Lecture 10: Face Recognition
• Human performance: 99.20%
• Local Binary Patterns: 95.17%
• DeepFace: 97.35%
• DeepID2: 99.15%
• FaceNet: 99.63%
Axes for Studying Face Recognition

Loss
- Euclidean distance
- Angular margin
- Softmax variations

Architecture
- Backbone Networks
- Assembled Networks

Data Process
- One to many augmentation
- Many to one normalization

Specific scenario
- Low shot
- Anti-spoofing
- Cross age
- Cross pose
- NIV-VIS
- Video
- Make-up
- 3D
- Pose-sketch
- Template-based

Domain adaptation

Data
- MS-Celeb-1M
- VGGFace 2
- CASIA-Webface
- IJB-A
- FG-Net
- CP/CA/S L-LFW
## Face Datasets

| Name       | Identities | Images       | Purpose                                                        |
|------------|------------|--------------|                                                               |
| LFW        | 5,749      | 13,233       | Small, used for testing, saturated                             |
| Celeb Faces| 10177      | 202,599      | Many identities, attribute labels                              |
| VGG-Face   | 2622       | 1,635,159    | Many examples per class, somewhat noisy                       |
| IJB-A      | 500        | 5k images, 2k videos | Challenging pose, lighting, quality                           |
| MS-1M      | 80k        | 7M           | Largest public dataset for training (currently)                |
| Facebook   | 4030       | 4.4M         | Proprietary, used in DeepFace                                  |
| Google     | 8M         | 200M         | Proprietary, used in FaceNet                                   |

**Diagram:**
- **FRGC**
- **ORL**
- **CASPEAL**
- **MORPH**
- **CASIA-HFBI**
- **CASIA-HFBI (NR-VIS)**
- **FRGCv2 (1D)**
- **BU-3DFE (2D)**
- **YuC (video)**
- **YT6 (video)**
- **PaSC (video)**
- **CASCAD (cross-age)**
- **VGGFace (2.6M/2.6K)**
- **CFP (frontal-profile)**
- **UMDFaces (video)**
- **VGGFace2 (3.3M/8K)**
- **SLLFW (face-ground)**
- **MegaFace (4.3M/900K)**
- **CALFW (cross-age)**
- **CPLFW (cross-age)**
- **MS-cub: 1M (10K, 100K)**

**Timeline:**
- 1994
- 1996
- 1998
- 2000
- 2002
- 2004
- 2006
- 2008
- 2010
- 2012
- 2014
- 2016
- 2018
Face Processing

**Augmentation**
- One-to-Many Augmentation: mitigate difficulty of diverse data collections
  - Generate 3D pose-variant faces from frontal inputs, use for training
  - Use GANs or other methods to generate faces with diverse attributes

**Normalization**
- Many-to-One Normalization: reduce variation in test-time inputs
  - Generate frontal face from pose-variant input
  - Use GANs or methods to generate faces with neutral attributes

[Zhu et al., “3DDFA”]
[Yin et al., “FF-GAN”]
Network Architectures

**Backbone networks**

- AlexNet (12/2012)
- VGGNet (2014)
- GoogleNet (6/2015)
- ResNet (6/2016)
- SENet (9/2017)

**Multi-tasked networks**

- Deepface (7/2014) (AlexNet)
- Facenet (6/2015) (GoogleNet)
- VGGface (9/2015) (VGGNet)
- SphereFace (7/2017) (ResNet)
- VGGFace2 (11/2017) (SENet)
Loss Functions

Large-margin losses and softmax variants

- DeepID2 (contrastive loss)
- DeepID (softmax)
- Deepface (softmax)
- DeepID2+ (contrastive loss)
- FaceNet (triplet loss)
- VGGface (triplet+softmax)
- L-softmax (large margin)
- TSE (triplet loss)
- Range loss
- L2 softmax (feature normalization)
- Normface (feature normalization)
- A-softmax (large margin)
- Cosface (large margin)
- CoCo loss (feature normalization)
- Arcface (large margin)
- Center invariant loss (center loss)
- Marginal loss
- Center loss
- Feature and weight normalization
- Large margin loss
DeepFace

Layer 4-6: Intuition
• Apply filters to different locations on the map
• Similar to a conv. layer but spatially dependent

• F8 calculates probability with softmax
  \[ p_k = \frac{\exp(o_k)}{\sum_h \exp(o_h)} \]

• Cross-entropy loss function:
  \[ L = -\sum_k \log(p_k) \]

