OCC Methods for Media Forensics

Shervin R. Arashloo

sh.rahimzadeh@hotmail.com

Outline

1. Face Presentation Attack Detection
   - Two-Class vs. One-Class Formulation
   - Client-Specific Modelling
   - Classifier Fusion
   - Kernel Fusion

2. Face Manipulation (Deepfake) Detection
   - One-Class Classification for Deepfake Detection
   - Large-margin Classification
   - Sparsity-Induced Classifier Fusion

Face Recognition

- Pose (out-of-plane rotation)
- Illumination
- Expression
- *etc.*

Face Presentation Attack Detection

Problem

An unauthorised subject tries to get illegitimate access to a face recognition system by presenting fake biometrics traits

Face Presentation Attack Detection

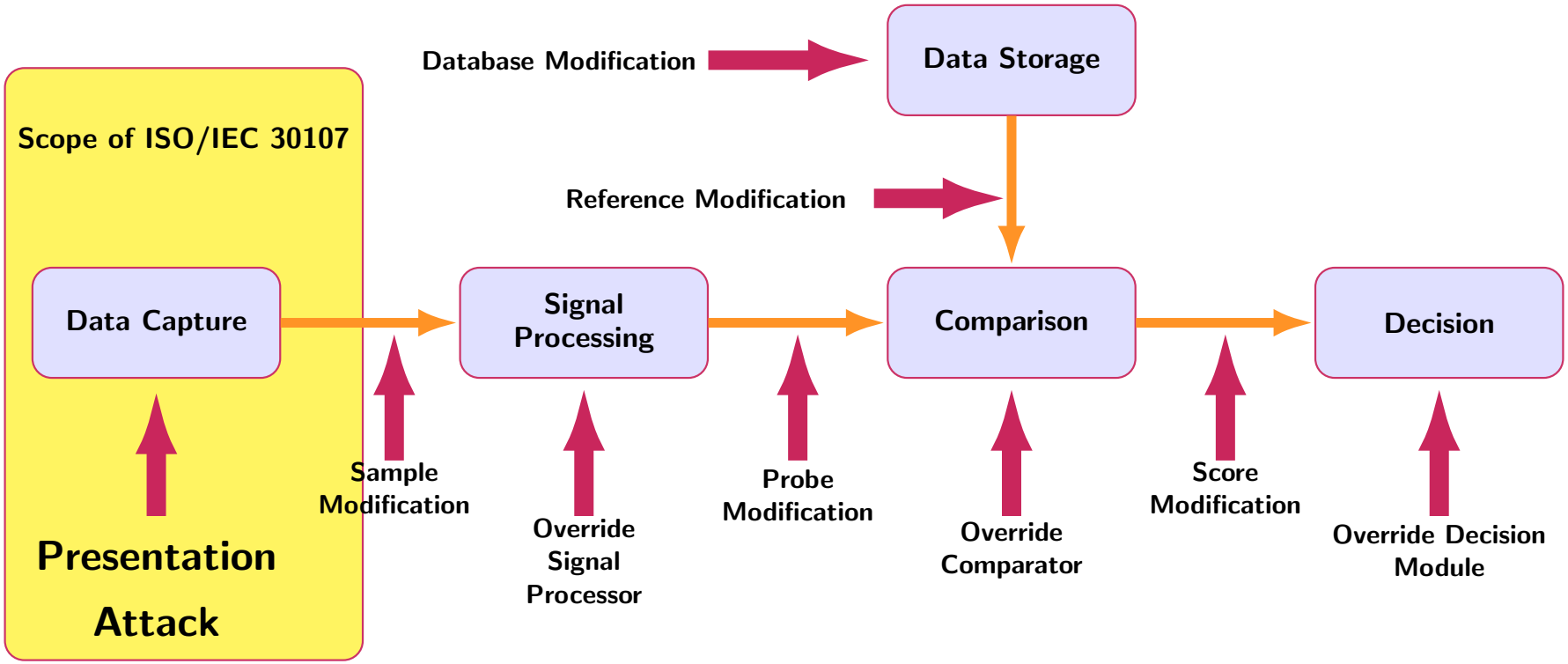
Problem

An unauthorised subject tries to get illegitimate access to a face recognition system by presenting fake biometrics traits

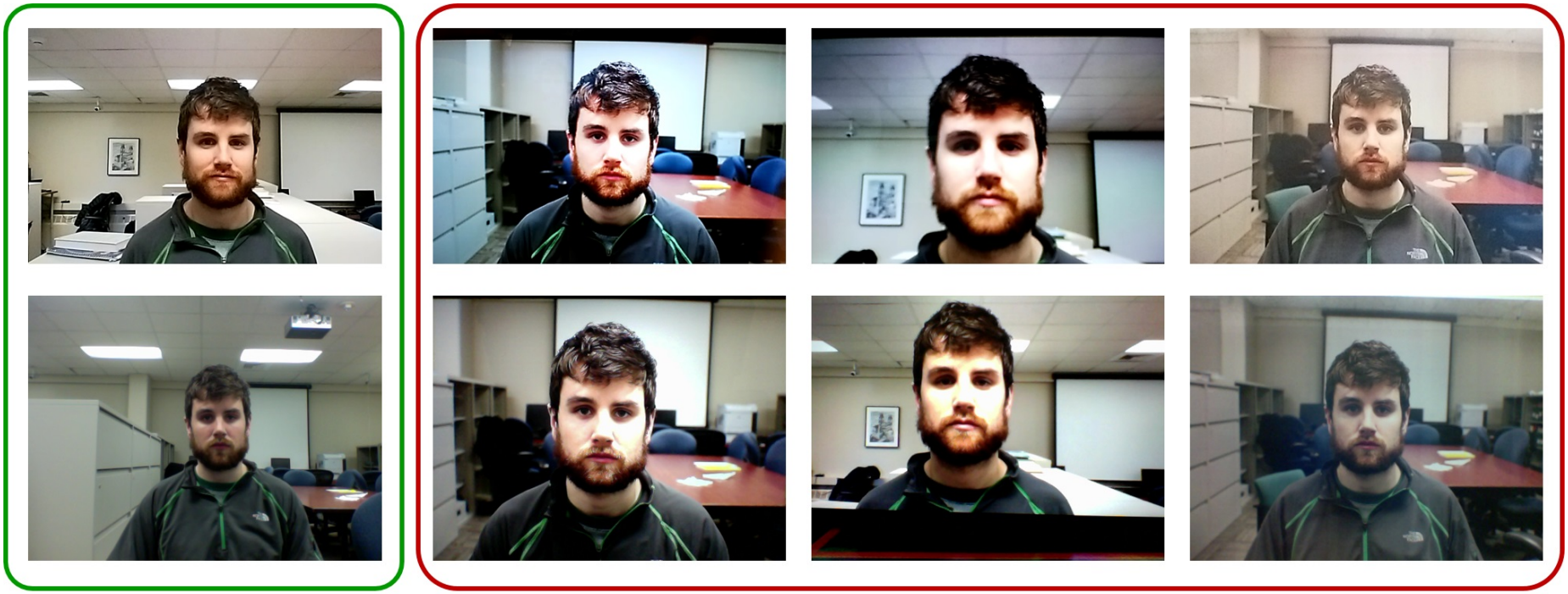
Typical face presentation attack instruments:

- Print
- Replay
- Mask
- *etc.*

Points of Attack to a Biometrics System



Samples captured by a recognition system



(a) (b) (c) (d)

(a) Genuine (bona fide) samples
(b),(c), and (d) Presentation Attacks

The Conventional approach

Two-Class classification

Collect both bona fide and attack samples and train a binary classifier to separate attacks from genuine samples

The Conventional approach

Two-Class classification

Collect both bona fide and attack samples and train a binary classifier to separate attacks from genuine samples

Drawbacks:

- High cost of collecting attack samples: deep models!

