A summary of deep models for face recognition

Qianli Liao

Face recognition

- Face recognition:

  Detection → Alignment → Recognition

- Face detection & alignment
- Face recognition

Face detection & alignment

- Detection
- Alignment (~= landmark localization)

Face detection

- Deformable Parts Models (DPMs)

  Most of the publicly available face detectors are DPMs. It is easy to find them online.

- CNNs (old ones)

  R. Vaillant, C. Monrocq and Y. LeCun: An Original approach for the localisation of objects in images, International Conference on Artificial Neural Networks, 26-30, 1993

  Garcia, Christophe, and Manolis Delakis. "A neural architecture for fast and robust face detection." Pattern Recognition, 2002. Proceedings. 16th International Conference on. Vol. 2. IEEE, 2002.
  Osadchy, Margarita, Yann Le Cun, and Matthew L. Miller. "Synergistic face detection and pose estimation with energy-based models." The Journal of Machine Learning Research 8 (2007): 1197-1215.

- CNNs (recent)

  Li, Haoxiang, et al. "A Convolutional Neural Network Cascade for Face Detection." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

  Farfade, Sachin Sudhakar, Mohammad Saberian, and Li-Jia Li. "Multi-view Face Detection Using Deep Convolutional Neural Networks." arXiv preprint arXiv:1502.02766 (2015).

Cascade CNN for face detection

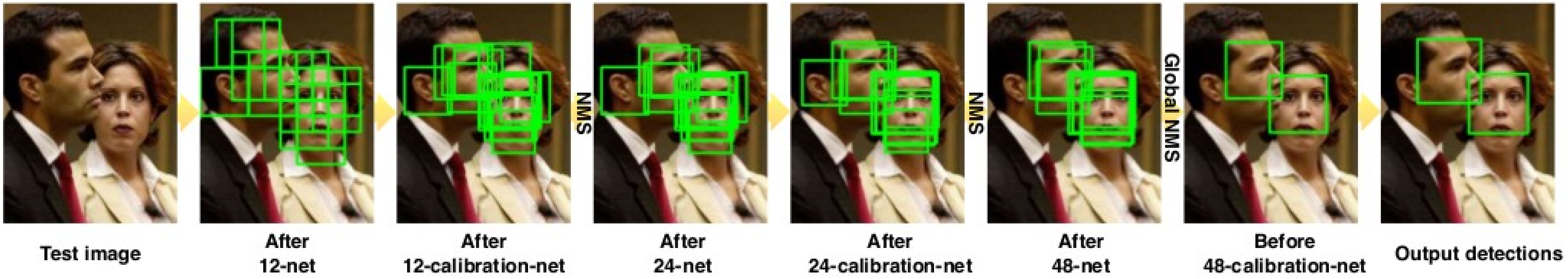
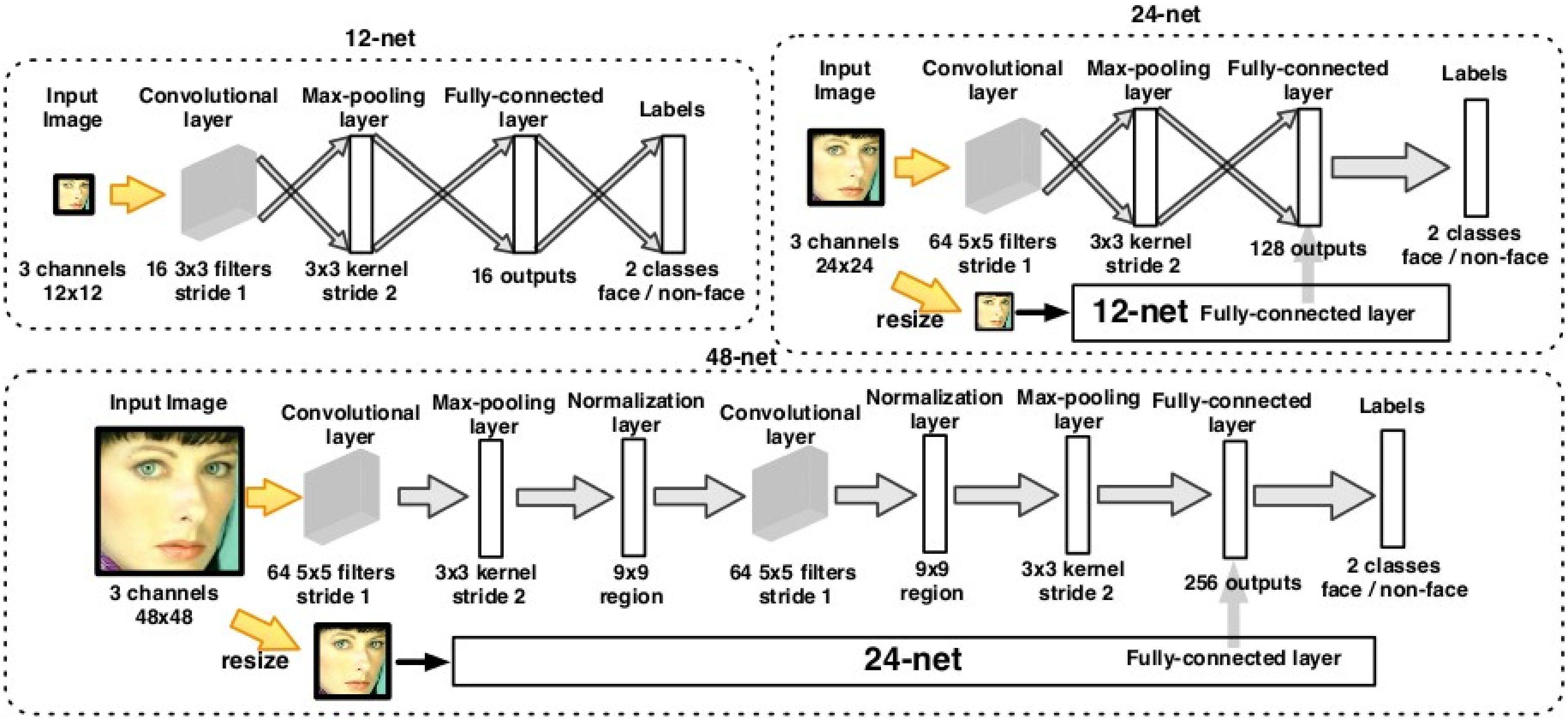
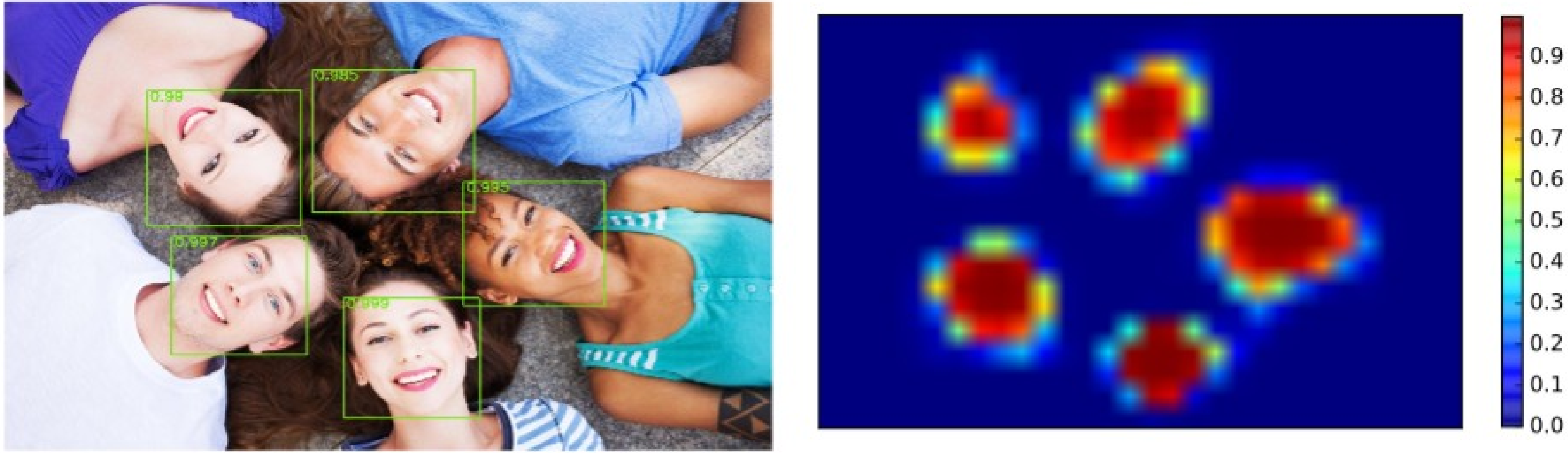


Figure 1: Test pipeline of our detector: from left to right, we show how the detection windows (green squares) are reduced and calibrated from stage to stage in our detector. The detector runs on a single scale for better viewing.



Multiview Face detection by fine-tuning AlexNet



Farfade, Sachin Sudhakar, Mohammad Saberian, and Li-Jia Li. "Multi-view Face Detection Using Deep Convolutional Neural Networks." arXiv preprint arXiv:1502.02766 (2015).

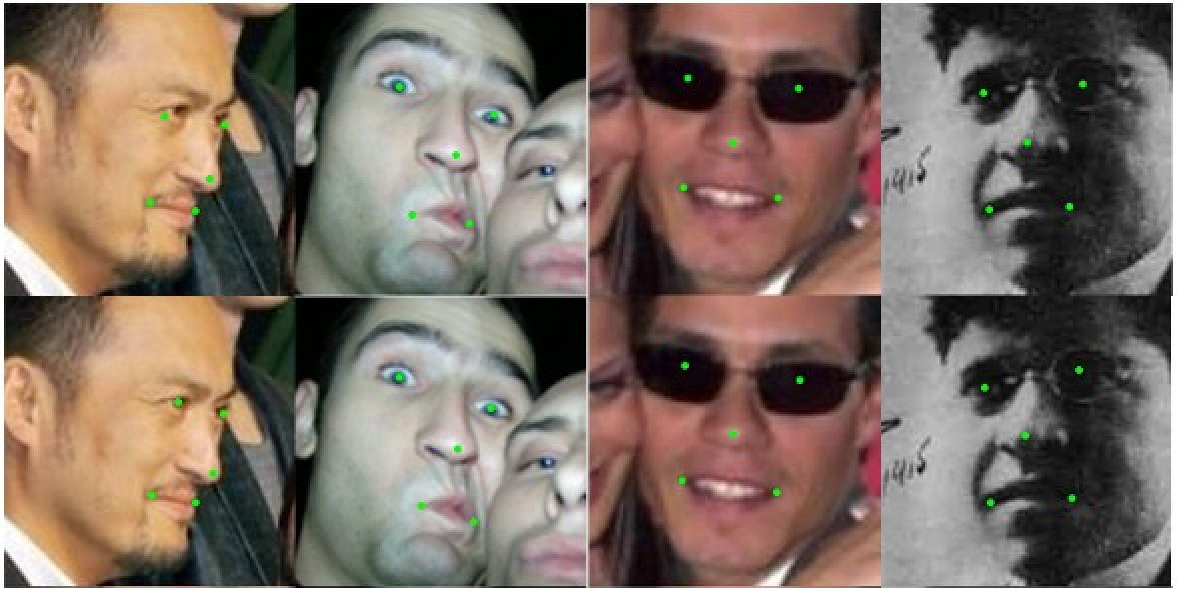
Face alignment

- There are many face alignment algorithms. I'll mainly talk about the ones used by DeepID models.

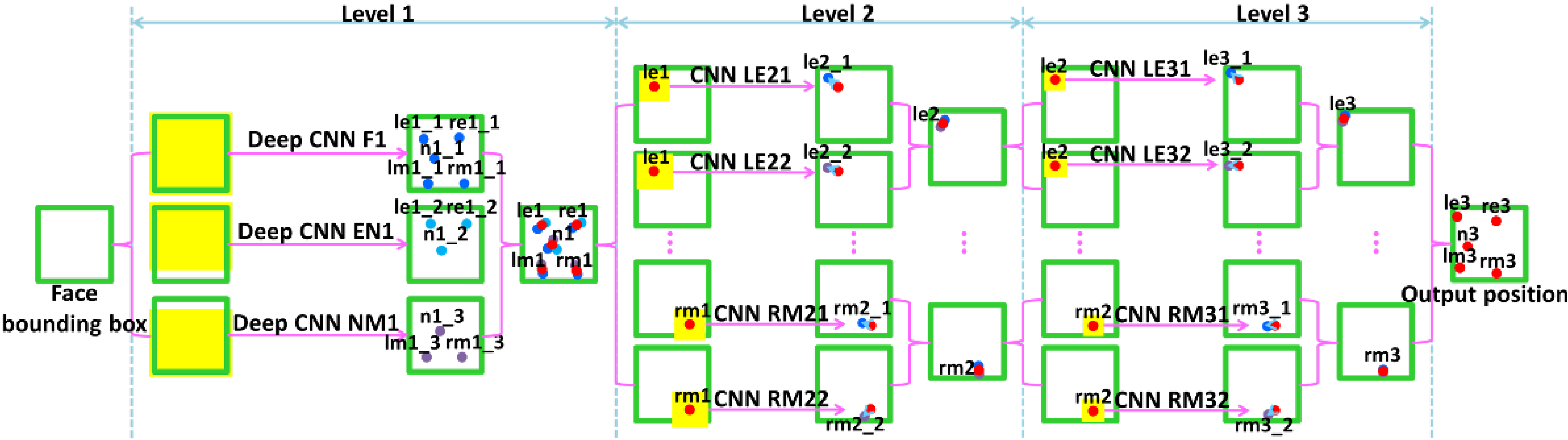
- DeepID 1: Sun, Yi, Xiaogang Wang, and Xiaoou Tang. "Deep convolutional network cascade for facial point detection." Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013.
- DeepID 2,2+,3 (**CMU Intraface**): Xiong, Xuehan, and Fernando De la Torre. "Supervised descent method and its applications to face alignment." Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013. (Not CNN)

Initial

Fine-tuned



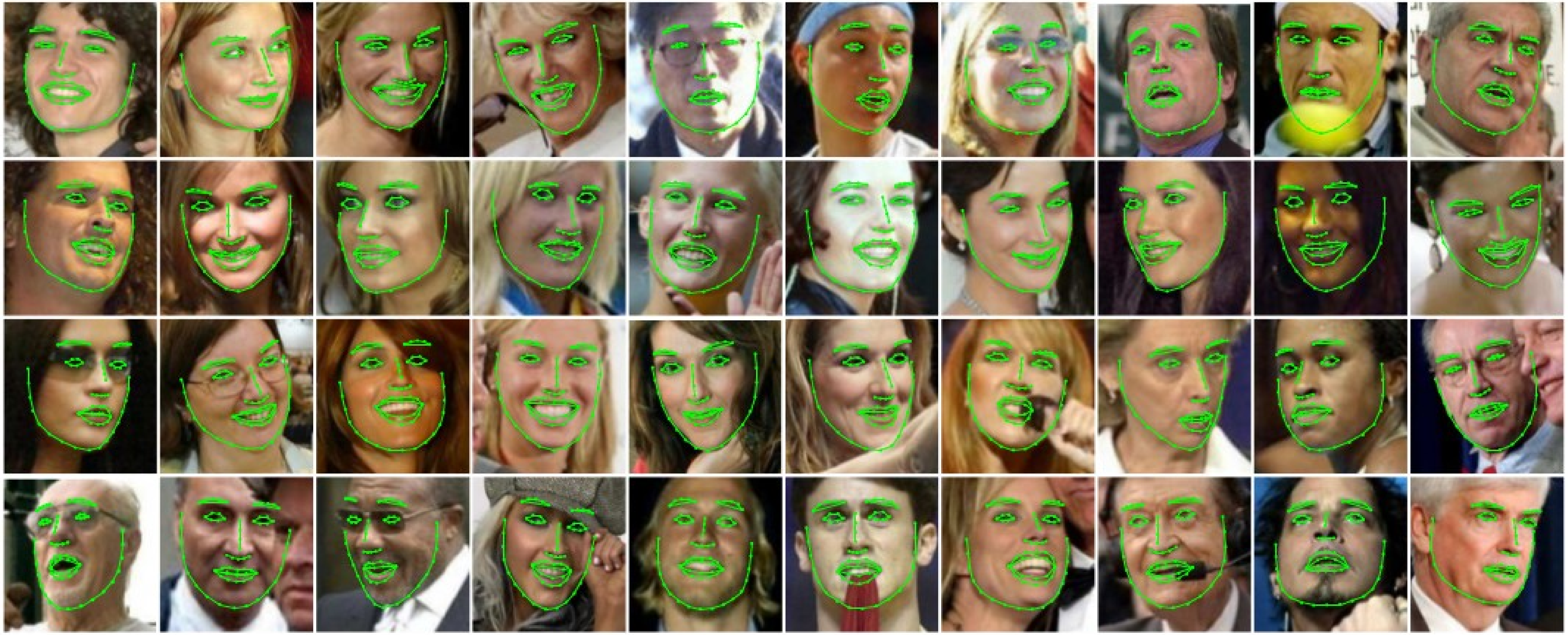
DeepID 1
Landmarks



- Sun, Yi, Xiaogang Wang, and Xiaoou Tang. "Deep convolutional network cascade for facial point detection." Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013.

