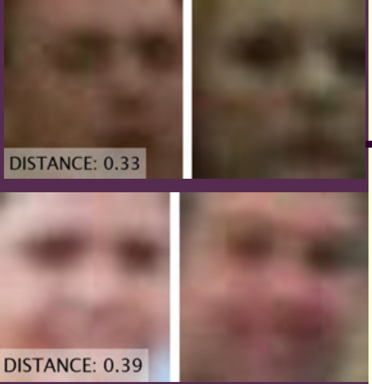


Lookalike Disambiguation in Face Recognition

Arun Ross

Professor

Michigan State University

rossarun@cse.msu.edu

http://iprobe.cse.msu.edu

Work done with Thomas Swearingen

Publication

- T. Swearingen and A. Ross, "**Lookalike Disambiguation: Improving Face Identification Performance at Top Ranks**," Proc. of 25th International Conference on Pattern Recognition (ICPR 2020), January 2021

Lookalike Faces

- In this presentation we are **not** considering “lookalike faces” from a human vision standpoint
- **Not** specifically considering twins, siblings, and other types of kinship relationships
- We are considering “lookalike faces” from a **computer vision** standpoint
  - Face images of different identities that are “confused” to be the same by a **face matcher**

Examples of Lookalike Faces

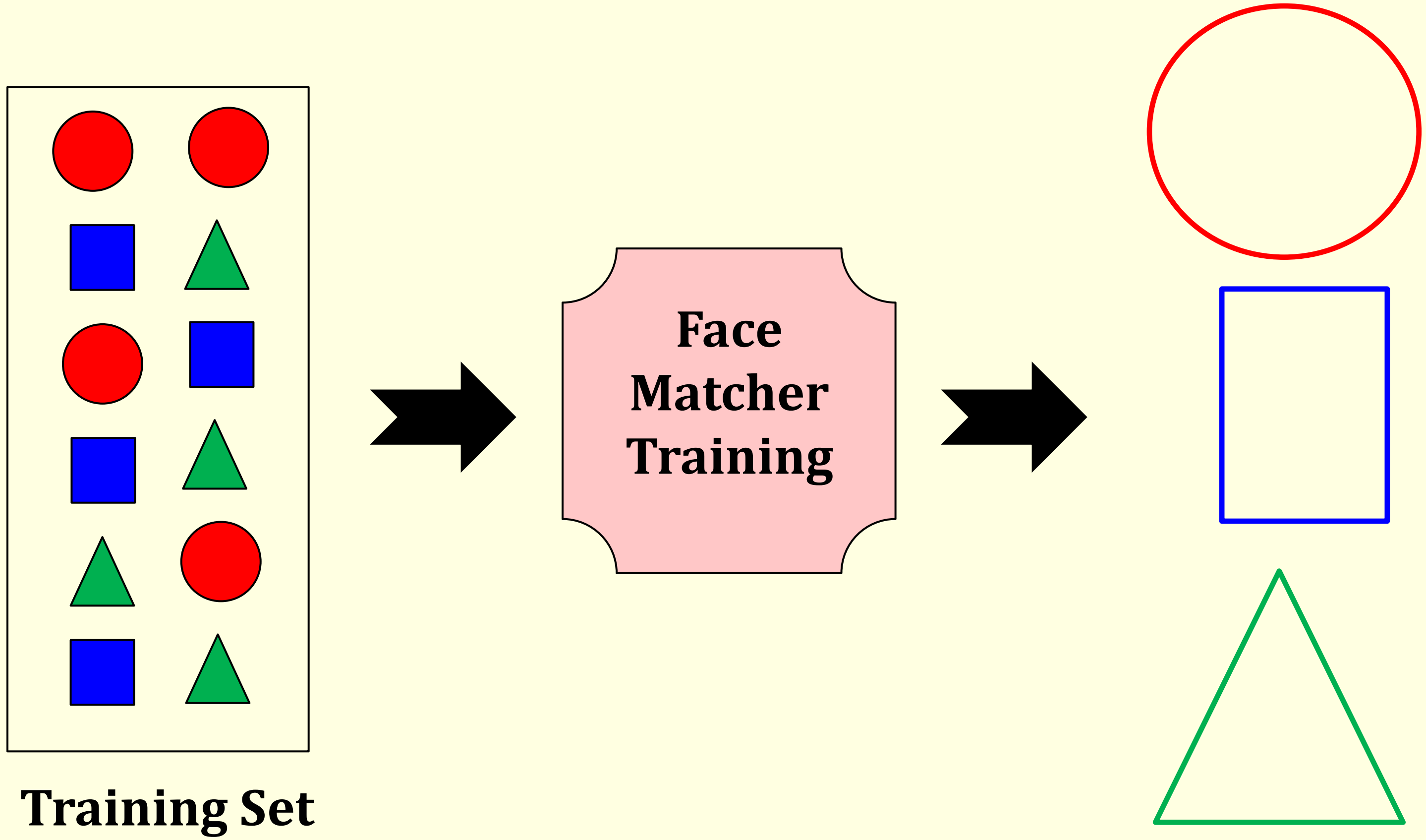
- LFW Dataset | COTS Matcher

Identification versus Similarity

- Most face recognition methods **do not** explicitly consider the notion of **similarity** during the training phase
- Face images are **labeled** with identifiers
  - **001; 001; 001; 002; 002; 003; 003; 003; 003; 003**
- Then the method attempts to **minimize intra-class** variations and **maximize inter-class** variations