The Conventional approach

Two-Class classification

Collect both bona fide and attack samples and train a binary classifier to separate attacks from genuine samples

Drawbacks:

- High cost of collecting attack samples: deep models!
- Poor generalisation
  - Different imaging conditions

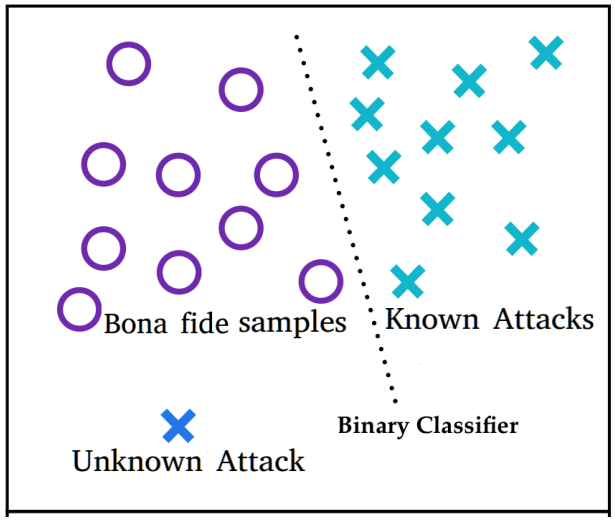
The Conventional approach

Two-Class classification

Collect both bona fide and attack samples and train a binary classifier to separate attacks from genuine samples

Drawbacks:

- High cost of collecting attack samples: deep models!
- Poor generalisation
  - Different imaging conditions
  - Novel attack types unseen during training!



Our Approach

One-Class formulation

One-class classifiers can be trained using only positive samples!

Our Approach

One-Class formulation

One-class classifiers can be trained using only positive samples!

Advantages:

- Normal access data can be collected with relative ease whereas attack data is demanding in terms of manpower resource

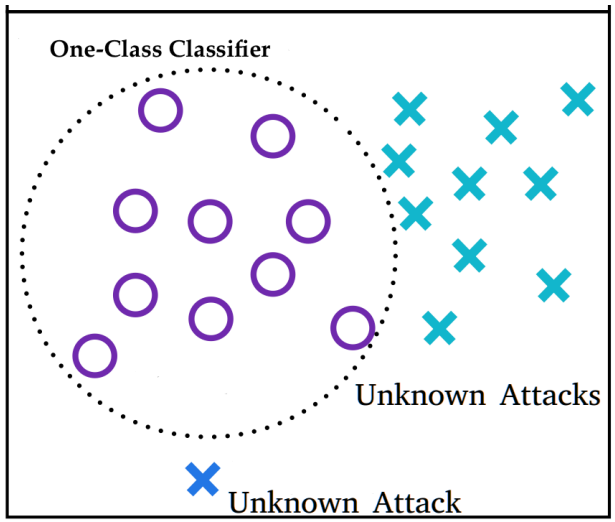
Our Approach

One-Class formulation

One-class classifiers can be trained using only positive samples!

Advantages:

- Normal access data can be collected with relative ease whereas attack data is demanding in terms of manpower resource
- Learns from genuine data $\Rightarrow$ not biased towards specific attack types!



S. R. Arashloo, J. Kittler and W. Christmas, "An Anomaly Detection Approach to Face Spoofing Detection: A New Formulation and Evaluation Protocol," in IEEE Access, vol. 5, pp. 13868-13882, 2017.

Client-Specific Modelling

The common approach: **Subject-Independent**

- A single classifier is trained to detect PA w.r.t. all subjects

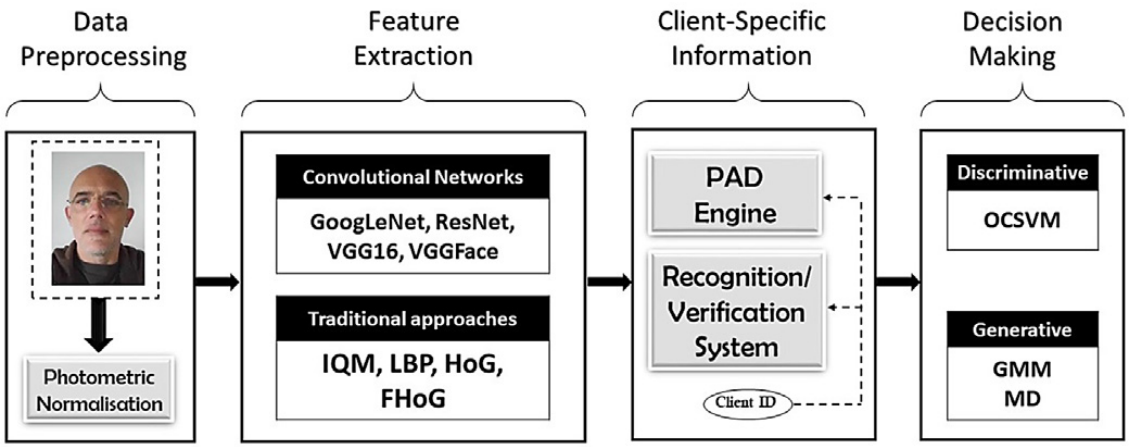
Client-Specific Modelling

The common approach: **Subject-Independent**

- A single classifier is trained to detect PA w.r.t. all subjects

Our approach:

- Deploying **client-specific** data for model training
- Subject-specific score distributions motivate a distinct threshold for each client