DeepID 2 Landmarks

- Using CMU IntraFace landmark detector (non-CNN)

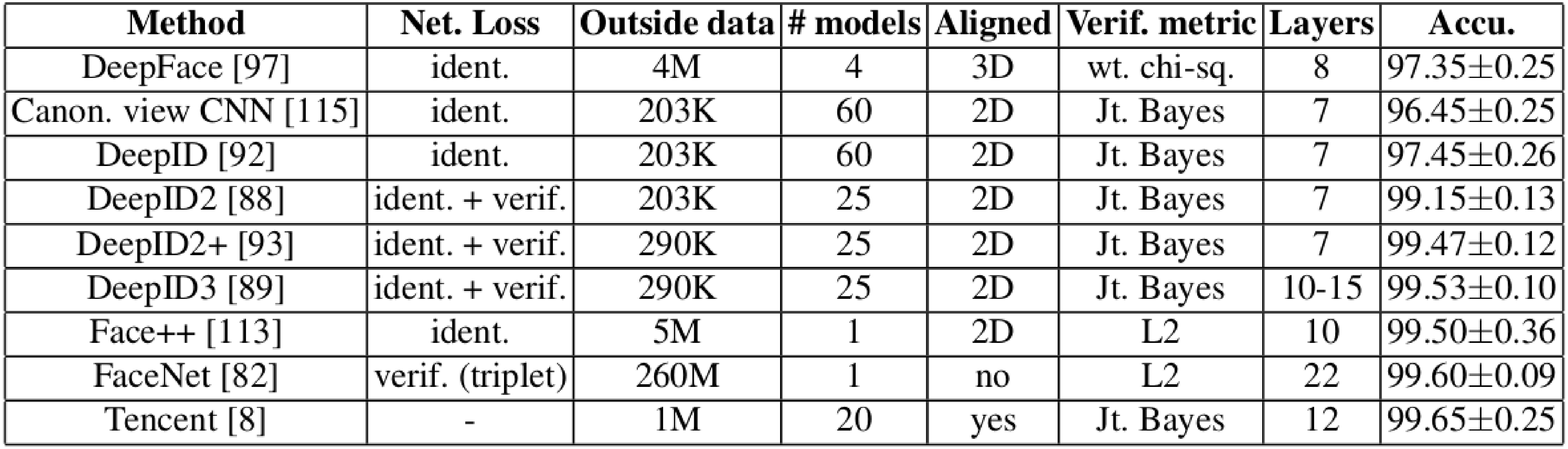


Alignment

- After the landmarks are detected, one could apply a simple similarity transformation
- This strategy is used by most of the models including DeepIDs
- But DeepFace uses 3D alignment.

Deep face recognition

1. DeepID
2. DeepID2
3. DeepID2+
4. DeepID3
5. DeepFace
6. Face++
7. FaceNet
8. Baidu

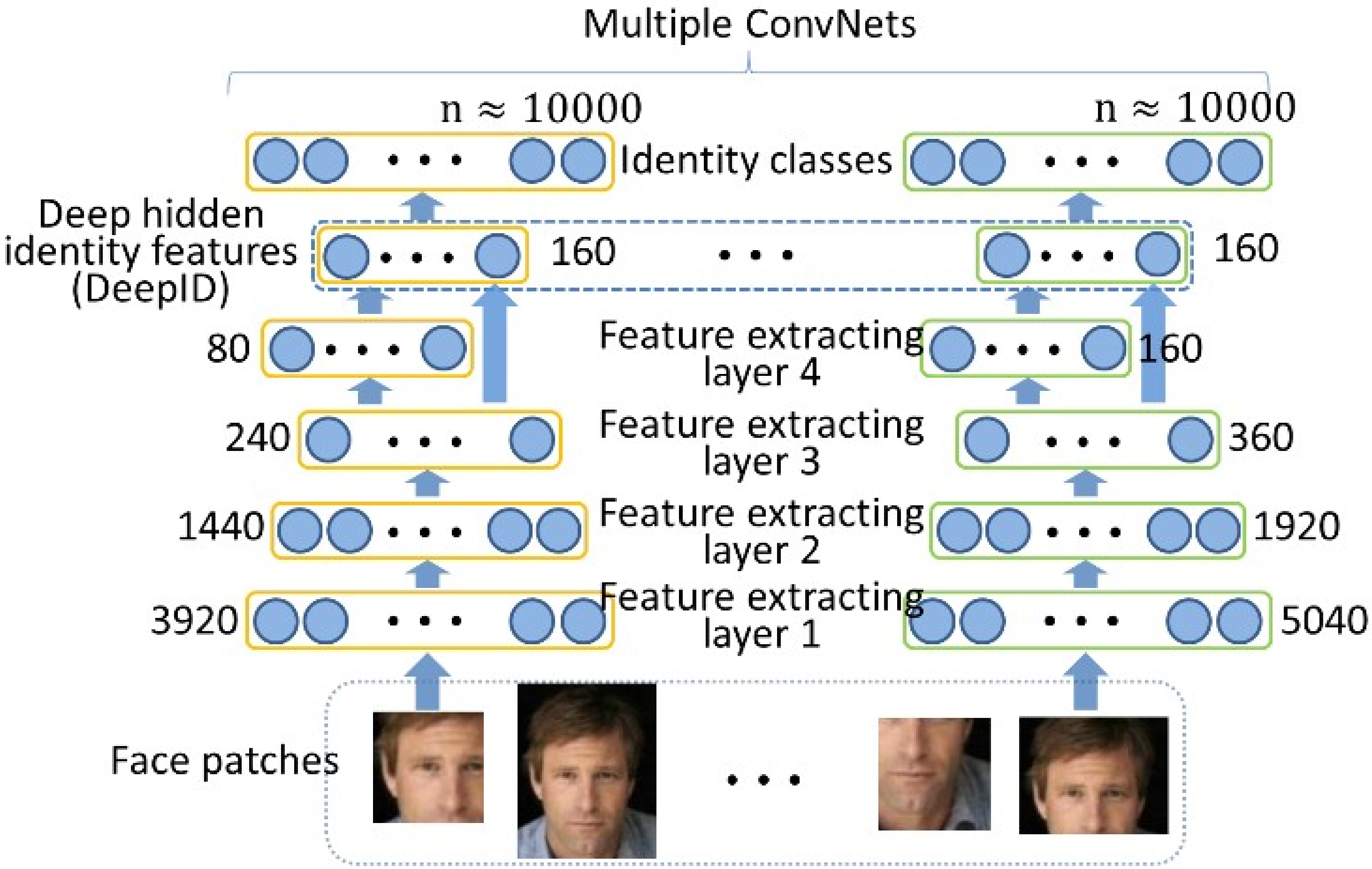


Method	Net. Loss	Outside data	# models	Aligned	Verif. metric	Layers	Accu.
DeepFace [97]	ident.	4M	4	3D	wt. chi-sq.	8	97.35±0.25
Canon. view CNN [115]	ident.	203K	60	2D	Jt. Bayes	7	96.45±0.25
DeepID [92]	ident.	203K	60	2D	Jt. Bayes	7	97.45±0.26
DeepID2 [88]	ident. + verif.	203K	25	2D	Jt. Bayes	7	99.15±0.13
DeepID2+ [93]	ident. + verif.	290K	25	2D	Jt. Bayes	7	99.47±0.12
DeepID3 [89]	ident. + verif.	290K	25	2D	Jt. Bayes	10-15	99.53±0.10
Face++ [113]	ident.	5M	1	2D	L2	10	99.50±0.36
FaceNet [82]	verif. (triplet)	260M	1	no	L2	22	99.60±0.09
Tencent [8]	-	1M	20	yes	Jt. Bayes	12	99.65±0.25

Learned-Miller, Erik, et al. "Labeled Faces in the Wild: A Survey.

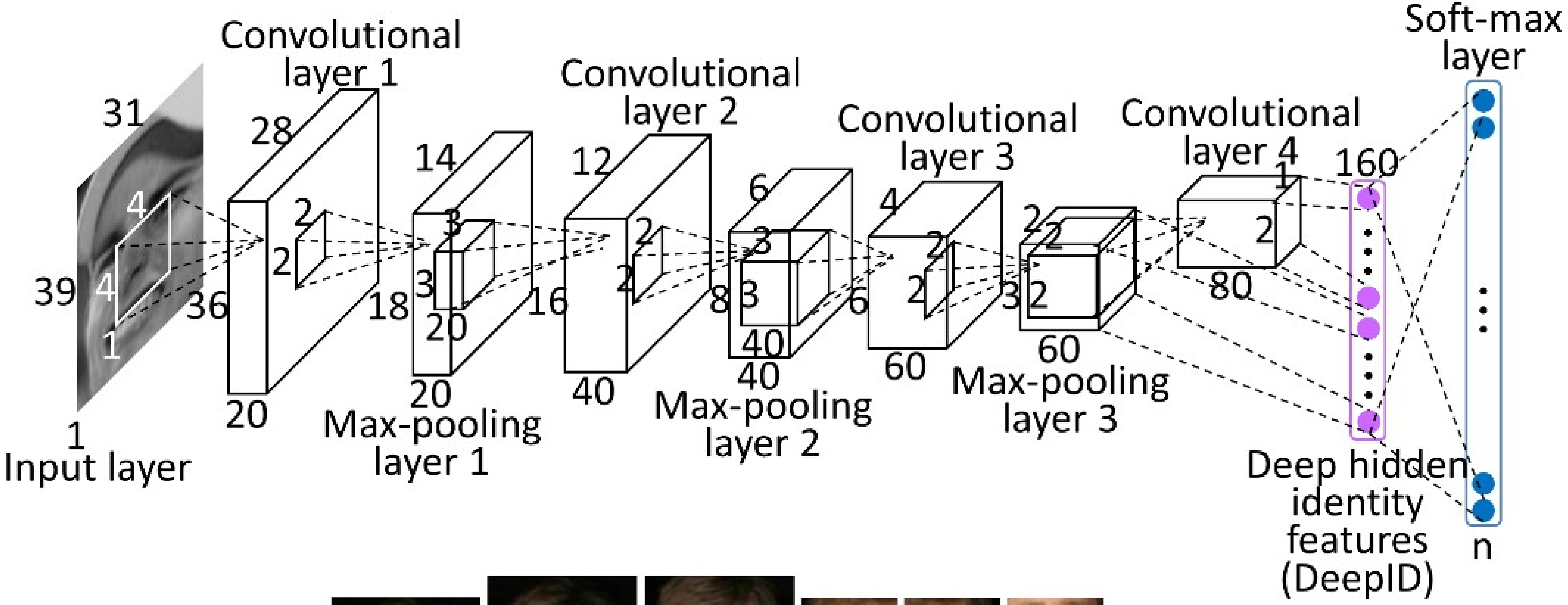
- DeepID seems most interesting since least data is used

DeepID 1 (CVPR 2014)


Multiple ConvNets
n ≈ 10000
n ≈ 10000
Identity classes
Deep hidden identity features (DeepID)
160
160
Feature extracting layer 4
80
160
Feature extracting layer 3
240
360
Feature extracting layer 2
1440
1920
Feature extracting layer 1
3920
5040
Face patches

One CNN for a landmark location (or a crop of the face at some scale). 60 CNNs in total. Concatenate all second-to-last layers. Reduce to 150 dim. by PCA.

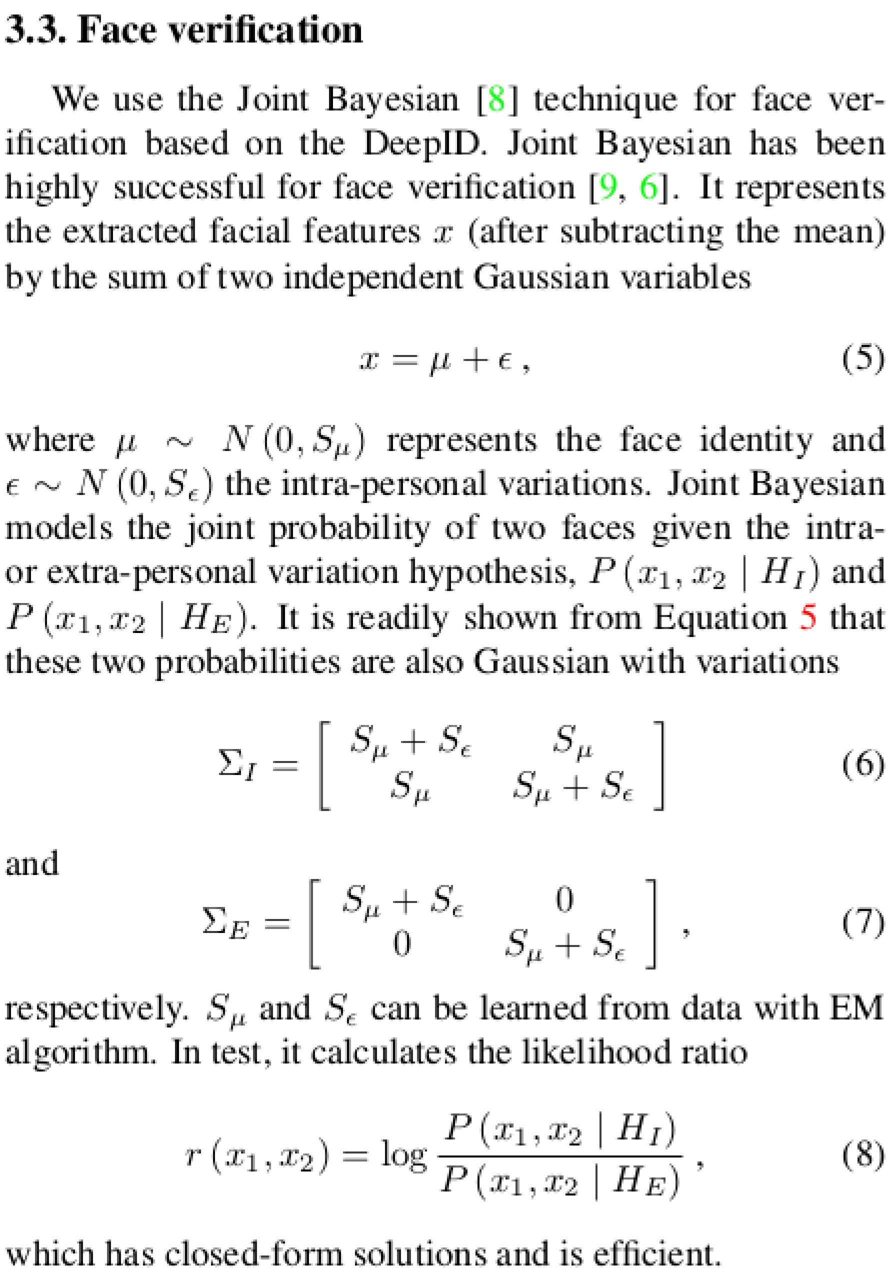
DeepID 1 (CVPR 2014)



