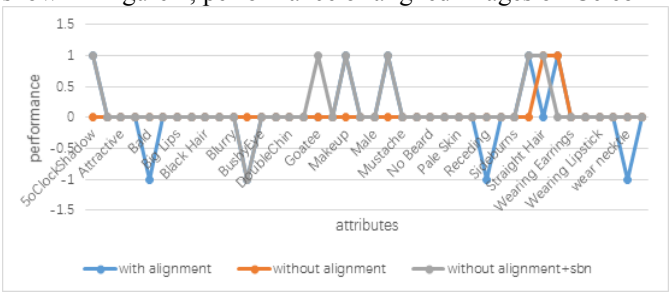

Fig. 4. Performance comparison on CelebA. Accuracy of images without alignment is set to 0, and accuracy of images with alignment is fluctuating with the accuracy of images with alignment. Result of images without alignment+spatial transformer network is higher than both of images with or without alignment except performance on bush-eye attribute.

Fig. 5. Performance comparison on LFWA. Accuracy of images without alignment is set to 0, and accuracy of images with alignment is fluctuating with the accuracy of images with alignment. Result of images without alignment +spatial transformer network is higher than both of images without alignment.

is lower than not aligned in attributes, such as smiling. We select two attributes: smiling and straight hair, in which smiling performs well in aligned images while straight hair performs well in not aligned images. Applying spatial transformation improves the performance of not aligned images on smiling attribute while keeping the best performance on straight hair attribute. Table 3 shows the result on CelebA of our method on the smiling attribute and straight hair attribute (Without alignment+ Spatial transformer network). Similar to CelebA, we select two attributes: blurry and big lips on LFWA. The attribute blurry performs well in image with alignment while big lips performs well in images without alignment. Result of our method is highlighted in Table 4(without alignment+ Spatial transformer network). The performance of images without alignment is lower than 86.2%. However, we improve this result to 86.9% by adding spatial transformer network.

Table 3 Performance comparison of spatial transformer network against traditional alignment method on smiling and straight hair attributes

	Smiling	Straight hair
alignment[14]	92.9	83.1
without aligment + spatial transformer network	**92.9**	**83.7**

Table 4 Performance comparison of spatial transformer network against traditional alignment method on blurry and big lips attributes

	Blurry	Big lips
alignment[14]	86.2	77.8
without aligment + spatial transformer network	**86.9**	**79.4**

IV. Conclusion

In this paper, we have proposed a new framework based on the spatial transformer network to improve the performance of face attributes recognition. Experiments on both CelebA [17] and LFWA [18] databases demonstrate that our approach consistently achieves the state-of-the-art performance on the both databases. Average performance on face attribute recognition is improved by 4%, 2% on CelebA and LFWA respectively.

References

[1] Z. Zhang, P. Luo, C.-C. Loy, and X. Tang, "Learning Social Relation Traits from Face Images," *Computer Science,* vol. 45, pp. 845-855, 2015.

[2] T. Berg and P. N. Belhumeur, "POOF: Part-Based One-vs.-One Features for Fine-Grained Categorization, Face Verification, and Attribute Estimation," 2013, pp. 955-962.

[3] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar, "Attribute and Simile Classifiers for Face Verification," in *IEEE International Conference on Computer Vision*, 2009, pp. 365-372.

[4] F. Song, X. Tan, and S. Chen, "Exploiting relationship between attributes for improved face verification ☆," *Computer Vision & Image Understanding,* vol. 122, pp. 143-154, 2014.

[5] S. Suchitra, S. Chitrakala, and J. Nithya, "A robust face recognition using automatically detected facial attributes," in *International Conference on Science Engineering and Management Research*, 2014.

[6] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 1778-1785.

[7] X. Yan, J. Yang, K. Sohn, and H. Lee, "Attribute2Image: Conditional Image Generation from Visual Attributes," *Computer Science,* 2015.

[8] N. Bellustin and Y. Kalafati, "Instant Human Face Attributes Recognition System," *International Journal of Advanced Computer Science & Applications,* vol. 1, pp. 269-282, 2011.

[9] N. Zhang, M. Paluri, M. A. Ranzato, T. Darrell, and L. Bourdev, "PANDA: Pose Aligned Networks for Deep Attribute Modeling," *Computer Science,* pp. 1637-1644, 2013.

[10] P. Luo, X. Wang, and X. Tang, "A Deep Sum-Product Architecture for Robust Facial Attributes Analysis," in *IEEE International Conference on Computer Vision*, 2013, pp. 2864-2871.

[11] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 3730-3738.

[12] Y. Zhong, J. Sullivan, and H. Li, "Face Attribute Prediction with classification CNN," *CoRR,* vol. abs/1602.01827, 2016.

[13] Y. Zhong, J. Sullivan, and H. Li, "Face Attribute Prediction Using Off-The-Shelf Deep Learning Networks," *CoRR,* vol. abs/1602.03935, 2016.

[14] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks," *CoRR,* vol. abs/1604.02878, 2016.

[15] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial Transformer Networks," *CoRR,* vol. abs/1506.02025, 2015.

[16] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in *British Machine Vision Conference*.

[17] Y. Sun, X. Wang, and X. Tang, "Deep Learning Face Representation by Joint Identification-Verification," *CoRR,* vol. abs/1406.4773, 2014.

[18] N. Kumar, P. N. Belhumeur, and S. K. Nayar, "FaceTracer: A Search Engine for Large Collections of Images with Faces," in *European Conference on Computer Vision (ECCV)*, pp. 340-353.

[19] J. Ashburner and K. Friston, "Spatial transformation of images," 1997.

[20] Y. Zhang, "Image processing using spatial transform," in *International Conference on Image Analysis and Signal Processing*, 2009, pp. 282-285.

[21] Y. Wen, Z. Li, and Y. Qiao, "Latent Factor Guided Convolutional Neural Networks for Age-Invariant Face Recognition," presented at the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[22] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, *et al.*, "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv preprint arXiv:1408.5093,* 2014.

[23] Y. Zhong, J. Sullivan, and H. Li, "Face Attribute Prediction Using Off-The-Shelf Deep Learning Networks," 2016.

[24] Y. Zhong, J. Sullivan, and H. Li, "Face Attribute Prediction with classification CNN," 2016.