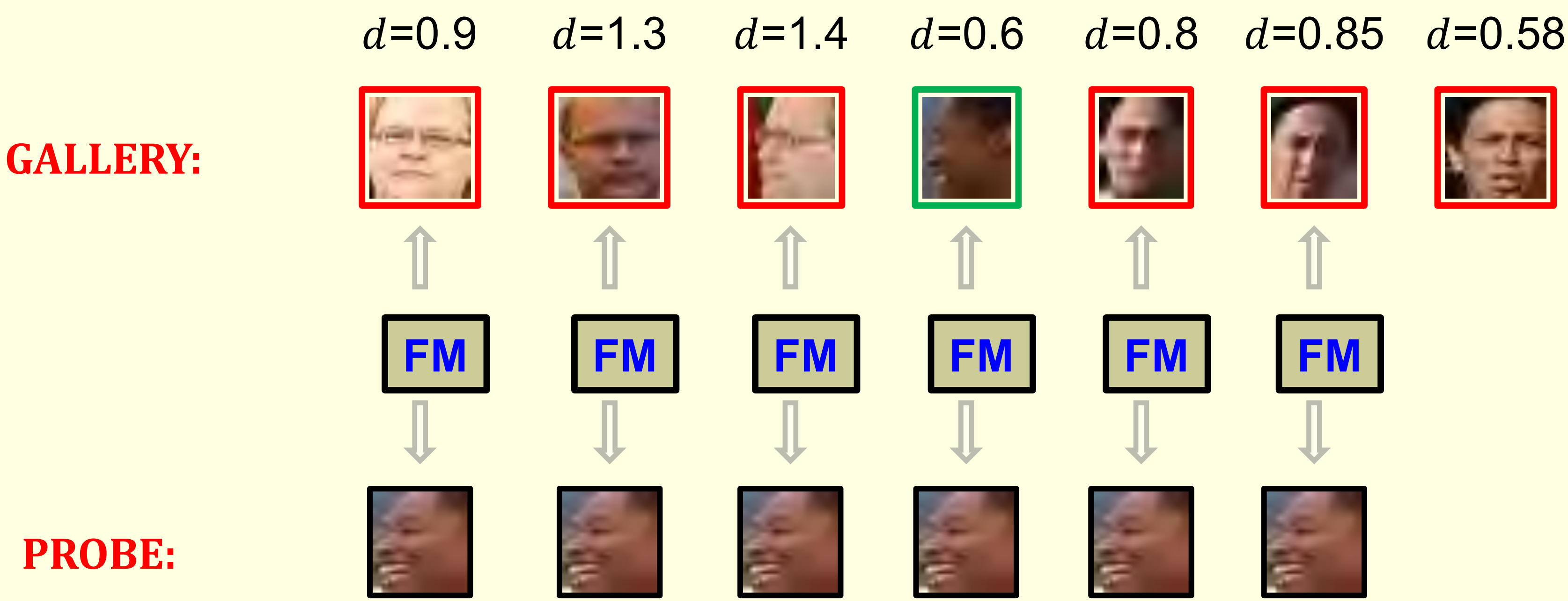
During this process, the degree of similarity between different identities is not explicitly used – since that information is typically not available during training

Training Stage



- The distance between different identities is not explicitly specified during the training phase; it is implicitly learned by the face matcher
- But see: Sadovnik, **Finding your Lookalike: Measuring Face Similarity Rather than Face Identity**, CVPRW 2018

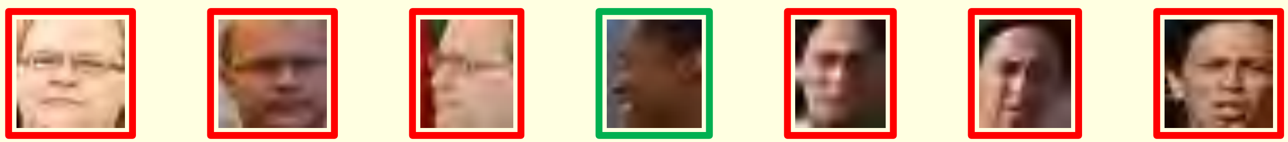
Identification Process



Ranked Match List

d=0.58 d=0.6 d=0.8 d=0.85 d=0.9 d=1.3 d=1.4

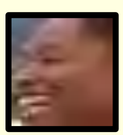
GALLERY:



Rank **1** Rank **2** Rank **3** Rank **4** Rank **5** Rank **6** Rank **7**

Ranked Match List

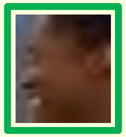
PROBE:



Matcher Confusion

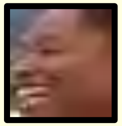
d=0.58 d=0.6

GALLERY:



Rank **1** Rank **2**

PROBE:



- Correct match occurs at rank 2, not rank 1
- Matcher "**confuses**" imposter face at rank 1 with genuine face at rank 2
- These 2 face could be **lookalikes**

