Face Model Compression by Distilling Knowledge from Neurons

Ping Luo[1,3*], **Zhenyao Zhu**[1*], **Ziwei Liu**[1], **Xiaogang Wang**[2,3], and **Xiaoou Tang**[1,3]
[1]Department of Information Engineering, The Chinese University of Hong Kong
[2]Department of Electronic Engineering, The Chinese University of Hong Kong
[3]Shenzhen Key Lab of Comp. Vis. & Pat. Rec., Shenzhen Institutes of Advanced Technology, CAS, China
{pluo,zz012,lz013,xtang}@ie.cuhk.edu.hk, {xgwang}@ee.cuhk.edu.hk

Abstract

The recent advanced face recognition systems were built on large Deep Neural Networks (DNNs) or their ensembles, which have millions of parameters. However, the expensive computation of DNNs make their deployment difficult on mobile and embedded devices. This work addresses model compression for face recognition, where the learned knowledge of a large teacher network or its ensemble is utilized as supervision to train a compact student network. Unlike previous works that represent the knowledge by the soften label probabilities, which are difficult to fit, we represent the knowledge by using the neurons at the higher hidden layer, which preserve as much information as the label probabilities, but are more compact. By leveraging the essential characteristics (domain knowledge) of the learned face representation, a neuron selection method is proposed to choose neurons that are most relevant to face recognition. Using the selected neurons as supervision to mimic the single networks of DeepID2+ and DeepID3, which are the state-of-the-art face recognition systems, a compact student with simple network structure achieves better verification accuracy on LFW than its teachers, respectively. When using an ensemble of DeepID2+ as teacher, a mimicked student is able to outperform it and achieves 51.6× compression ratio and 90× speed-up in inference, making this cumbersome model applicable on portable devices.

Introduction

As the emergence of big training data, Deep Neural Networks (DNNs) recently attained great breakthroughs in face recognition [23, 20, 21, 22, 19, 15, 29, 30, 28] and become applicable in many commercial platforms such as social networks, e-commerce, and search engines. To absorb massive supervision from big training data, existing works typically trained a large DNN or a DNN ensemble, where each DNN consists of millions of parameters. Nevertheless, as face recognition shifts toward mobile and embedded devices, large DNNs are computationally expensive, which prevents them from being deployed to these devices. It motivates research of using a small network to fit very large training data. This work addresses model compression of DNNs for face recognition, by incorporating *domain knowledge* of learning face representation.

There have been several attempts [1, 7, 18] in literature to compress DNNs, so as to make their deployments easier, where a single network (*i.e.* a student) was trained by using the knowledge learned with a large DNN or a DNN ensemble (*i.e.* a teacher) as supervision. This knowledge can be simply represented as the probabilities of label predictions by employing the softmax function [10]. Compared with the original 1-of-K hard labels, the label probabilities encode richer relative similarities among training samples and can train a DNN more effectively. However, this representation loses much information because most of the probabilities are close to zeros after squashed by softmax. To overcome this problem, Ba and Caruana [1] represented the learned knowledge by using the logits, which are the values before softmax activation but zero-meaned, revealing the relationship between labels as well as the similarities among samples in the logit space. However, as these unconstrained values (*e.g.* the large negatives) may contain noisy information that overfits the training data, using them as supervision limits the generalization ability of the student. Recently, Hinton et al. [7] showed that both the label probabilities and zero-meaned logits are two extreme outputs of the softmax functions, where the temperature becomes one and positive infinity, respectively. To remove target noise, they empirically searched for a suitable temperature in the softmax function, until it produced soften probabilities that were able to disclose the similarity structure of data. As these soften target labels comprise much valuable information, a single student trained on them is able to mimic the performance of a cumbersome network ensemble. Despite the successes of [7], our empirical results show that training on soft targets is difficult to converge when compressing DNNs for face recognition. Previous studies [23, 24, 20, 19] have shown that the face representation learned from classifying larger amount of identities in the training data (*e.g.* 250 thousand in [24]) may have better generalization capacity. In face recognition, it seems difficult to fit soft targets with high dimensionality, which makes convergence slow.

