

AdvKin: Adversarial Convolutional Network for Kinship Verification

IEEE TRANSACTIONS ON CYBERNETICS, VOL. 51, NO. 12, DECEMBER 2021



报告人：冉涛 黄霄宇 邹俊豪

2024/5/22

目录

01 引言与相关工作

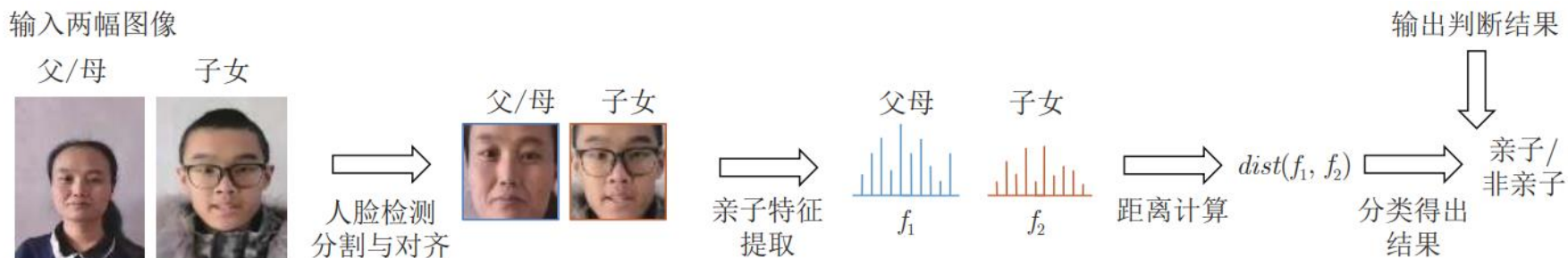
02 实现原理

03 实验



引言与相关工作

人脸亲属关系验证的目的：旨在通过图像特征提取和机器学习方法, 分析两幅不同的人脸图像的面部特征相似性来判断其是否具有某种亲属关系。



亲属关系验证的一般流程

现存的挑战：

1. 同一人不同形态差异的影响
2. 亲属长相差异的影响
3. 亲属/非亲属分类边界模糊
4. 图像高维特征提取与数据样本量少之间的矛盾

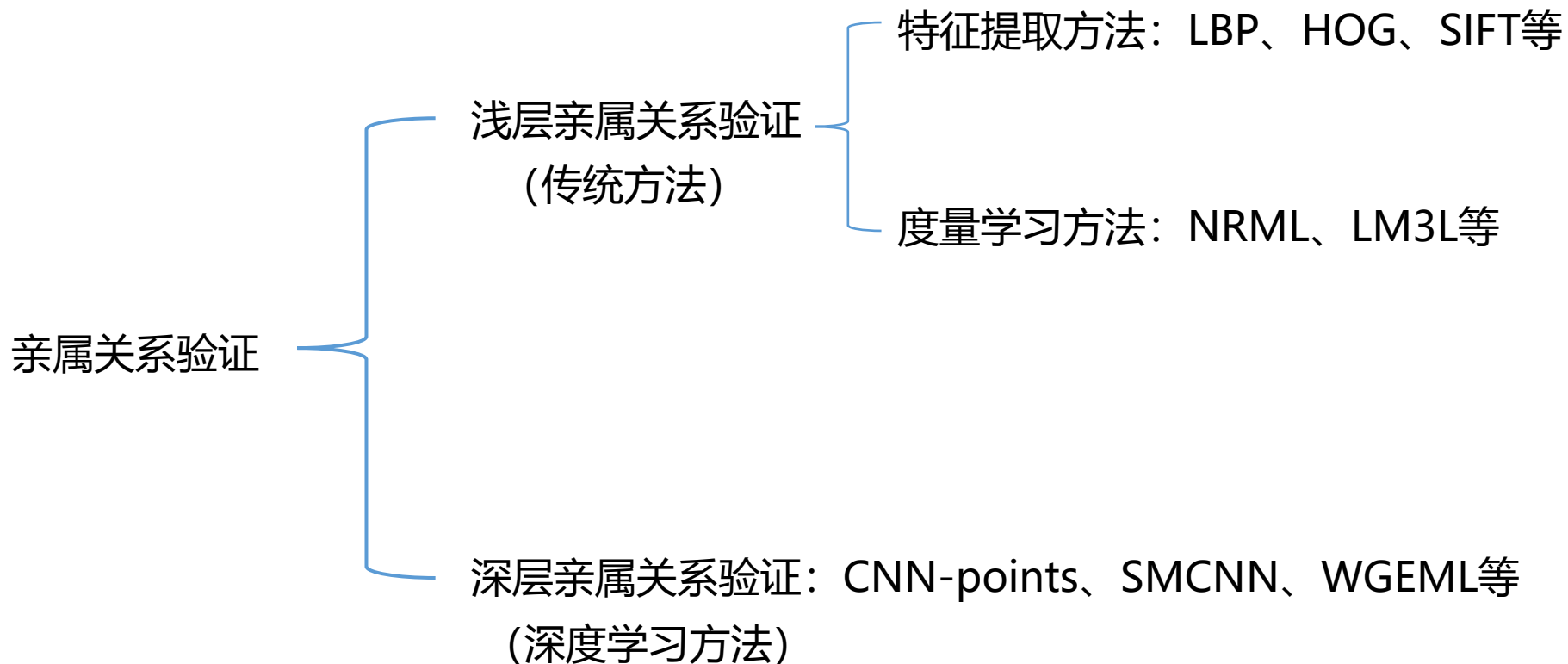


两大关键难题：

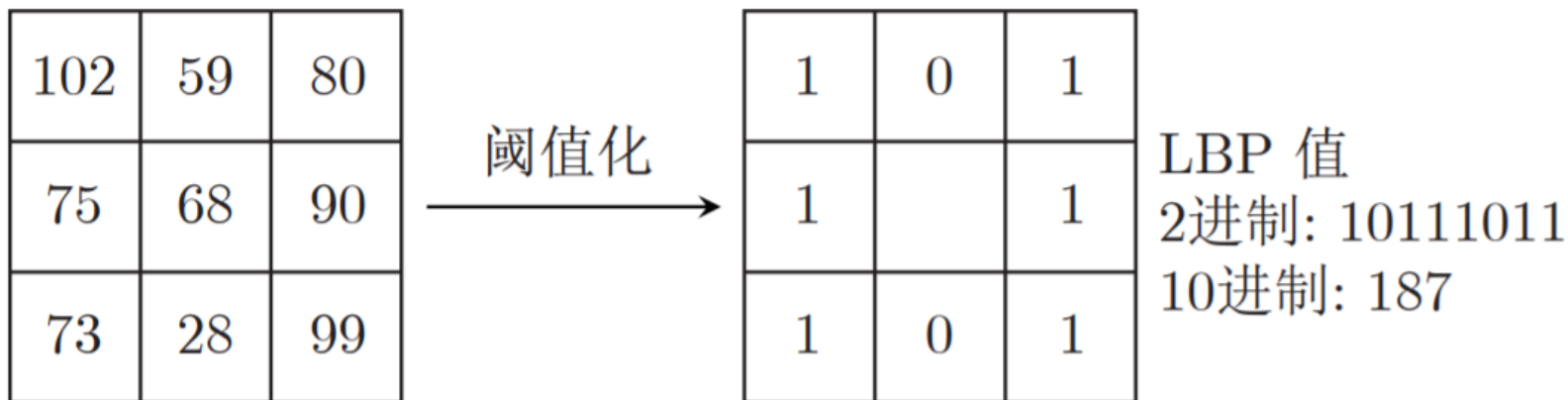
1. 亲属特征的有效表示方法
2. 正、负样本对的准确分离方法



AdvKin模型的诞生：一种运用GAN的对抗思想的双流共享卷积神经网络模型。



LBP^[2] (Local Binary Pattern,局部二值模式): 一种用来描述图像局部纹理特征的算子, 采用圆形领域编码方法。



[2] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification," IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 2, pp. 331–345, Feb. 2014.

LBP的计算公式:

$$LBP_{P,R}(x_c, y_c) = \sum_{P=0}^{P-1} s(g_p - g_c) 2^P$$
$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

通常, 在 LBP 特征提取前会将图像进行分块, 计算每个子块中每个像素的 LBP 值, 对子块进行直方图统计得到该子图像块的 LBP 值, 利用子图像块的 LBP 值即可以描述该图像的纹理特征。

HOG^[6] (Histogram of Oriented Gradient,方向梯度直方图):一种特殊性描述子, 通过计算图像局部区域的梯度方向直方图来构成特征。

梯度值的计算公式:

$$G_x = U(x+1, y) - U(x-1, y)$$

$$G_y = U(x, y+1) - U(x, y-1)$$

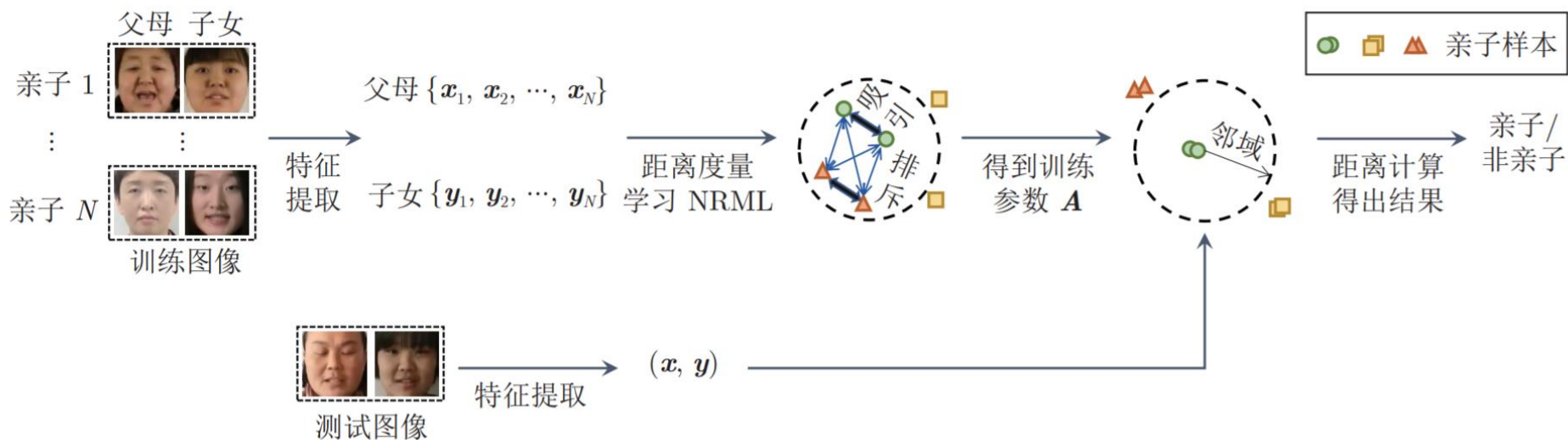
$$N(x, y) = \sqrt{G_x^2 + G_y^2}$$

梯度方向的计算公式:

$$\theta(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

[6] R. Fang, K. D. Tang, N. Snavely, and T. Chen, "Towards computational models of kinship verification," in Proc. ICIP, 2010, pp. 1577-1580.