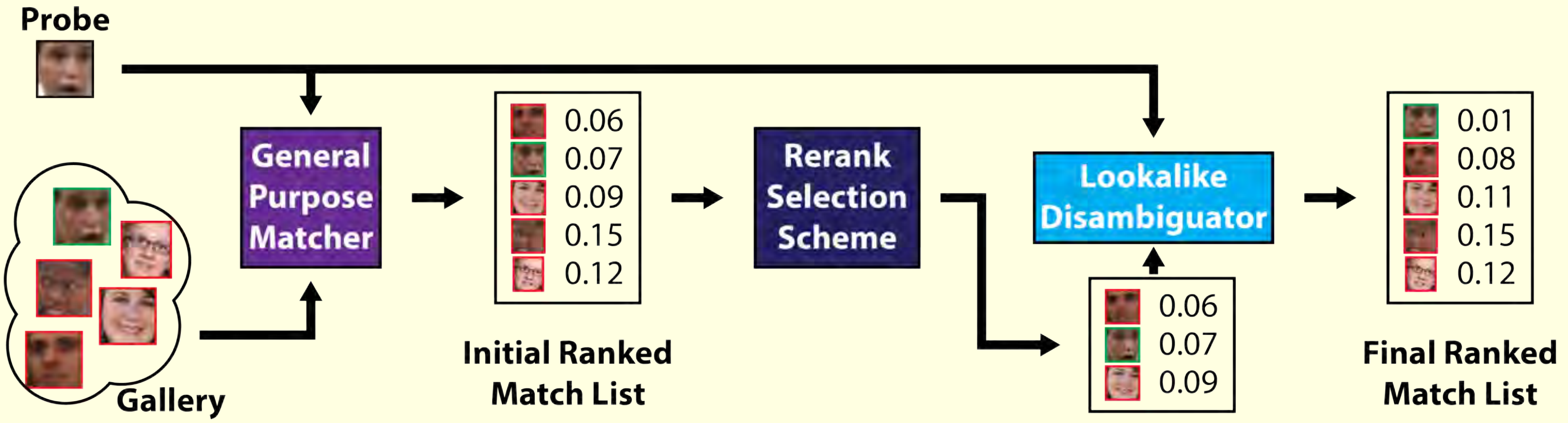
Related Work

Year	Work	Approach
2012	Srinivas: Analysis of facial marks to distinguish between identical twins	Use facial marks to distinguish twins
2012	Le: A facial aging approach to identification of identical twins	Face aging to distinguish twins
2018	Sun: Deep Siamese convolutional neural networks for identical twins and look-alike identification	Develop CNN to distinguish twins
2011	Lambda: Face recognition for look-alikes: A preliminary study	Match face regions independently
2017	Smirnov: Doppelgänger mining for face representation learning	Refine mini-batch selection of a general-purpose matcher using a list of lookalikes
2017	Moeini: Open-set face recognition across look-alike faces in real-world scenarios	Use 3D models to distinguish lookalikes

- **Identical twins were an early interest where approaches focused on a specific aspect of the face**
- **Later approaches, focused on lookalikes more generally**

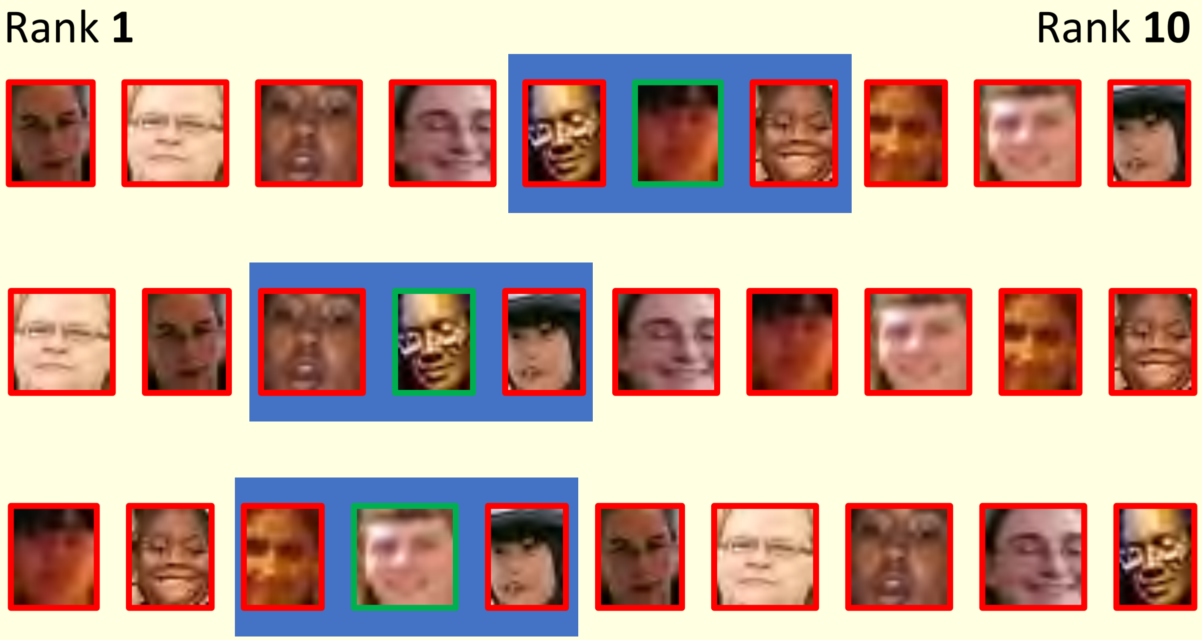
Our Approach

- Choose **some** of the top ranked faces on initial ranked match list to **re-rank**
- Re-rank them using a lookalike disambiguator (LD)
  - **LD matcher** specifically trained to distinguish lookalike face images



Selecting Gallery Samples to Re-rank

- Conduct an analysis to determine how the **scores** in the ranked match list vary in the **vicinity** of a correct match



Match-Vicinity Analysis

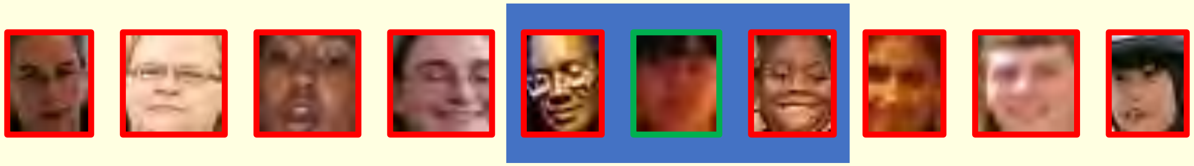
- Find the **match-vicinity scores** for a given probe image p in a ranked match list