DeepID 1

- 5 landmarks: two eye centers, the nose tip, and the two mouth corners
- Globally aligned by similarity transformation
- 10 Regions * 3 scales * RGB/Gray = 60 patches
- 60 ConvNets, each of which extracts two 160-dimensional vectors from a particular patch and its horizontally flipped counterpart (the flipped counterpart of the patch centered on the left eye is derived by flipping the patch centered on the right eye).
- The total length of DeepID is 19,200 (160 × 2 × 60), which is reduced to 150 by PCA for the purpose of verification

Joint Bayesian



3.3. Face verification

We use the Joint Bayesian [8] technique for face verification based on the DeepID. Joint Bayesian has been highly successful for face verification [9, 6]. It represents the extracted facial features x (after subtracting the mean) by the sum of two independent Gaussian variables

$$x = \mu + \epsilon \, , \tag{5}$$

where $\mu \sim N(0, S_\mu)$ represents the face identity and $\epsilon \sim N(0, S_\epsilon)$ the intra-personal variations. Joint Bayesian models the joint probability of two faces given the intra- or extra-personal variation hypothesis, $P(x_1, x_2 \mid H_I)$ and $P(x_1, x_2 \mid H_E)$. It is readily shown from Equation 5 that these two probabilities are also Gaussian with variations

$$\Sigma_I = \begin{bmatrix} S_\mu + S_\epsilon & S_\mu \\ S_\mu & S_\mu + S_\epsilon \end{bmatrix} \tag{6}$$

and

$$\Sigma_E = \begin{bmatrix} S_\mu + S_\epsilon & 0 \\ 0 & S_\mu + S_\epsilon \end{bmatrix} , \tag{7}$$

respectively. S_μ and S_ϵ can be learned from data with EM algorithm. In test, it calculates the likelihood ratio

$$r(x_1, x_2) = \log \frac{P(x_1, x_2 \mid H_I)}{P(x_1, x_2 \mid H_E)} , \tag{8}$$

which has closed-form solutions and is efficient.

Joint Bayesian

DeepID1

- ~8000 identities are used for training CNN
- ~2000 identities are held-out for training JB

DeepID2, 2+, 3

- ~10000 identities are used for training CNN
- ~2000 identities are held-out for training JB

DeepID 1

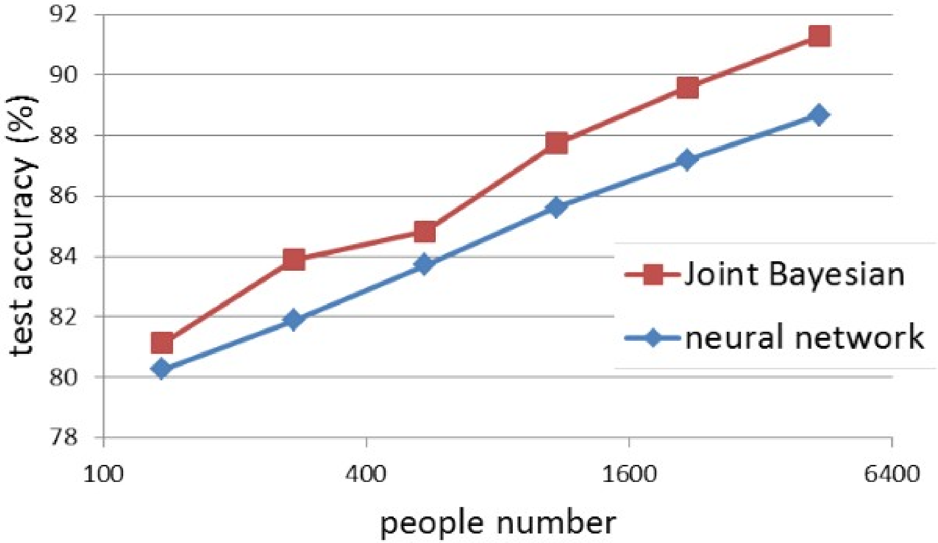


Figure 6. Face verification accuracy of Joint Bayesian (red line) and neural network (blue line) learned from the DeepID, where the ConvNets are trained with 136, 272, 544, 1087, 2175, and 4349 classes, respectively.

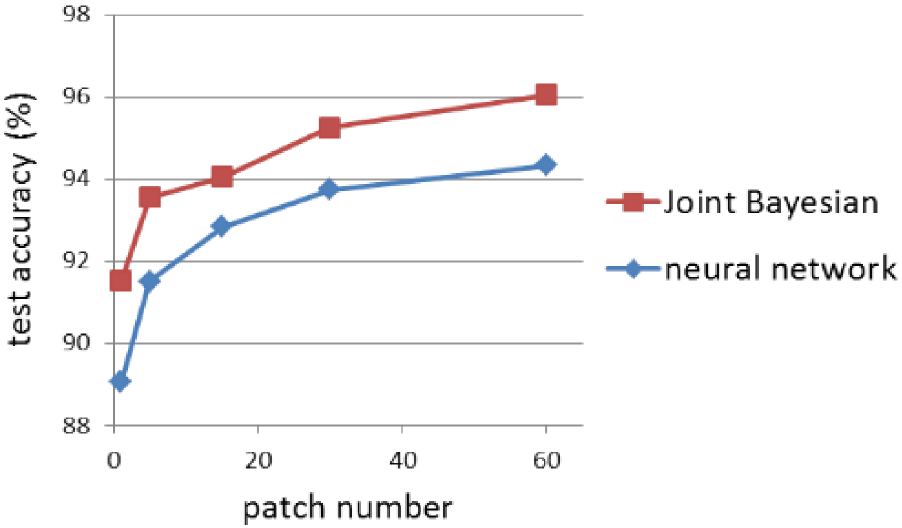
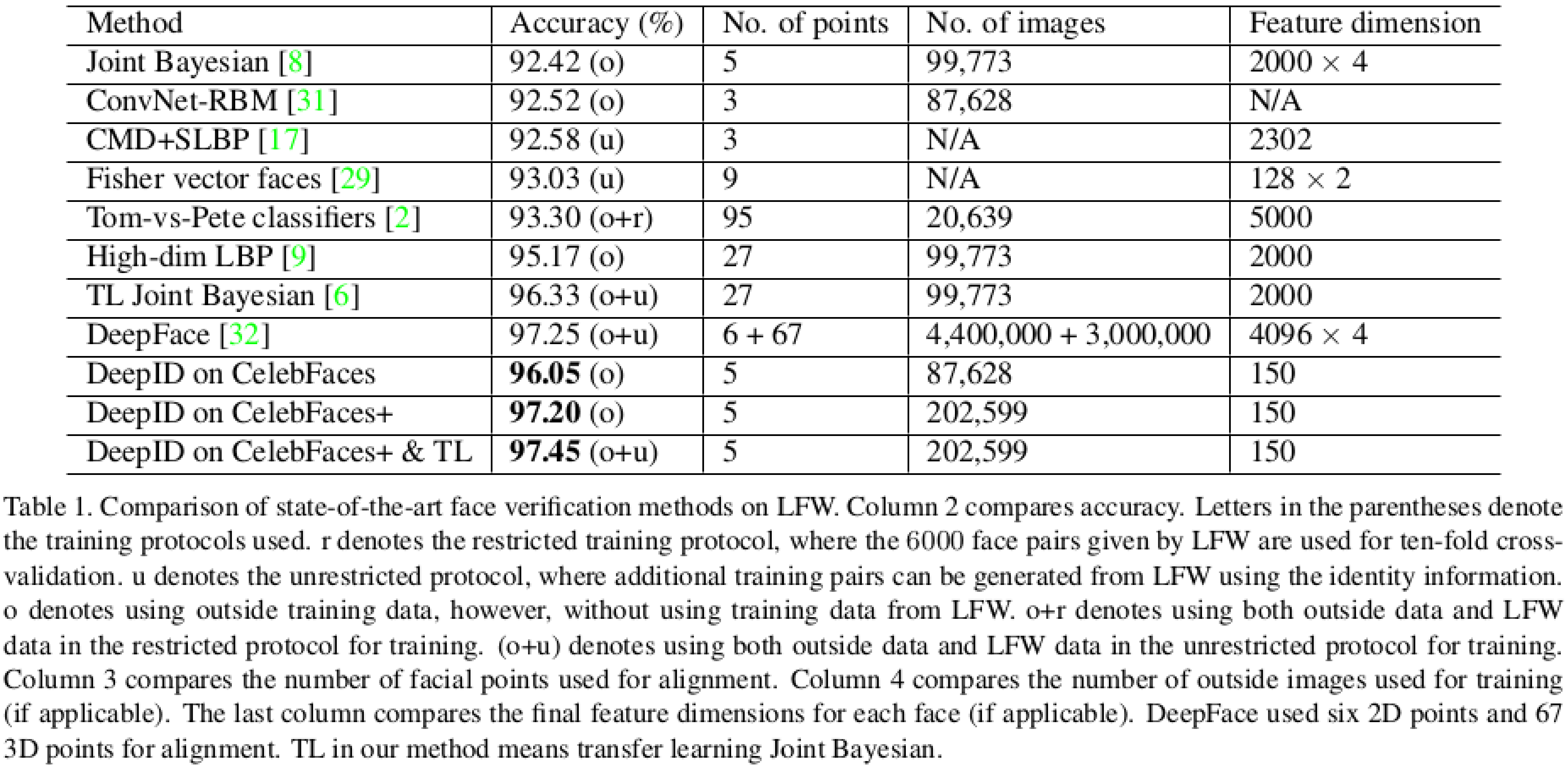


Figure 9. Test accuracy of Joint Bayesian (red line) and neural networks (blue line) using features extracted from 1, 5, 15, 30, and 60 patches. Performance consistently improves with more features. Joint Bayesian is approximately 1.8% better on average than neural networks.

DeepID 1



Method	Accuracy (%)	No. of points	No. of images	Feature dimension
Joint Bayesian [8]	92.42 (o)	5	99,773	2000 × 4
ConvNet-RBM [31]	92.52 (o)	3	87,628	N/A
CMD+SLBP [17]	92.58 (u)	3	N/A	2302
Fisher vector faces [29]	93.03 (u)	9	N/A	128 × 2
Tom-vs-Pete classifiers [2]	93.30 (o+r)	95	20,639	5000
High-dim LBP [9]	95.17 (o)	27	99,773	2000
TL Joint Bayesian [6]	96.33 (o+u)	27	99,773	2000
DeepFace [32]	97.25 (o+u)	6 + 67	4,400,000 + 3,000,000	4096 × 4
DeepID on CelebFaces	**96.05** (o)	5	87,628	150
DeepID on CelebFaces+	**97.20** (o)	5	202,599	150
DeepID on CelebFaces+ & TL	**97.45** (o+u)	5	202,599	150

Table 1. Comparison of state-of-the-art face verification methods on LFW. Column 2 compares accuracy. Letters in the parentheses denote the training protocols used. r denotes the restricted training protocol, where the 6000 face pairs given by LFW are used for ten-fold cross-validation. u denotes the unrestricted protocol, where additional training pairs can be generated from LFW using the identity information. o denotes using outside training data, however, without using training data from LFW. o+r denotes using both outside data and LFW data in the restricted protocol for training. (o+u) denotes using both outside data and LFW data in the unrestricted protocol for training. Column 3 compares the number of facial points used for alignment. Column 4 compares the number of outside images used for training (if applicable). The last column compares the final feature dimensions for each face (if applicable). DeepFace used six 2D points and 67 3D points for alignment. TL in our method means transfer learning Joint Bayesian.

DeepID 2 (NIPS 2014)

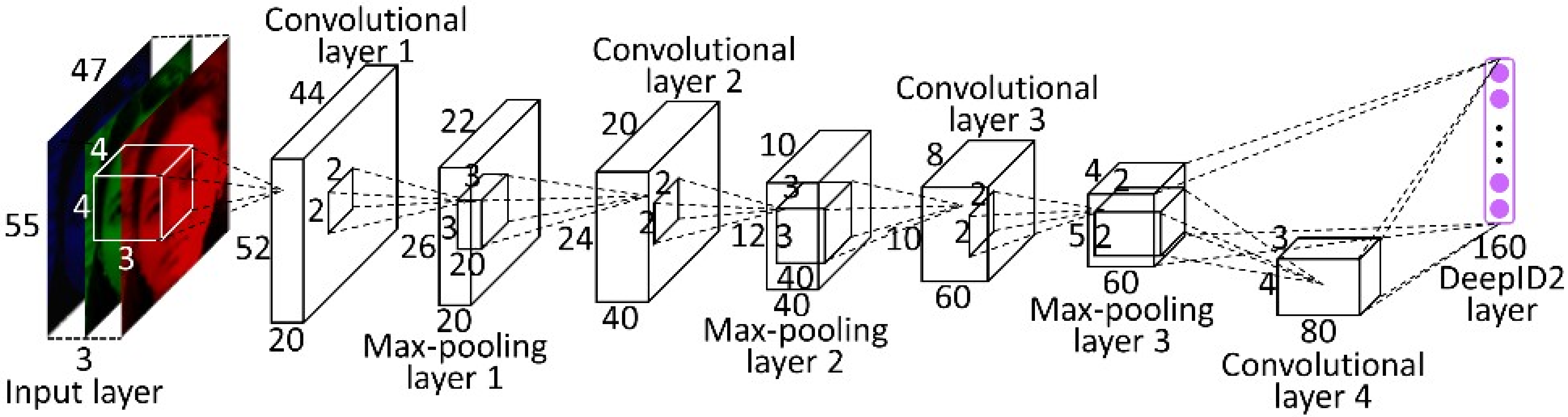


Figure 1: The ConvNet structure for DeepID2 extraction.

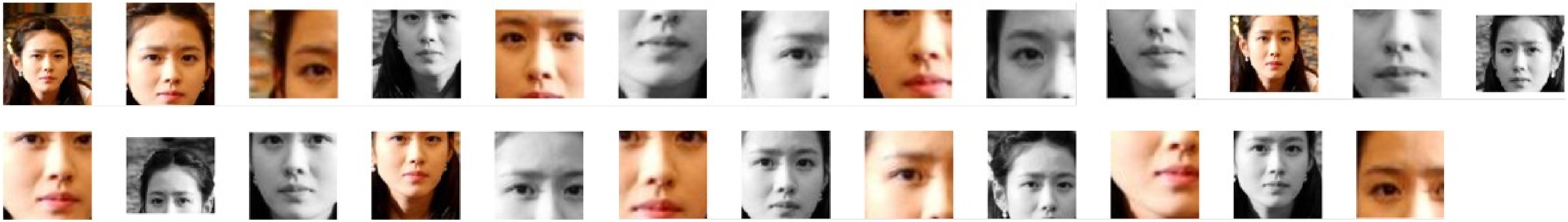


Figure 2: Patches selected for feature extraction.