[Taigman et al., DeepFace, CVPR 2014]
DeepID2: Verification and Identification Signals

- **Verification**:
  - Match two images of an individual across large appearance variations
  - Favors tight clusters for each identity

- **Identification**:
  - Distinguish images of one identity from another identity
  - Favors large distance between clusters
DeepID2: Verification and Identification Signals

Learn face representations from

Prediction becomes richer
Prediction becomes more challenging
Supervision becomes stronger
Feature learning becomes more effective

Predicting binary labels (verification)
Predicting multi-class labels (identification)
Predicting thousands of real-valued pixels (multi-view) reconstruction
Balancing Identification and Verification

- Balance required between signals to learn good features

[Sun et al., DeepID2, NIPS 2014]
Balancing Identification and Verification

- When only identification signal is used ($\lambda = 0$):
  - High diversity in both inter-personal and intra-personal features
  - Good for identification since it helps distinguish different identities
  - But large intra-personal variance is noise for verification

- When only verification signal is used ($\lambda$ approaches $+\infty$):
  - Both intra-personal and inter-personal variance collapse to few directions
  - Good for verification, but cannot distinguish many classes in identification

\[
\sum_{i=1}^{c} n_i \cdot (\bar{x}_i - \bar{x}) (\bar{x}_i - \bar{x})^T
\]

\[
\sum_{i=1}^{c} \sum_{x \in D_i} (x - \bar{x}_i) (x - \bar{x}_i)^T
\]
Balancing Identification and Verification

- When both verification and identification signals are used ($\lambda = 0.05$):
  - Inter-personal variations stay high
  - Intra-personal variations reduce in diversity and magnitude
Balancing Identification and Verification

- Visualize features for 6 identities
- With only identification signal:
  - Cluster centers are well-separated, but large cluster size causes overlap
- With only verification signal:
  - Cluster sizes become small, but cluster centers also collapse
- With both signals:
  - Clusters sizes become small and cluster centers are reasonably separated
Visualization with t-SNE

- Map high-dimensional data with t-distributed stochastic neighbor embedding

[van der Maaten and Hinton, Visualizing Data Using t-SNE, JMLR 2008]
Visualization with t-SNE

- Map high-dimensional data with t-distributed stochastic neighbor embedding
- Represent similarity for data points as a probability
- Assume neighbor $x_j$ for point $x_i$ picked based on Gaussian density

$$p_{ij} = \frac{\exp\left(-\frac{||x_i - x_j||^2}{2\sigma^2}\right)}{\sum_{k \neq l} \exp\left(-\frac{||x_k - x_l||^2}{2\sigma^2}\right)}$$
Visualization with t-SNE

- Map high-dimensional data with t-distributed stochastic neighbor embedding
- Represent similarity for data points as a probability
- Assume neighbor \( x_j \) for point \( x_i \) picked based on Gaussian density
  \[
  p_{ij} = \frac{\exp \left( -\frac{\|x_i - x_j\|^2}{2\sigma^2} \right)}{\sum_{k \neq l} \exp \left( -\frac{\|x_k - x_l\|^2}{2\sigma^2} \right)}
  \]
- Assume a t-student distribution for low-dimensional points
  \[
  q_{ij} = \frac{\left(1 + \frac{\|y_i - y_j\|^2}{2}\right)^{-1}}{\sum_{k \neq l} \left(1 + \frac{\|y_k - y_l\|^2}{2}\right)^{-1}}
  \]
- Heavier tail means points moderately far \( x \) are mapped further in \( y \)
  - More volume at greater distance in high dimensions
  - More “space” available in low dimension for nearby points
Visualization with t-SNE

- Map high-dimensional data with t-distributed stochastic neighbor embedding
- Represent similarity for data points as a probability
- Assume neighbor $x_j$ for point $x_i$ picked based on Gaussian density

$$p_{ij} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)}{\sum_{k \neq l} \exp\left(-\frac{\|x_k - x_l\|^2}{2\sigma^2}\right)}$$

- Assume a t-student distribution for low-dimensional points

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}$$

- Heavier tail means points moderately far $x$ are mapped further in $y$
  - More volume at greater distance in high dimensions
  - More “space” available in low dimension for nearby points

- Minimize a cost given by the KL-divergence using gradient descent:

$$C = KL(P || Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$
Learn an Embedding for Face Recognition

- **Face verification**: determine whether two images are of the same person
- **Face identification**: determine identity of person in an image
- **Face clustering**: find the same person among a collection of faces