- Normalize score with respect to the score at **position of correct gallery match** ($d_p^{(c)}$)
  - $s_p^{(i)} = d_p^{(i)} - d_p^{(c)}$

- Normalized score
  - **before** rank c must be **non-positive**
  - **after** rank c must be **non-negative**

Match-Vicinity Analysis

Correct Match
Rank **c**



$$d_p^{(c-1)} \quad d_p^{(c)} \quad d_p^{(c+1)}$$

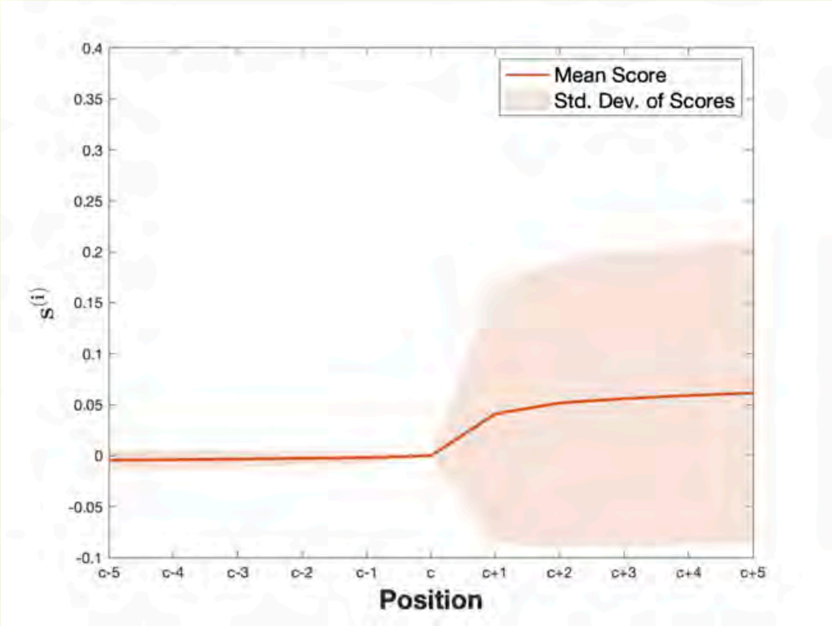
$$s_p^{(c-1)} = d_p^{(c-1)} - d_p^{(c)}$$

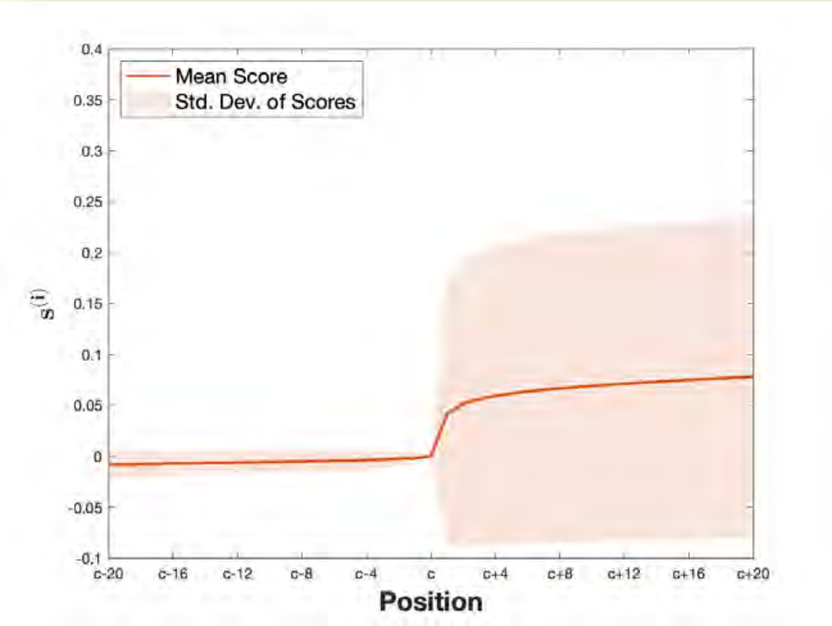
$$s_p^{(c+1)} = d_p^{(c+1)} - d_p^{(c)}$$

$$s_p^{(c)} = d_p^{(c)} - d_p^{(c)} = 0$$

Match-Vicinity Plot (MVP)

- MVP shows **mean** and **SD** of normalized match scores in the match vicinity for 3,728 probe images queries
- **Dataset**: TinyFace dataset | **Matcher**: ArcFace matcher