DeepID 2 (NIPS 2014)

- During training, 200 patches are cropped initially with varying positions, scales, color channels
- Each patch and its horizontal flip are fed into a ConvNet. Two 160 dimensional features are extracted from the patch and its mirror-flip.
- Greedily select best 25 patches (shown below). Discard other models.

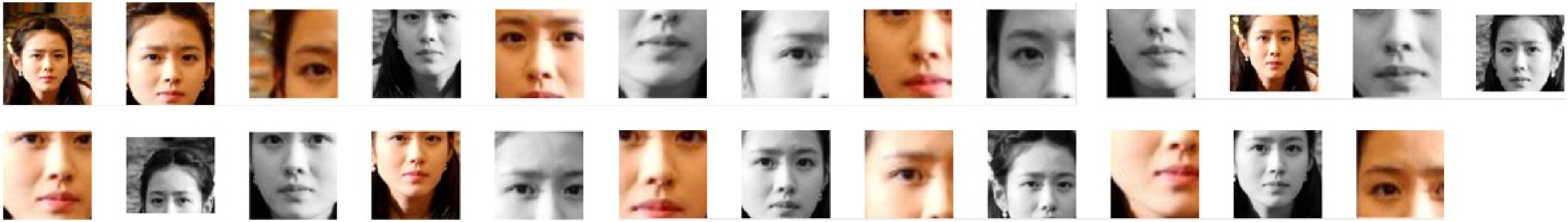
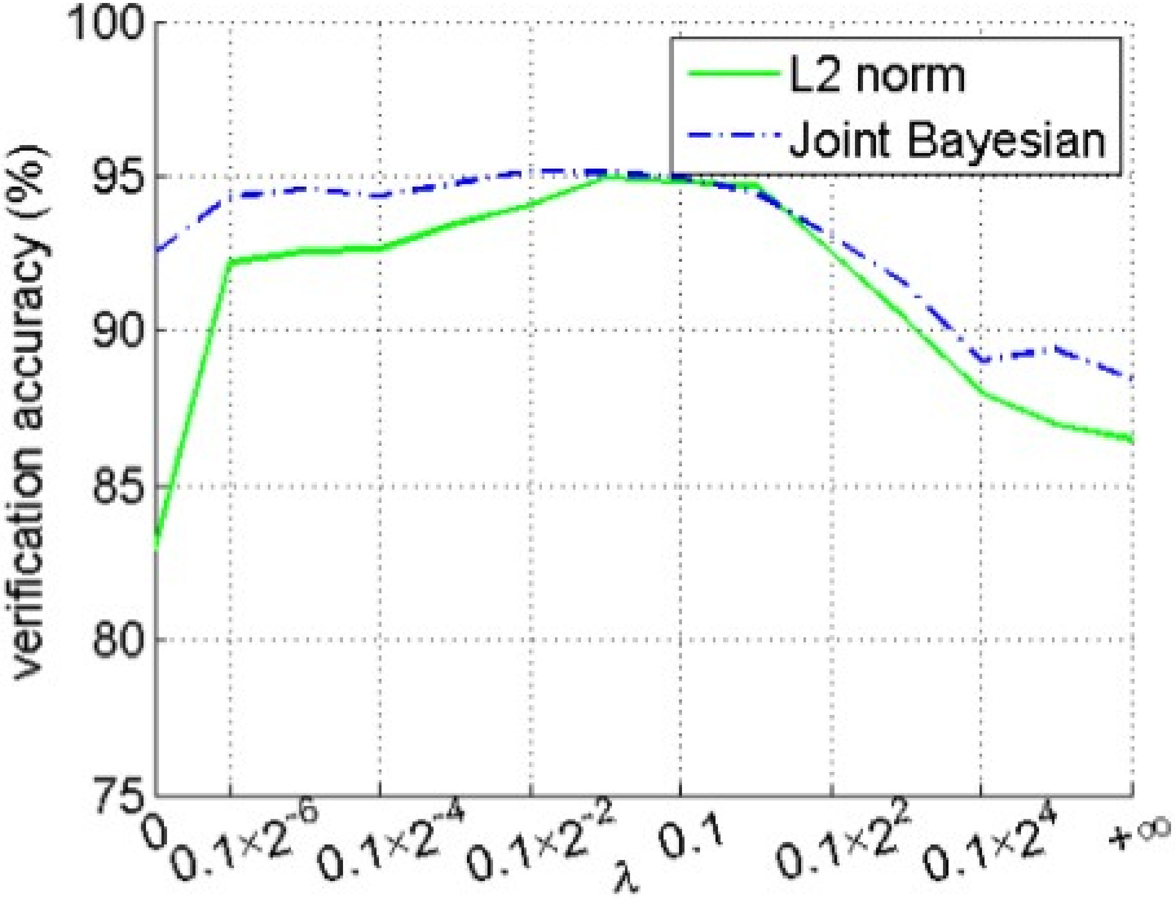


Figure 2: Patches selected for feature extraction.

DeepID 2

- Identification + verification for training CNN



Only identification <------> Only verification

DeepID 2

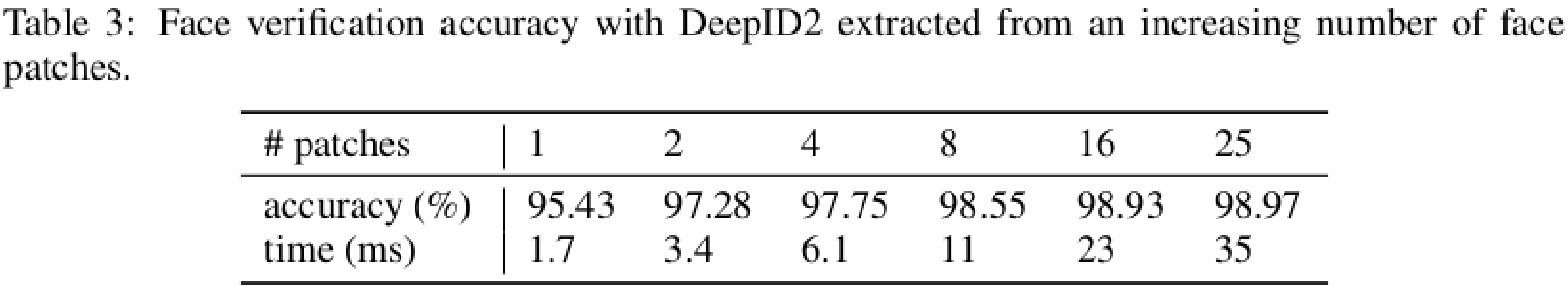


Table 3: Face verification accuracy with DeepID2 extracted from an increasing number of face patches.

# patches	1	2	4	8	16	25
accuracy (%)	95.43	97.28	97.75	98.55	98.93	98.97
time (ms)	1.7	3.4	6.1	11	23	35

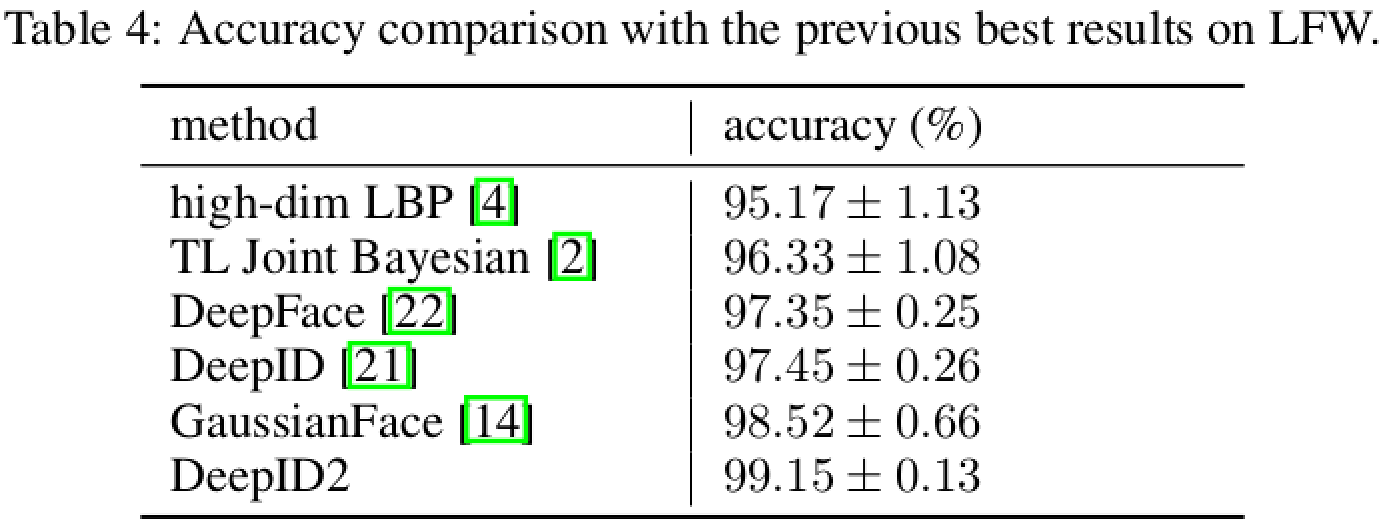


Table 4: Accuracy comparison with the previous best results on LFW.

method	accuracy (%)
high-dim LBP [4]	95.17 ± 1.13
TL Joint Bayesian [2]	96.33 ± 1.08
DeepFace [22]	97.35 ± 0.25
DeepID [21]	97.45 ± 0.26
GaussianFace [14]	98.52 ± 0.66
DeepID2	99.15 ± 0.13

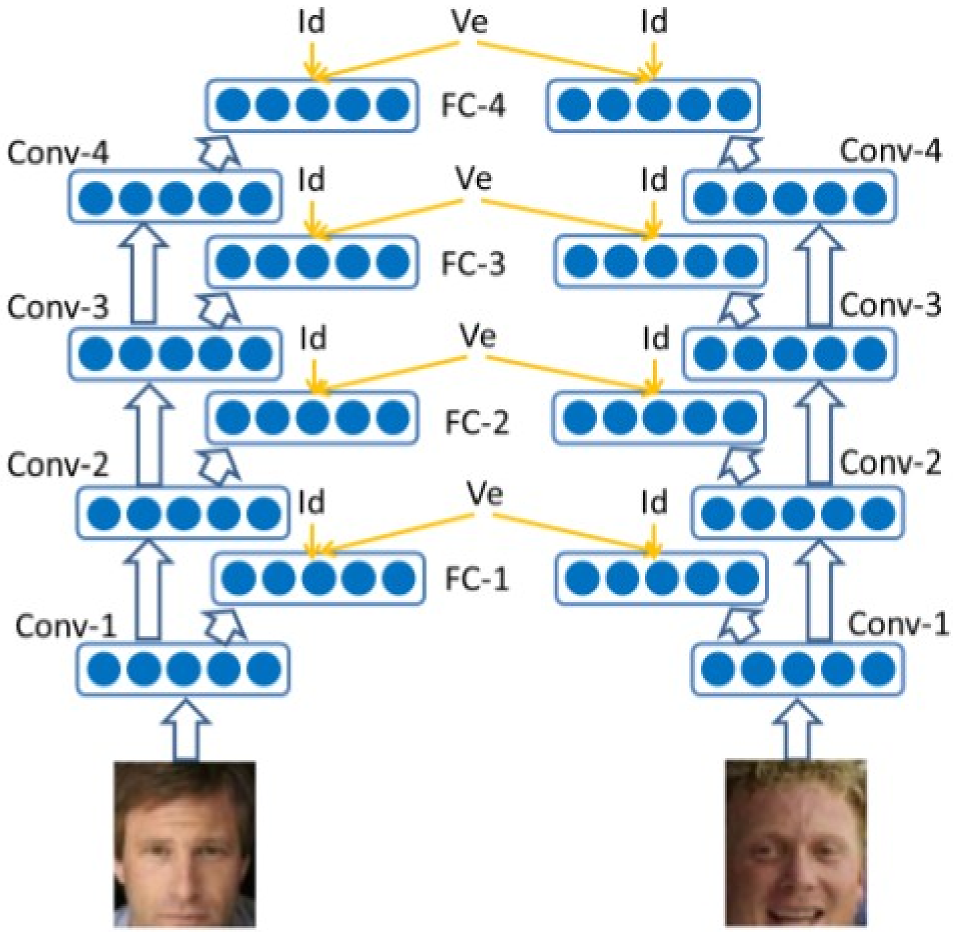
DeepID 2

Summary of differences from DeepID 1:

- Better landmark detector and more landmarks/patches
- Greedy selection of patches (this was even done 7 times when training the ensemble model for the best performance: 98.97% → 99.15%)
- Verification + identification loss. L2 loss seems the best for generating verification signals

DeepID 2+ (arXiv 2014)

- More data (CelebFace + WFRef, both private) =12k ID, 290k images
- Larger network
- Supervision at every layer

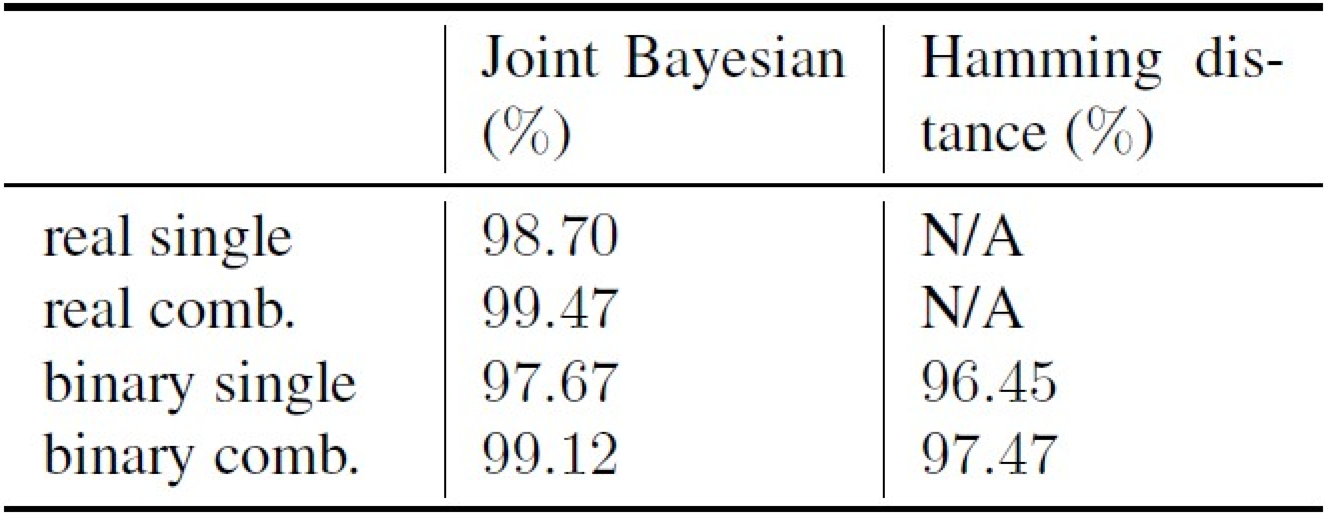


DeepID 2+ (arXiv 2014)

- Properties of the neurons in the DeepID2+ network: there are units tuned to identities (e.g., George W. Bush) and attributes: (male, female, white, black, asian, young, senior, etc.)