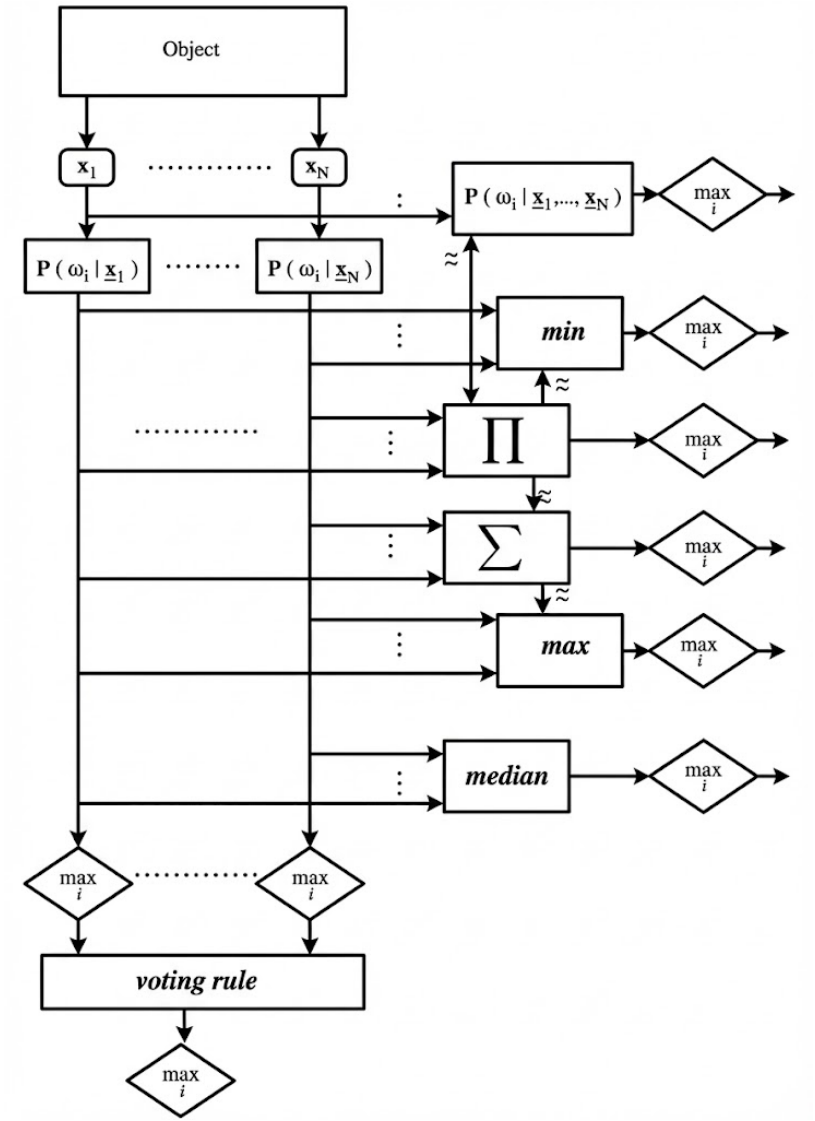


S. Fatemifar, S. R. Arashloo, M. Awais, J. Kittler, "Client-specific anomaly detection for face presentation attack detection, Pattern Recognition," Volume 112, 2021, 107696.

Client-specific vs. client-independent modelling

Dataset	Client-Specific			Client-Independent		
	Video-Based Scenario					
	APCER	BPCER	HTER	APCER	BPCER	HTER
Replay-Attack	0	0	0	9.54	9.14	8.45
Replay-Mobile	14.32	3.96	8.58	20.98	25.78	17.63
Rose-Youtu	17.33	10.00	8.13	20.00	0	11.48
	Frame-Based Scenario					
Replay-Attack	1.85	0	1.46	13.23	14.19	12.75
Replay-Mobile	23.78	5.69	13.56	32.11	12.43	17.43
Rose-Youtu	31.25	15.06	14.69	20.60	15.29	17.95

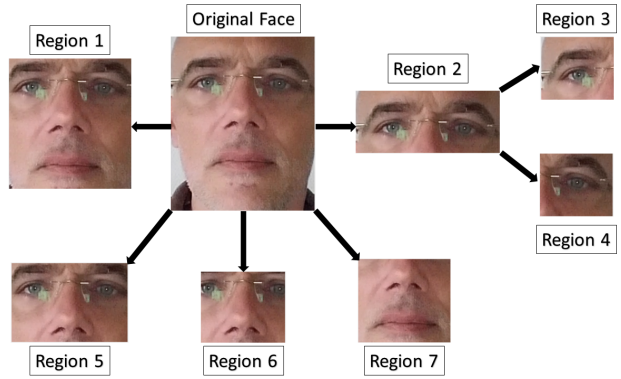
Fixed-rule Classifier Fusion



J. Kittler, M. Hatef, R. P. W. Duin and J. Matas, "On combining classifiers," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pp. 226-239, March 1998.

Diversity in Representations and Classifiers

Multiple regions



Multiple Deep Features:

- GoogleNet
- ResNet50
- VGG16

Multiple One-Class Learners:

- Support Vector Data Description
- Mahalanobis distance (MD)
- Gaussian mixture model

$$7(\text{regions}) \times 3(\text{features}) \times 3(\text{OCCs})$$

Table: Sum-rule vs. single best classifier in terms of HTER (%).

	Single Best Classifier	Sum Rule
Replay-Mobile	13.14	12.19
Replay-Attack	2.49	1.57
Rose-Youtu	11.73	11.21

S. Fatemifar, M. Awais, S. R. Arashloo and J. Kittler, "Combining Multiple one-class Classifiers for Anomaly based Face Spoofing Attack Detection," 2019 International Conference on Biometrics (ICB), Crete, Greece, 2019, pp. 1-7.

One-Class Fisher Discriminant Analysis

The Fisher classifier:

$$F(\beta) = \frac{\beta^\top \mathbf{\Sigma_b} \beta}{\beta^\top \mathbf{\Sigma_w} \beta}$$

$\mathbf{\Sigma}_b$: between-class scatter matrix
$\mathbf{\Sigma}_w$: within-class scatter matrix
β: Fisher discriminant

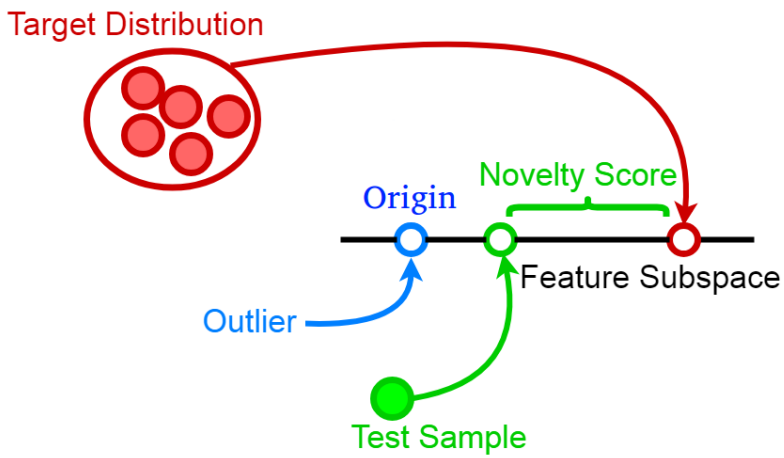
One-Class Fisher Discriminant Analysis

The Fisher classifier:

$$F(\beta) = \frac{\beta^\top \mathbf{\Sigma_b} \beta}{\beta^\top \mathbf{\Sigma_w} \beta}$$

$\mathbf{\Sigma}_b$: between-class scatter matrix
$\mathbf{\Sigma}_w$: within-class scatter matrix
β: Fisher discriminant

Originally developed for two-class classification but can be adapted to a one-class setting!