NRML^[2] (Neighborhood repulsed metric learning, 领域驳斥度量学习)算法的思想是, 正样本相邻的负样本会对分类器产生干扰, 因此方法对正样本 k 个邻域中的负样本进行排斥, 同时使正样本互相吸引, 从而分离正负样本。



NRML框架图

亲属特征采用 m 维列向量表示,两者之间的距离用马氏距离表示:

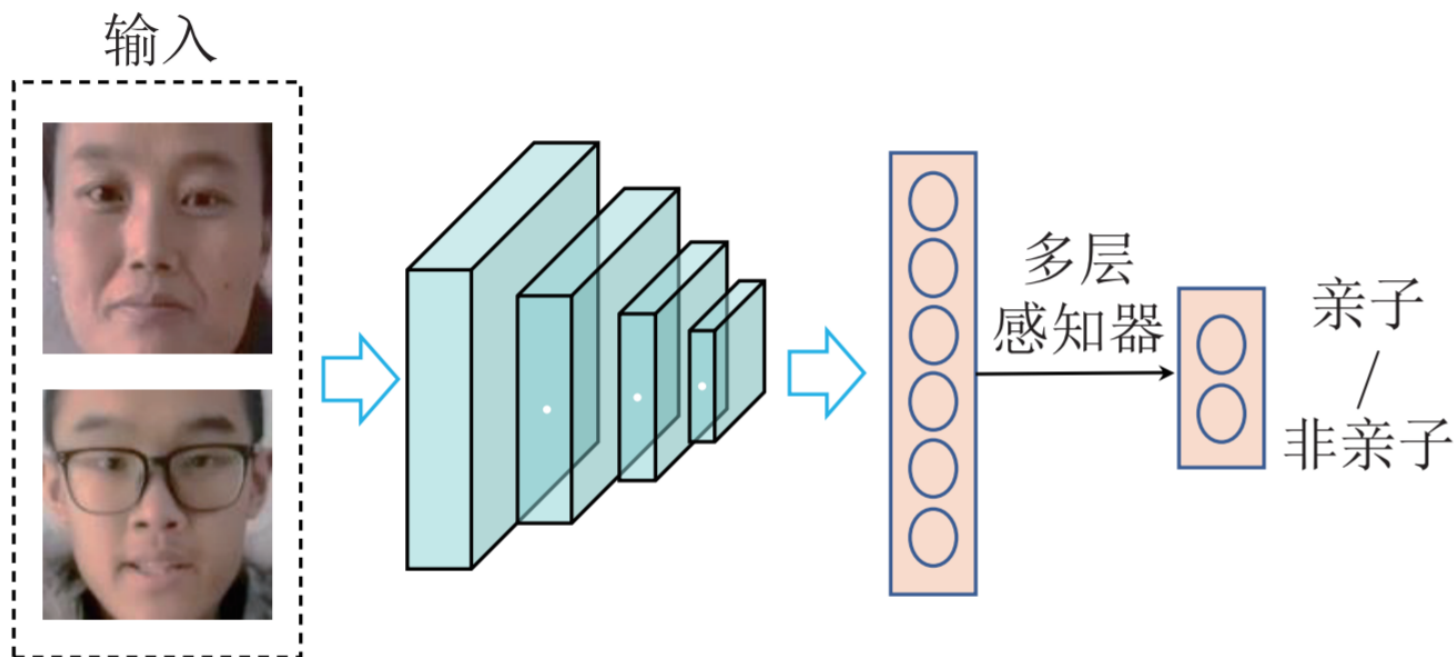
$$d(\mathbf{x}_i, \mathbf{y}_j) = \sqrt{(\mathbf{x}_i - \mathbf{y}_j)^T \mathbf{A} (\mathbf{x}_i - \mathbf{y}_j)}$$

\mathbf{A} 是一个 $m \times m$ 维的正定矩阵, 找到一个合适的 \mathbf{A} , 使 $i=j$ 时的距离尽可能小; $i \neq j$ 时, 距离尽可能大。

对此, NRML对 \mathbf{A} 构造了如下目标函数:

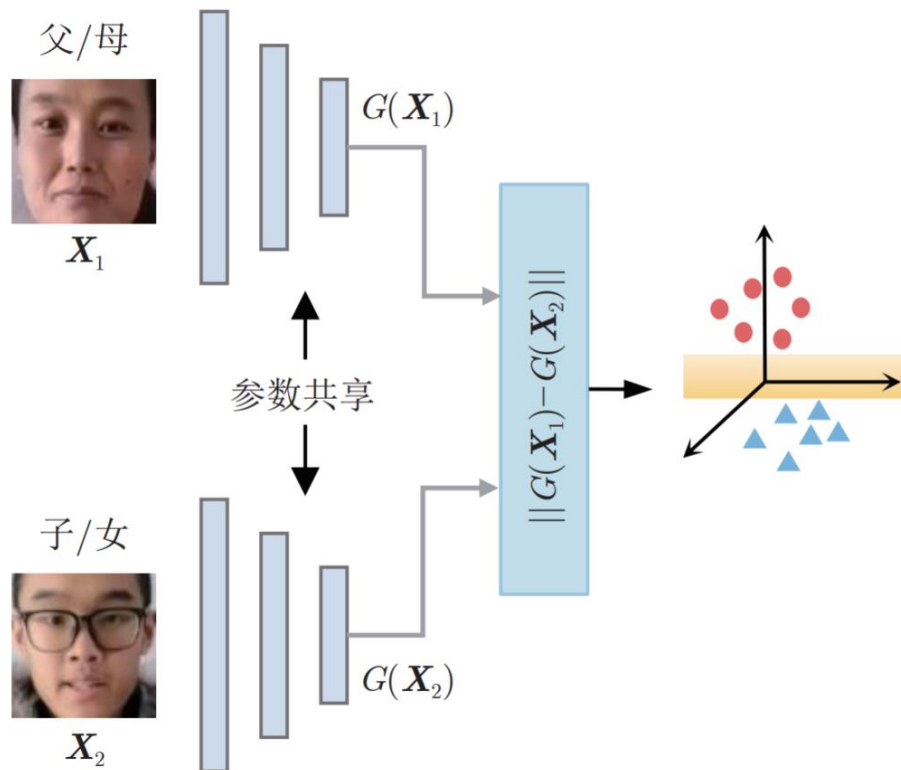
$$\begin{aligned}\max_{\mathbf{A}} J(\mathbf{A}) = J_1(\mathbf{A}) + J_2(\mathbf{A}) - J_3(\mathbf{A}) = & \frac{1}{Nk} \sum_{i=1}^N \sum_{t_1=1}^k d^2(\mathbf{x}_i, \mathbf{y}_{it_1}) + \\ & \frac{1}{Nk} \sum_{i=1}^N \sum_{t_2=1}^k d^2(\mathbf{x}_{it_2}, \mathbf{y}_i) - \\ & \frac{1}{N} \sum_{i=1}^N d^2(\mathbf{x}_i, \mathbf{y}_i)\end{aligned}$$

Zhang等^[26] 开启了基于端到端 (End-to-end) 深度学习的人脸亲子关系验证方法 (CNN-points):



CNN-points框架图

基于相似度量的卷积神经网络^[15] (Similarity metric based convolutional neural networks, SMCNN)



SMCNN框架图

[15] L. Li, X. Feng, X. Wu, Z. Xia, and A. Hadid, "Kinship verification from faces via similarity metric based convolutional neural network," in Proc. ICIAR, 2016, pp. 539–548.

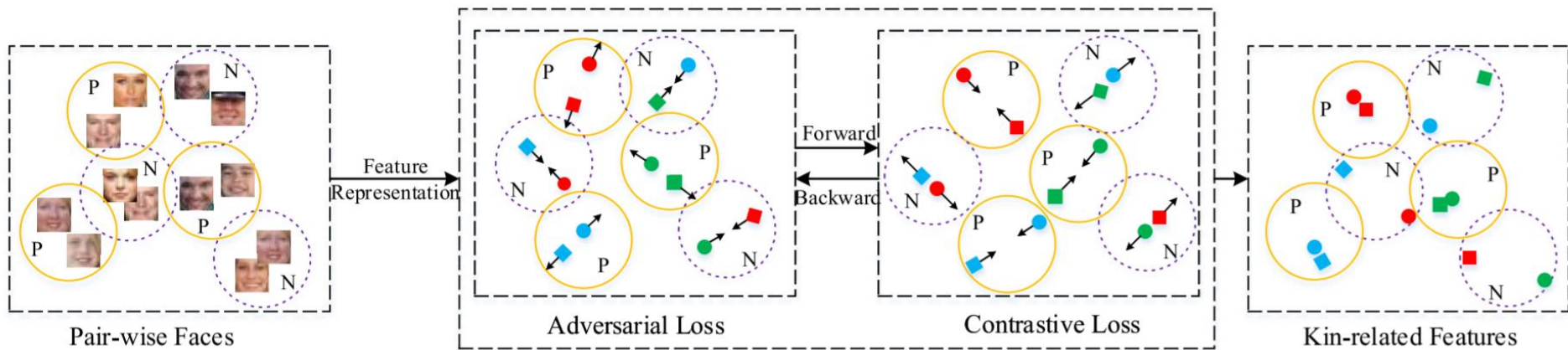
两幅图像经过参数共享的网络得到特征的嵌入表示, 特征间的距离用L1范数表示:

$$D(\mathbf{X}_1, \mathbf{X}_2) = \|G(\mathbf{X}_1) - G(\mathbf{X}_2)\|_1$$

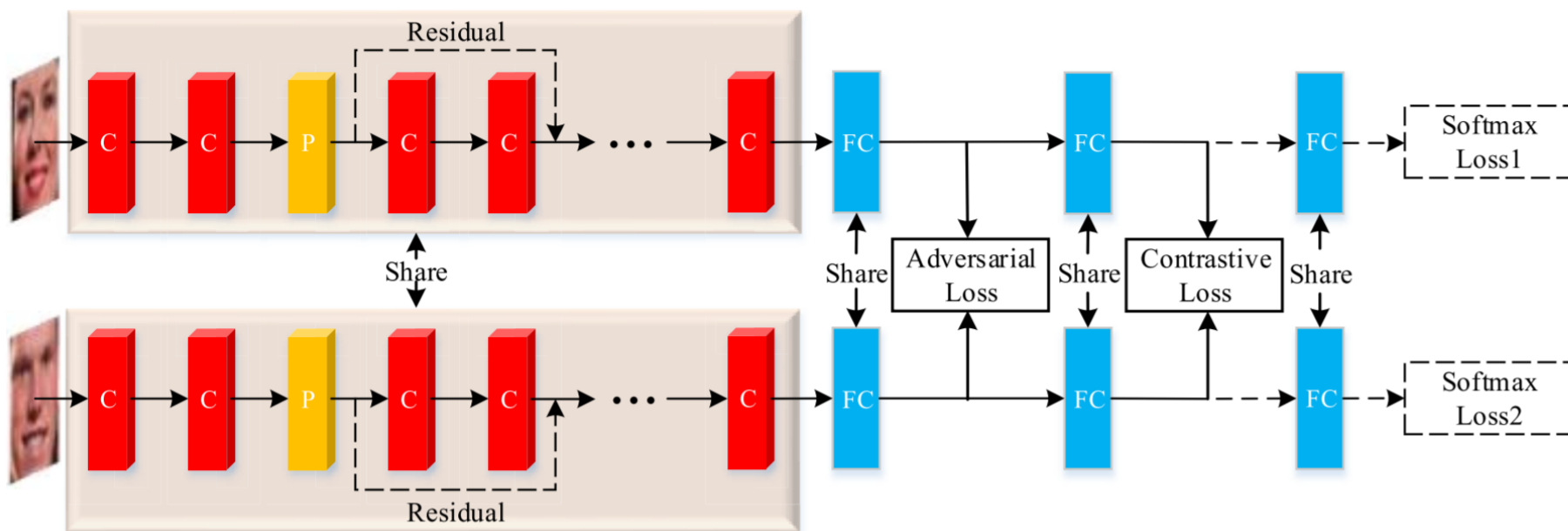
为了更好区分正负样本对,设置阈值参数和正负样本对标签,得到对比损失函数 (Contrastive loss):

$$L_c = yd^2 + (1 - y)(\alpha - d)_+^2$$

$$(x)_+ = \max(x, 0)$$



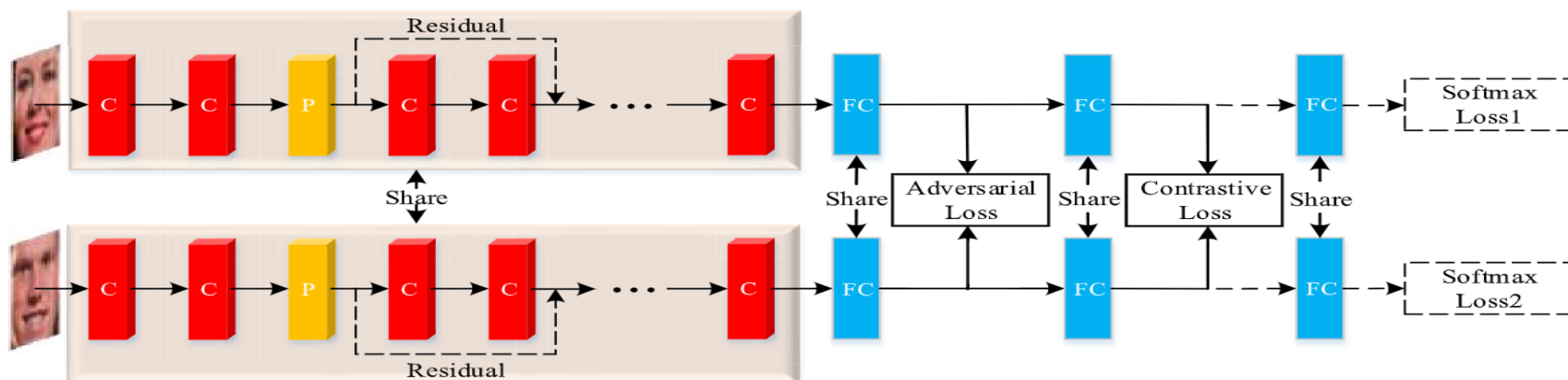
AdvKin的核心思想



AdvKin模型框架



实现原理



基于家族ID的对比损失(Contrastive Loss)

VS

基于家族ID的对抗性损失(Adversarial Loss)

$$\min L_C = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) d^2 + \delta(y_n^1 \neq y_n^2) \max(\text{margin} - d, 0)^2 \right)$$

- ◆ 作用: 最小化类内差异, 最大化类间差异
- ◆ 目的: 增强模型提取亲属共同特征能力

$$\min L_A = -\frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_{\tilde{h}}^2 - \delta(y_n^1 \neq y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_{\tilde{h}}^2 \right)$$

- ◆ 作用: 最大化类内差异, 最小化类间差异
- ◆ 目的: 放大亲属差异特征, 引入自对抗机制

基于家族ID的对抗性损失

最大均值差异MMD

$$\text{MMD}^2(s, t) = \sup_{\|\phi\|_h \leq 1} \|E_{\mathbf{x}^s \sim s}[\phi(\mathbf{x}^s)] - E_{\mathbf{x}^t \sim t}[\phi(\mathbf{x}^t)]\|_h^2.$$

- ◆ 将低维数据映射到高维希尔伯特空间，以评估两个分布之间的相似度



基于MMD的成对损失

$$\min L_{\text{MMD}} = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) \|\phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2)\|_h^2 - \delta(y_n^1 \neq y_n^2) \|\phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2)\|_h^2 \right).$$

- ◆ 降低不同亲属域间分布差异，方向与对比性损失一致



基于NMMD的自对抗损失

$$\min L_A = -\frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) \|\phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2)\|_h^2 - \delta(y_n^1 \neq y_n^2) \|\phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2)\|_h^2 \right)$$

- ◆ 受GAN模型对抗机制的启发，通过构造NMMD损失，引入自对抗机制
- ◆ 对抗性损失AL与对比性损失CL相结合，促使模型学习更多判别特征

基于家族ID的对抗性损失

$$\min L_A = -\frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_h^2 - \delta(y_n^1 \neq y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_h^2 \right)$$

↓ $\phi(\cdot)$ 映射函数展开

$$L_A = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 \neq y_n^2) - \delta(y_n^1 = y_n^2) \right) \times \left(k(\mathbf{x}_n^1, \mathbf{x}_n^1) + k(\mathbf{x}_n^2, \mathbf{x}_n^2) - 2k(\mathbf{x}_n^1, \mathbf{x}_n^2) \right)$$

↓ 高斯核函数展开

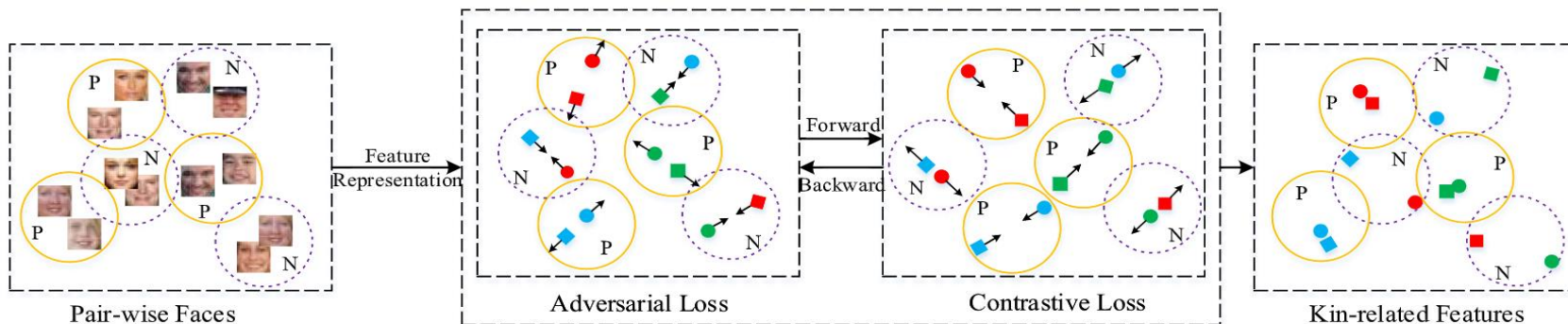
$$L_A = \frac{1}{N} \sum_{n=1}^N \left(\delta(y_n^1 \neq y_n^2) - \delta(y_n^1 = y_n^2) \right) \times \left(1 - \exp\left(-\frac{\|\mathbf{x}_n^1 - \mathbf{x}_n^2\|_2^2}{2\sigma^2}\right) \right).$$

对抗性损失 L_A 梯度计算:

$$\begin{aligned} \frac{\partial L_A}{\partial \mathbf{x}_n^1} &= \frac{1}{N\sigma^2} \left(\delta(y_n^1 \neq y_n^2) - \delta(y_n^1 = y_n^2) \right) \\ &\quad \times \exp\left(-\frac{\|\mathbf{x}_n^1 - \mathbf{x}_n^2\|_2^2}{2\sigma^2}\right) (\mathbf{x}_n^1 - \mathbf{x}_n^2) \\ \frac{\partial L_A}{\partial \mathbf{x}_n^2} &= \frac{1}{N\sigma^2} \left(\delta(y_n^1 \neq y_n^2) - \delta(y_n^1 = y_n^2) \right) \\ &\quad \times \exp\left(-\frac{\|\mathbf{x}_n^1 - \mathbf{x}_n^2\|_2^2}{2\sigma^2}\right) (\mathbf{x}_n^2 - \mathbf{x}_n^1) \end{aligned}$$