In this work, we show that instead of using soft targets in the output layer, the knowledge of the teacher can also be obtained from the neurons in the top hidden layer, which

*indicates co-first authors who contributed equally.

preserve as much information as the soft targets (as the soft targets are predicted from these neurons) but are more compact, *e.g.* 512 versus 12,994 according to the net structure in [21]. As these neurons may contain noise or information not relevant to face recognition, they are further selected according to the usefulness of knowledge captured by them. In particular, the selection is motivated by three original *observations* (domain knowledge) of face representation disclosed in this work, which are naturally generalized to all DNNs trained by distinguishing massive identities, such as [19, 23, 24, 22]. (1) Deeply learned face representation by the face recognition task is a *distributed representation [6] over face attributes*, including the identity-related attributes (IA), such as gender, race, and shapes of facial components, as well as the identity non-related attributes (NA), such as expression, lighting, and photo quality. This observation implies that each attribute concept is explained by having some neurons being activated while each neuron is involved in representing more than one attribute, although attribute labels are not provided during training. (2) However, a certain amount of neurons are selective to NA or both NA and IA, implying that the distributed representation is *neither invariant nor completely factorized*, because attributes in NA are variations that should be removed in face recognition, whereas these two factors (NA and IA) are presented and coupled in some neurons. (3) Furthermore, a small amount of neurons are inhibitive to all attributes and server as noise. With these observations, we cast neuron selection as inference on a fully-connected graph, where each node represents attribute-selectiveness of neuron and each edge represents correlation between neurons. An efficient mean field algorithm [9] enables us to select neurons that are *more selective or discriminative to IA*, but *less correlated* with each other. As a result, the features of the selected neurons are able to maintain the inter-personal discriminativeness (*i.e.* distributed and factorized to explain IA), while reducing intra-personal variations (*i.e.* invariant to NA). We employ the features after neuron selection as regression targets to train the student.

To evaluate neuron selection, we employ DeepID2+ [21] as a teacher (T1), which achieved state-of-the-art performance on LFW benchmark [8]. This work is chosen as an example because it successfully incorporated multiple complex components for face recognition, such as local convolution [12], ranking loss function [19], deeply supervised learning [13], and model ensemble [17]. The effectiveness of all these components in face recognition have been validated by many existing works [19, 23, 24, 27]. Evaluating neuron selection on it demonstrates its capacity and generalization ability on mimicking functions induced by different learning strategies in face recognition. With neuron selection, a student with simple network structure is able to outperform a single network of T1 or its ensemble. Interestingly, this simple student generalizes well to mimic a deeper teacher (T2), DeepID3 [22], which is a recent extension of DeepID2+. Although there are other advanced methods [24, 19] in face recognition, [21, 22] are more suitable to be taken as baselines. They outperformed [24] and achieved comparable result with [19] on LFW with much smaller size of training data and identities, *i.e.* 290K images [21] compares to 7.5M images [24] and 200M images [19]. We cannot compare with [24, 19] because their data are unavailable.

Three main *contributions* of this work are summarized as below. (1) We demonstrate that more compact supervision converge more efficiently, when compressing DNNs for face recognition. Soft targets are difficult to fit because of high dimensionality. Instead, neurons in the top hidden layers are proper supervision, as they capture as much information as soft targets but more compact. (2) Three valuable observations are disclosed from the deeply learned face representation, identifying the usefulness of knowledge captured in these neurons. These observations are naturally generalized to all DNNs trained on face images. (3) With these observations, an efficient neuron selection method is proposed for model compression and its effectiveness is validated on T1 and T2.

Face Model Compression

Training Student via Neuron Selection

The merit behind our method is to select informative neurons in the top hidden layer of a teacher, and adopt the features (responses) of the chosen neurons as supervision to train a student, mimicking the teacher's feature space. We formulate the objective function of model compression as a regression problem given a training set $\mathcal{D} = \{\mathbf{I}_i, \mathbf{f}_i\}_{i=1}^{M}$,

$$L(\mathcal{D}) = \frac{1}{2M}\sum_{i=1}^{M} \|\mathbf{f}_i - g(\mathbf{I}_i; \mathbf{W})\|_2^2, \qquad (1)$$

where $\mathbf{I}_i$ and $\mathbf{f}_i$ represent the i-th face image and its corresponding selected features, respectively. $\mathbf{f}_i$ is obtained from a well training large DNN, which is the teacher. When dealing with an ensemble of DNNs, $\mathbf{f}_i$ is selected from the top layers of all the DNNs. $\mathbf{W}$ denotes a set of parameters of the student network and $g(\cdot)$ indicates a non-linear transformation from the input image to the features. Eqn.(1) is the objective function of training student network, which can be optimized by the stochastic gradient descent with standard back-propagation (BP) [10].