Regression-Based Formulation

- Not convenient to impose regularisation on the discriminant for improved generalisation performance
- Not straightforward to extend to kernel space

Regression-Based Formulation

- Not convenient to impose regularisation on the discriminant for improved generalisation performance
- Not straightforward to extend to kernel space

Regularised regression-based reformulation in the kernel space

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (1 - \boldsymbol{\theta}^{\top} \boldsymbol{\upsilon}(\mathbf{x}_i))^2$$

Regression-Based Formulation

- Not convenient to impose regularisation on the discriminant for improved generalisation performance
- Not straightforward to extend to kernel space

Regularised regression-based reformulation in the kernel space

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (1 - \boldsymbol{\theta}^{\top} \boldsymbol{\upsilon}(\mathbf{x}_i))^2 + \sigma \|\boldsymbol{\theta}\|_2^2$$

Empirical loss

Tikhonov regularisation

- Not convenient to impose regularisation on the discriminant for improved generalisation performance
- Not straightforward to extend to kernel space

Regularised regression-based reformulation in the kernel space

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (1 - \boldsymbol{\theta}^{\top} \boldsymbol{\upsilon}(\mathbf{x}_i))^2 + \sigma \|\boldsymbol{\theta}\|_2^2$$

Empirical loss

Tikhonov regularisation

- Not convenient to impose regularisation on the discriminant for improved generalisation performance
- Not straightforward to extend to kernel space

Regularised regression-based reformulation in the kernel space

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (1 - \boldsymbol{\theta}^\top \boldsymbol{\upsilon}(\mathbf{x}_i))^2 + \sigma \|\boldsymbol{\theta}\|_2^2$$

The **dual** problem is

$$\max_{\boldsymbol{\omega}} -\boldsymbol{\omega}^\top \mathbf{K} \boldsymbol{\omega} - \sigma \boldsymbol{\omega}^\top \boldsymbol{\omega} + 2\boldsymbol{\omega}^\top \mathbf{1}$$

$\mathbf{K}$: kernel matrix

S. R. Arashloo and J. Kittler, "Robust One-Class Kernel Spectral Regression," in IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 3, pp. 999-1013, March 2021.

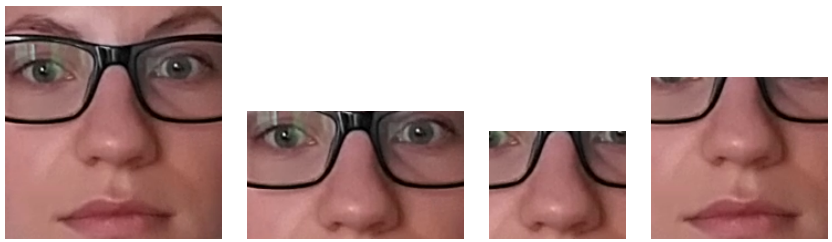
Kernel Fusion: sum rule

Fusing multiple representations via a **sum rule**:

$$\mathbf{K} = \mathbf{K}_1 + \mathbf{K}_2 + \cdots + \mathbf{K}_J$$

Diversity in the representations:

- Multiple Regions
- Multiple Deep Features
  - GoogleNet
  - ResNet50
  - VGG16



Kernel Fusion: Unseen attack evaluation protocol

Comparison on the Replay-Attack dataset in terms of AUC

Method	AUC (%)
OCSVM+IMQ [7]	80.76
OCSVM+BSIF [7]	81.94
NN4+LBP [9]	91.26
GMM+LBP [9]	90.06
OCSVM+LBP [9]	87.90
AE+LBP [9]	86.12
DTL [11]	99.80
One-Class MD [10]	99.75
SVDD	97.50
KPCA	100
GP	100
Our work	**100**

Comparison on the MSU-MFSD dataset in terms of AUC

Method	AUC (%)
OCSVM+IMQ [7]	67.77
OCSVM+BSIF [7]	75.64
NN4+LBP [9]	81.59
GMM+LBP [9]	81.34
OCSVM+LBP [9]	84.47
AE+LBP [9]	87.63
DTL [11]	93.00
SVDD	97.5
KPCA	100
GP	100
Our work	**100**

Comparison on the OULU-NPU dataset protocol IV (%)

Method	APCER	BPCER	ACER
Massy HNU [78]	35.8±55.5	8.3±4.1	22.1±17.6
GRADIANT [78]	5.0±4.5	15.0±7.1	10.0±5.0
FAS-BAS [20]	9.3±5.6	10.4±6.0	9.8±6.0
LBP-SVM [62]	41.67±27.03	55.0±21.21	48.33±6.07
IQM-SVM [62]	34.17±25.89	39.17±23.35	36.67±12.13
DeepPixBiS [62]	36.67±29.67	13.33±16.75	25.0±12.67
the work in [77]	0.9±1.8	4.2±5.3	2.6±2.8
SVDD	25.0±17.32	8.33±6.83	16.67±10.68
KPCA	13.33±14.72	11.67±11.25	12.5±12.94
GP	15.83±16.25	2.5±4.18	9.17±8.76
Our work	**11.67±13.66**	**0.83±2.04**	**6.25±6.85**

Comparison on the Replay-Mobile dataset in terms of HTER

Method	HTER (%)
GoogleNet+SVDD [10]	14.34
ResNet50+SVDD [10]	21.76
VGG16+SVDD [10]	18.78
GoogleNet+MD [10]	13.70
ResNet50+MD [10]	21.81
VGG16+MD [10]	19.84
GoogleNet+GMM [10]	14.21
ResNet50+GMM [10]	21.53
VGG16+GMM [10]	18.05
SVDD	16.14
KPCA	17.05
GP	16.36
Our work	**11.88**

S. R. Arashloo, "Unseen Face Presentation Attack Detection Using Sparse Multiple Kernel Fisher Null-Space," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 4084-4095, Oct. 2021.

Multiple Kernel Learning

The idea:

Learn combination weights instead of using fixed equal weights

Objective function

$$\max_{\boldsymbol{\omega}} \quad -\boldsymbol{\omega}^\top \mathbf{K}\boldsymbol{\omega} - \delta\boldsymbol{\omega}^\top\boldsymbol{\omega} + 2\boldsymbol{\omega}^\top\mathbf{1}$$

Multiple Kernel Learning

The idea:

Learn combination weights instead of using fixed equal weights

Objective function

$$\begin{aligned} & \max_{\omega} && -\omega^\top \mathbf{K}\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ \min_{\beta} \quad & \max_{\omega} && -\omega^\top (\sum_j \beta_j \mathbf{K}_j)\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ & \text{s.t.} && \beta \geq 0, \mathcal{R}(\beta) \end{aligned}$$

Multiple Kernel Learning

kernel weights

The idea:

Learn combination weights instead of using fixed equal weights

Objective function

$$\begin{aligned} & \max_{\omega} && -\omega^\top \mathbf{K}\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ \min_{\beta} \quad & \max_{\omega} && -\omega^\top \Big(\sum_j \beta_j \mathbf{K}_j\Big)\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ & \text{s.t.} && \beta \geq 0, \mathcal{R}(\beta) \end{aligned}$$

Multiple Kernel Learning

kernel weights

The idea:

Learn combination weights instead of using fixed equal weights

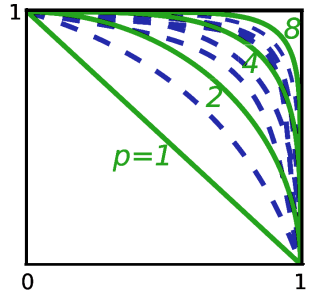
Objective function

$$\begin{aligned} & \max_{\omega} && -\omega^\top \mathbf{K}\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ \min_{\beta} \quad & \max_{\omega} && -\omega^\top \Big(\sum_j \beta_j \mathbf{K}_j\Big)\omega - \delta\omega^\top\omega + 2\omega^\top \mathbf{1} \\ & \text{s.t.} && \beta \geq 0, \mathcal{R}(\beta) \end{aligned}$$