DeepID 2+ (arXiv 2014)

- Binary features for faster testing/search



	Joint Bayesian (%)	Hamming distance (%)
real single	98.70	N/A
real comb.	99.47	N/A
binary single	97.67	96.45
binary comb.	99.12	97.47

DeepID 2+

- Occlusion tolerance

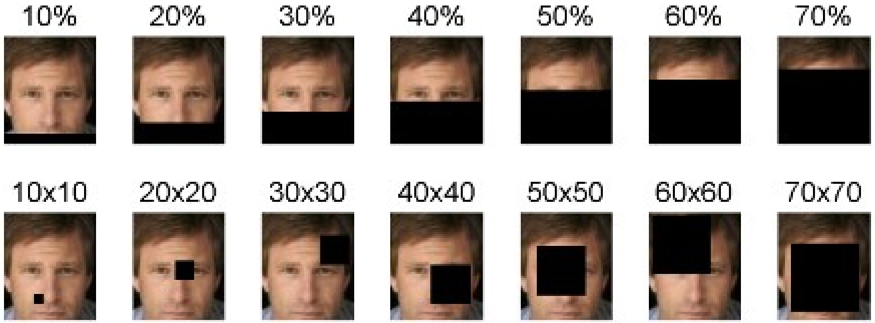
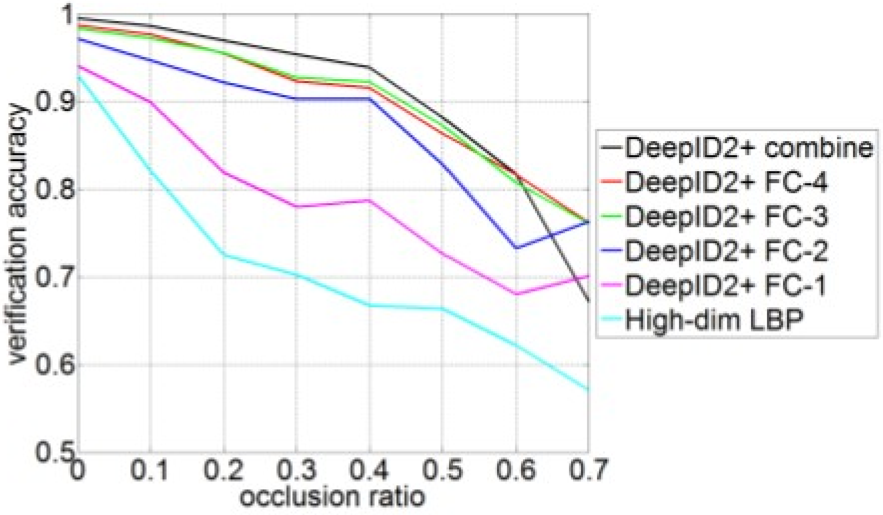
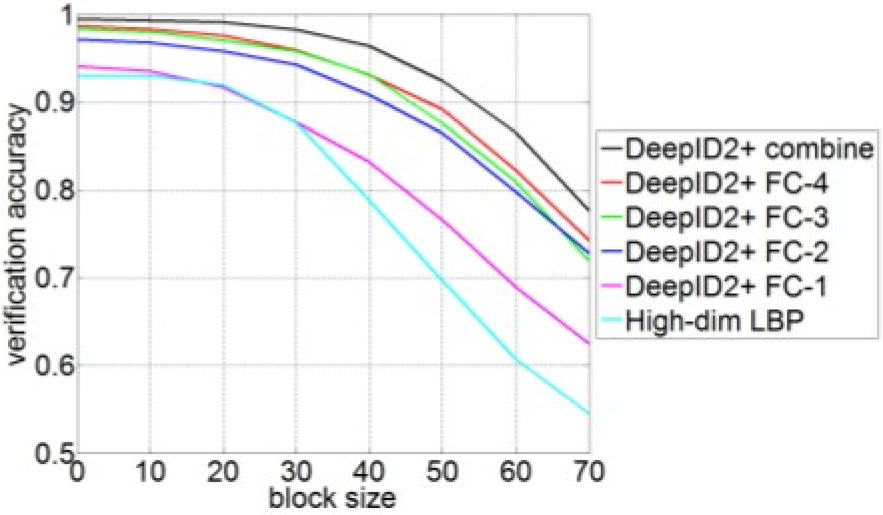


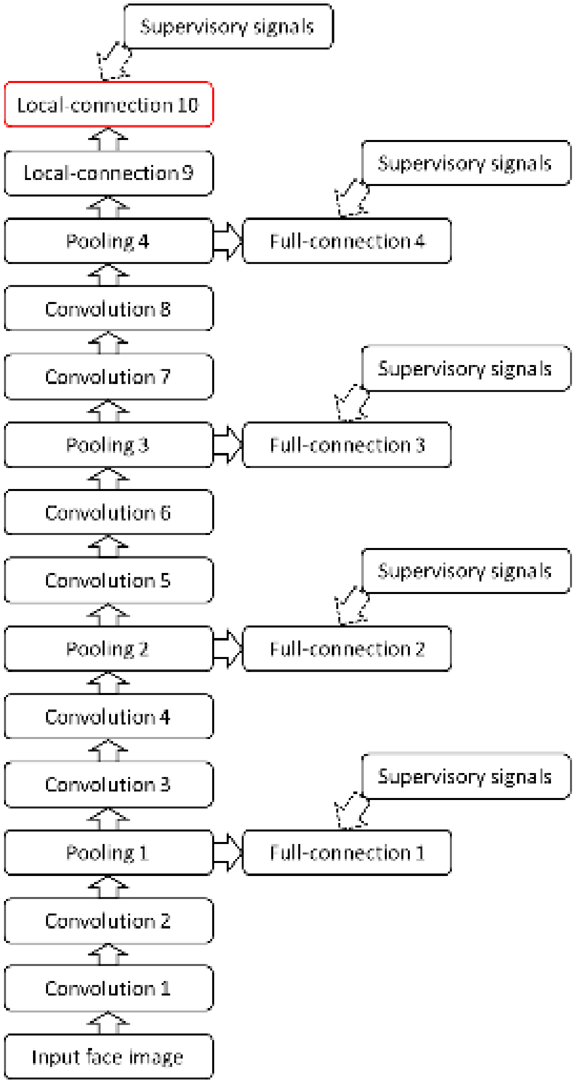
Figure 15: The occluded images tested in our experiments. First row: faces with 10% to 70% areas occluded, respectively. Second row: faces with 10 × 10 to 70 × 70 random block occlusions, respectively.



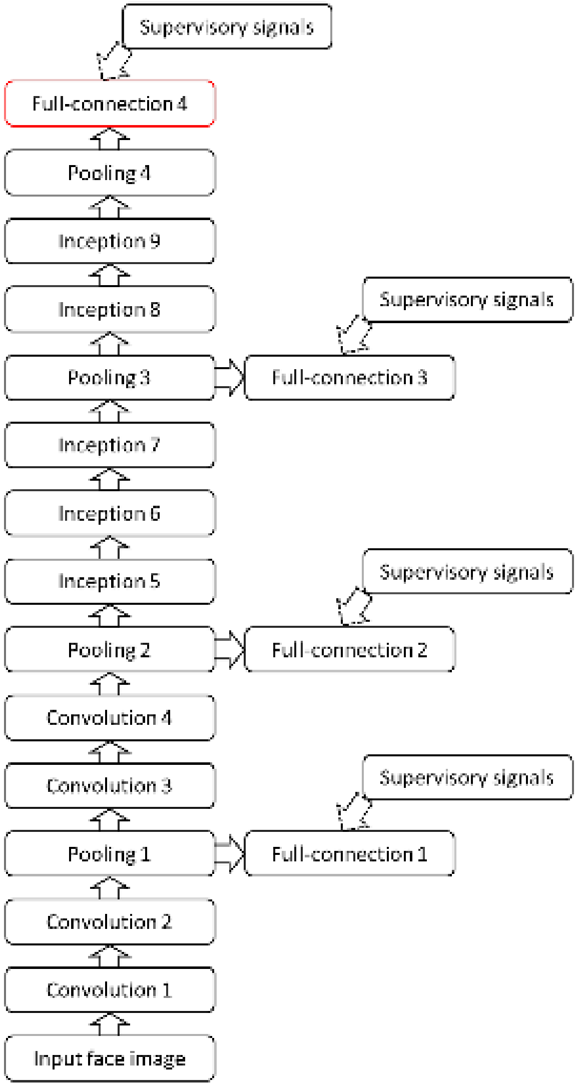


DeepID 3 (arXiv 2015)

- A deeper version of DeepID 2+



DeepID3 net1



DeepID3 net2

Face verification performance of DeepID models

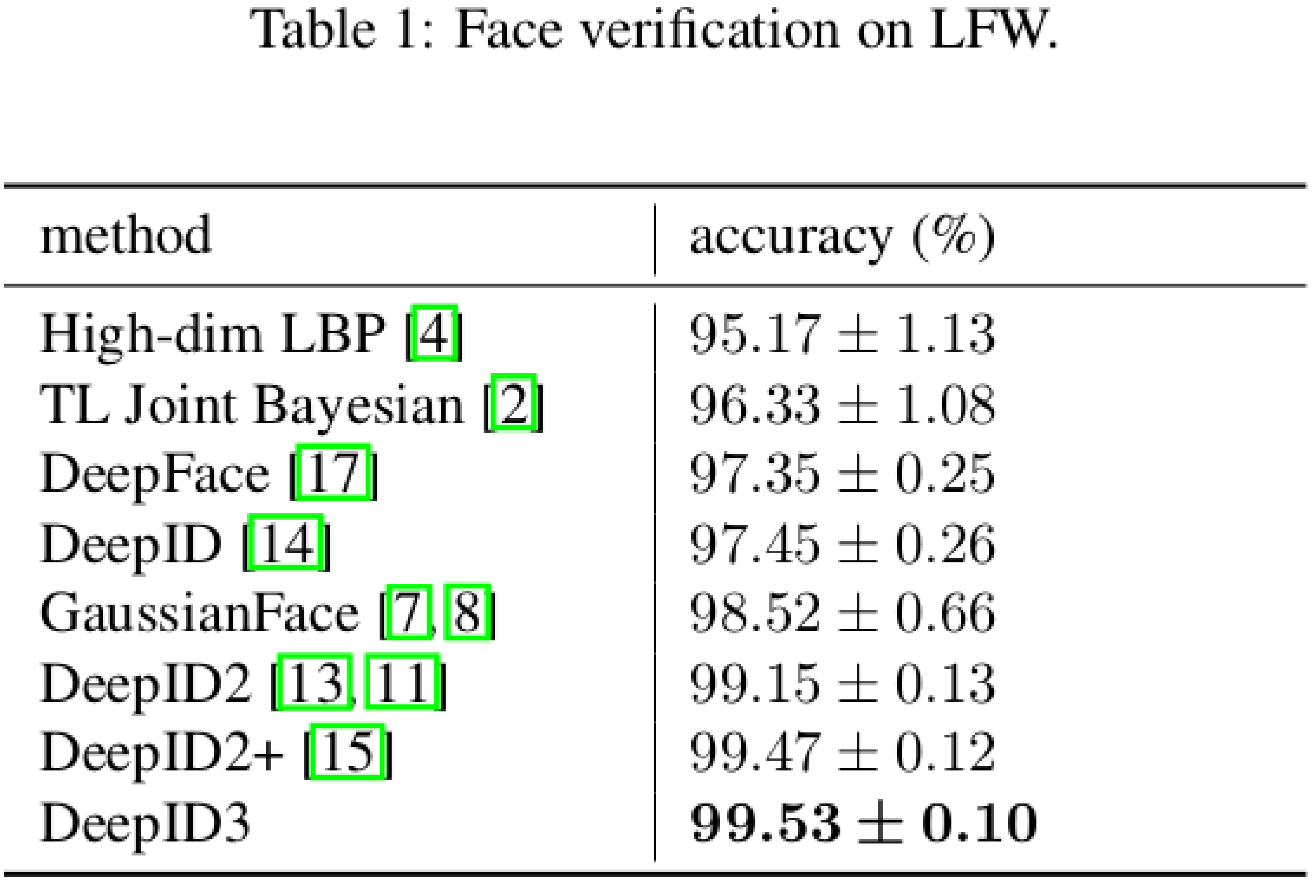


Table 1: Face verification on LFW.

method	accuracy (%)
High-dim LBP [4]	95.17 ± 1.13
TL Joint Bayesian [2]	96.33 ± 1.08
DeepFace [17]	97.35 ± 0.25
DeepID [14]	97.45 ± 0.26
GaussianFace [7, 8]	98.52 ± 0.66
DeepID2 [13, 11]	99.15 ± 0.13
DeepID2+ [15]	99.47 ± 0.12
DeepID3	**99.53 ± 0.10**

Face identification

- Close-set identification

  The gallery set contains 4249 subjects with a single face image per subject, and the probe set contains 3143 face images from the same set of subjects in the gallery.

- Open-set identification

  The gallery set contains 596 subjects with a single face image per subject, and the probe set contains 596 genuine probes and 9494 imposter ones.

Face identification performance of DeepID models

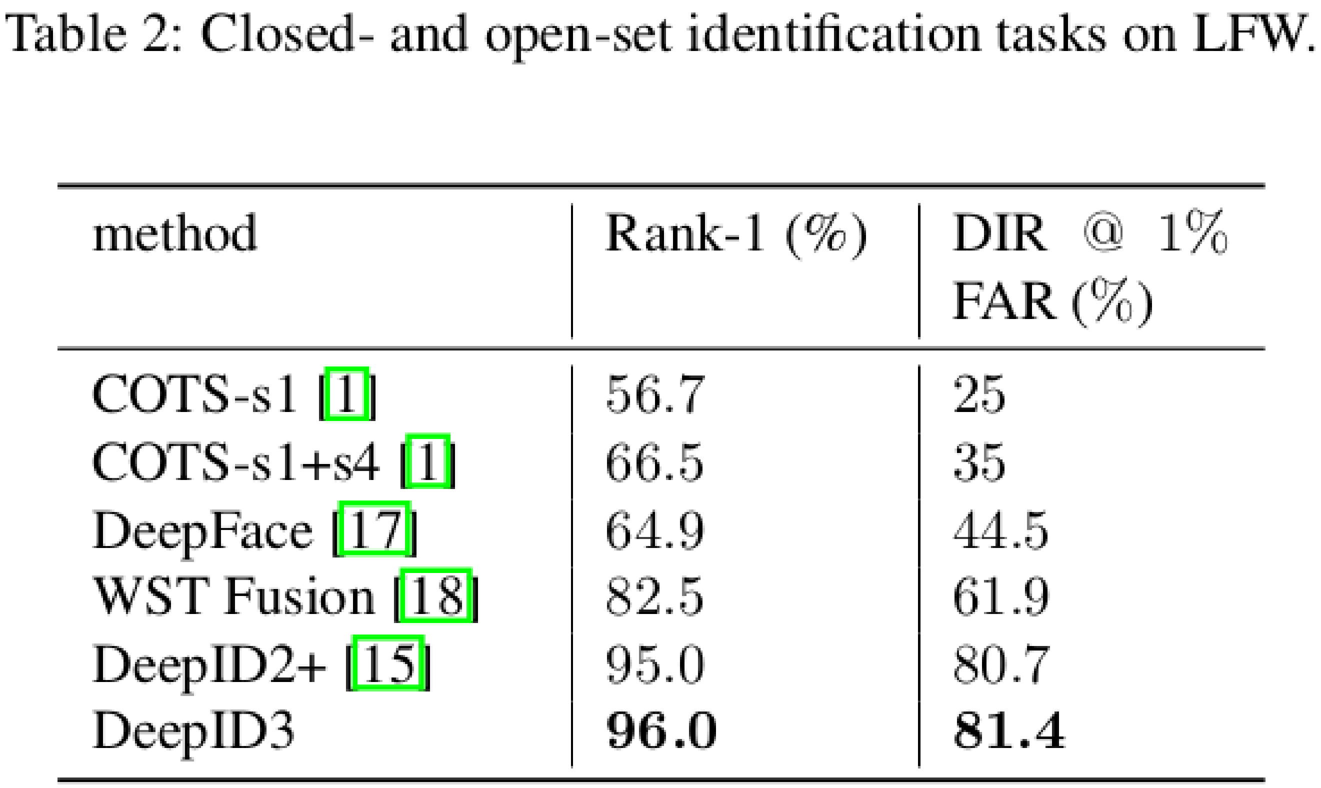


Table 2: Closed- and open-set identification tasks on LFW.

method	Rank-1 (%)	DIR @ 1% FAR (%)
COTS-s1 [1]	56.7	25
COTS-s1+s4 [1]	66.5	35
DeepFace [17]	64.9	44.5
WST Fusion [18]	82.5	61.9
DeepID2+ [15]	95.0	80.7
DeepID3	**96.0**	**81.4**

- What would be the human performance?