- Train a network such that embedding distances directly represent similarity
  - Faces of same person: small distances
  - Faces of different persons: large distances

- Once embedding is learned, above problems are all solvable
  - **Verification**: threshold distance between two embeddings
  - **Identification**: can be posed as k-NN classification
  - **Clustering**: can be solved using methods like k-means
Learn an Embedding for Face Recognition

- Goal: learn d-dimensional embedding $f(x)$ for face image $x$
- Constrain embedding to lie on unit sphere: $\|f(x)\|_2 = 1$

- Goal for triplet loss:
  - Minimize distance between anchor and a positive (from same class)
  - Maximize distance between anchor and a negative (from different classes)

[Schroff et al., FaceNet, CVPR 2015]
Triplet Loss for Training

• Goal for triplet loss:
  • Minimize distance between anchor image $x_i^a$ and a positive $x_i^p$
  • Maximize distance between anchor $x_i^a$ and a negative $x_i^n$

![Diagram showing anchor, positive, and negative examples with learning process]

$$\| f(x_i^a) - f(x_i^p) \|_2^2 + \alpha < \| f(x_i^a) - f(x_i^n) \|_2^2$$

• Total loss to minimize:

$$\sum_{i=1}^{N} \left[ \| f(x_i^a) - f(x_i^p) \|_2^2 - \| f(x_i^a) - f(x_i^n) \|_2^2 + \alpha \right]_+$$

• Challenge: too many triplets satisfy the margin easily
• Need to select hard examples that are active and improve the model
Triplet Selection

- To ensure fast convergence, given an anchor \( x_i^a \):
  - Select **hard positive** \( x_i^p \) such that \( \arg\max_{x_i^p} \| f(x_i^a) - f(x_i^p) \|_2^2 \)
  - Select **hard negative** \( x_i^n \) such that \( \arg\min_{x_i^n} \| f(x_i^a) - f(x_i^n) \|_2^2 \)
Triplet Selection

• To ensure fast convergence, given an anchor $x_i^a$:
  
  • Select **hard positive** $x_i^p$ such that
    $$\argmax_{x_i^p} \| f(x_i^a) - f(x_i^p) \|_2^2$$
  
  • Select **hard negative** $x_i^n$ such that
    $$\argmin_{x_i^n} \| f(x_i^a) - f(x_i^n) \|_2^2$$

• **Inefficient** (or infeasible) to compute argmin and argmax over training set

• Might lead to **poor training** as mislabeled or poorly imaged examples dominate
Triplet Selection

- Two courses of action: offline and online selection of triplets
- **Offline**: every n steps, use current feature for argmin and argmax on subset
Triplet Selection

- Two courses of action: offline and online selection of triplets

- **Offline**: every n steps, use current feature for argmin and argmax on subset

- **Online**: selecting hard positive and negative examples in mini-batch
  - Use large mini-batch with several thousand examples
  - Use several examples per identity for meaningful anchor-positive distances
  - Randomly sample negatives from other identities
Triplet Selection

• Two courses of action: offline and online selection of triplets

• **Offline**: every n steps, use current feature for argmin and argmax on subset

• **Online**: selecting hard positive and negative examples in mini-batch
  • Use large mini-batch with several thousand examples
  • Use several examples per identity for meaningful anchor-positive distances
  • Randomly sample negatives from other identities

• In practice:
  • Use all anchor-positive pairs, instead of just hard positives
  • Use **semi-hard negatives** at beginning of training

\[ \|f(x_i^a) - f(x_i^p)\|_2^2 < \|f(x_i^a) - f(x_i^n)\|_2^2 \]

  • Not hardest negatives, can be within margin, but further than positives
  • Hardest negatives at beginning can cause feature to collapse to \( f(x) = 0 \)
Comparison of DeepFace, DeepID2, FaceNet

- Benefit over DeepFace:
  - Learns an embedding, can be used for multiple tasks
  - Only 128-dimensional representation, efficient for inference
Comparison of DeepFace, DeepID2, FaceNet

- Benefit over DeepFace:
  - Learns an embedding, can be used for multiple tasks
  - Only 128-dimensional representation, efficient for inference
- Intuitive benefit of triplet loss over pairwise loss in DeepID2
  - Pairwise: map all faces from one identity to same point
  - Triplet: margin between each pair of faces of an identity and all other faces
  - Triplet loss allows an identity manifold, with distance from other identities
Evaluation of Face Verification