Here, we introduce how to obtain the features $\mathbf{f}_i$ in Eqn.(1) by selecting informative neurons. We formulate neuron selection as an inference problem on a fully-connected graph, where each node represents a neuron and each edge represents the correlation between a pair of neurons. Each node is associated with a binary latent variable, $y_i \in \{0, 1\}$, indicating whether neuron i has been chosen. Given a set of variables of N neurons, $\mathbf{y} = \{y_i\}_{i=1}^{N}$, the graph is optimized by minimizing the following energy function

$$E(\mathbf{y}) = \sum_{i=1}^{N} \Phi(y_i) + \lambda \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} \Psi(y_i, y_j), \qquad (2)$$

where $\Phi(y_i)$ and $\Psi(y_i, y_j)$ denote the unary and pairwise costs of selecting neuron i and both neurons i and j, respectively. λ is a constant weight. The first cost function is defined as $\Phi(y_i) = f(\mathbf{x}_i)$, where $f(\cdot)$ is a penalty function and $\mathbf{x}_i$ is a vector measuring the attribute discriminativeness

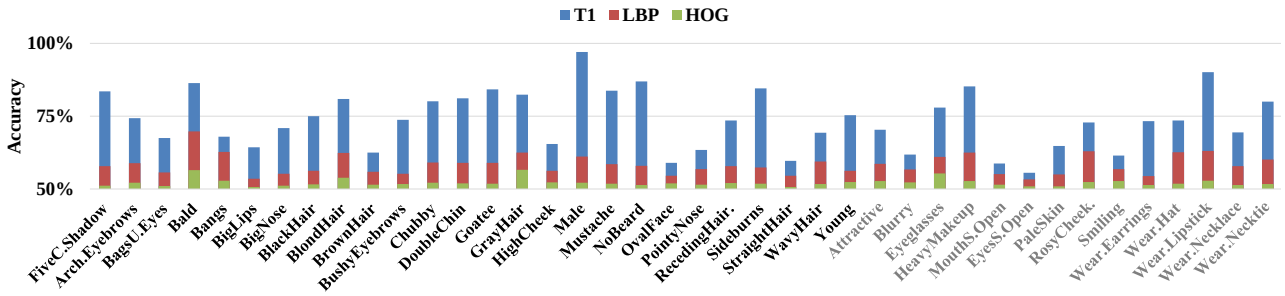

Figure 1: Attribute classification accuracies of single neurons in T1 are compared with the accuracies of single features from HOG and LBP.

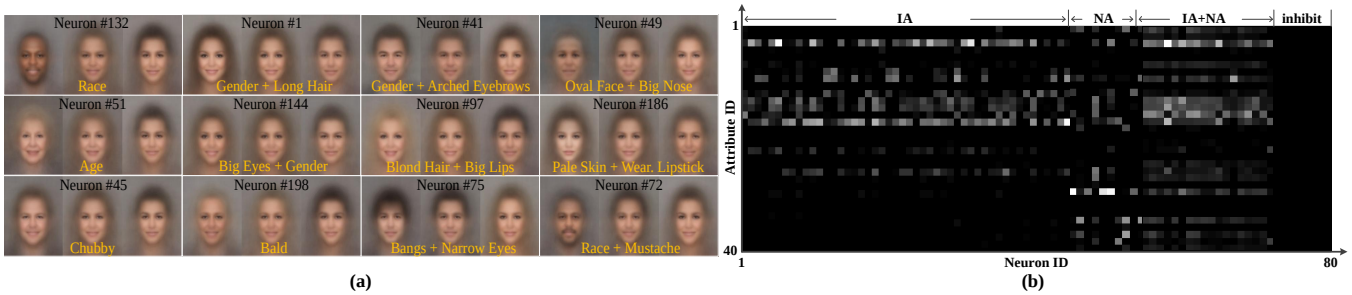

Figure 2: (a) visualizes several neurons in the top hidden layer of T1, where the top one or two most dominative attributes for each neuron are outlined in yellow. For each neurons, images with low, medium, and high responses are grouped and averaged for visualization. (b) shows four different patterns of neuron activations. From left to right: neurons that are discriminative or selective to IA, NA, NA+IA, and the inhibitive neurons, respectively. Larger intensity indicates higher classification accuracy.

of neuron i. The second one measures the similarity between neurons, *penalizing* large correlation between them, *i.e.* $\Psi(y_i, y_j) = \exp\{-\frac{1}{2}\|\mathbf{x}_i - \mathbf{x}_j\|_2^2\}$. These two terms demonstrate that we select neurons, which are more discriminative but less correlated. The representation of $\mathbf{x}$ is discussed in the next section. The graphical model in Eqn.(2) can be solved by using the mean field algorithm [9].