DeepFace
(by Facebook, CVPR 2014)

Pros: At the time of publication, it was the best (as good as DeepID 1)

Cons: large dataset; not as good as DeepID 2&3, the latest Face++ and FaceNet; 3D alignment is also somewhat complicated

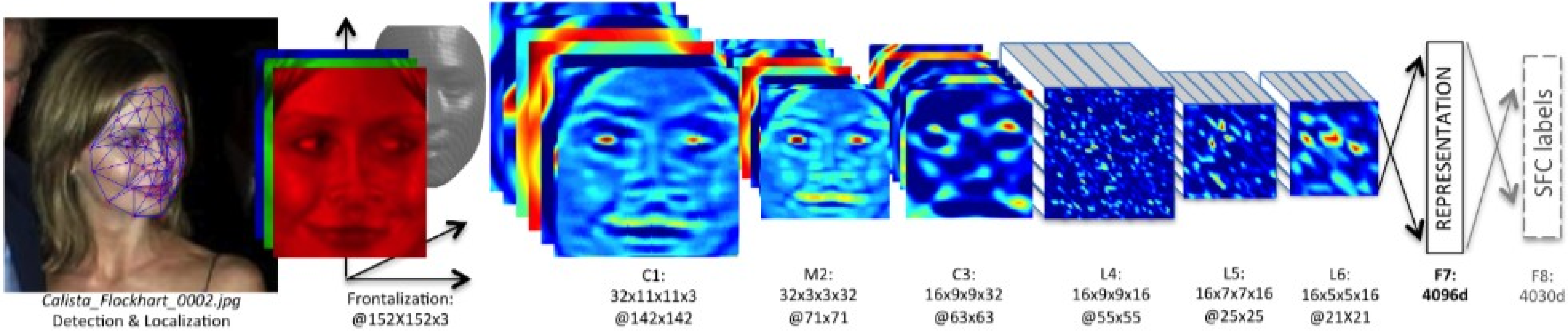
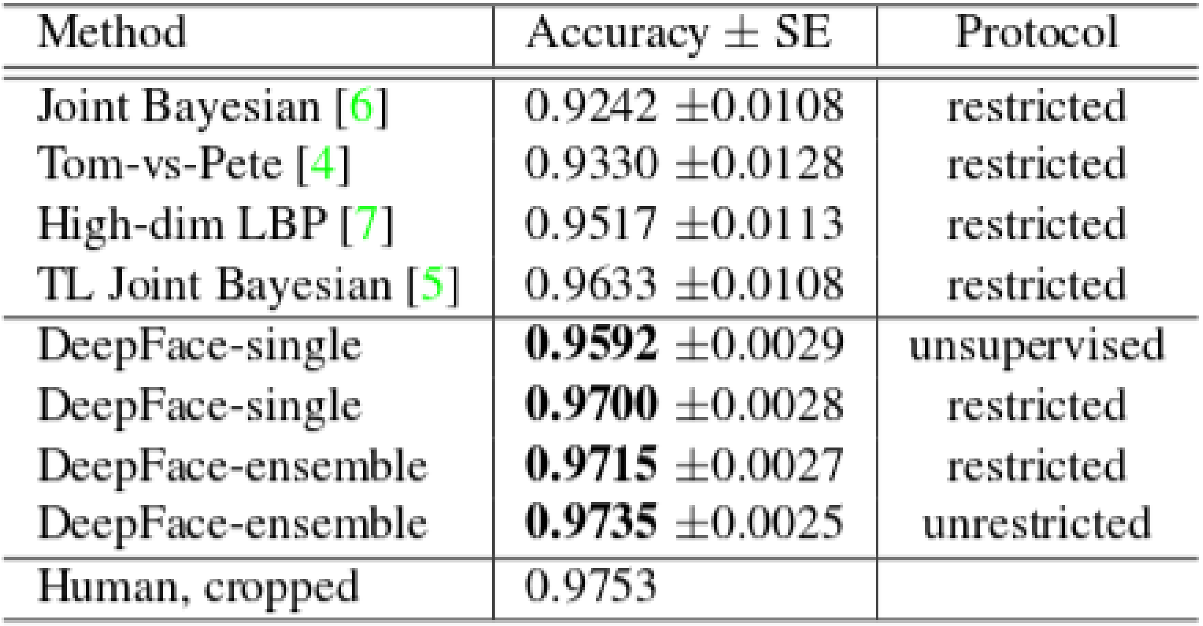


Figure 2. **Outline of the *DeepFace* architecture**. A front-end of a single convolution-pooling-convolution filtering on the rectified input, followed by three locally-connected layers and two fully-connected layers. Colors illustrate feature maps produced at each layer. The net includes more than 120 million parameters, where more than 95% come from the local and fully connected layers.

Performance of DeepFace



Method	Accuracy ± SE	Protocol
Joint Bayesian [6]	0.9242 ±0.0108	restricted
Tom-vs-Pete [4]	0.9330 ±0.0128	restricted
High-dim LBP [7]	0.9517 ±0.0113	restricted
TL Joint Bayesian [5]	0.9633 ±0.0108	restricted
DeepFace-single	**0.9592** ±0.0029	unsupervised
DeepFace-single	**0.9700** ±0.0028	restricted
DeepFace-ensemble	**0.9715** ±0.0027	restricted
DeepFace-ensemble	**0.9735** ±0.0025	unrestricted
Human, cropped	0.9753	

Table 3. Comparison with the state-of-the-art on the *LFW* dataset.

Face++ (the latest one, 2015)

- A naive CNN trained on a large dataset

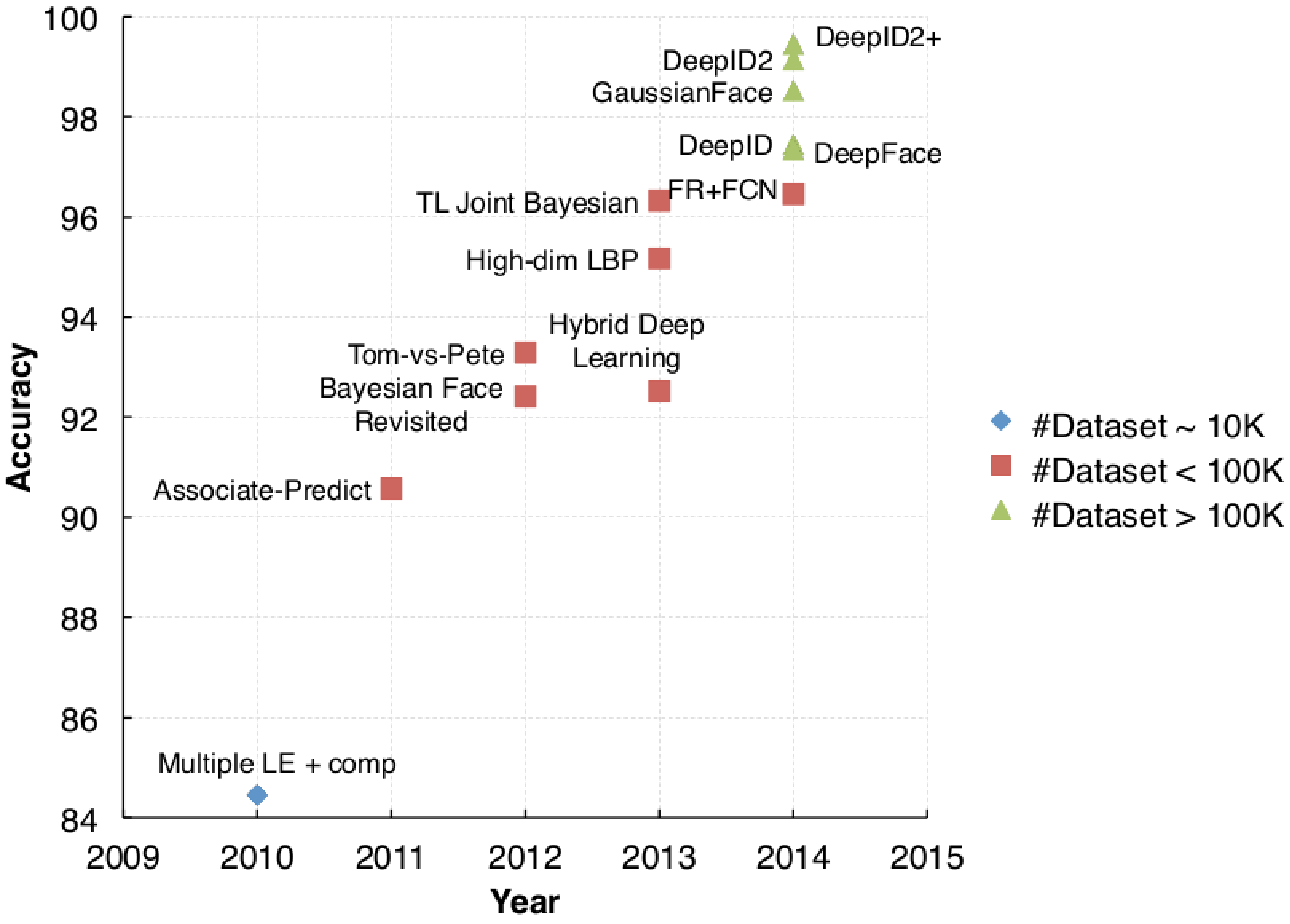
Pros:

- No joint identification and verification
- No 3D alignment
- No Joint Bayesian

Cons:

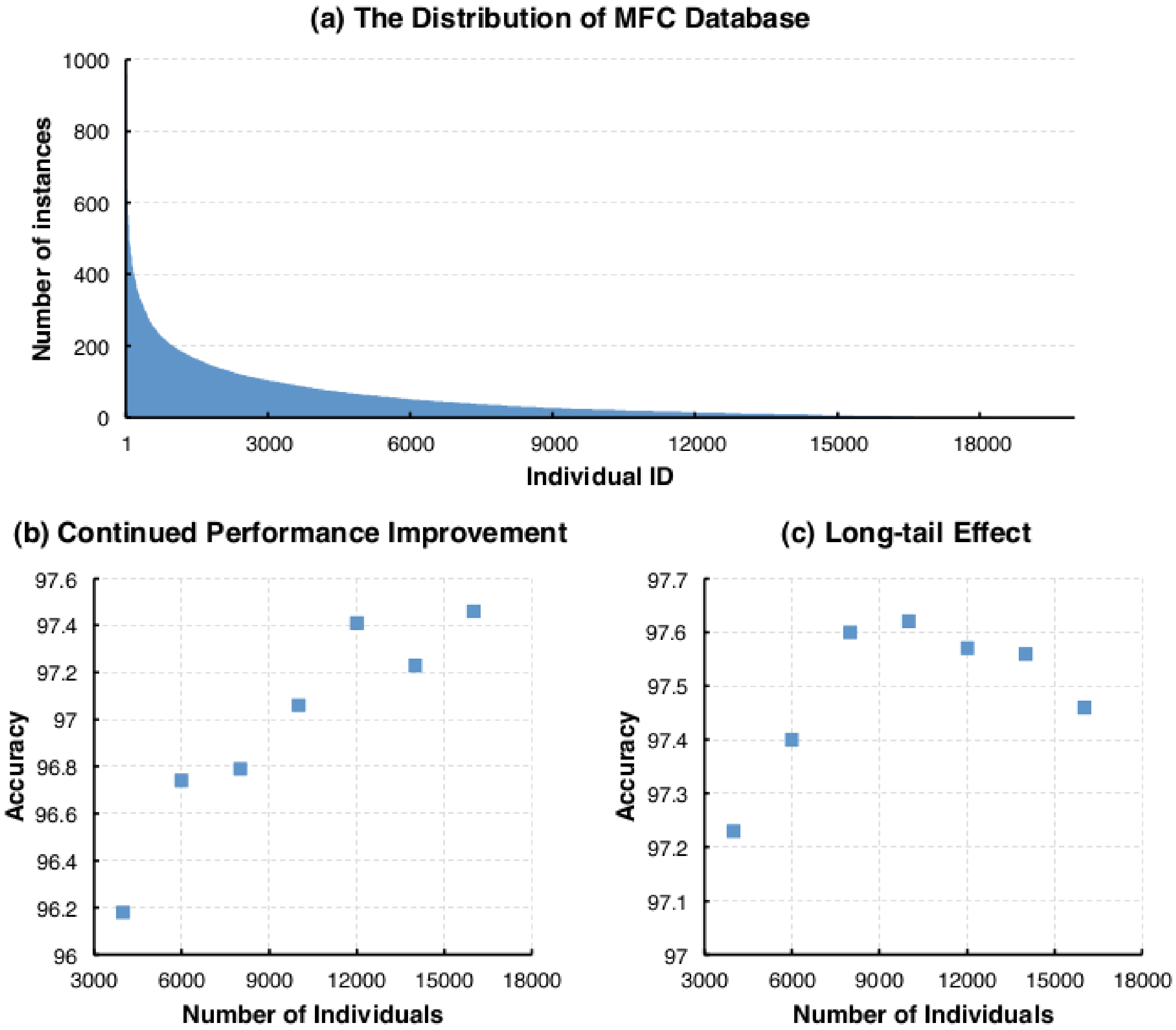
- Trained on a much larger dataset than DeepID

Summarized by Face++



Face++

- Dataset: Megvii Face Classification (MFC) database. It has 5 million labeled faces with about 20,000 individuals. Private.



(a) The Distribution of MFC Database

(b) Continued Performance Improvement

(c) Long-tail Effect

Face++ System

- CNN architecture: “a simple 10 layers deep convolutional neural network”. Details not revealed but they claim the specific choices are not important.