- ◆ 上式表明整个模型可以采用最常用的反向传播算法(如SGD)更新网络参数

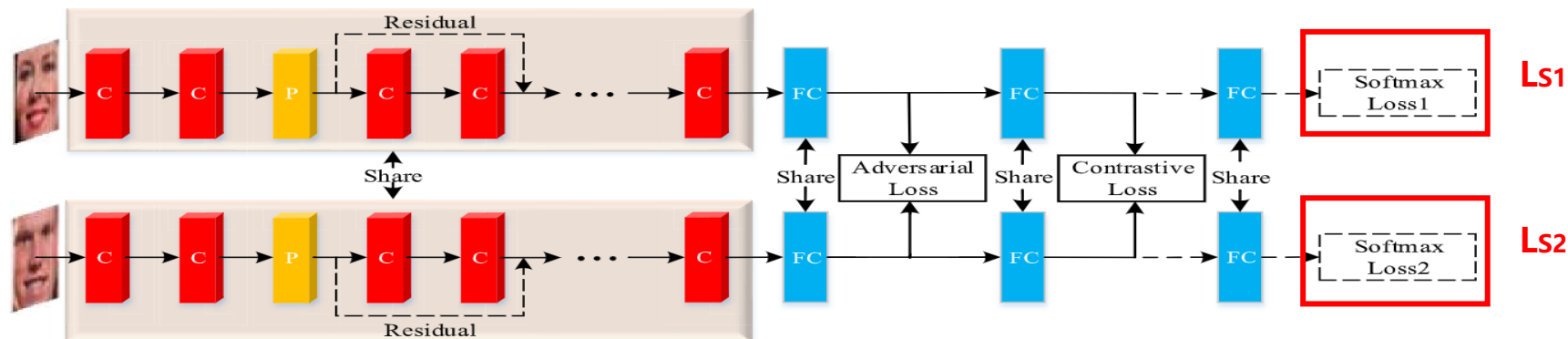
整体损失函数



$$\begin{aligned}
 L &= L_C + \lambda L_A \\
 &= \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) d^2 + \delta(y_n^1 \neq y_n^2) \right. \\
 &\quad \times \max(\text{margin} - d, 0)^2 \Big) \\
 &\quad - \lambda \left(\frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) - \delta(y_n^1 \neq y_n^2) \right) \right. \\
 &\quad \times \left. \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_h^2 \right)
 \end{aligned}$$

- ◆ 通过在AL对抗层中生成“硬特征”，即在特征空间中排斥相似对，吸引不相似对，促使网络更加仔细地学习CL对比层
- ◆ 本质上，该模型通过增加训练难度来改进网络性能的

基于家族ID的Softmax联合损失

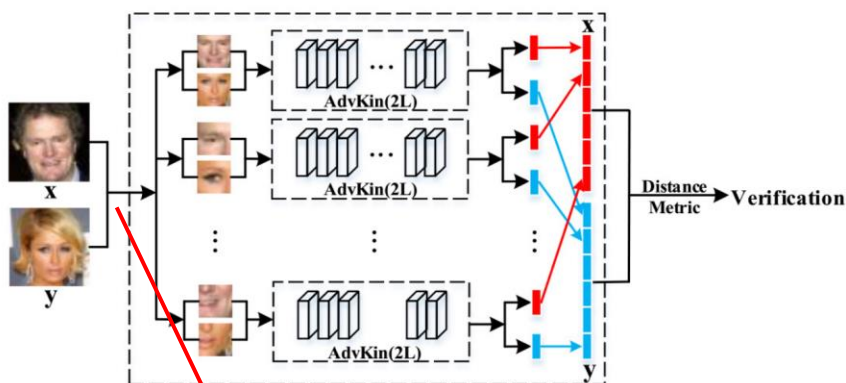


$$L = L_C + \lambda L_A + L_{S1} + L_{S2}$$

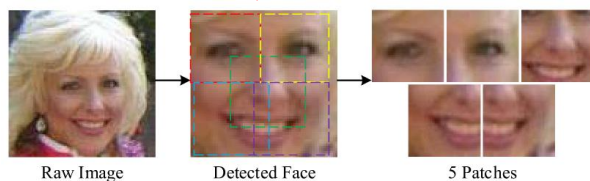
- ◆ 对于大规模亲属关系验证，输入数据中还包含明显的家族类别信息，这种类别特征可以通过引入Softmax损失，促进模型进一步发现深层特征的隐式亲缘关系
- ◆ L_{S1} 与 L_{S2} 分别表示输入 x_n^1 与 x_n^2 的交叉熵损失

针对不同规模亲属验证的E-AdvKin网络

小规模亲属验证 Patch-Wise E-AdvKin

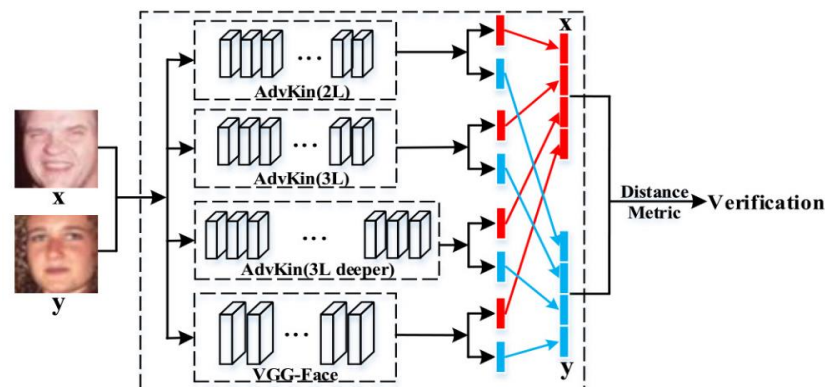


关键点分割



- ◆ Patch-Wise E-AdvKin包含6个深度相同AdvKin网络, 每个AdvKin网络产生80-D亲属相关的深度特征

大规模亲属验证 E-AdvKin

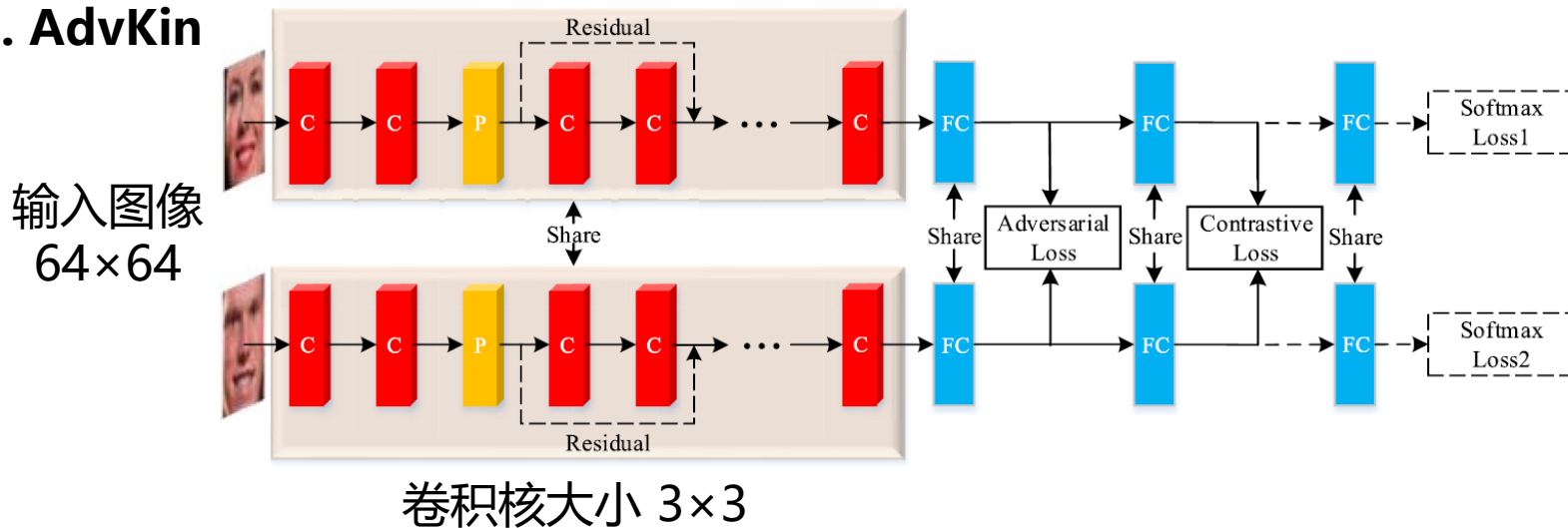


- ◆ 大规模亲属验证E-AdvKin由3个损失与深度不同的AdvKin网络和一个VGG-Face网络构成
- ◆ 引入VGG-Face网络增强E-AdvKin对于卷积网络中呈现层次分布特征的提取能力

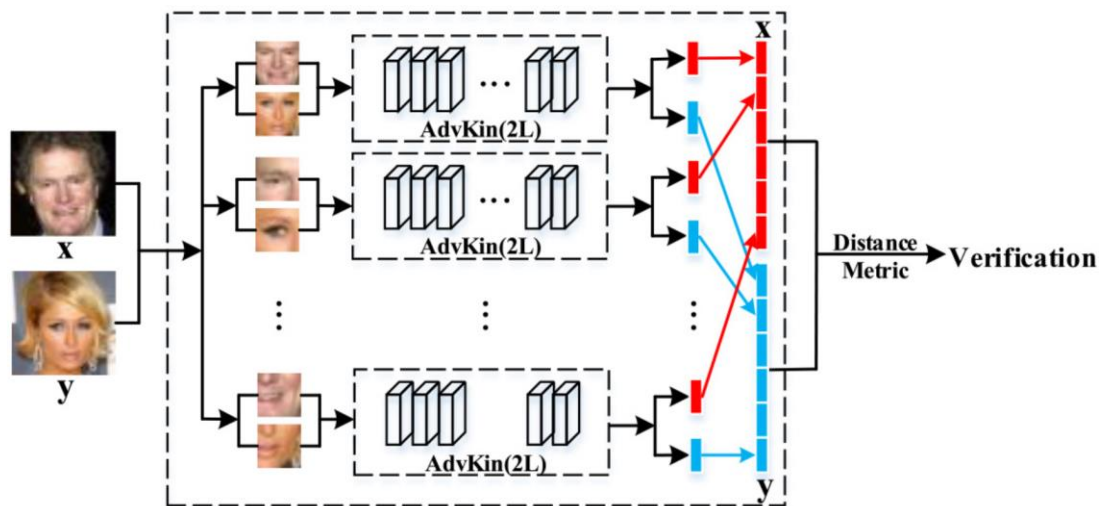


实验部分

1. AdvKin



2. E-AdvKin



四种小规模亲属关系验证任务基准：

- ◆ KinFaceW-I
- ◆ KinFaceW-II
- ◆ Cornell KinFace
- ◆ UB KinFace (三元组)

