Feature Ensemble Networks with Re-ranking for Recognizing Disguised Faces in the Wild

Arulkumar Subramaniam[1], Ajay Narayanan[1] and Anurag Mittal

Indian Institute of Technology Madras (IITM),
Indian Institute of Information Technology Design & Manufacturing Kancheepuram (IIITDM)

December 30, 2021

[1]equal contribution

Overview

Challenges

Challenges in Face recognition task include,

- Natural challenges (as any other CV tasks):
  - Illumination
  - Pose
  - Background Clutter

Challenges

Challenges in Face recognition task include,

- Natural challenges (as any other CV tasks):
  - Illumination
  - Pose
  - Background Clutter
- Subject-specific challenges:
  Intentional or un-intentional disguises such as
  - Wearables like Eye-glasses, Masks, Hats etc.,
  - Make-up
  - Plastic surgery

Prior deep learning approaches in Face Recognition

- FaceNet[2]
  - ZF-Net[3] and GoogleNet[4] architectures with Triplet loss
  - L2 distance comparison
- IR50
  - Extension of SE-ResNet50 architecture with ArcFace loss[5] and Focal loss[6].

[2] Florian Schroff, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 815–823.

[3] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *European conference on computer vision*. Springer. 2014, pp. 818–833.

[4] Christian Szegedy et al. "Going deeper with convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1–9.

[5] Jiankang Deng et al. "ArcFace: Additive Angular Margin Loss for Deep Face Recognition". In: *arXiv preprint arXiv:1801.07698* (2018).

[6] Tsung-Yi Lin et al. "Focal Loss for Dense Object Detection". In: *arXiv:1708.02002* (2017).

- Instead of comparing individual images, what if we **take the neighborhood** of the Gallery (or database) images into account?

[7] Zhun Zhong et al. "Re-ranking person re-identification with k-reciprocal encoding". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1318–1327.

Application of "Re-ranking" to retrieval tasks

- Instead of comparing individual images, what if we **take the neighborhood** of the Gallery (or database) images into account?
- **Re-ranking methods** exploit the neighborhood information among the query and gallery instances

[7] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Application of "Re-ranking" to retrieval tasks

- Instead of comparing individual images, what if we **take the neighborhood** of the Gallery (or database) images into account?
- **Re-ranking methods** exploit the neighborhood information among the query and gallery instances
- Prevalent in retrieval tasks like Person Re-Identification to improve performances in an unsupervised way.

[7] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Application of "Re-ranking" to retrieval tasks

- Instead of comparing individual images, what if we **take the neighborhood** of the Gallery (or database) images into account?
- **Re-ranking methods** exploit the neighborhood information among the query and gallery instances
- Prevalent in retrieval tasks like Person Re-Identification to improve performances in an unsupervised way.
- **k-reciprocal nearest neighbor re-ranking**[7] is popular in retrieval tasks

[7] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

"Re-ranking" intuition

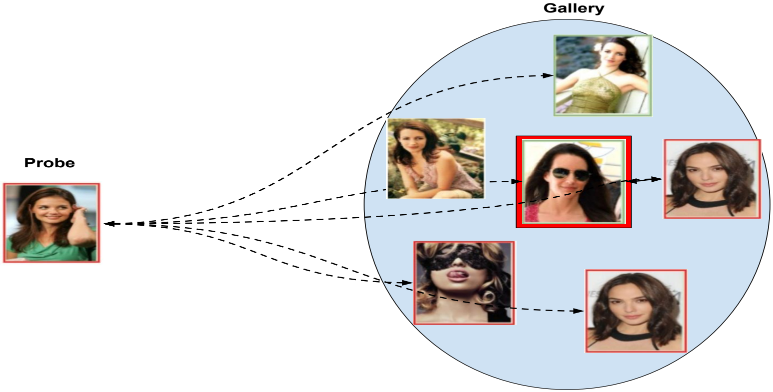

Figure: Probe-to-Gallery comparison

“Re-ranking” intuition

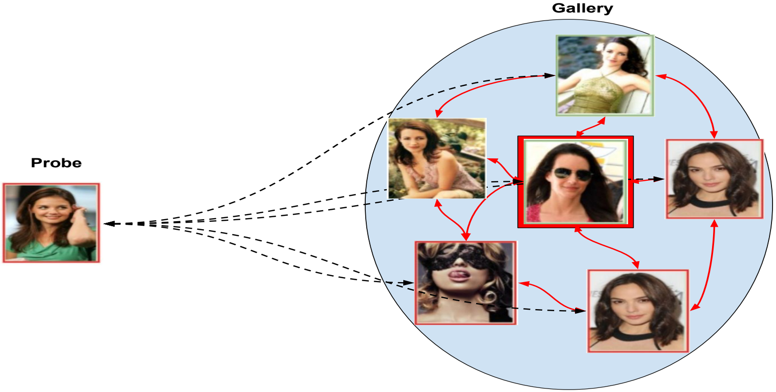

Figure: Probe-to-Gallery comparison and exploit **neighborhood within gallery**

Contributions

Our contributions are as follows:

- We propose a **F**eature **E**nsem**B**le **Net**work (FEBNet)- an ensemble of multiple state-of-the-art face recognition networks
- Two loss functions
  - Impersonator Triplet loss
  - Category loss
- Usage of re-ranking strategy

Feature Ensemble Network (FEBNet) Pipeline

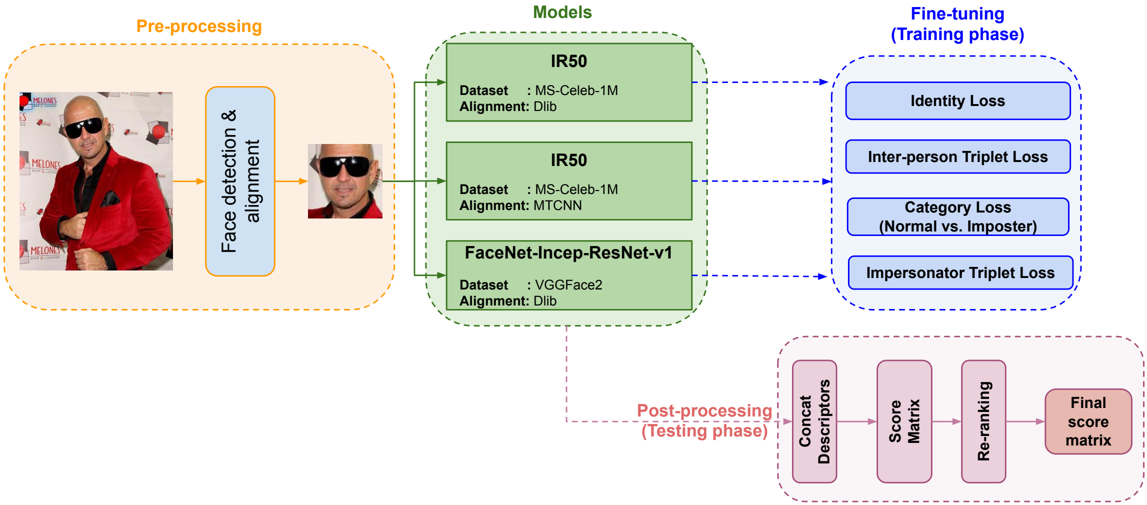

Figure: Illustration of FEBNet model pipeline

Pre-processing & Base models

We use two methods for landmark detection and alignment:

- dlib[8]
- MTCNN[9]

Three pretrained base models:

- $\mathbf{IR50}_D$ = IR50[10] + dlib (pre-processing)
- $\mathbf{IR50}_M$ = IR50 + MTCNN (pre-processing)
- FaceNet-Incep-ResNet-v1[11]

[8] Davis E. King. "Dlib-ml: A Machine Learning Toolkit". In: *Journal of Machine Learning Research* 10 (2009), pp. 1755–1758.

[9] K. Zhang et al. "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks". In: *IEEE Signal Processing Letters* 23.10 (Oct. 2016), pp. 1499–1503. ISSN: 1070-9908. DOI: 10.1109/LSP.2016.2603342.

[10] Jian Zhao. *High-Performance Face Recognition Library on PyTorch*. https://github.com/ZhaoJ9014/face.evoLVe.PyTorch. 2018.

[11] Szegedy et al., "Going deeper with convolutions"; Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

IR50:

- an extension of SE-ResNet50[12] model
- pretrained on MS-Celeb-1M[13] dataset
- **pretraining objective functions:** ArcFace loss[14] and Focal loss[15]

FaceNet-Incep-ResNet-v1:

- Inception model with residual connections
- **pretraining datasets**: "VGGFace2"[16]
- **pretraining objective functions:** person classification loss (cross-entropy) & Triplet loss

[12] Jie Hu, Li Shen, and Gang Sun. "Squeeze-and-excitation networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7132–7141.

[13] Adam Harvey and Jules LaPlace. *MegaPixels: Origins, Ethics, and Privacy Implications of Publicly Available Face Recognition Image Datasets*. 2019. URL: https://megapixels.cc/ (visited on 04/18/2019).

[14] Deng et al., "ArcFace: Additive Angular Margin Loss for Deep Face Recognition".

[15] Lin et al., "Focal Loss for Dense Object Detection".

[16] Q. Cao et al. "VGGFace2: A dataset for recognising faces across pose and age". In: *International Conference on Automatic Face and Gesture Recognition*. 2018.