Different possibilities for sparsity regularisation $\mathcal{R}(\beta)$:

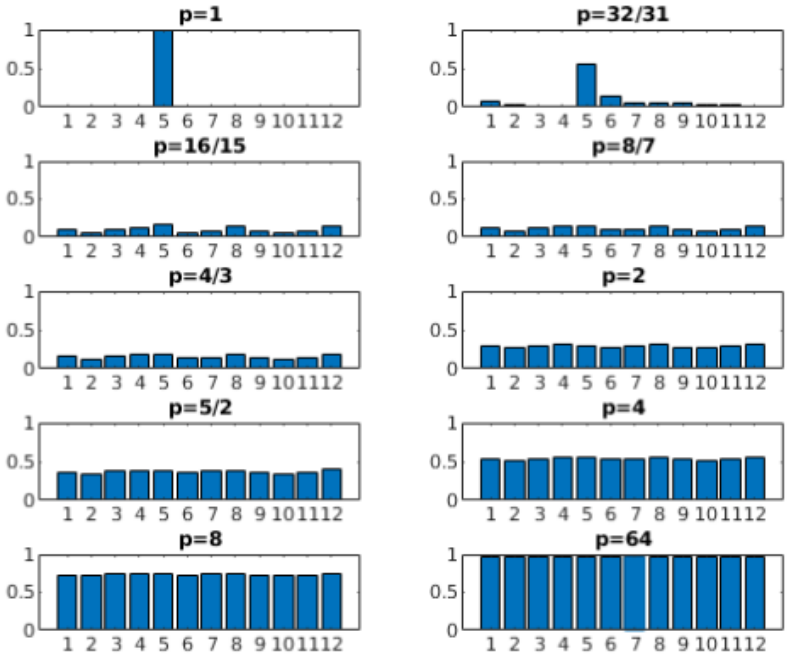
- ℓ_p-norm $\|\beta\|_p^p \leq 1;\ p \geq 1$
- mixed (r,p)-norm $\left\|\beta\beta^\top\right\|_{r,p} \leq 1;\ r,p \geq 1$

Both regularisations lead to convex optimisation problems!



green: unit ℓ_p-norm balls for $p \in \{1, 2, 4, 8\}$; green&blue: unit (r, p)-norm balls for $r, p \in \{1, 2, 4, 8\}$

ℓ_p and (r, p)-norm Evaluation Results



Sample kernel weights for ℓ_p regularisation.

Unseen face PAD results on protocol IV of the Oulu-NPU dataset.

Method	**ACER (mean $\pm$ std) %**
Product-FN	4.5 $\pm$ 5.3
Average-FN	5.0 $\pm$ 3.9
Product-GP	5.8 $\pm$ 6.4
Average-GP	6.2 $\pm$ 4.4
Product-KPCA	4.5 $\pm$ 5.3
Average-KPCA	5.4 $\pm$ 3.6
MK-SVDD	7.1 $\pm$ 6.2
MK-OCSVM	7.9 $\pm$ 6.4
Slim-MK-SVDD	6.2 $\pm$ 4.4
Slim-MK-OCSVM	6.2 $\pm$ 4.4
SAPLC [45]	9.3 $\pm$ 4.4
OCA-FAS [46]	4.1 $\pm$ 2.7
The work in [47]	3.7 $\pm$ 2.1
The work in [48]	9.8 $\pm$ 4.2
ℓ_p MK-FN	**3.3 $\pm$ 3.4**
(r, p)-norm MK-FN	**2.5 $\pm$ 2.2**

- S. R. Arashloo, "Matrix-Regularized One-Class Multiple Kernel Learning for Unseen Face Presentation Attack Detection," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 4635-4647, 2021.
- S. R. Arashloo, "One-Class Classification Using ℓp-Norm Multiple Kernel Fisher Null Approach," in IEEE Transactions on Image Processing, vol. 32, pp. 1843-1856, 2023.

Face Manipulation (Deepfake) Detection

Entirely or partially modified photorealistic face images.

Conventional face recognition systems are vulnerable to Deepfakes and may confuse Deepfakes with genuine images!

Deepfakes may have harmful impacts on

- Politics
- Economy
- Erosion of public trust
- *etc.*

H. Dang, F. Liu, J. Stehouwer, X. Liu and A. K. Jain, "On the Detection of Digital Face Manipulation," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 5780-5789

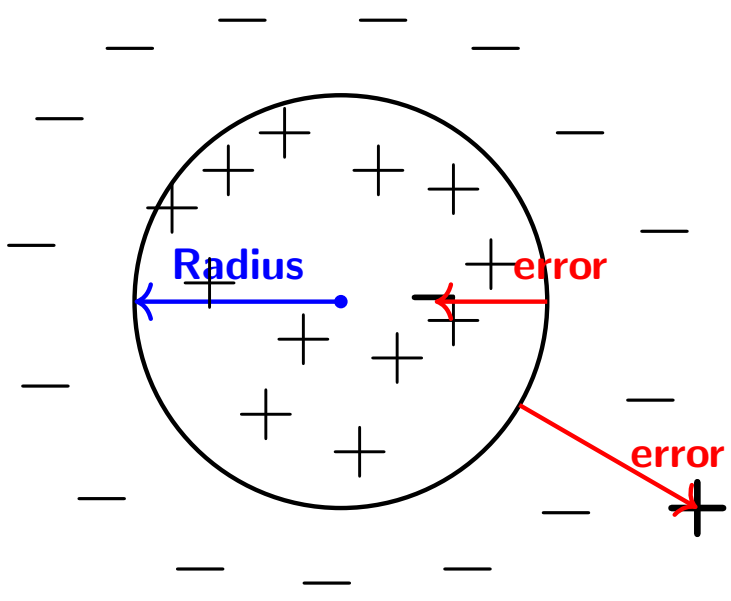
Real or Fake?

One-Class Classification for Deepfake Detection

- Genuine samples considered as "*normal*" samples
- Deepfakes as "*anomalies*" deviating from normality
- One-Class Classification may be deployed to learn the support of normal observations

One-Class Classification for Deepfake Detection

- Genuine samples considered as "*normal*" samples
- Deepfakes as "*anomalies*" deviating from normality
- One-Class Classification may be deployed to learn the support of normal observations



Support Vector Data Description (SVDD)

$$\min_{r,\mathcal{C},\boldsymbol{\varepsilon}} E(r,\mathcal{C},\varepsilon) = r^2 + c_1 \sum_i \varepsilon_i + c_2 \sum_I \varepsilon_I$$

$$\text{s.t.} \quad \|\mathbf{o}_i - \mathcal{C}\|_2^2 \leq r^2 + \varepsilon_i, \;\; ,\varepsilon_i \geq 0, \forall i$$

$$\|\mathbf{o}_I - \mathcal{C}\|_2^2 \geq r^2 - \varepsilon_I, \;\; \varepsilon_I \geq 0, \forall I$$

Tax, D.M., Duin, R.P., "Support Vector Data Description," Machine Learning 54, 45-66, 2004