包含四种不同类型的亲属关系：

- ◆ F-S (父子)
- ◆ F-D (父女)
- ◆ M-S (母子)
- ◆ M-D (母女)



F-D

F-S



M-D

M-S

- ◆ 模型性能评估方法：5折交叉验证
- ◆ 评估指标：5折的平均准确率
- ◆ 亲属关系距离度量：余弦距离

K折交叉验证

是机器学习中一种常用的模型评估技术。它将原始数据集分成K个子集，称为折（fold），在每次训练中，其中一个折被作为验证集，而剩下的K-1个折被用作训练集。模型在K个不同的训练集上进行K次训练和验证。

Fold	KinFaceW-I				KinFaceW-II	UB	Cor
	F-S	F-D	M-S	M-D	all subset	all subset	all subset
1	[1,31]	[1,27]	[1,23]	[1,25]	[1,50]	[1,40]	[1,29]
2	[32,62]	[28,54]	[24,46]	[26,50]	[51,100]	[41,80]	[30,58]
3	[63,93]	[55,81]	[47,69]	[51,75]	[101,150]	[81,120]	[59,87]
4	[94,124]	[82,108]	[70,92]	[76,100]	[151, 200]	[121,160]	[88,115]
5	[125,156]	[109,134]	[93,116]	[101,127]	[201,250]	[161, 200]	[116, 143]

四个数据集每折的图像索引集

- ◆ 模型性能评估方法：5折交叉验证
- ◆ 评估指标：5折的平均准确率
- ◆ 亲属关系距离度量：余弦距离
- ◆ 超参数设置：

1. 优化器：mini-batch SGD
2. 初始学习率： 10^{-2}
3. 批量大小：151
4. CL中的阈值 margin：1

$$\min L_C = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) d^2 + \delta(y_n^1 \neq y_n^2) \max(\text{margin} - d, 0)^2 \right) \quad (1)$$



[2] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, and J. Zhou, "Neighborhood repulsed metric learning for kinship verification," IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 2, pp. 331–345, Feb. 2014.

[15] L. Li, X. Feng, X. Wu, Z. Xia, and A. Hadid, "Kinship verification from faces via similarity metric based convolutional neural network," in Proc. ICIAR, 2016, pp. 539–548.

[25] M. Wang, Z. Li, X. Shu, and J. Wang, "Deep kinship verification," in Proc. IEEE Int. Workshop MSP, 2015, pp. 1–6.

[26] K. Zhang, Y. Huang, C. Song, H. Wu, and L. Wang, "Kinship verification with deep convolutional neural networks," in Proc. BMVC, 2015, pp. 1–12.

[27] M. Dawson, A. Zisserman, and C. Nellaker, "From same photo: Cheating on visual kinship challenges," in Proc. ACCV, 2018, pp. 654–668.

[32] H. Yan, J. Lu, and X. Zhou, "Prototype-based discriminative feature learning for kinship verification," IEEE Trans. Cybern., vol. 45, no. 11, pp. 2535–2545, Nov. 2015.

[33] X. Zhou, Y. Shang, H. Yan, and G. Guo, "Ensemble similarity learning for kinship verification from facial images in the wild," Inf. Fusion, vol. 32, pp. 40–48, Nov. 2016.

[35] H. Yan, "Learning discriminative compact binary face descriptor for kinship verification," Pattern Recognit. Lett., vol. 117, pp. 146–152, Jan. 2019.

[43] J. Liang, Q. Hu, C. Dang, and W. Zuo, "Weighted graph embeddingbased metric learning for kinship verification," IEEE Trans. Image Process., vol. 28, no. 3, pp. 1149–1162, Mar. 2019.

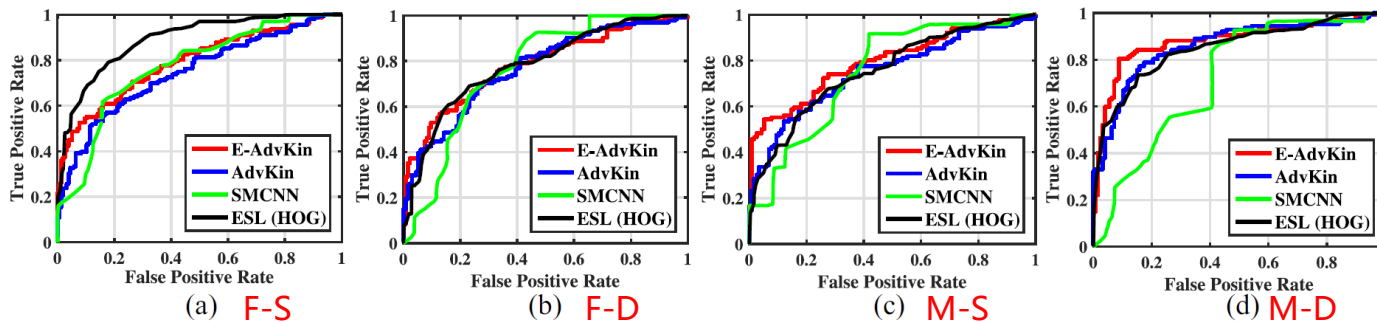
[56] J. Lu, J. Hu, and Y.-P. Tan, "Discriminative deep metric learning for face and kinship verification," IEEE Trans. Image Process., vol. 26, no. 9, pp. 4269–4282, Sep. 2017.

Methods	KinFaceW-I					KinFaceW-II					UB			Cor
	F-S	F-D	M-S	M-D	Mean	F-S	F-D	M-S	M-D	Mean	0-1	0-2	Mean	-
Human A [32]	62.0	60.0	68.0	72.0	65.6	63.0	63.0	71.0	75.0	68.0	-	-	-	-
Human B [32]	68.0	66.5	74.0	75.0	70.9	72.0	72.5	77.0	80.0	75.4	-	-	-	-
MNRML [2]	72.5	66.5	66.2	72.0	69.9	76.9	74.3	77.4	77.6	76.5	67.3	66.8	67.1	71.6
MPDFL [32]	73.5	67.5	66.1	73.1	70.1	77.3	74.7	77.8	78.0	77.0	67.5	67.0	67.3	71.9
ESL (HOG) [33]	<u>83.9</u>	76.0	73.5	81.5	78.6	81.2	73.0	75.6	73.0	75.7	-	-	-	-
D-CBFD [35]	79.6	73.6	76.1	81.5	77.6	79.0	74.2	75.4	77.3	78.5	-	-	-	-
SMCNN [15]	75.0	75.0	68.7	72.2	72.7	75.0	79.0	78.0	85.0	79.3	-	-	-	-
DKV [25]	71.8	62.7	66.4	66.6	66.9	73.4	68.2	71.0	72.8	71.3	-	-	-	-
CNN-Points [26]	76.1	71.8	78.0	84.1	77.5	89.4	81.9	89.9	92.4	88.4	-	-	-	-
DDMML (All) [56]	86.4	79.1	81.4	87.0	83.5	87.4	83.8	83.2	83.0	84.3	-	-	-	-
FSP [27]	74.6	74.9	78.3	86.0	76.8	92.3	84.5	90.3	94.8	90.2	-	-	-	76.7
WGEML [43]	78.5	73.9	<u>80.6</u>	81.9	78.7	88.6	77.4	83.4	81.6	82.8	-	-	-	-
AdvKin	75.7	<u>78.3</u>	77.6	83.1	78.7	88.4	85.8	88.0	89.8	88.0	75.0	75.0	75.0	81.4
E-AdvKin	76.6	<u>77.3</u>	78.4	<u>86.2</u>	<u>79.6</u>	<u>91.6</u>	<u>85.2</u>	<u>90.2</u>	<u>92.4</u>	<u>89.9</u>	-	-	-	<u>80.4</u>

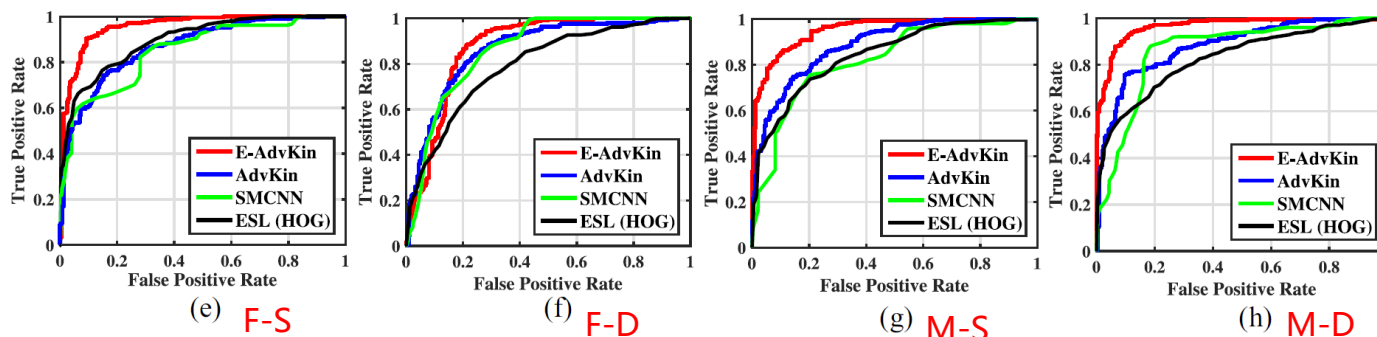
Note: The best results are highlighted in bold type, and the second-best results are underlined.