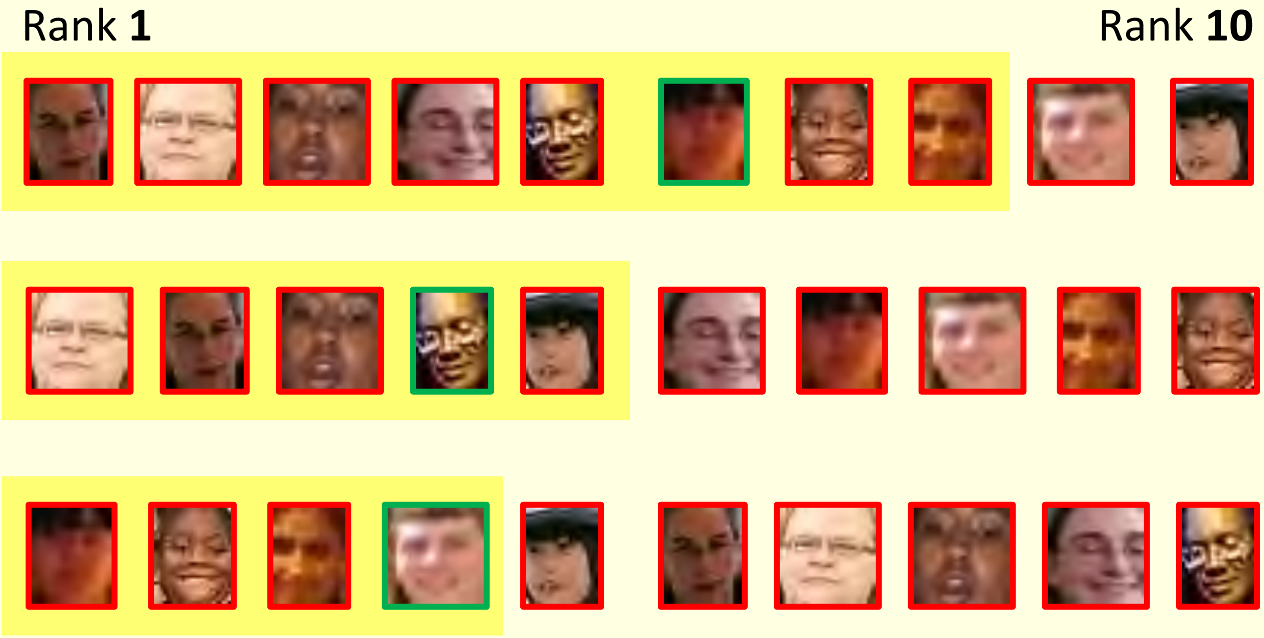


$\mathbf{c \pm 5}$

$\mathbf{c \pm 20}$

Adaptive Re-Ranking

- MVP: distance score increases at a **higher rate** from one rank to the next **after** encountering the correct match
- Use **sharp increases** in distance score to determine subset selection



Adaptive Re-Ranking

- Given a probe image, p, and a gallery set, $\mathcal{G} = \{g_1, g_2, \dots, g_n\}$
- Compare p to each gallery image g_i to obtain ranked list, $\mathcal{L} = \left(d^{(1)}, d^{(2)}, \dots, d^{(n)}\right)$
- Calculate **rolling sum** over consecutive distance scores, S_k
- Re-rank the **top $\boldsymbol{k}$** matches
    - the smallest value of k such that $S_k > \tau$

General Purpose Matcher (GPM)

- **ArcFace** is a publicly-available face matcher
  - High performance on LFW dataset (99.8% accuracy)
- Outputs a **512-dimensional** representation for a given input image
- Compare representations using **Euclidean distance**

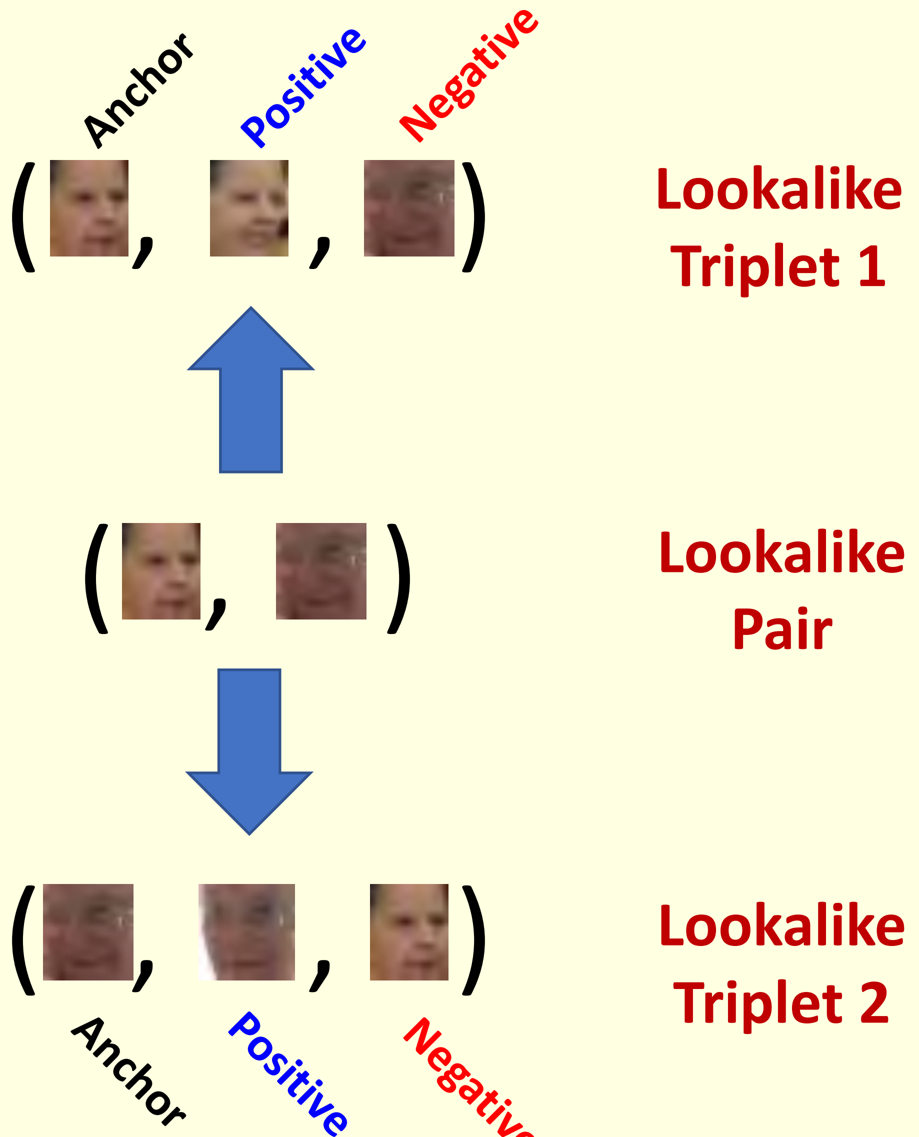
Deng et al., "Arcface: Additive angular margin loss for deep face recognition," CVPR 2019