- Given a pair of face images:
  - A squared L2 distance $D(x_i, x_j)$ is used to determine same or different
  - Good embedding: true matches will lie within a small value of $D(x_i, x_j)$
Evaluation of Face Verification

- Given a pair of face images:
  - A squared L2 distance $D(x_i, x_j)$ is used to determine same or different
  - Good embedding: true matches will lie within a small value of $D(x_i, x_j)$

- True accepts: set of face pairs correctly classified as same within a threshold $d$

$$\text{TA}(d) = \{(i, j) \in P_{\text{same}}, \text{with } D(x_i, x_j) \leq d\}$$

All face pairs in test set that belong to same identities
Evaluation of Face Verification

• Given a pair of face images:
  • A squared L2 distance $D(x_i, x_j)$ is used to determine same or different
  • Good embedding: true matches will lie within a small value of $D(x_i, x_j)$

• True accepts: set of face pairs correctly classified as same within a threshold $d$

$$TA(d) = \{(i, j) \in \mathcal{P}_{\text{same}}, \text{ with } D(x_i, x_j) \leq d\}$$

All face pairs in test set that belong to same identities

• False accepts: set of face pairs incorrectly classified as same for threshold $d$

$$FA(d) = \{(i, j) \in \mathcal{P}_{\text{diff}}, \text{ with } D(x_i, x_j) \leq d\}$$

All face pairs in test set that belong to different identities
Evaluation of Face Verification

- Given a pair of face images:
  - A squared L2 distance $D(x_i, x_j)$ is used to determine same or different
  - Good embedding: true matches will lie within a small value of $D(x_i, x_j)$

- True accepts: set of face pairs correctly classified as same within a threshold $d$

  $$TA(d) = \{(i, j) \in P_{\text{same}}, \text{with } D(x_i, x_j) \leq d\}$$

  All face pairs in test set that belong to same identities

- False accepts: set of face pairs incorrectly classified as same for threshold $d$

  $$FA(d) = \{(i, j) \in P_{\text{diff}}, \text{with } D(x_i, x_j) \leq d\}$$

  All face pairs in test set that belong to different identities

- Determine validation rate and false accept rate:

  $$VAL(d) = \frac{|TA(d)|}{|P_{\text{same}}|}, \quad \text{FAR}(d) = \frac{|FA(d)|}{|P_{\text{diff}}|}$$
Identification Harder than Verification

- Closed set identification: assign probe image one of gallery identities
- Galleries can be very large, high chance of similar appearances

Face Identification

Tom Cruise
Identification Harder than Verification

- Closed set identification: assign probe image one of gallery identities
- Galleries can be very large, high chance of similar appearances

- True accept rate: percentage of probes matched correctly to gallery
- False accept rate: percentage of probes matched incorrectly to gallery
- Aim: achieve high TAR at low FAR
Identification Harder than Verification

- Closed set identification: assign probe image one of gallery identities
- Galleries can be very large, high chance of similar appearances

- True accept rate: percentage of probes matched correctly to gallery
- False accept rate: percentage of probes matched incorrectly to gallery
- Aim: achieve high TAR at low FAR

- Suppose a verification system achieves 99% accuracy
- Incorrectly verifies 2 cases among 100 matched and 100 unmatched pairs
- Now use for identification against gallery of 901 subjects
- Expect 1 correct and 9 incorrect candidates, all of which look similar
- Even if resolve further by 50%, still have a 50% error rate
Identification Harder than Verification

- Closed set identification: assign probe image one of gallery identities
- Galleries can be very large, high chance of similar appearances

- True accept rate: percentage of probes matched correctly to gallery
- False accept rate: percentage of probes matched incorrectly to gallery
- Aim: achieve high TAR at low FAR

- Suppose a verification system achieves 99% accuracy
- Incorrectly verifies 2 cases among 100 matched and 100 unmatched pairs
- Now use for identification against gallery of 901 subjects
- Expect 1 correct and 9 incorrect candidates, all of which look similar
- Even if resolve further by 50%, still have a 50% error rate

- Even harder case: open set identification
  - Probe identity may or may not exist in the gallery
Evaluation of Face Verification