1. AdvKin优于采用手工特征集成和度量学习的最先进浅层方法;
2. AdvKin优于基于深度学习的人脸识别方法;
3. 在具有相似深度的架构下, 本文方法的性能总体上优于其他方法;
4. AdvKin方法也优于人类的评估结果;
5. E-AdvKin优于AdvKin, 因此可以通过patch的集成学习到更多细粒度的亲属相关特征。
6. 可以通过从人脸识别到亲属关系验证任务的迁移学习来提高性能。

KinFaceW-I



KinFaceW-II



KinFaceW-I 和 KinFaceW-II数据集上结果的ROC曲线

ROC曲线 (Receiver Operating Characteristic Curve) 是一种描绘分类器性能的图形工具，它显示了分类器的**真阳性率** (True Positive Rate, TPR) 和**假阳性率** (False Positive Rate, FPR) 之间的关系。

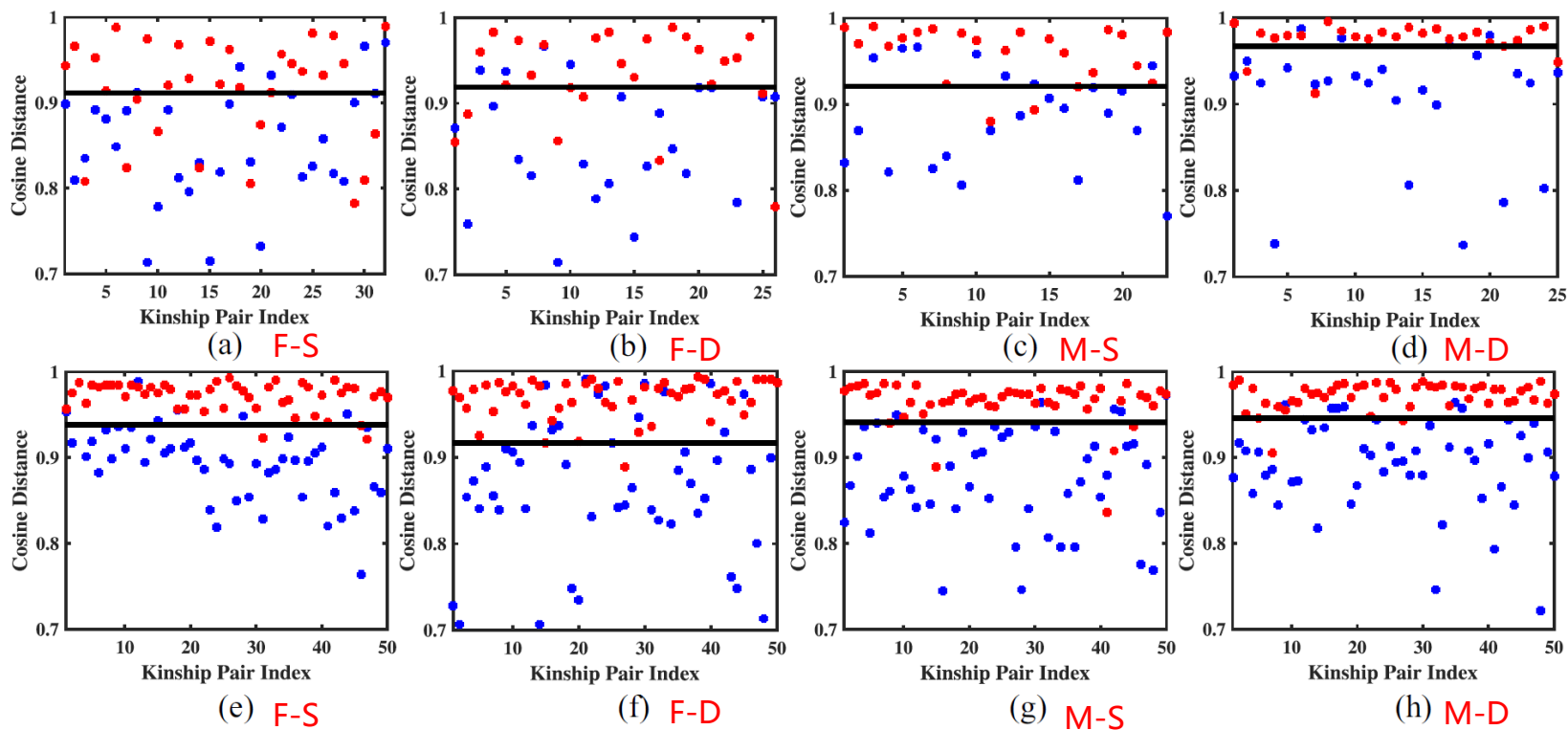
		预测值	
		Positive	Negative
真实值	True	TP	TN
	False	FP	FN

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{TN + FP}$$

余弦距离可视化

KinFaceW-I



KinFaceW-II

● 亲属对

● 非亲属对

Methods	KinFaceW-I					KinFaceW-II					UB			Cor
	F-S	F-D	M-S	M-D	Mean	F-S	F-D	M-S	M-D	Mean	0-1	0-2	Mean	-
CL	74.7	77.6	72.4	81.1	76.5	85.8	85.8	84.0	83.8	84.9	58.3	60.0	59.2	76.2
ML+CL	77.3	74.6	78.0	83.6	78.4	85.8	84.6	86.6	88.0	86.3	59.8	61.0	60.4	78.3
AdvKin	<u>75.7</u>	78.3	<u>77.6</u>	<u>83.1</u>	78.7	88.4	85.8	88.0	89.8	88.0	75.0	75.0	75.0	81.4

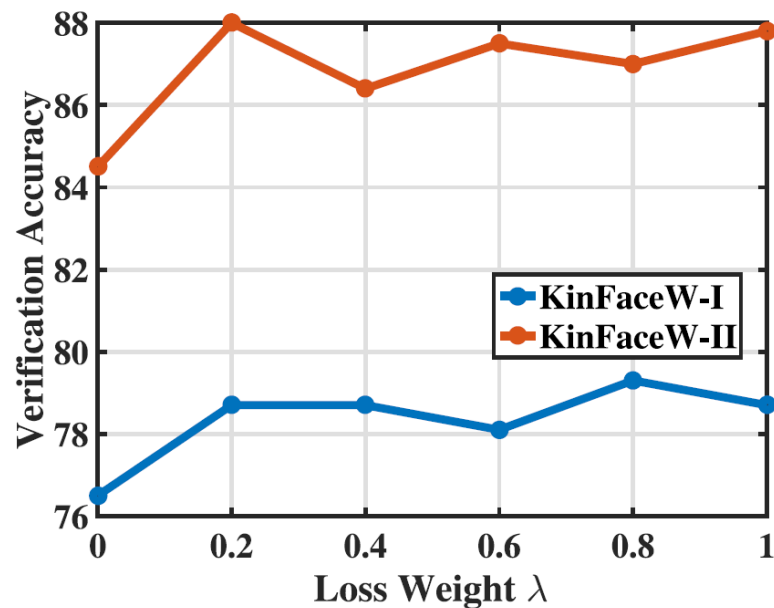
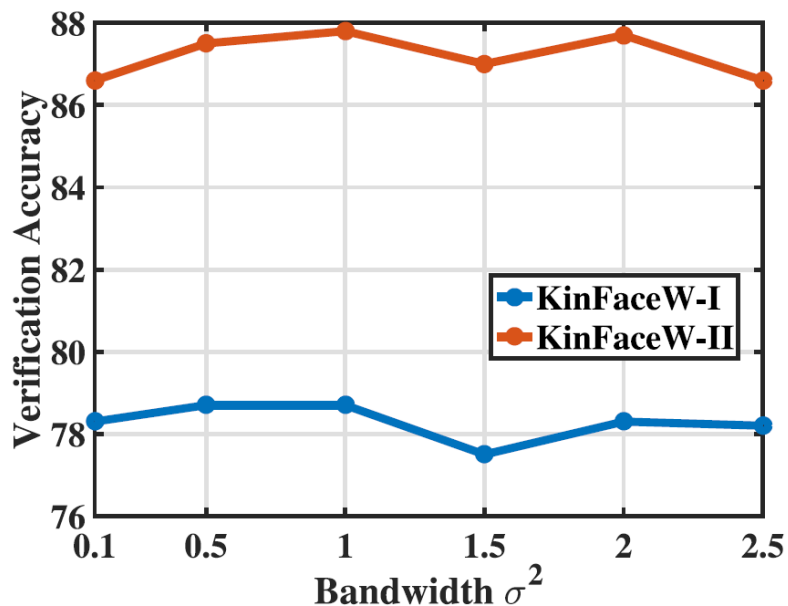
◆ CL :
$$\min L_C = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) d^2 + \delta(y_n^1 \neq y_n^2) \max(\text{margin} - d, 0)^2 \right) \quad (1)$$

◆ ML (基于MMD) :

$$\min L_{\text{MMD}} = \frac{1}{2N} \sum_{n=1}^N \left(\delta(y_n^1 = y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_{\hbar}^2 - \delta(y_n^1 \neq y_n^2) \left\| \phi(\mathbf{x}_n^1) - \phi(\mathbf{x}_n^2) \right\|_{\hbar}^2 \right). \quad (3)$$

Methods	KinFaceW-I					KinFaceW-II				
	F-S	F-D	M-S	M-D	Mean	F-S	F-D	M-S	M-D	Mean
PDFL (LE) [32]	68.2	63.5	61.3	69.5	65.6	77.0	74.3	77.0	77.2	76.4
CNN-Basic [26]	75.7	70.8	73.4	79.4	74.8	<u>84.9</u>	79.6	88.3	<u>88.5</u>	<u>85.3</u>
DDML (LPQ) [56]	83.8	<u>77.0</u>	<u>78.1</u>	86.6	81.4	84.8	<u>82.6</u>	79.4	81.8	82.2
WGEML (CNN) [43]	<u>77.0</u>	69.1	78.8	78.7	75.9	83.4	75.2	80.2	79.9	79.7
AdvKin	<u>75.7</u>	78.3	77.6	<u>83.1</u>	<u>78.7</u>	88.4	85.8	<u>88.0</u>	89.8	88.0
MPDFL (Fusion) [32]	73.5	67.5	66.1	73.1	70.1	77.3	74.7	77.8	78.0	77.0
CNN-Points (Fusion) [26]	76.1	71.8	78.0	84.1	77.5	<u>89.4</u>	81.9	<u>89.9</u>	92.4	<u>88.4</u>
DDMML (Fusion) [56]	86.4	79.1	81.4	87.0	83.5	87.4	<u>83.8</u>	83.2	83.0	84.3
WGEML (Fusion) [43]	<u>78.5</u>	73.9	<u>80.6</u>	81.9	78.7	88.6	77.4	83.4	81.6	82.8
E-AdvKin (Fusion)	<u>76.6</u>	<u>77.3</u>	78.4	<u>86.2</u>	<u>79.6</u>	91.6	85.2	90.2	92.4	89.9