Attribute Discriminativeness of Neurons

We determine the values of $\mathbf{x}_i$ for each neuron i, according to its selectiveness with respect to face attributes, which are implicitly captured by the learned face representation. To show this, we take the features of T1 (outputs of top hidden layer) as an example. Nevertheless, the following observations are naturally generalized to all DNNs trained to distinguish faces.

• Firstly, deeply learned face representation is a *distributed representation over face attributes*. This observation is inspired by [21], which showed that specific neurons at the higher layer are discriminative to 13 binary attributes that are closely related to face identity, such as gender and race. To further characterize the features, we employ the CelebA dataset [14] as a validation set, which contains 20 thousand face images and each image is annotated with 40 attributes. These attributes can mostly describe a face image as suggested by [11]. As shown in Fig.1, they include both identity-related attributes (IA) and non-related attributes (NA), plotted in black and gray respectively. We define each element j in vector $\mathbf{x}_i$ as the mean classification accuracy of the j-th attribute, *i.e.* $\forall \mathbf{x}_i \in \mathbb{R}^{1\times 40}$ and $\mathbf{x}_{i(j)} = \frac{TP_j + TN_j}{2}$, where TP_j and TN_j represent the true positive and negative rates of attribute j, respectively. Fig.1 compares the maximum classification accuracy for each attribute achieved by using the neurons in T1 with the hand-crafted face descriptors like LBP [5] and HOG[1] [4], showing that deep features are automatically learned to distinguish *not only the attributes of IA but also NA*, although these concepts are not provided during training. However, LBP and HOG are not selective to neither of them. In this case, each identity is essentially represented by having a few neurons turned on (*i.e.* a distributed representation over attributes), while each neuron may be involved in representing more than one attribute as shown in Fig.2 (a), where several neurons are visualized by averaging face images of high, medium, and low responses.

• Secondly, this distributed representation is *neither invariant nor well factorized*, implying that some neurons are selective to NA or both IA and NA. As shown in Fig.2 (b), we randomly choose 80 neurons and partition them into four groups referring to their attribute-selectiveness, where larger intensity indicates higher accuracy. Most of the neurons are selective to IA, because they are trained to classify identities, but a quarter of neurons are sensitive to NA or IA+NA, implying that they are over-fitting the training data, since the attributes in NA such as 'smiling', 'earrings', and 'lipstick' belong to intra-personal variations to be removed in face recognition, while some neurons are not able to disentangle these factors.

• Thirdly, as illustrated at the rightmost side of Fig.2 (b), a small amount of neurons are inhibitive to all attributes, capturing no knowledge related to face and serving as noise.

[1]Each one-dimensional single feature of LBP or HOG is considered as an attribute classifier. The highest accuracy for each attribute achieved with a best single feature is reported.

(a) **Teacher-1** [21] (T1): 55×47×3 *input image*; 1×12K *output labels*

	1	2*	3	4*	5	6*	7	8*
layer	**conv**	**max**	**conv**	**max**	**lconv**	**max**	**lconv**	2×**fc**
neuron	relu	–	relu	–	relu	–	relu	relu
filter–stride	4-1	2-2	3-1	2-2	3-1	2-2	2-1	–
#channel	128	128	128	128	128	128	128	1
size	52×44	26×22	24×20	12×10	10×8	5×4	4×3	512, 12K
#param	6K	37M, 6M	262K	8M, 6M	12M	1M, 6M	2M	1M, 6M

(b) **Teacher-2** [22] (T2): 112×96×3 *input image*; 1×12K *output labels*

1	2	3	4*	5	6*	7	8*	9	10*
2×**conv**	**max**	2×**conv**	**max**	2×**conv**	**max**	2×**conv**	**max**	2×**lconv**	2×**fc**
relu	–	relu	–	relu	–	relu	–	relu	relu
3-1	2-2	3-1	2-2	3-1	2-2	3-1	2-2	3-1	–
64	64	96	96	192	192	256	256	256	1
112×96	56×48	56×48	28×24	28×24	14×12	14×12	7×6	5×4, 3×2	512, 12K
39K	0	138K	17M, 3M	498K	8M, 3M	1M	3M, 3M	12M, 4M	1M, 6M