For a few different network architectures

[Schroff et al., FaceNet, CVPR 2015]
Evaluation of Face Verification

Validation rates at threshold 0.001

<table>
<thead>
<tr>
<th>jpeg q</th>
<th>val-rate</th>
<th>#pixels</th>
<th>val-rate</th>
<th>#dims</th>
<th>VAL</th>
<th>#training images</th>
<th>VAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>67.3%</td>
<td>1,600</td>
<td>37.8%</td>
<td>64</td>
<td>86.8% ± 1.7</td>
<td>2,600,000</td>
<td>76.3%</td>
</tr>
<tr>
<td>20</td>
<td>81.4%</td>
<td>6,400</td>
<td>79.5%</td>
<td>128</td>
<td>87.9% ± 1.9</td>
<td>26,000,000</td>
<td>85.1%</td>
</tr>
<tr>
<td>30</td>
<td>83.9%</td>
<td>14,400</td>
<td>84.5%</td>
<td>256</td>
<td>87.7% ± 1.9</td>
<td>52,000,000</td>
<td>85.1%</td>
</tr>
<tr>
<td>50</td>
<td>85.5%</td>
<td>25,600</td>
<td>85.7%</td>
<td>512</td>
<td>85.6% ± 2.0</td>
<td>260,000,000</td>
<td>86.2%</td>
</tr>
<tr>
<td>70</td>
<td>86.1%</td>
<td>65,536</td>
<td>86.4%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>86.5%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Image quality | Image resolution | Embedding dimension | Training images

[Schroff et al., FaceNet, CVPR 2015]
Qualitative Results

False accept

False reject

Clustering
Training Objectives for Face Recognition

- Metric learning
  \[
  \sum_{i}^{N} \left[ \| f(x_i^a) - f(x_i^p) \|^2_2 - \| f(x_i^a) - f(x_i^n) \|^2_2 + \alpha \right]_+
  \]

- Softmax:
  - Labeled data \((x_i, y_i)\) for \(i = 1, \ldots, m\) in mini-batch and classes \(j = 1, \ldots, n\)
  - \(W, b\): Weight matrix and bias for last fully-connected layer
  \[
  \mathcal{L}_S = - \sum_{i=1}^{m} \log \frac{e^{W_{y_i}^{T} x_i + b_y_i}}{\sum_{j=1}^{n} e^{W_{j}^{T} x_i + b_j}}
  \]
Training Objectives: Center Loss

- Softmax pushes features apart for distinct classes
- Introduce a pull term to encourage features of a class to cluster together

\[- \sum_{i=1}^{m} \log \frac{e^{W_{y_i}^T x_i + b_y_i}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}} + \frac{\lambda}{2} \sum_{i=1}^{m} \|x_i - c_{y_i}\|_2^2\]

- Enhance discrimination power
- Sequentially update $W$, CNN parameters and $c$ in each mini-batch

[Wen et al., ECCV 2016]
Training Objectives: Angular Softmax

- Demand angular margin between classes, instead of Euclidean margin
  - Consider softmax when $W$ is normalized (and $b$ is 0)

$$
\sum_i - \log \left( \frac{e^{\|x_i\| \cos(\theta_{y_i,i})}}{\sum_j e^{\|x_i\| \cos(\theta_{j,i})}} \right)
$$

- Angle between data point $x_i$ and vector $W_j$: $\theta_{j,i}$

- To correctly classify class 1 against 2:
  $$
  \cos(\theta_1) > \cos(\theta_2)
  $$

- A-softmax: more stringent lower bound
  $$
  \cos(\theta_1) > \cos(m\theta_1) > \cos(\theta_2)
  $$

$$
\sum_i - \log \left( \frac{e^{\|x_i\| \cos(m\theta_{y_i,i})}}{e^{\|x_i\| \cos(m\theta_{y_i,i})} + \sum_{j \neq y_i} e^{\|x_i\| \cos(\theta_{j,i})}} \right)
$$

[Liu et al., SphereFace, CVPR 2017]
Training Objectives: Additive Angular Margins

- Demand additive angular margin between classes
- CosFace: maintains large angular margin even for visually similar classes

$$\sum_i \log \frac{e^{s(\cos(\theta_{y_i,i})-m)}}{e^{s(\cos(\theta_{y_i,i})-m)} + \sum_{j \neq y_i} e^{s \cos(\theta_{j,i})}}$$

- ArcFace imposes a margin on $\theta_{j,i}$, more representative of geodesic distance

$$\sum_{i=1}^{N} \log \frac{e^{s(\cos(\theta_{y_i,i}+m))}}{e^{s(\cos(\theta_{y_i,i}+m))} + \sum_{j=1, j \neq i}^{n} e^{s \cos \theta_j}}$$

[Wang et al., CosFace, CVPR 2018] [Deng et al., ArcFace]