- ◆ 除 DDMMML 外，AdvKin 在 KinFaceW-I 上的表现仍然优于其他方法
- ◆ E-AdvKin 表现最好 (89.9%)



- ◆ 当 σ^2 设置为 1.0 时, AdvKin 获得了最佳性能
- ◆ 当 λ 设置为 0.2 时, AdvKin 获得了最佳性能

输入图像
224×224



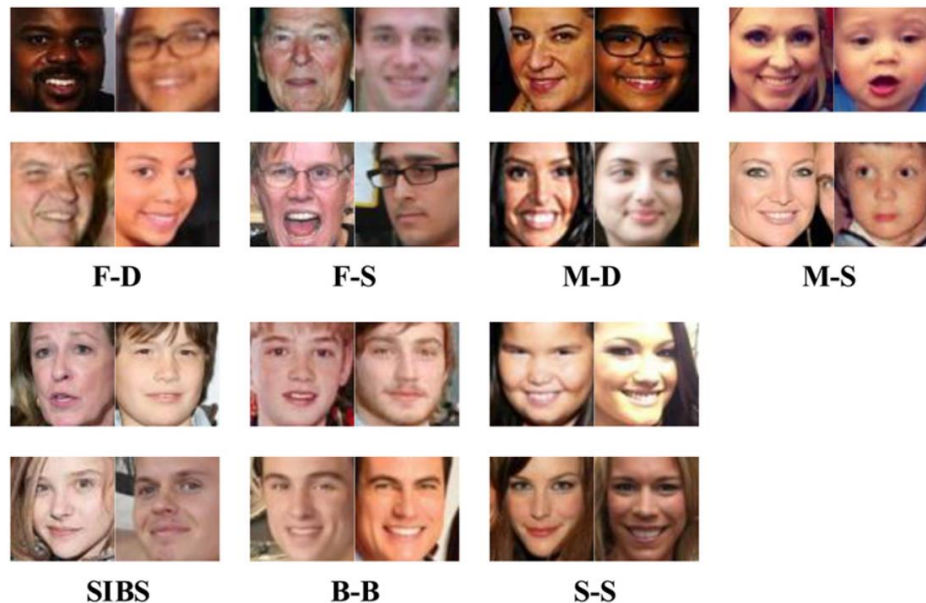
The diagram illustrates the proposed AdvKin framework for face verification. Two input face images, X and y , are fed into four parallel processing branches. Each branch contains a sequence of convolutional layers (represented by 3D blocks) and a final fully connected layer. The branches are labeled: AdvKin(2L), AdvKin(3L), AdvKin(3L deeper), and VGG-Face. The outputs of these branches are feature vectors, represented by red and blue vertical bars. These vectors are compared using a Distance Metric to produce the final Verification result.

大规模亲属关系验证任务数据集 FIW [9]:

- ◆ 来自 1001 个家庭的 12 000 多张照片

- ◆ 包含七种亲属关系:

F-D 父女、F-S 父子、M-D 母女、M-S 母子、SIBS 姐弟或兄妹、B-B 兄弟、S-S 姐妹



[9] J. P. Robinson, M. Shao, Y. Wu, and Y. Fu, "Families in the wild (FIW): Large-scale kinship image database and benchmarks," in Proc. ACM MM, 2016, pp. 242–246.

◆ 模型性能评估方法：划分为训练集、验证集和测试集（无标签）

数据集划分表（对）

	F-D	F-S	M-D	M-S	SIBS	B-B	S-S
训练集	42458	53974	34828	38312	40846	52482	19286
验证集	11460	13696	10698	9816	7434	17342	6218
测试集	23506	45988	20674	47954	15076	19946	6524

◆ 评估指标：测试集上的平均准确率

◆ 亲属关系距离度量：欧式距离

◆ 超参数设置：

1. 优化器：mini-batch SGD
2. 初始学习率： 10^{-2}
3. 批量大小：22
4. CL中的阈值 margin：1

与深度亲属验证模型的比较

Methods	M-D	M-S	S-S	B-B	SIBS	F-S	F-D	Mean
VGG-Face [40]	65.99	58.88	<u>74.59</u>	71.99	64.69	64.71	62.87	66.25
ResNet-29 [19]	59.55	59.08	51.74	64.81	59.39	58.21	56.54	58.47
ResNet-22(finetime) [11]	71.09	68.63	69.54	69.88	<u>69.54</u>	<u>67.73</u>	68.15	<u>69.22</u>
E-AdvKin	<u>69.93</u>	<u>67.33</u>	77.44	<u>71.76</u>	69.80	68.77	<u>67.82</u>	70.41

1. VGGFace^[40] 使用了 VGG-16 模型在 2622 名不同名人的 260 万张人脸图像上进行了预训练;
2. ResNet-29^[19] 是在 CASIA-WebFace 数据集上训练的 29 层残差卷积神经网络;
3. ResNet-22 (finetime) 是在 FIW 数据集上微调后的 ResNet-22 。

[40] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in Proc. BMVC, 2015, pp. 1–12.

[19] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in Proc. ECCV, 2016, pp. 499–515.

Index	Loss	Model	M-D	M-S	S-S	B-B	SIBS	F-S	F-D	Mean
0	CL	AdvKin (CL)	61.06	61.95	62.45	65.35	62.05	61.33	59.18	61.91
1	2L	AdvKin (2L)	60.50	64.07	64.17	63.76	61.99	62.23	60.53	62.46
2	3L	AdvKin (3L)	64.11	<u>65.65</u>	<u>64.53</u>	65.80	<u>64.82</u>	<u>63.42</u>	<u>63.18</u>	<u>64.50</u>
3	3L	AdvKin (3L deeper)	<u>63.56</u>	66.80	65.48	<u>65.77</u>	65.35	64.14	63.59	64.97
4	SL	VGG-Face [40]	65.99	58.88	74.59	71.99	64.69	64.71	62.87	66.25
1+2+3	Joint	E-AdvKin	64.20	67.55	65.71	66.82	66.45	64.78	64.04	65.65
2+3+4	Joint	E-AdvKin	70.07	65.60	77.52	<u>71.88</u>	<u>69.72</u>	68.79	<u>67.56</u>	<u>70.16</u>
1+2+3+4	Joint	E-AdvKin	<u>69.93</u>	<u>67.33</u>	<u>77.44</u>	71.76	69.80	<u>68.77</u>	67.82	70.41

$$L = L_C + \lambda L_A + \boxed{L_{S1} + L_{S2}} \quad \text{SL} \quad (10)$$

$$2L = CL + AL$$

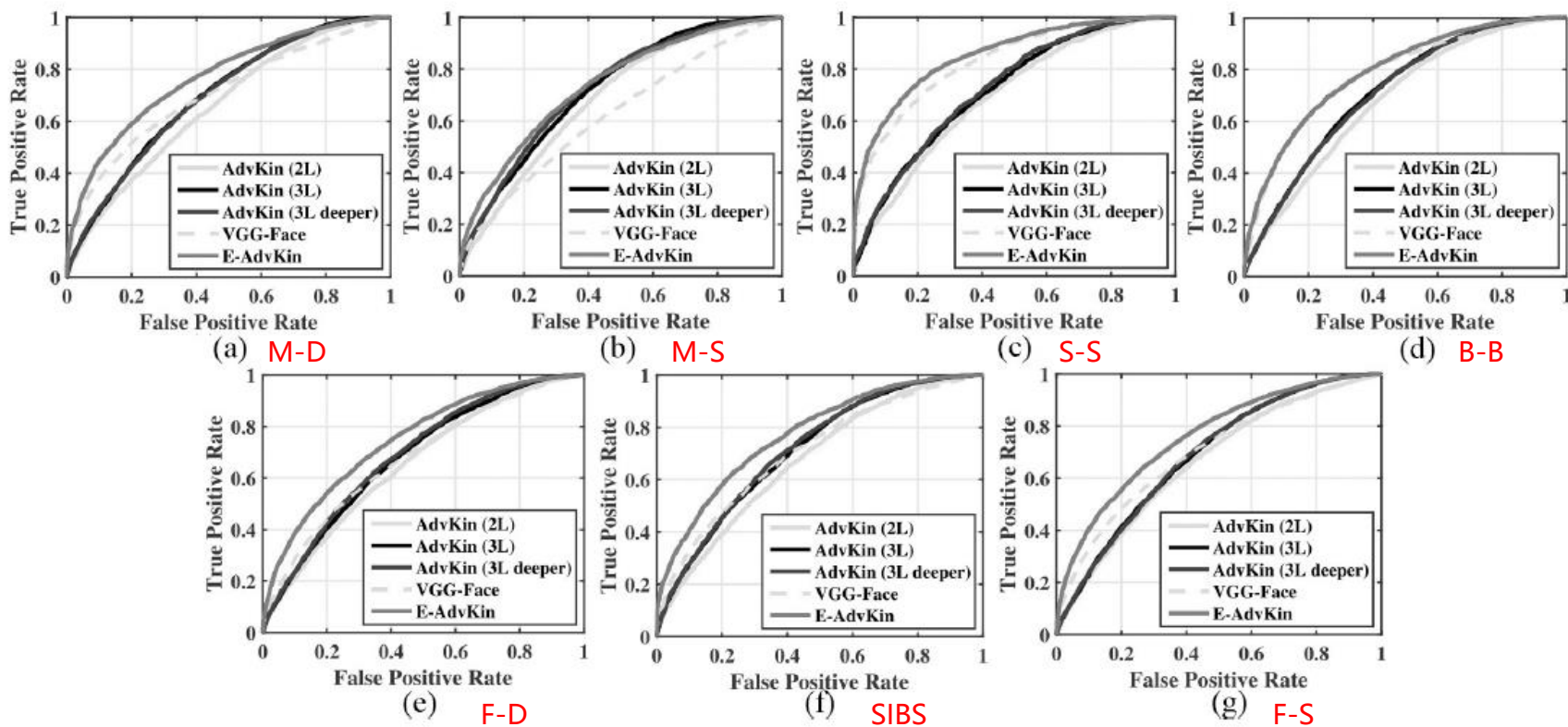
$$3L = CL + AL + SL$$

CNN	Conv1_x	Conv2_x	Conv3_x	Conv4_x	Conv5_x	Conv6_x	FC1	FC2	Softmax
AdvKin	$3 \times 3, 32$ $3 \times 3, 64$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 1$ $3 \times 3, 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ $3 \times 3, 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 5$ $3 \times 3, 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	-	1024	512	300
AdvKin (deeper)	$3 \times 3, 32$ $3 \times 3, 64$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 1$ $3 \times 3, 128$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$ $3 \times 3, 256$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 5$ $3 \times 3, 512$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	1024	512	300