Lookalike disambiguator (LD)

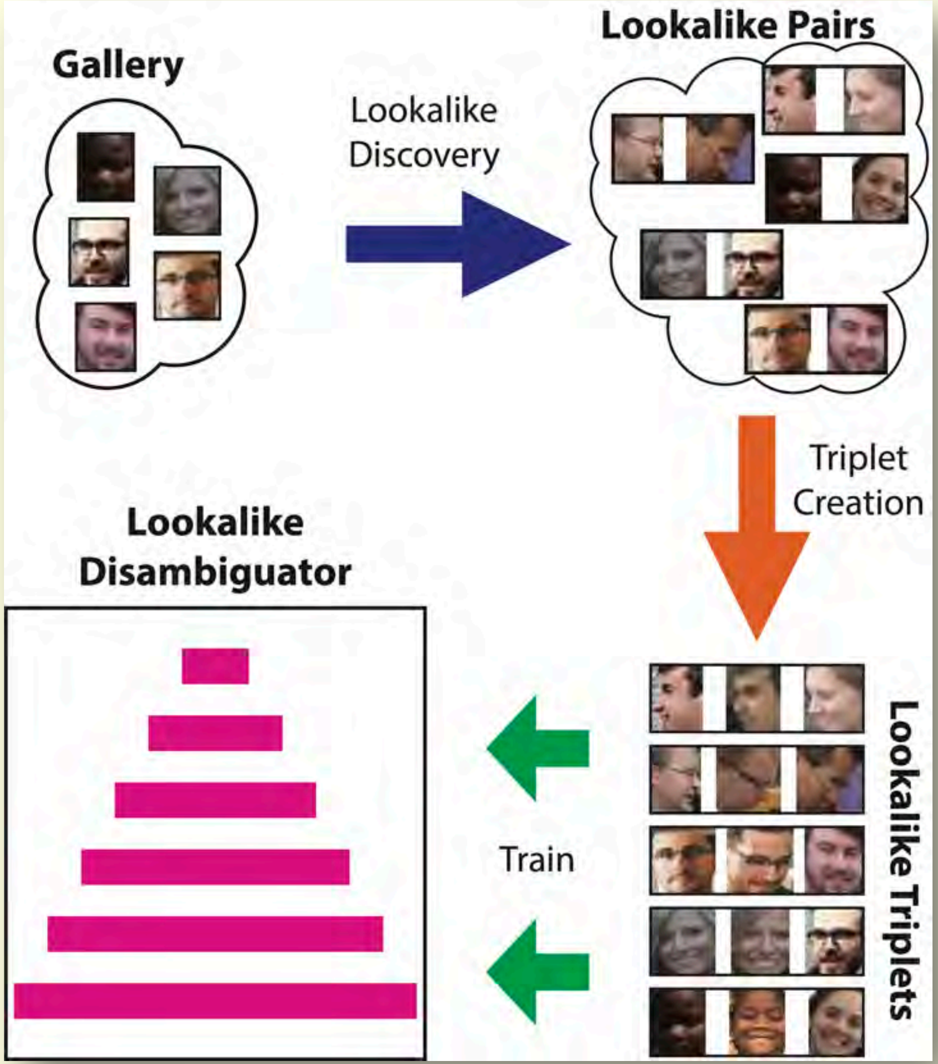
- **Finetunes** GPM using lookalike **triplets**
- Lookalike triplet consists of **anchor**, **positive**, and **negative** samples
  - **Anchor** & **positive** sample of same subject
  - **Anchor** & **negative** samples of different subjects, but judged by GPM to be **lookalikes**
- Loss function

$$\mathrm{L} = \sum_{\{I_a, I_p, I_n\}} \left\| f(I_a) - f\left(I_p\right) \right\|_2 - \| f(I_a) - f(I_n) \|_2 + \alpha_{\mathrm{margin}}$$

2 triplets from 1 pair



Lookalike disambiguator (LD)



Training Parameters

- PyTorch environment
- Stochastic gradient descent with Adam optimizer
- $\alpha_{\text{margin}} = 0.2$
- Batch Size: 32
- Learning Rate: 0.01

TinyFace Dataset

- Dataset consisting of **small** face images
  - Average size 20 x 16 pixels
- ***Gallery-match*** and ***Probe*** sets used
- Gallery contains multiple images of the same subject
- Identification experiments are **closed-set**

Set	Num. Images	Num. Subjects
Probe	3,728	2,569
Gallery-Match	4,443	
~~Gallery-Distractor~~	~~153,428~~	~~Unknown~~