(c) **Student** (S): 55×47×3 *input image*; 1×N *output*

	1	2	3	4	5	6	7	8*
layer	**conv**	**max**	**conv**	**max**	**conv**	**max**	**fc**	2×**fc**
neuron	relu	–	relu	–	relu	–	relu	relu
filter–stride	4-1	2-2	3-1	2-2	3-1	2-2	–	–
#channel	128	128	128	128	128	128	1	1
size	52×44	26×22	24×20	12×10	10×8	5×4	500	512, N
#param	6K	0	262K	0	262K	0	1M	256K, 512×N

Table 1: Comparisons among the network architectures of T1, T2, and the student model. Each table contains seven rows, representing the '*type of layer*', '*type of neuron*', '*size of filter*'–'*stride*', '*number of channels*', '*size of response map*', and '*number of parameters*', respectively. Furthermore, '**conv**', '**lconv**','**max**', and '**fc**' represent the convolution, local convolution, max pooling, and fully-connected layers respectively, while 'relu' indicates the rectified linear unit [16]. For simplicity, thousand and million are denoted as 'K' and 'M'.

As a good face representation should be both invariant and factorized to explain identity-related concepts, we select neurons discriminative to IA. To this end, the unary term can be written as $f(\mathbf{x}_i) = \frac{\max\{\mathbf{x}_{i(j)}\}_{\forall j \in \mathrm{NA}} - \mathrm{avg}\{\mathbf{x}_{i(j)}\}_{\forall j \in \mathrm{NA}}}{\max\{\mathbf{x}_{i(j)}\}_{\forall j \in \mathrm{IA}} - \mathrm{avg}\{\mathbf{x}_{i(j)}\}_{\forall j \in \mathrm{IA}}}$, where $\max\{\cdot\}$ and $\mathrm{avg}\{\cdot\}$ look for the maximum and averaged values, respectively. If a neuron is more selective to NA compared to IA, $f(\cdot)$ produces large penalty, implying neurons discriminative to IA are more likely to be selected. Furthermore, the chosen neurons should have small correlations so as to explain different concepts. This constraint is modeled by the similarity between neurons as defined in the previous section. With the above definitions, we are able to select neurons by solving Eqn.(2).

Network Structures of Teachers and Student

This section introduces the structures of T1, T2, and a simple structure to mimic them.

Teachers The architectures of T1 and T2 are summarized in Table 1 (a) and (b) respectively, where the first two rows represent the types of layer and neuron, while 'x-y' in the third row represents the filter size and the stride of convolution. The last three rows represent number of channels, size of response maps, and number of parameters, respectively. As listed in Table 1 (a), T1 learns 512-dimensions face features by classifying 12K identities with images of $55 \times 47 \times 3$ as input. It contains four convolutional layers, three max-pooling layers, and two fully-connected layers. These layers can be partitioned into eight groups, each of which covers one or more homogenous layers. The superscript ('$*$') over the group index indicates supervisory signals are propagated to this group. For instance, the second pooling layer is also connected to two 'fc' layers, which have the same hyper-parameters as group-8, leading to two large weight matrixes of $128 \times 26 \times 22 \times 512 \approx$ 37M and $512 \times$ 12K$\approx$ 6M parameters, respectively. Similarly, the supervision is also propagated into group-4 and 6 respectively. In other words, T1 was trained in a deeply supervised manner, leading to more discriminative low- and middle-level representations. As a result, T1 has 85M parameters. T2 contains smaller number of parameters but deeper structure compared to T1. As listed in Table 1 (b), it has 62M parameters and 16-layers depth.

Student As shown in Table 1 (c), the structure of the student network (S) is simply derived from T1, where the local convolutional layers ('lconv') in group-5 and 7 are replaced by a convolutional layer and a fully-connected layer respectively, reducing the number of parameters by 11 times. All the supervision evaluated in our experiments are trained with S. For the soft targets [7] and the logits [1], S is learned to minimize the cross-entropy and squared losses, respectively. Thus, the output dimension N equals 12K. For neuron selection, S predicts the features by minimizing Eqn.(1). In this case, N is typically smaller than 512, and therefore the student only contains about 2M parameters compared to the parameters of 85M in T1 and 62M in T2.