Our approach: Large-margin ℓ_p-SVDD

Primal:

$$\min_{r,\mathcal{C},\boldsymbol{\varepsilon},\rho} E = r^2 + c_1 \sum_i \varepsilon_i^p + c_2 \sum_l \varepsilon_l^p - \nu\rho^2$$

$$\text{s.t.} \quad \|\mathbf{o}_i - \mathcal{C}\|_2^2 \leq r^2 - \rho^2 + \varepsilon_i, \quad \|\mathbf{o}_l - \mathcal{C}\|_2^2 \geq r^2 + \rho^2 - \varepsilon_l,$$
$$\varepsilon_i \geq 0, \quad \varepsilon_l \geq 0, \quad \forall i, l$$

Our approach: Large-margin ℓ_p-SVDD

Primal:

$$\min_{r,\mathcal{C},\boldsymbol{\varepsilon},\rho} E = r^2 + c_1 \sum_i \varepsilon_i^p + c_2 \sum_l \varepsilon_l^p - \nu\rho^2$$

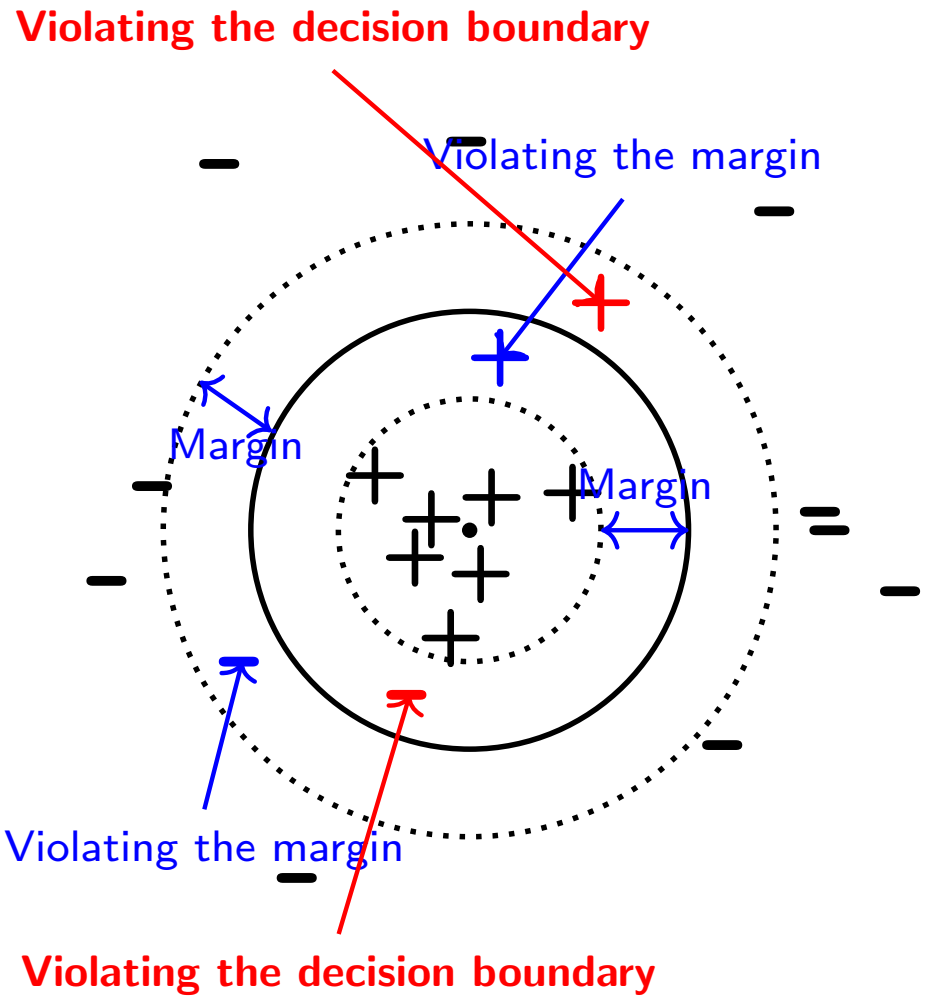
$$\text{s.t.} \quad \|\mathbf{o}_i - \mathcal{C}\|_2^2 \leq r^2 - \rho^2 + \varepsilon_i, \quad \|\mathbf{o}_l - \mathcal{C}\|_2^2 \geq r^2 + \rho^2 - \varepsilon_l,$$
$$\varepsilon_i \geq 0, \quad \varepsilon_l \geq 0, \quad \forall i, l$$

Dual:

$$\min_{\boldsymbol{\beta}} \quad \bar{c}_1 \|\boldsymbol{\beta} \odot (\mathbf{1} + \mathbf{t})\|_q^q + \bar{c}_2 \|\boldsymbol{\beta} \odot (\mathbf{1} - \mathbf{t})\|_q^q + (\boldsymbol{\beta} \odot \mathbf{t})^\top \mathbf{K} (\boldsymbol{\beta} \odot \mathbf{t})$$
$$\text{s.t. } \boldsymbol{\beta} \geq 0, \ \mathbf{1}^\top \boldsymbol{\beta} = \nu, \ \mathbf{t}^\top \boldsymbol{\beta} = 1$$

- Optimised by applying a **new Frank-Wolfe-based approach** to the dual problem
- Probability of misclassification shown to be theoretically reduced based on **Rademacher complexities**

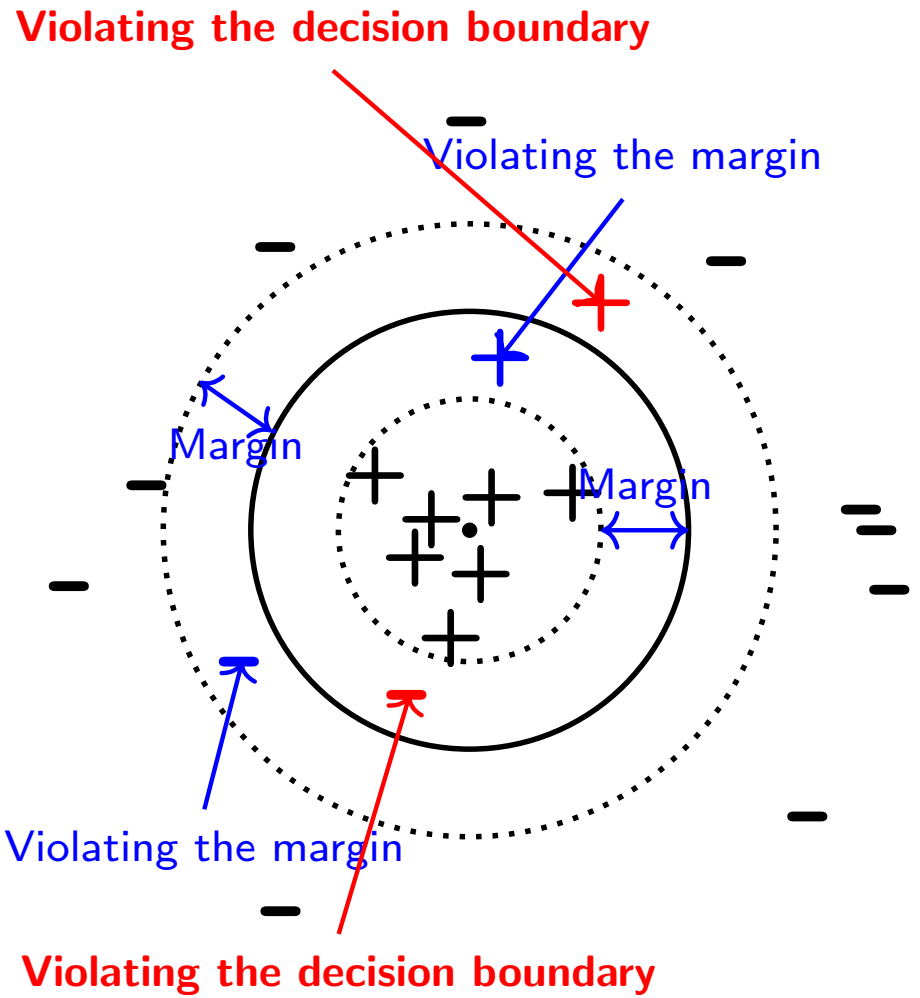
Our approach: Large-margin ℓ_p-SVDD



Why **large-margin**?