Performance of base models before fine-tuning

Models	GAR[17]					
	@1%FAR[18]			@0.1%FAR		
	Protocol			Protocol		
	1	2	4	1	2	4
$IR50_D$	96.47	80.42	80.73	44.70	70.32	69.85
$IR50_M$	67.58	79.22	81.27	40.83	72.62	70.61
FaceNet	79.83	72.48	72.61	45.04	50.15	49.17

Table: Performance of base models without fine-tuning on training dataset

[17] GAR = Genuine Acceptance Rate
[18] FAR = False Acceptance Rate

Objective functions

The pretrained base models are fine-tuned using training dataset[19] with the aid of four objective functions as follows:

- Identity Loss
- Inter-person Triplet Loss
- Category Loss
- Impersonator Triplet Loss

[19] Maneet Singh et al. "Recognizing Disguised Faces in the Wild". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science, Volume 1, No. 2.* 2019, pp. 97–108.

Objective functions

- **Cross-entropy loss $\mathbf{L}_{id}$:** loss between the softmax probabilty output p_i from the model and the target identity.

$$L_{id} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} t_{ij} \log p_{ij} \tag{1}$$

  Here, N = number of face images in the mini-batch, M = number of identities in train-set.

- **Inter-person Triplet Loss $\mathbf{L}_{trip}$:** To promote small intra-class distance and high inter-class distance.

$$L_{trip} = \frac{1}{N}\sum_{i=1}^{N} max(0, d(I_i, I_{i+}) - d(I_i, I_{i-}) + m) \tag{2}$$

  Here m = margin parameter, $d(i,j)$ = distance between embeddings i & j (Here, we use Euclidean distance).

Objective functions

- **Category Loss $\mathbf{L}_{cat}$:** To discriminate the impersonator images of the identities. Two classes namely 1) Normal-validation-disguise class, 2) Impersonator class.

$$L_{cat} = -y \log p - (1 - y) \log(1 - p) \tag{3}$$

- **Impersonator Triplet Loss $\mathbf{L}_{imp}$:** loss to distinguish a particular identity from it's impersonator.

$$L_{imp} = \frac{1}{N} \sum_{i=1}^{N} max(0, d(I_i, I_{i+}) - d(I_i, I_{imp}) + m) \tag{4}$$

Here m = margin parameter, $d(i, j)$ = distance between embeddings i & j (In this paper, Euclidean distance).

Overall Objective function

The overall objective function/total loss is given by:

$$L = \gamma_1 L_{id} + \gamma_2 L_{trip} + \gamma_3 L_{imp} + \gamma_4 L_{cat} \tag{5}$$

The ratios $\gamma_1 = 1.0, \gamma_2 = 0.5, \gamma_3 = 0.1, \gamma_4 = 0.01$ are selected using validation set.

Ensemble method

- L2-normalized feature vectors are extracted from the base models independently
- Concatenate them to get the final feature descriptor
- Euclidean distance to get distance matrix
- Apply Re-ranking[20] to get the final distance matrix.

[20] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Performance of ensemble of fine-tuned models

Architecture			GAR					
			@1%FAR			@0.1%FAR		
			Protocol			Protocol		
$IR50_D$	$IR50_M$	FaceNet	1	2	4	1	2	4
		✓	80.33	73.80	74.37	45.37	52.57	51.87
	✓		66.38	81.81	82.27	05.71	73.87	72.97
	✓	✓	91.93	83.11	83.50	52.77	71.86	70.07
✓			93.94	83.16	83.37	48.40	70.12	69.05
✓		✓	93.61	84.30	84.44	53.10	71.24	69.66
✓	✓		94.62	85.42	85.56	53.44	75.07	73.72
✓	✓	✓	**95.79**	**86.19**	**86.25**	**56.30**	**75.25**	**73.42**

Table: Performance of various configurations of ensemble architectures

Analysis of objective functions

Losses		GAR					
		@1%FAR			@0.1%FAR		
		Protocol			Protocol		
L_{cat}	L_{imp}	1	2	4	1	2	4
		95.46	86.22	**86.42**	54.95	75.10	73.33
	✓	95.79	**86.37**	86.34	54.11	75.13	73.37
✓		95.12	86.31	86.39	55.63	75.16	73.29
✓	✓	**95.79**	86.19	86.25	**56.30**	**75.25**	**73.42**

Table: Performance comparison of various configurations of ensemble architectures with the proposed objective functions: Impersonator Triplet loss (L_{imp}), Category loss (L_{cat})

Re-ranking[21]

Input: Calculated distance matrix D_{orig} $(Q \times G)$,
Q = number of query images, G = number of gallery images

[21] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Re-ranking[21]

Input: Calculated distance matrix D_{orig} $(Q \times G)$,
Q = number of query images, G = number of gallery images
Steps:

1. k-reciprocal nearest neighbors pruning:
   - Only keep the gallery entries which are reciprocal k-reciprocal (hyper parameter $= k_1$) neighbor to the probe

[21] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Re-ranking[21]

Input: Calculated distance matrix D_{orig} $(Q \times G)$,
Q = number of query images, G = number of gallery images

Steps:

1. k-reciprocal nearest neighbors pruning:
   - Only keep the gallery entries which are reciprocal k-reciprocal (hyper parameter = k_1) neighbor to the probe
2. New Feature formulation
   - For each image (probe (query) and gallery), formulate a G-dim descriptor

$$f_{p,g_i} = \begin{cases} e^{-d(p,g_i)} & \text{if } g_i \in \text{k1-NN of probe } p \\ 0 & \text{otherwise} \end{cases}$$

[21] Zhong et al., "Re-ranking person re-identification with k-reciprocal encoding".

Re-ranking - continuation

3. local query expansion
   - each image's feature is approximated by

$$f_p = \frac{1}{k_2} \sum_{i=0}^{k_2} f_{NN_i}$$

4. Jaccard distance (D_{jac}) calculation

$$d_{jac}(p, g_i) = 1 - \frac{\sum_{j=1}^{N} \min(f_{(p,g_j)}, f_{(g_i,g_j)})}{\sum_{j=1}^{N} \max(f_{(p,g_j)}, f_{(g_i,g_j)})}$$

Re-ranking - continuation

3. local query expansion
   - each image's feature is approximated by

$$f_p = \frac{1}{k_2} \sum_{i=0}^{k_2} f_{NN_i}$$

4. Jaccard distance (D_{jac}) calculation

$$d_{jac}(p, g_i) = 1 - \frac{\sum_{j=1}^{N} \min(f_{(p,g_j)}, f_{(g_i,g_j)})}{\sum_{j=1}^{N} \max(f_{(p,g_j)}, f_{(g_i,g_j)})}$$

Essentially, it is similar to,

$$d_{jac}(p, g_i) = 1 - \frac{\text{Count of intersection of neighbours}}{\text{Count of union of neighbors}}$$

5. Distance fusion : $D_{final} = (1 \text{ - } \lambda)\ D_{jac} + \lambda\ D_{orig}$

Experiments

Application of re-ranking in face recognition

Hyper-parameters			GAR					
			@1% FAR			@0.1% FAR		
			Protocol			Protocol		
k_1	k_2	λ	1	2	4	1	2	4
23	5	0.6	95.46	88.64	88.69	56.97	83.88	82.57
		0.7	96.30	88.35	88.49	57.14	82.88	81.68
	6	0.6	95.29	88.74	88.83	54.11	84.13	**82.88**
		0.7	95.96	88.41	88.60	53.78	83.21	82.00
24	5	0.6	95.46	88.68	88.75	**57.64**	83.85	82.44
		0.7	**96.47**	88.27	88.42	56.97	82.85	81.70
	6	0.6	95.83	**88.77**	**88.87**	56.13	**84.13**	82.77
		0.7	96.30	88.42	88.54	55.29	83.13	81.90
FEBNet (No re-ranking)			95.79	86.19	86.25	56.30	75.25	73.42

Table: Hyper parameter search for re-ranking method on the final model. Here, k_1 = the count for finding k-reciprocal nearest neighbors, k_2 = count for k-reciprocal nearest neighbor expansion, λ = ratio of importance given to original distance matrix with respect to jaccard distance during re-ranking.

Comparison with state-of-art

Models	GAR					
	@1%FAR			@0.1%FAR		
	Protocol			Protocol		
	1	2	4	1	2	4
MiRA-Face	95.46	**90.65**	**90.62**	51.09	80.56	79.26
UMDNets	94.28	86.62	86.75	53.27	74.69	72.90
FEBNet (Ours)	**95.83**	88.77	88.87	**56.13**	**84.13**	**82.77**

Table: Comparisons of FEBNet with state-of-art on DFW2018 dataset

Model	GAR							
	@0.1% FAR				@0.01% FAR			
	Protocol				Protocol			
	1	2	3	4	1	2	3	4
ResNet-50	47.6	35.4	46.4	35.9	38.4	16.4	22.4	16.9
LightCNN-29v2	**74.4**	55.6	69.2	55.7	**51.2**	36.9	47.2	36.5
FEBNet (ours)	54.8	**92.3**	**78.8**	**90.8**	42.4	**87.7**	**47.6**	**73.7**

Table: Test dataset (DFW2019 dataset) results

Conclusion

- Transfer learning based ensemble model
- Two new loss functions apart from prevalent person-id based cross entropy and inter-person triplet loss
- Application of re-ranking to DFW

Future work: What if we augment the face images with disguising effects? Will it help?

Thank you!