Experiments

MNIST We first evaluate the effectiveness of targeting the features in the top hidden layer as a learning objective for model compression. To this end, we test on MNIST similar

networks	#target length	AUC
T1 [21] (single)	12,294	97.83
S-1-of-K	12,294	91.23
S-soft target (t=1) [7]		94.42
S-soft target (t=10) [7]		97.15
S-soft target (t→ +∞) [1]		96.75
S-neurons	512	97.73
S-selection	422	**98.18**

Figure 3: Mimicking a single network of T1.

networks	#target length	AUC
T2 [22] (single)	12,294	98.27
S-soft target (t=1) [7]	12,294	91.88
S-soft target (t=10) [7]		97.27
S-soft target (t→ +∞) [1]		96.77
S-neurons	512	97.90
S-selection	377	98.12
S-selection† (unsupervised)	377	**98.37**

Figure 4: Mimicking a single network of T2.

networks	#target length	AUC
T1 [21] (an ensemble of 6 networks)	12,294	98.37
S-soft target (t=1) [7]	12,294	91.88
S-soft target (t=5) [7]		96.57
S-soft target (t=10) [7]		96.07
S-soft target (t=15) [7]		96.33
S-soft target (t→ +∞) [1]		96.65
S-neurons	3,072	98.07
S-selection	960	98.25
S-selection† (unsupervised)	960	**98.43**

Figure 5: Mimicking an ensemble of T1.

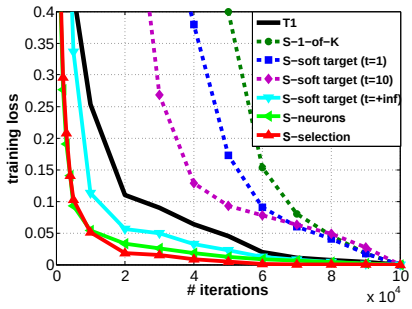

Figure 6: Comparisons of convergence rates.

to [7] by distilling the learned knowledge of a teacher (T) with 1200 neurons to a student (S) with 800 neurons. As a result, T and S achieve 63 and 168 test errors[2] respectively, when both of them are trained using 1-of-K as targets. Since it is difficult to employ domain knowledge on MNIST, we select features to train S by simply removing the inhibitive neurons of T. In this case, S achieves 82 errors, which outperforms all soft targets when the temperature $t = 1$, 2, 5, 10, 20, and $t = +\infty$ (*i.e.* the logits). Their errors are 124, 104, 111, 119, 120, and 92 respectively.

Face Data For face model compression, we train all the students using data the same as [21, 22], which combined two face databases for training, CelebFaces+ [20] and WDRef [3], resulting in a training set of 290 thousands face images of 12,294 identities. In test, we evaluate all the models on LFW [8], which is the most well known benchmark for face recognition, containing 13,233 face images of 5,749 identities collected from the Internet. Note that the identities in training and test are exclusive. The face verification performance on LFW is reported as the Area under ROC curve (AUC) with respect to 3,000 positive and 3,000 negative face pairs.

For face verification, feature vectors in the top hidden layers are first extracted from a pair of face images and then the Euclidean distance[3] between them is calculated for face verification. Unlike [23, 21] trained SVM or Joint Bayesian [3] for face verification, the Euclidean distance is used throughout the experiments to directly show the benefit from better supervision utilized to train students, other than strong classifiers with additional supervision.

Compressing a Single Model of T1 We train many students (S) with different targets to compress T1. Note that the structures of different S's are the same except the last output layers. The architectures of S and T1 are given in Table 1. As shown in Fig.3, a student trained with knowledge distilled from selected neurons ('S-selection') achieves the best performance. It even outperforms its teacher T1, showing that selected neurons can preserve similarities among samples as well as remove noisy information in the features of T1. However, students supervised by the other targets have different losses in accuracy. Specifically, S is merely dropped by 0.68% compared to T1 when being trained with soft targets ($t = 10$), but dropped by 6.6% with hard labels ('1-of-K'). Small networks cannot be well trained with hard labels. We examine different temperatures for soft targets. Note that when $t = 1$ and $t = +\infty$, soft targets turn into label probabilities (after softmax) and logits, respectively. The best performance of soft targets is achieved when $t = 10$. 'S-neurons' directly mimics the features of neurons in the top hidden layer of T1, and with 0.1% drop in the verification accuracy compared with T1. Neuron selection increases the accuracy by 0.35%. Furthermore, to verify the improvement is come from better targets, we train 'S-1-of-K' with attributes as additional supervision. The accuracy is 91%, showing that predicting identities and attributes jointly does not help face recognition, because attributes can be implicitly learned by classifying identities.