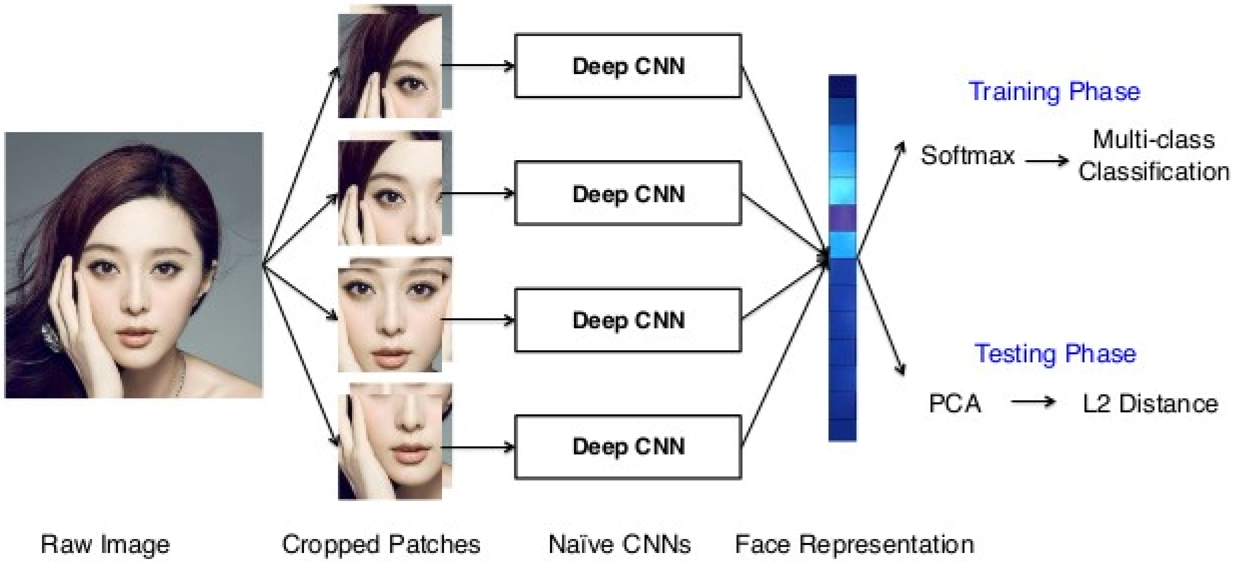
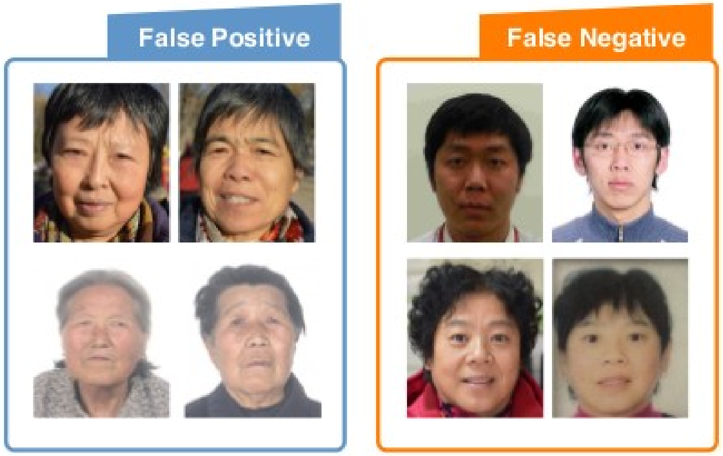


Figure 3. **Overview of Megvii Face Recognition System.** We design a simple 10 layers deep convolutional neural network for recognition. Four face regions are cropped for representation extraction. We train our networks on the MFC database under the traditional multi-class classification framework. In testing phase, a PCA model is applied for feature reduction, and a simple L2 norm is used for measuring the pair of testing faces.

Performance of Face++

- 99.50% on LFW
- Not good enough on a Chinese identification task: 10^-5 FPR, 66% TPR

“Results show that 90% failed cases can be solved by human. There still exists a big gap between machine recognition and human level.”



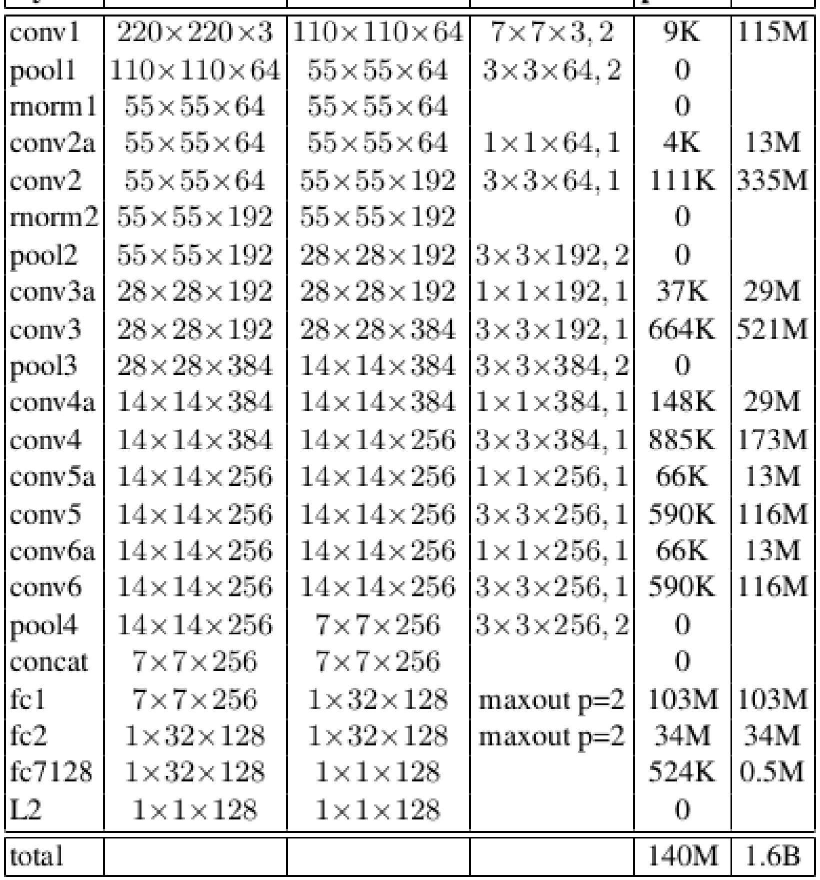
FaceNet (Google 2015)

- Extremely large dataset (260M)
- Very deep model
- Closely cropped, but no alignment other than the crop

- Cons: nobody else has 260M face images!!!

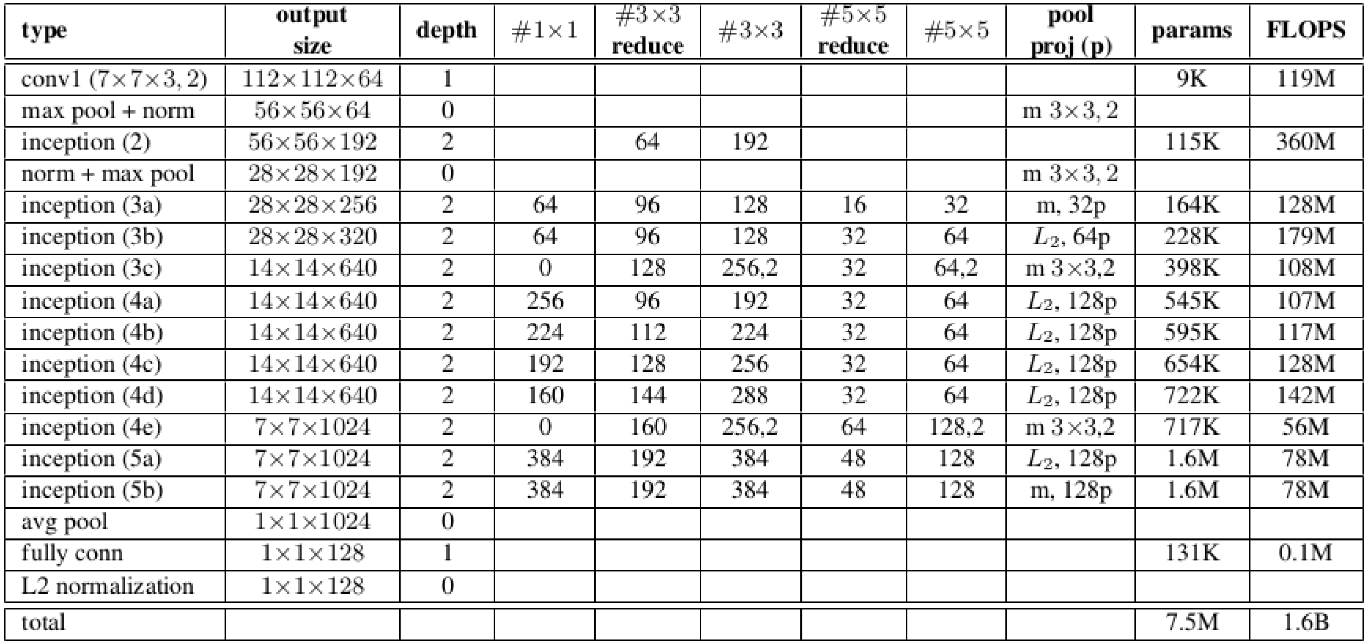
FaceNet two CNN architectures

- Zeiler & Fergus



conv1	220×220×3	110×110×64	7×7×3, 2	9K	115M
pool1	110×110×64	55×55×64	3×3×64, 2	0	
rnorm1	55×55×64	55×55×64		0	
conv2a	55×55×64	55×55×64	1×1×64, 1	4K	13M
conv2	55×55×64	55×55×192	3×3×64, 1	111K	335M
rnorm2	55×55×192	55×55×192		0	
pool2	55×55×192	28×28×192	3×3×192, 2	0	
conv3a	28×28×192	28×28×192	1×1×192, 1	37K	29M
conv3	28×28×192	28×28×384	3×3×192, 1	664K	521M
pool3	28×28×384	14×14×384	3×3×384, 2	0	
conv4a	14×14×384	14×14×384	1×1×384, 1	148K	29M
conv4	14×14×384	14×14×256	3×3×384, 1	885K	173M
conv5a	14×14×256	14×14×256	1×1×256, 1	66K	13M
conv5	14×14×256	14×14×256	3×3×256, 1	590K	116M
conv6a	14×14×256	14×14×256	1×1×256, 1	66K	13M
conv6	14×14×256	14×14×256	3×3×256, 1	590K	116M
pool4	14×14×256	7×7×256	3×3×256, 2	0	
concat	7×7×256	7×7×256		0	
fc1	7×7×256	1×32×128	maxout p=2	103M	103M
fc2	1×32×128	1×32×128	maxout p=2	34M	34M
fc7128	1×32×128	1×1×128		524K	0.5M
L2	1×1×128	1×1×128		0	
total				140M	1.6B

- GoogLeNet



type	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj (p)	params	FLOPS
conv1 (7×7×3, 2)	112×112×64	1							9K	119M
max pool + norm	56×56×64	0						m 3×3, 2		
inception (2)	56×56×192	2		64	192				115K	360M
norm + max pool	28×28×192	0						m 3×3, 2		
inception (3a)	28×28×256	2	64	96	128	16	32	m, 32p	164K	128M
inception (3b)	28×28×320	2	64	96	128	32	64	L_2, 64p	228K	179M
inception (3c)	14×14×640	2	0	128	256,2	32	64,2	m 3×3,2	398K	108M
inception (4a)	14×14×640	2	256	96	192	32	64	L_2, 128p	545K	107M
inception (4b)	14×14×640	2	224	112	224	32	64	L_2, 128p	595K	117M
inception (4c)	14×14×640	2	192	128	256	32	64	L_2, 128p	654K	128M
inception (4d)	14×14×640	2	160	144	288	32	64	L_2, 128p	722K	142M
inception (4e)	7×7×1024	2	0	160	256,2	64	128,2	m 3×3,2	717K	56M
inception (5a)	7×7×1024	2	384	192	384	48	128	L_2, 128p	1.6M	78M
inception (5b)	7×7×1024	2	384	192	384	48	128	m, 128p	1.6M	78M
avg pool	1×1×1024	0								
fully conn	1×1×128	1							131K	0.1M
L2 normalization	1×1×128	0								
total									7.5M	1.6B

FaceNet (Google 2015)

- Loss function: verification only (same as metric learning)
- Why? Too many identities!

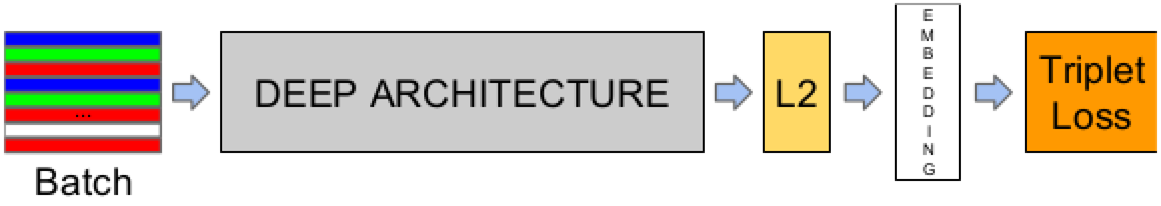


Figure 2. **Model structure.** Our network consists of a batch input layer and a deep CNN followed by L_2 normalization, which results in the face embedding. This is followed by the triplet loss during training.

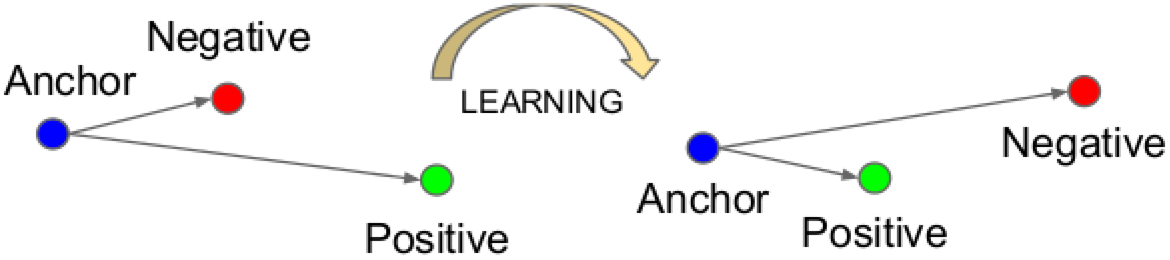
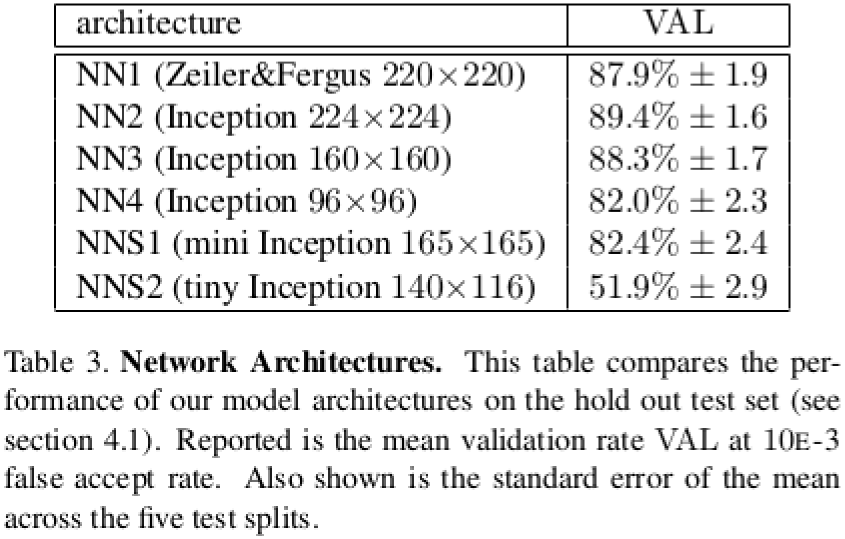


Figure 3. The **Triplet Loss** minimizes the distance between an *anchor* and a *positive*, both of which have the same identity, and maximizes the distance between the *anchor* and a *negative* of a different identity.

Selecting triplets may be tricky

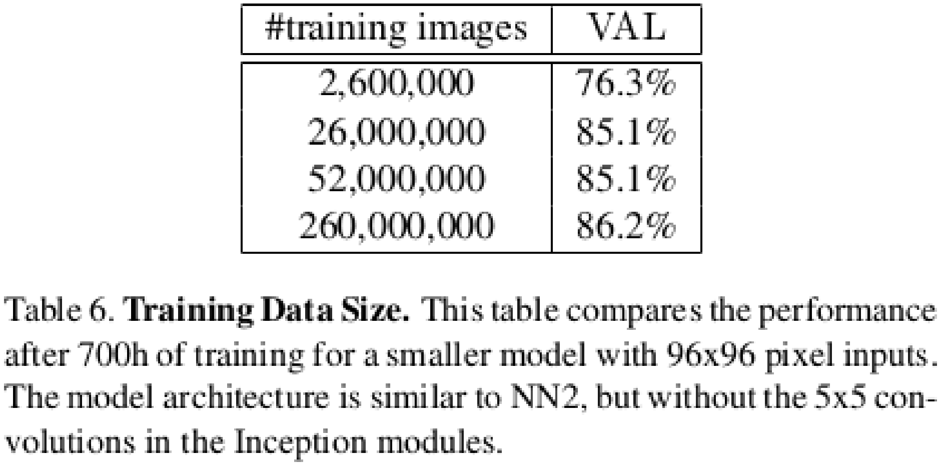
- Select the hard positive/negative exemplars from within a mini-batch by using large mini-batches in the order of a few thousand exemplars and compute the argmin and argmax within a mini-batch.
- Some tricks of selecting “semi-hard” negative exemplars.