- Feels safest
- Decreased probability of misclassification
- Empirically better performance

Our approach: Large-margin ℓ_p-SVDD



Why **large-margin**?

- Feels safest
- Decreased probability of misclassification
- Empirically better performance

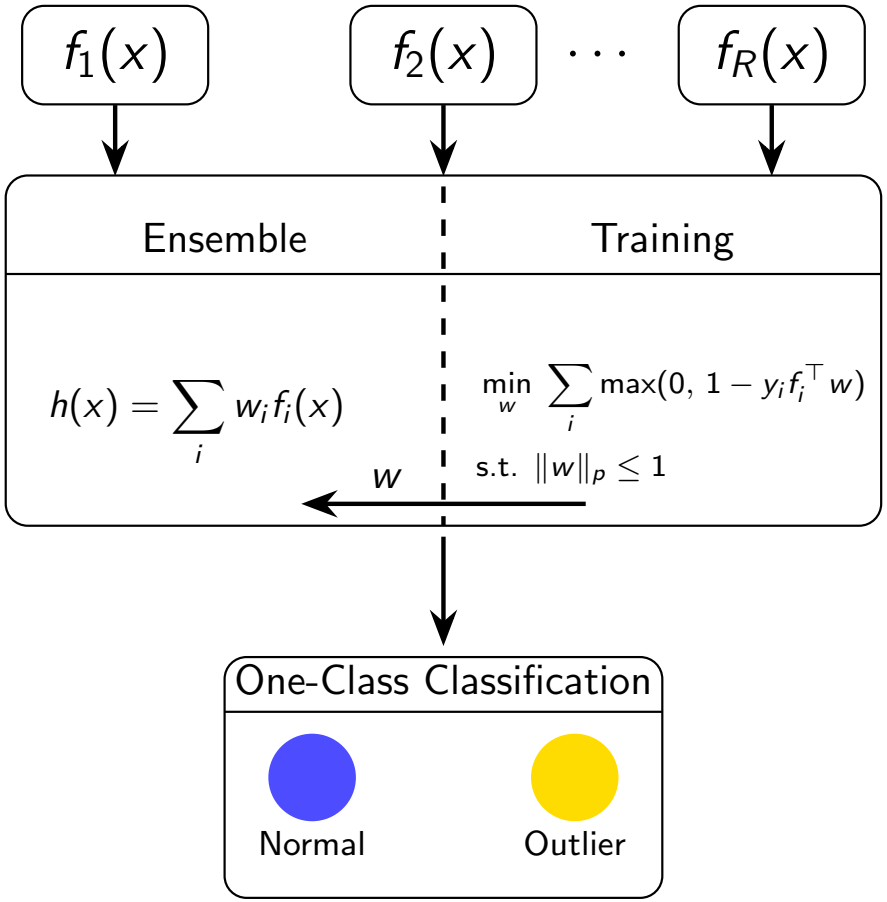
Why ℓ_p?

$$\|\mathbf{x}\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_d|^p)^{1/p}$$

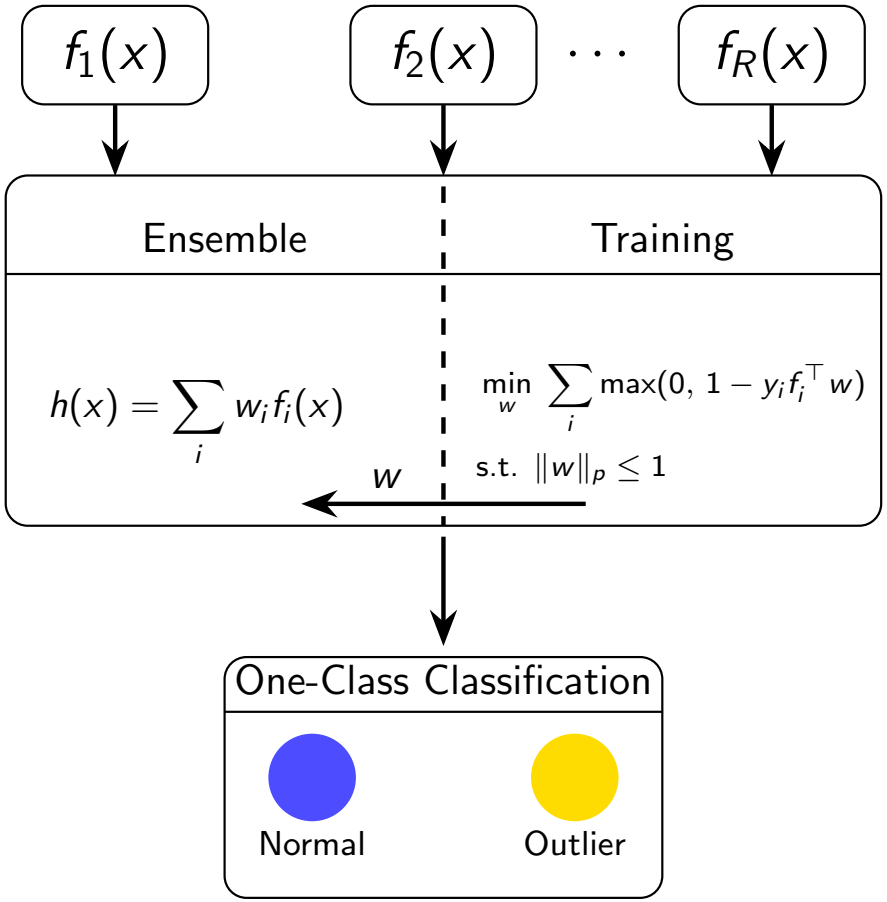
- Penalise errors non-linearly
- Free parameter for penalising errors of different magnitudes

S.R. Arashloo, "Large-margin multiple kernel ℓp-SVDD using Frank–Wolfe algorithm for novelty detection," Pattern Recognition, Volume 148, 2024, 110189.