Fig.6 compares the training losses of T1 and different students. Several *valuable facts* can be observed. First, when using hard labels as targets, a larger network converges faster (*e.g.* comparing T1 and 'S-1-of-K'), since it has larger fitting capability. Second, S's trained with compact and informative targets converge faster than long targets, *e.g.* 'S-neurons' and 'S-selection' have 512 and 422 dimensional targets respectively, while 'S-soft target' has 12,294 dimensional

[2]The test error indicates how many test samples are misclassified.

[3]If the Euclidean distance between features extracted from a pair of face images exceeds a threshold, it is negative pair; otherwise, it is positive. The threshold is determined by cross-validation on LFW following [8].

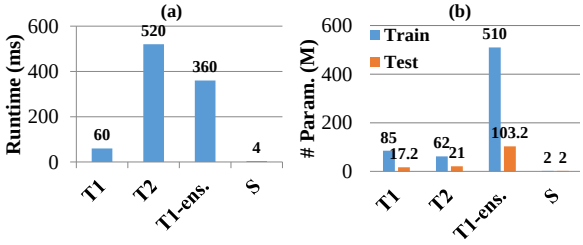

Figure 7: (a) Comparisons of running times (milliseconds per image). (b) Comparisons of number of parameters (millions).

targets. With these targets, a small student is able to learn the knowledge of a large teacher with a smaller amount of training time compared to that used to train the teacher. For example, 'S-selection' outperforms T1 with 42 times fewer parameters and learns at least 5 times faster. Third, convergence rate increases when the temperature of soft target increases. To the extreme, training with logits (*i.e.* $t = +\infty$) is easy, but training with 1-of-K labels is the most difficult, because the hard label vector has high dimensionality (*e.g.* 12K identities) and only one of its entries is non-zero. So the mimic network may produce wrong prediction at many iterations and converge slowly. In contrast, when temperature increases, the soft targets contain more non-zero values and become more informative.

Compressing a Deeper Teacher (T2) Fig.4 shows that the student (S) with shallow structure generalizes well to compress a deep teacher (T2), where T2 is 2× deeper than S. In this case, 'S-selection†' outperforms T2 by 0.1%. It is obtained by fine-tuning 'S-selection' with ten-folds cross-validation on LFW [8]. This is done in an unsupervised manner without leveraging the identity labels of LFW, but using the selected features of T2. Without fine-tuning, the accuracy of 'S-selection' decreases by 0.15% compared to T2, indicating that deeper teacher is more difficult to fit. However, fine-tuning on more unsupervised data can improve performance. When compressing T2, S trained with neuron selection also outperforms those trained with the other supervision.

Compressing an Ensemble of T1 As listed in Fig.5, S is employed to compress an ensemble of six T1 networks, each of which was trained on different face regions, including two eyes, nose, two mouth corners, and the entire face region. This ensemble outperforms the best single network of T1 by 0.54%. For each student trained with soft targets, the supervision is obtained by averaging the soft targets of all the networks in this ensemble. When training 'S-selection' with the same data as T1 ensemble, its performance decreases by 0.12%. This result is similar to that of compressing T2, implying that not only deep teacher, an ensemble of teachers is also difficult to mimic. However, the performance can still be improved by unsupervised fine-tuning, *e.g.* 'S-selection' is increased by 0.18% after fine-tuning, outperforming T1 ensemble.

Complexity and Efficiency Fig.7 compare the efficiencies and complexities between T1, T2, T1 ensemble, and the student network (S). Efficiency is measured with implementation on a Intel Core 2.0GHz CPU. To simulate the environment of embedded or portable devices, the runtime is evaluated on CPU instead of GPU. As shown in Fig.7 (a), S achieves 90× speed-up compared to T1 ensemble. The model complexities are measured by the numbers of parameters during training and testing, respectively. *The former indicates learning capacity, while the latter indicates complexity in inference.* As shown in Fig.7 (b), if we compare S with T1 ensemble, S reduces the inference complexity by 51.6× and increases the performance by 0.06%, using a network structure with much smaller learning capacity, *i.e.* 1/255. In general, with neuron selection, the student S is able to outperform its corresponding teacher models by using much fewer parameters and process much faster. Specifically, it occupies 4 megabytes storage and processes face images with 250 frames per second, making T1 ensemble applicable on embedded or portable devices.