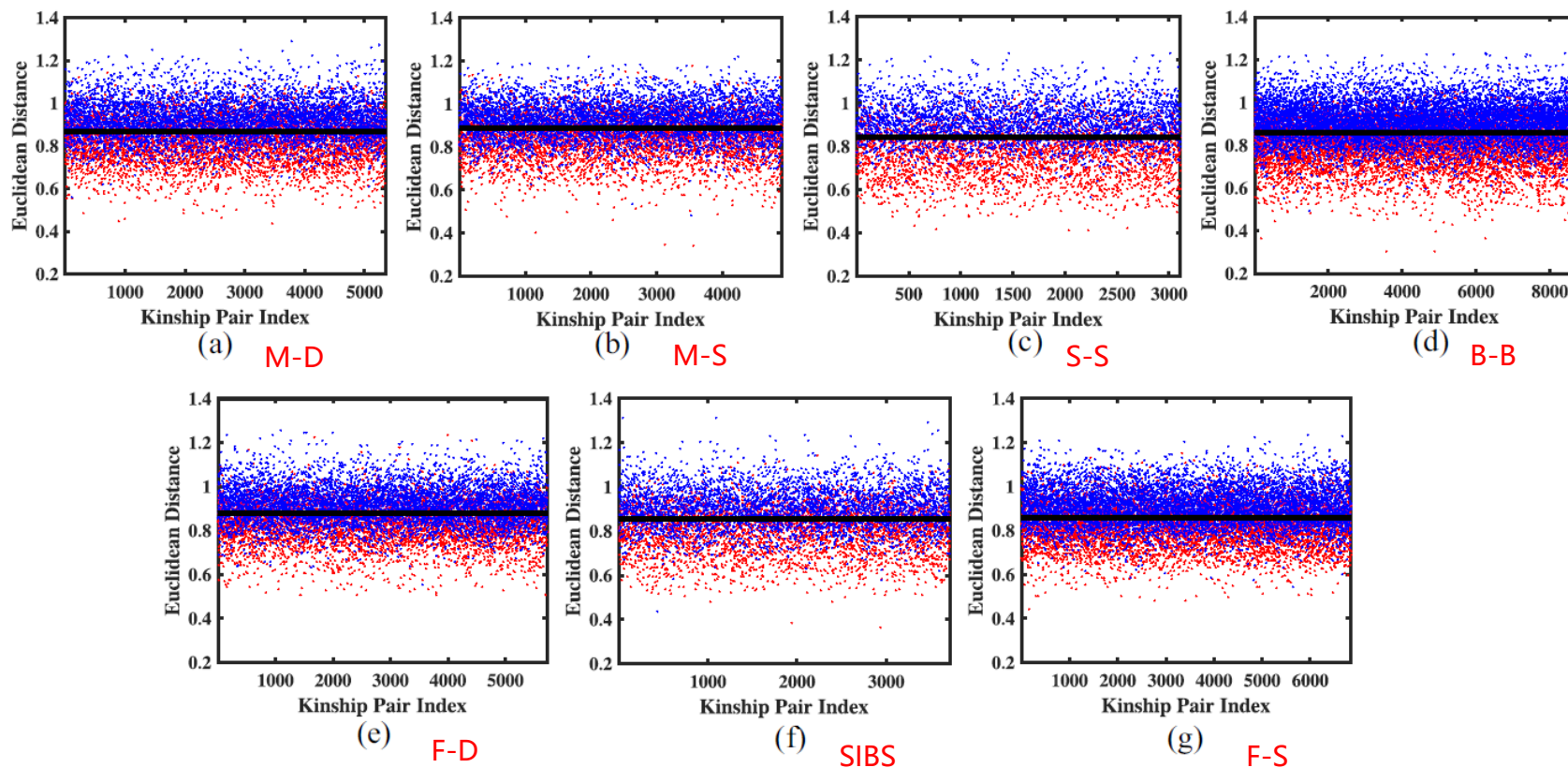
Index	Loss	Model	M-D	M-S	S-S	B-B	SIBS	F-S	F-D	Mean
0	CL	AdvKin (CL)	61.06	61.95	62.45	65.35	62.05	61.33	59.18	61.91
1	2L	AdvKin (2L)	60.50	64.07	64.17	63.76	61.99	62.23	60.53	62.46
2	3L	AdvKin (3L)	64.11	<u>65.65</u>	<u>64.53</u>	65.80	<u>64.82</u>	<u>63.42</u>	<u>63.18</u>	<u>64.50</u>
3	3L	AdvKin (3L deeper)	<u>63.56</u>	66.80	65.48	<u>65.77</u>	65.35	64.14	63.59	64.97
4	SL	VGG-Face [40]	65.99	58.88	74.59	71.99	64.69	64.71	62.87	66.25
1+2+3	Joint	E-AdvKin	64.20	67.55	65.71	66.82	66.45	64.78	64.04	65.65
2+3+4	Joint	E-AdvKin	70.07	65.60	77.52	<u>71.88</u>	<u>69.72</u>	68.79	<u>67.56</u>	<u>70.16</u>
1+2+3+4	Joint	E-AdvKin	<u>69.93</u>	<u>67.33</u>	<u>77.44</u>	71.76	69.80	<u>68.77</u>	67.82	70.41

- 索引 0、1、2、3 和 4 代表单一特征（无增强）
- 索引 1+2+3 和 2+3+4 代表其中三个模型的特征拼接在一起，增强特征的维度为 1536 (512×3)
- 索引 1+2+3+4 表示四个网络模型的特征都拼接在一起
- VGG-Face 模型提取的特征维度为 4096

各个模型的ROC曲线



正负样本对的欧氏距离可视化



● 亲属对

● 非亲属对

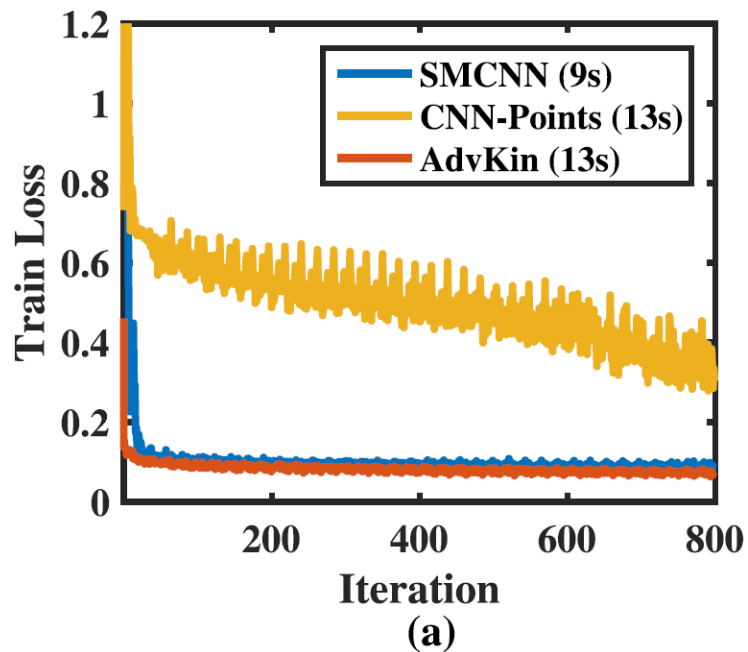
◆ 使用 E-AdvKin 和 VGG-Face 模型

◆ 测试集上的最终验证准确率分别为：

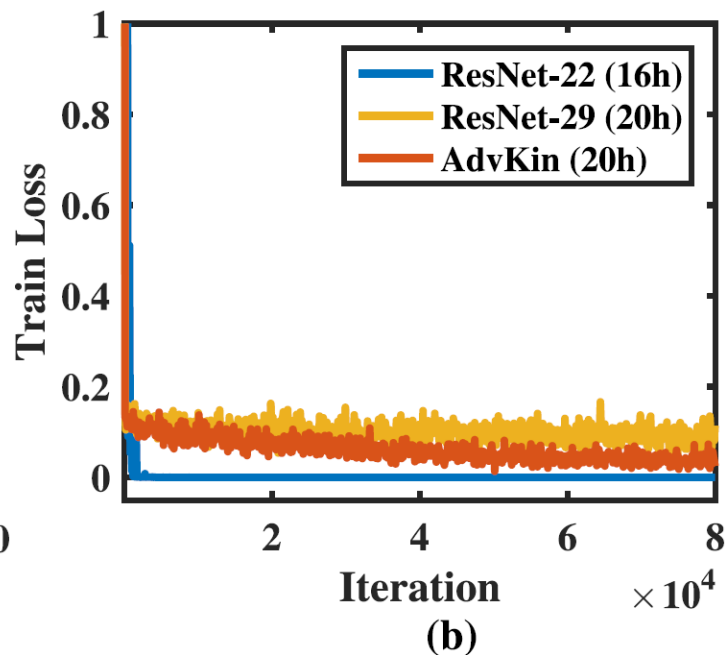
M-D 70.66%, M-S 65.22%, S-S 72.10%, B-B 63.59%,
SIBS 66.51%, F-S 63.38%, F-D 64.60%

七种亲属关系验证任务的平均准确率为 **66.58%**，排名第三

◆ 由于测试集的标签是“盲”的（不可用），本文未提供与其他方法的比较



小规模数据集



大规模数据集

◆ 模型的收敛速度和训练时间与其他方法相当

创新点：

- ◆ 基于对抗学习思想，提出了一种**双流网络模型 AdvKin**，用于小规模和大规模的亲属关系验证任务；
- ◆ 提出一个与**基于家庭ID的CL**和**SL**共同工作的**AL**损失，实现了自对抗学习机制；
- ◆ 提出了**E-AdvKin**，采用了两种特征增强方式：**补丁级 (patch-level)**融合和**网络级融合**。

未来展望：

- ◆ 在**卷积模块**中加入更多的**自对抗层**，以便通过多重自对抗训练策略更好地提升亲属关系特征的区分能力；
- ◆ 在**自对抗学习**中引入更多具有挑战性的**骨干网络**。

Thanks

Q & A