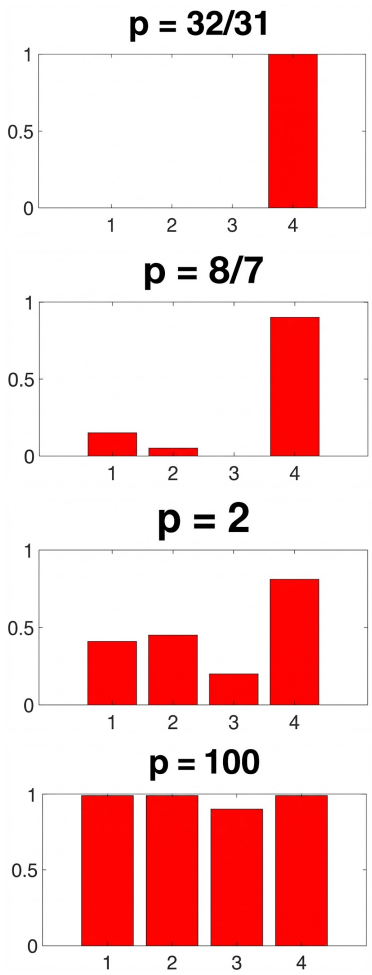
Learning Classifier Fusion Weights Subject to Sparsity



Learning Classifier Fusion Weights Subject to Sparsity



Key parameter: p



S. Nourmohammadi, S. R. Arashloo, J. Kittler, "ℓp-norm constrained one-class classifier combination," Information Fusion, Vol. 114, 2025, 102700.

Sparsity-Induced Classifier Fusion: Insights

Opinions of multiple experts combined without a careful manual pre-selection of base learners!

Sparsity-Induced Classifier Fusion: Insights

Opinions of multiple experts combined without a careful manual pre-selection of base learners!

Special cases:

- $p \to \infty$ yields a uniform weight vector that corresponds to the `sum rule` for classifier fusion

Sparsity-Induced Classifier Fusion: Insights

Opinions of multiple experts combined without a careful manual pre-selection of base learners!

Special cases:

- $p \to \infty$ yields a uniform weight vector that corresponds to the `sum rule` for classifier fusion
- $p \to 1^+$ chooses only the most confident classifier, *i.e.* the one with the maximum average margin

Sparsity-Induced Classifier Fusion: Insights

Opinions of multiple experts combined without a careful manual pre-selection of base learners!

Special cases:

- $p \to \infty$ yields a uniform weight vector that corresponds to the `sum rule` for classifier fusion
- $p \to 1^+$ chooses only the most confident classifier, *i.e.* the one with the maximum average margin
- $p = 2$ yields the conventional soft-margin linear SVM

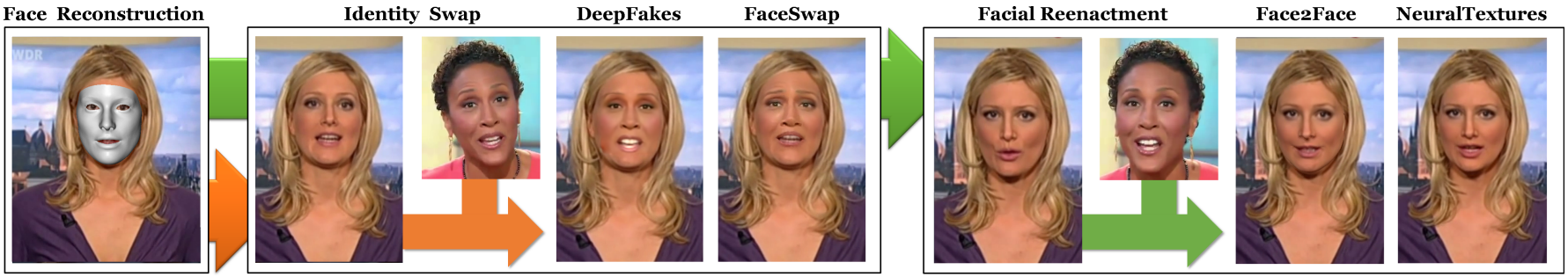
Sparsity-Induced Classifier Fusion: Insights

Opinions of multiple experts combined without a careful manual pre-selection of base learners!

Special cases:

- $p \to \infty$ yields a uniform weight vector that corresponds to the `sum rule` for classifier fusion
- $p \to 1^+$ chooses only the most confident classifier, *i.e.* the one with the maximum average margin
- $p = 2$ yields the conventional soft-margin linear SVM
- Varying p in $(1, \infty)$ sweeps the entire spectrum of base learners, starting with the single most confident one to the case of uniformly weighting all classifiers

FaceForensics++ dataset

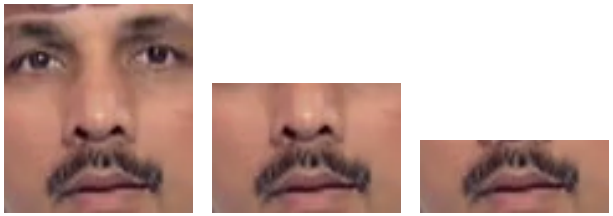


- The data has been gathered from Youtube and all videos contain a trackable mostly frontal face without occlusions
- 1000 original video sequences, manipulated with 4 automated face manipulation methods: Deepfakes, Face2Face, FaceSwap and NeuralTextures
- binary masks are available so the data can be used for image and video classification as well as segmentation

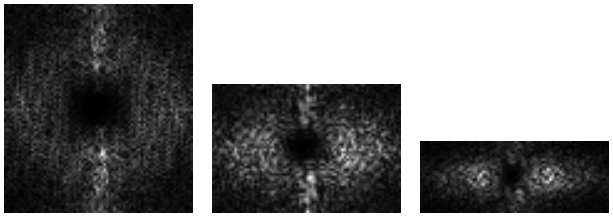
---

A. Rössler et al., "FaceForensics++: Learning to Detect Manipulated Facial Images", International Conference on Computer Vision (ICCV), 2019.

Representations & Classifiers



Multiple regions



Filtered Fourier spectrums

Features: 3 pre-trained CNNs (AlexNet, Inception-v3 and DarkNet-19) $\Rightarrow$ 9 sets of features

Base classifiers: SVDD, GP, KPCA, GMM

In total 36 classifiers!

C. Miao, Z. Tan, Q. Chu, N. Yu and G. Guo, "Hierarchical Frequency-Assisted Interactive Networks for Face Manipulation Detection," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 3008-3021, 2022.

FaceForensics++ dataset: leave-one-type-out detection results

Video-level unknown/unseen face manipulation detection on the FaceForensics++ dataset (C23 quality) (AUC %)

Dataset	DF	FF	FS	NT
LTW	92.70	80.20	64.00	77.30
UIA-ViT	96.70	94.20	70.70	82.80
F^2-Trans-B	98.92	94.08	–	–
3D Decom. & Comp.	99.45	94.64	83.67	79.22
FD^2Net	98.51	89.01	68.60	71.11
Large-margin	**100**	**96.60**	**99.80**	**97.50**
Classifier fusion	**99.90**	**96.20**	**99.90**	**97.90**

Video-level unknown/unseen face manipulation detection on the FaceForensics++ dataset (C40 quality) (AUC %)

Dataset	DF	FF	FS	NT
LTW	75.60	72.40	68.10	60.80
F^2-Trans-B	88.77	77.73	–	–
HFI-Net	86.80	73.01	55.00	–
Constr. learning	81.80	72.50	69.90	62.60
Trans. & adaptation	81.80	68.60	71.00	58.20
Large-margin	**99.00**	**87.70**	**99.40**	**91.70**
Classifier fusion	**99.20**	**73.80**	**97.20**	**74.20**

Both methods outperform existing approaches with the large-margin method demonstrating an edge on low-quality videos!