Conclusions and Discussions

This work demonstrates several interesting results towards model compression for face recognition. (1) In face recognition, both hard and soft labels are difficult to fit because of high dimensionality in the output layer, as well as the zero entries they contain. Instead, neurons in the top hidden layer are more suitable supervision because they capture as much information as soft targets, but are more compact. Experiments validate its effectiveness and its superior convergence rate. (2) Valuable observations show that the deeply learned face representation is *neither invariant nor well factorized*. Therefore, employing all the features as targets is not a beneficial solution because they may contain noise or knowledge that is not relevant to face recognition. A neuron selection method is proposed to select neurons, so as to obtain a face representation that maintains the inter-personal discriminativeness, while reduces intra-personal variations. This is the goal of all the face recognition algorithms [25, 2, 26, 19, 21]. (3) As more informative and compact representation can be obtained by neuron selection, a mimic student trained on the selected features outperform its teacher, by using much smaller number of parameters and shallower network structure (*e.g.* comparing T2 and S-selection), making the student easy to be parallelized or distributed. Similarly, [1] trained shallow student with logits, but our experiments show that students trained in this way cannot outperform their teachers for face recognition. In contrast, [18] trained deeper and thinner student to improve the teacher, but sacrificed distributed processing because a deep network has to be processed sequentially through each of its layer. (4) When using an ensemble as teacher, a small student is also able to outperform this teacher. This result has not been disclosed in previous works [1, 7, 18].

Acknowledgement This work is partially supported by the National Natural Science Foundation of China (61503366, 91320101, 61472410), Guangdong Innovative Research Team Program (No.201001D0104648280), and Shenzhen Basic Research Program (KQCX2015033117354153, JCYJ20120903092050890, JCYJ20130402113127496).

References

[1] L. Jimmy Ba and R. Caruana. Do deep nets really need to be deep? In *NIPS*, 2014.

[2] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *TPAMI, vol.19*, pages 711–720, 1997.

[3] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *ECCV*, 2012.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[5] DC. He and L. Wang. Texture unit, texture spectrum, and texture analysis. In *IEEE TGRS, vol.28*, 1990.

[6] G. E. Hinton. Learning distributed representations of concepts. In *Proc. 8th Conf. Cog. Sc. Society*, 1986.

[7] G. E. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *arXiv:1503.02531*, 2015.

[8] Gary B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *University of Massachusetts, Amherst, Technical Report*, 2007.

[9] Michael I. Jordan, Z. Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. In *Machine Learning, vol.37, no.2*, 1999.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[11] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.

[12] Quoc V. Le, J. Ngiam, Z. Chen, D. Chia, P. Koh, and Andrew Y. Ng. Tiled convolutional neural networks. In *NIPS*, 2010.

[13] C. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu. Deeply-supervised nets. In *AISTATS*, 2015.

[14] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[15] P. Luo, X. Wang, and X. Tang. Hierarchical face parsing via deep learning. In *CVPR*, 2012.

[16] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[17] D. Opitz and R. Maclin. Popular ensemble methods: An empirical study. In *Journal of Artificial Intelligence Research*, 1999.

[18] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

[19] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *arXiv:1503.03832*, 2015.

[20] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014.

[21] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *arXiv:1412.1265*, 2014.

[22] Y. Sun, D. Liang, X. Wang, and X. Tang. Deepid3: Face recognition with very deep neural networks. In *arXiv:1502.00873*, 2015.

[23] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014.

[24] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Web-scale training for face identification. In *arXiv:1406.5266*, 2014.

[25] M. Turk and A. Pentland. Eigenfaces for recognition. In *J. of Cognitive Neuroscience, vol.3, no.1*, 1991.

[26] X. Wang and X. Tang. A unified framework for subspace face recognition. In *TPAMI, vol.26, no.9*, pages 1222–1228, 2004.

[27] E. Zhou, Z. Cao, and Q. Yin. Naive-deep face recognition: Touching the limit of lfw benchmark or not? In *arXiv:1501.04690*, 2015.

[28] Z. Zhu, P. Luo, X. Wang, and X. Tang. Deep learning identity-preserving face space. In *ICCV*, 2013.

[29] Z. Zhu, P. Luo, X. Wang, and X. Tang. Multi-view perceptron: a deep model for learning face identity and view representations. In *NIPS*, 2014.

[30] Z. Zhu, P. Luo, X. Wang, and X. Tang. Recover canonical-view faces in the wild with deep neural networks. In *arXiv:1404.3543*, 2014.