TinyFace Dataset

Cheng et al., "Low-resolution face recognition," ACCV 2018

Filtered TinyFace Dataset

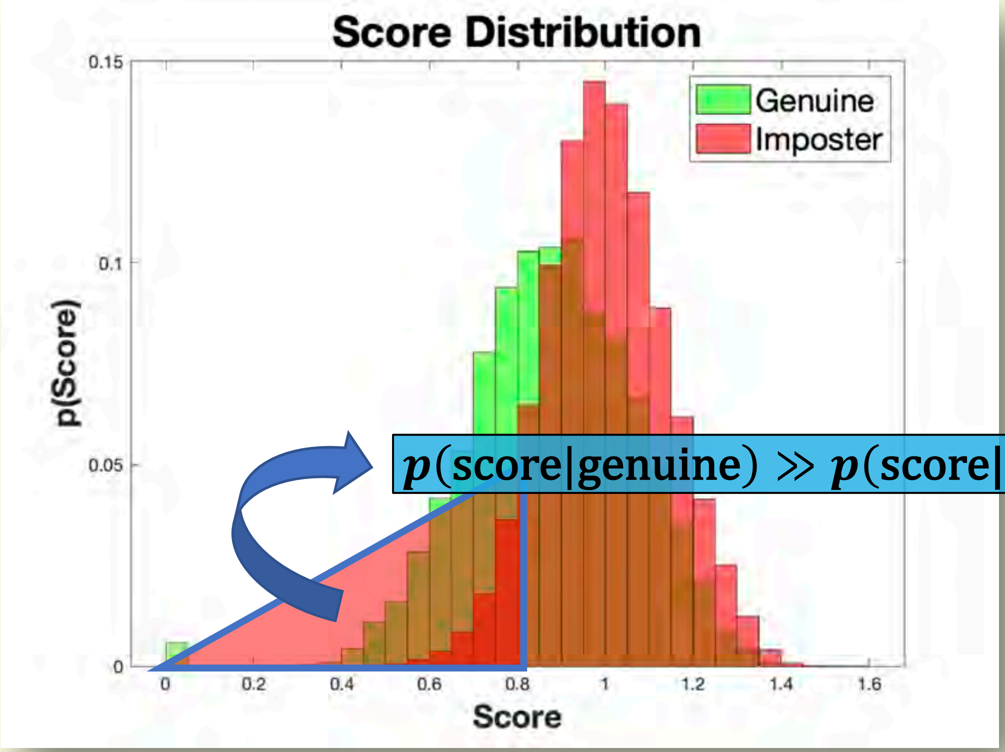
- Dataset **manually filtered** to exclude profile-view faces

- Filtered dataset contains **1,145** subjects
  - 2,081 images in probe subset
  - 2,461 images in gallery subset

- Experiments conducted on filtered dataset

Lookalike Discovery

- Match **gallery** against itself using **GPM**
- Select **imposter pairs** in the distance score range [0,0.8]
- Results in **~679K lookalike** pairs
  - 6.9% of all imposter pairs



Evaluation Metrics

Re-rank Subset Selection

1. **Hit Rate**

   Fraction of probes for which the selection scheme chooses a gallery subset that **includes the correct match**

2. **Surplus Size**

   Number of samples included in the subset with rank **higher than the rank of the correct match**

3. **Pool Size**

   Number of **gallery samples** selected to be reranked

Parameter Selection (using gallery)

- Estimate q and τ from gallery dataset (filtered)
- Rolling sum calculated for those gallery samples that have at least 1 other gallery sample of the same subject
  - 1,897 such images
- $\boldsymbol{\tau}$ is the average value of the rolling sum taken at position of correct match ($\boldsymbol{S_c}$)

q	τ	Surplus Size		Hit Rate
		Total	Per Search	
1	0.7695	270,276	142.5	55.77%
2	1.378	294,003	155.0	61.68%
3	1.958	295,173	155.6	62.20%
4	2.511	296,353	156.2	62.63%
5	3.049	297,541	156.8	63.05%
6	3.574	298,737	157.5	63.52%
7	4.090	299,942	158.1	63.78%
8	4.597	301,152	158.8	63.94%
9	5.094	302,365	159.4	64.21%
10	5.584	303,583	160.0	64.63%

Fixed versus Adaptive

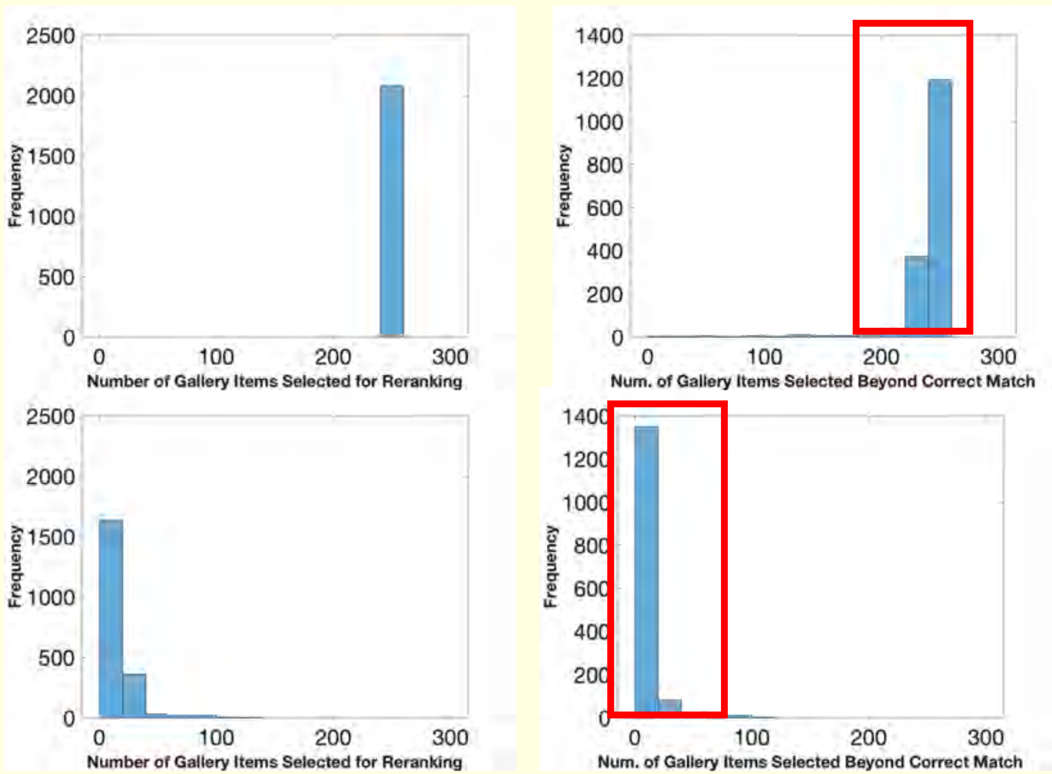
- Compare **Fixed** and **Adaptive** selection schemes
- For **adaptive** scheme, $q = 10$ and $\tau = 5.584$
- For **fixed** scheme, top 10% of matches are reranked (246)
- A **small pool size** is generally better
  - Not inherently bad: Could be that correct match occurs at a higher rank