FaceNet Performance



architecture	VAL
NN1 (Zeiler&Fergus 220×220)	87.9% ± 1.9
NN2 (Inception 224×224)	89.4% ± 1.6
NN3 (Inception 160×160)	88.3% ± 1.7
NN4 (Inception 96×96)	82.0% ± 2.3
NNS1 (mini Inception 165×165)	82.4% ± 2.4
NNS2 (tiny Inception 140×116)	51.9% ± 2.9

Table 3. **Network Architectures.** This table compares the performance of our model architectures on the hold out test set (see section 4.1). Reported is the mean validation rate VAL at 10E-3 false accept rate. Also shown is the standard error of the mean across the five test splits.



#training images	VAL
2,600,000	76.3%
26,000,000	85.1%
52,000,000	85.1%
260,000,000	86.2%

Table 6. **Training Data Size.** This table compares the performance after 700h of training for a smaller model with 96x96 pixel inputs. The model architecture is similar to NN2, but without the 5x5 convolutions in the Inception modules.

FaceNet Performance

- LFW verification:

  No alignment: 98.87%±0.15

  With alignment: 99.63%±0.09
- Youtube Faces DB: 95.12%±0.39 (state-of-the-art, DeepID 2: 93.2%)

Baidu (2015)

- Multiple patches
- Training data: 1.2M face images from 18K people

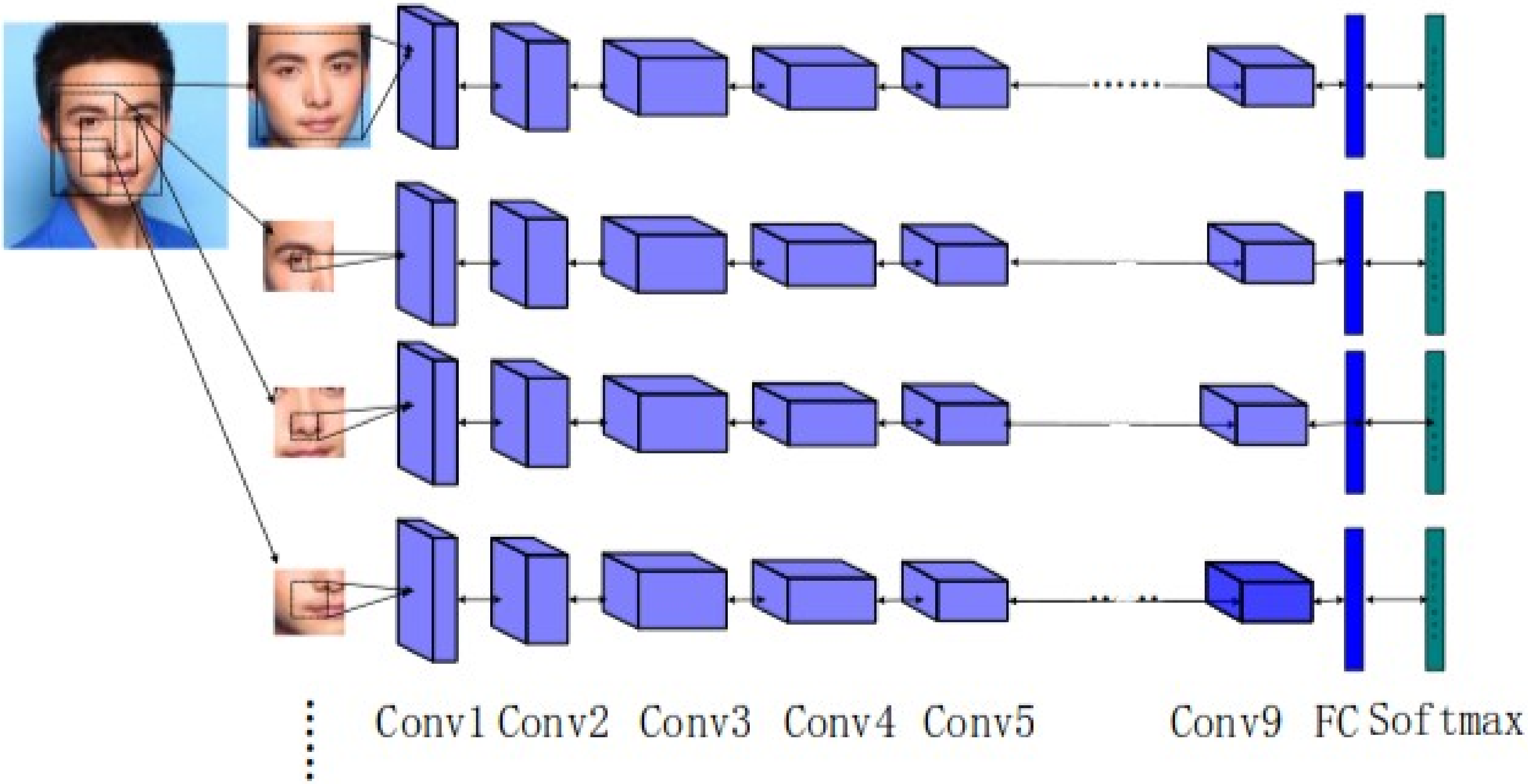
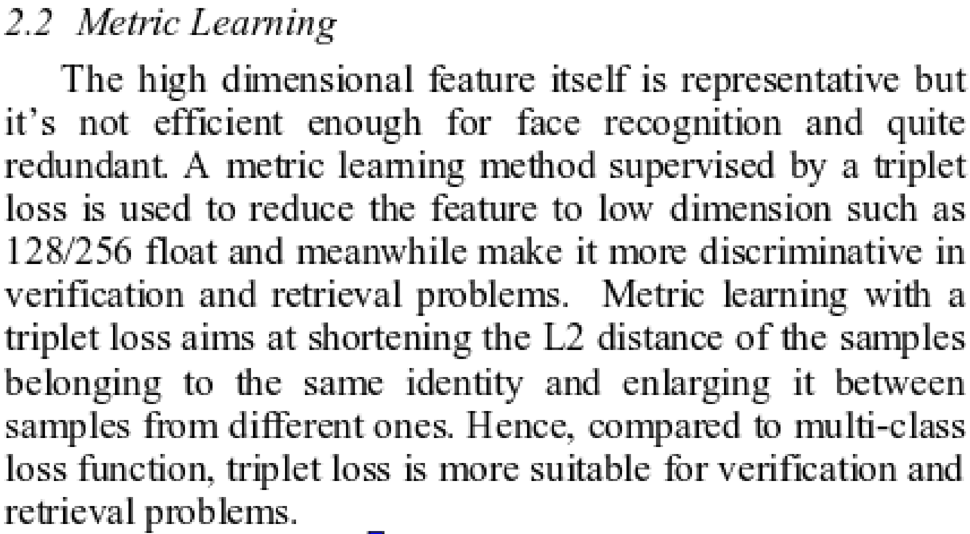


Figure 1. Overview of deep CNN structure on multi-patch.

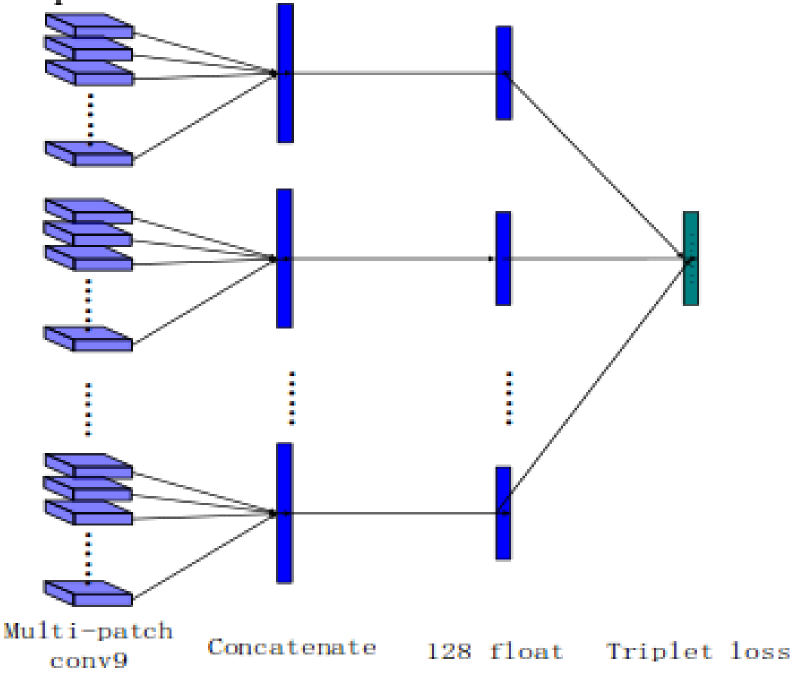
Baidu (2015)

- Loss function



2.2 Metric Learning

The high dimensional feature itself is representative but it's not efficient enough for face recognition and quite redundant. A metric learning method supervised by a triplet loss is used to reduce the feature to low dimension such as 128/256 float and meanwhile make it more discriminative in verification and retrieval problems. Metric learning with a triplet loss aims at shortening the L2 distance of the samples belonging to the same identity and enlarging it between samples from different ones. Hence, compared to multi-class loss function, triplet loss is more suitable for verification and retrieval problems.



Baidu (2015)

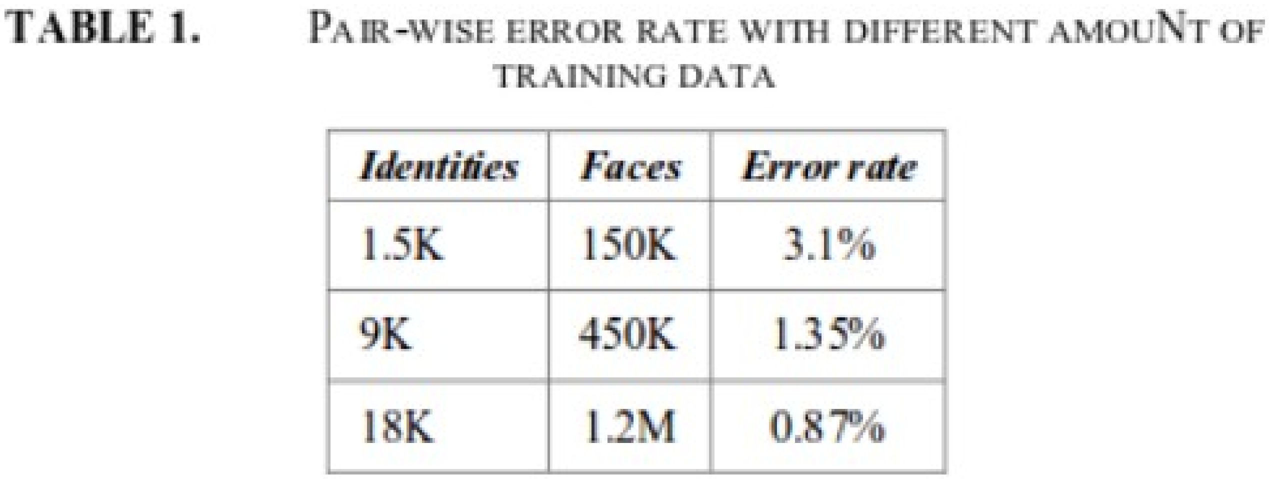


TABLE 1. PAIR-WISE ERROR RATE WITH DIFFERENT AMOUNT OF TRAINING DATA

Identities	*Faces*	*Error rate*
1.5K	150K	3.1%
9K	450K	1.35%
18K	1.2M	0.87%

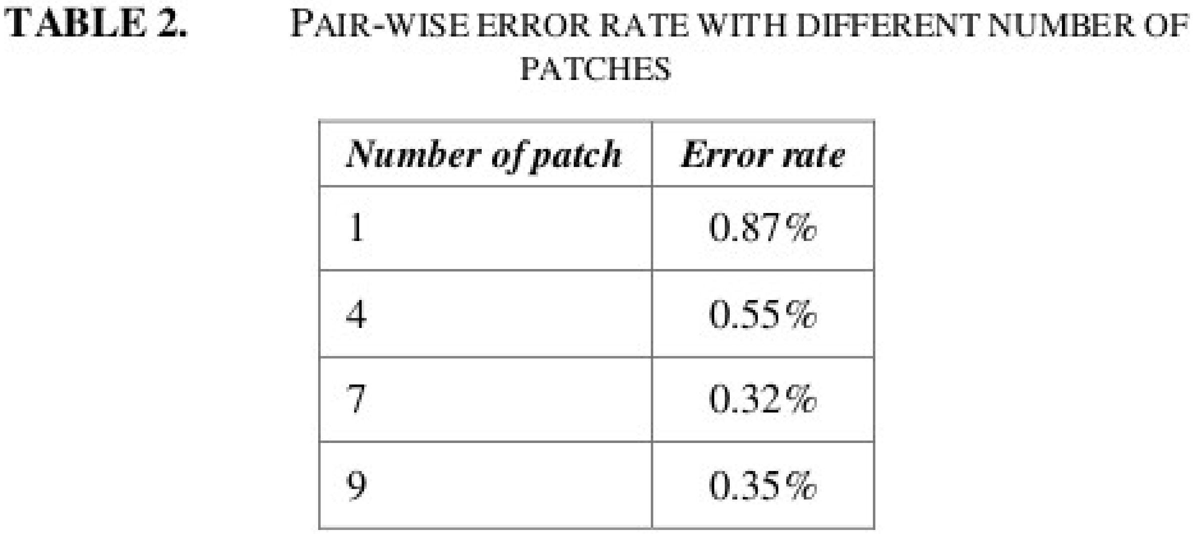


TABLE 2. PAIR-WISE ERROR RATE WITH DIFFERENT NUMBER OF PATCHES

Number of patch	*Error rate*
1	0.87%
4	0.55%
7	0.32%
9	0.35%

TABLE 3. COMPARISONS WITH OTHER METHODS ON SEVERAL EVALUATION TASKS

Method	*Performance on tasks*				
	Pair-wise Accuracy(%)	*Rank-1(%)*	*DIR(%) @ FAR =1%*	*Verification(%)@ FAR=0.1%*	*Open-set Identification(%)@ Rank = 1,FAR = 0.1%*
IDL Ensemble Model	*99.77*	*98.03*	*95.8*	*99.41*	*92.09*
IDL Single Model	*99.68*	*97.60*	*94.12*	*99.11*	*89.08*
FaceNet[12]	*99.63*	*NA*	*NA*	*NA*	*NA*
DeepID3[9]	*99.53*	*96.00*	*81.40*	*NA*	*NA*
Face++[2]	*99.50*	*NA*	*NA*	*NA*	*NA*
Facebook[15]	*98.37*	*82.5*	*61.9*	*NA*	*NA*
Learning from Scratch[4]	*97.73*	*NA*	*NA*	*80.26*	*28.90*
HighDimLBP[10]	*95.17*	*NA*	*NA*	*41.66(reported in [4])*	*18.07(reported in [4])*

Sumnmary

- 1. Dataset is the key: the more training data the better
- 2. Multiple patches
- 3. Joint Bayesian helps
- 4. Alignment helps
- 5. Loss function(s): either verification (metric learning) or identification or both
- 6. Not yet human performance on identification?