Fixed versus Adaptive

POOL SIZE | SURPLUS SIZE

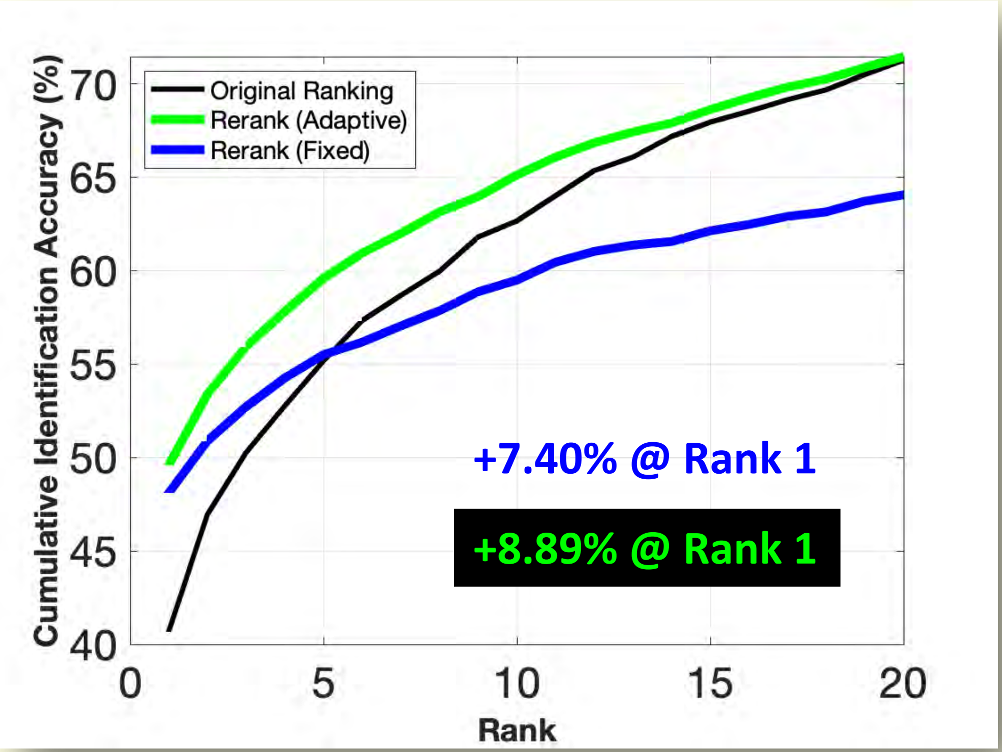
FIXED

ADAPTIVE



Scheme	Pool Size (min/mean/median/max)	Hit Rate
Fixed	246 | 246 | 246 | 246	80.1%
Adaptive	5 | 20.66 | 18 | 121	71.3%

Identification Performance



- Given a probe: Use **GPM** to rank gallery samples
- Select gallery samples to **re-rank** using fixed and adaptive schemes
- Re-rank top gallery samples using **LD**

Rank-1 identification accuracy improves from ~40.7% to ~49.6%

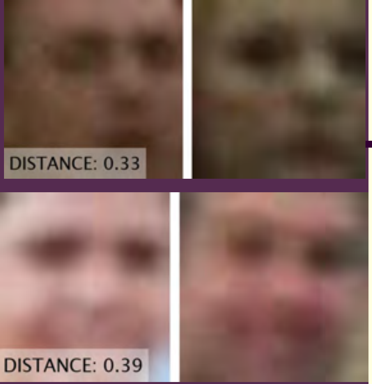
Summary

- Proposed an **adaptive gallery selection scheme** based on match scores generated using a face matcher
- Proposed the use of a **separate matcher** for re-ranking lookalike face images
- Observed an improvement in identification accuracy when using a **Lookalike Disambiguator** on the selected gallery samples

Preliminary results presented; Experiments with other datasets and matchers are ongoing; Motion can help in disambiguating as well

Publication

- T. Swearingen and A. Ross, "**Lookalike Disambiguation: Improving Face Identification Performance at Top Ranks**," Proc. of 25th International Conference on Pattern Recognition (ICPR 2020), January 2021



Lookalike Disambiguation in Face Recognition

Arun Ross

Professor

Michigan State University

rossarun@cse.msu.edu

http://iprobe.cse.msu.edu

Work done with Thomas Swearingen