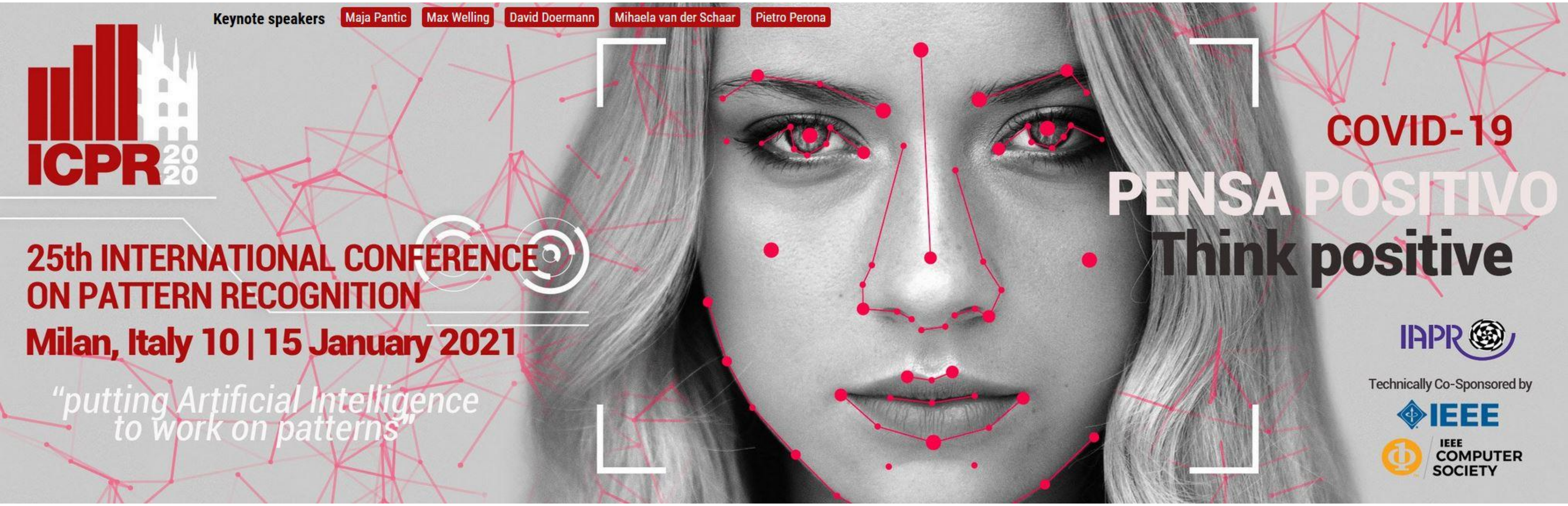
ICPR2020 Corporate Sponsors

BeanTech
Covision Lab
Deep Vision Consulting
Elsevier S.p.A.
Energy Way
EyeCan
Neosperience S.p.A.
SACMI
SAIMA Sicurezza S.p.A.
UniCredit



Peter Bellmann, Patrick Thiam, and Friedhelm Schwenker
Ulm University, Germany

Using Meta Labels for the Training of Weighting Models in a Sample-Specific Late Fusion Classification Architecture

Agenda

- Late Fusion Architectures
- Proposed Approach
- Results
- Conclusion & Future Work

LATE FUSION ARCHITECTURES

Late Fusion Architectures – Formalisation

- $X \subset \mathbb{R}^d$, $d \in \mathbb{N}$: d-dimensional data set
- $\Omega = \{\omega_1, \ldots, \omega_c\}$, $c \geq 2$: class label set
- $m \in \mathbb{N}$: number of feature subsets

- CM_i: classification model that is trained on feature subset i
- The outputs of $\{CM_1, \ldots, CM_m\}$ are combined for the final prediction

- In general: each CM_i is a **strong** model (LF combines *ensembles*)

Late Fusion Architectures – Example

Affect recognition or pain detection tasks with person-specific data

- Audio
- Video
- Physiology

$\Rightarrow m = 3$

Train 3 strong models in combination with each modality

PROPOSED APPROACH

Proposed Approach – Basic Idea

- Divide the training subsets $X_1, \ldots, X_m$ into
  - $T_1, \ldots, T_m$: Training Sets
  - $V_1, \ldots, V_m$: Validation Sets
- The output of CM_i on V_i defines the labels for Weighting Model WM_i

$$\tilde{y}_{i,j} := \begin{cases} 1, & \text{if } \mathrm{CM}_i(v_j) \rightsquigarrow y_j, \\ 0, & \text{otherwise.} \end{cases}$$

- The class-support vector for input $x \in \mathbb{R}^d$ is calculated as

$$\mu(x) = \sum_{i=1}^{m} s_i^{(1)}(x) \cdot \mathrm{CM}_i(x)$$

Proposed Approach – Training Phase

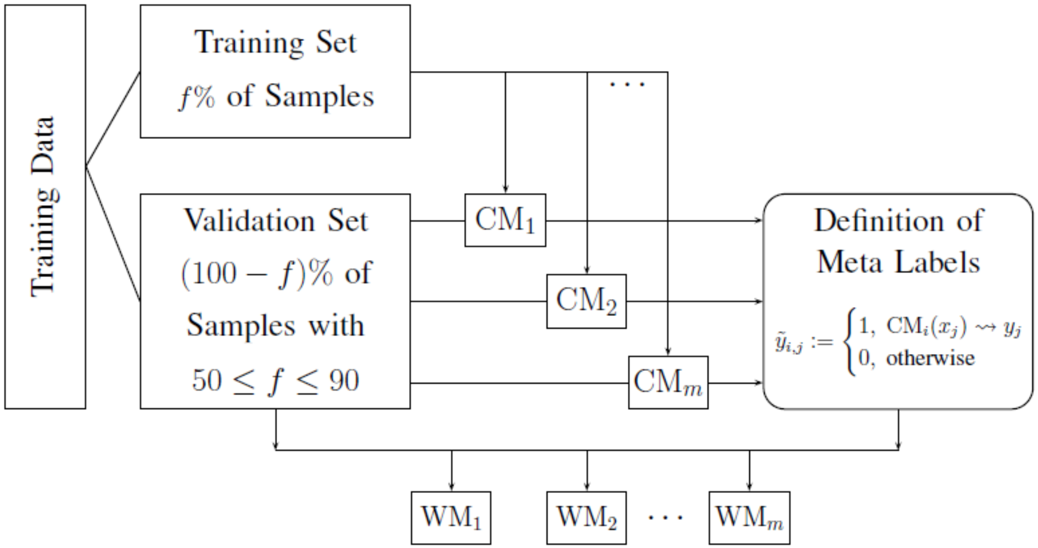


Fig. 2: **Training Phase of our Approach - System Diagram**. CM_i: Classification model i. WM_i: Weighting model i. Each model WM_i is trained on the validation set, in combination with an individual vector of corresponding meta labels.

$$\mu(x) = \sum_{i=1}^{m} s_i^{(1)}(x) \cdot \mathrm{CM}_i(x)$$

RESULTS

Data Sets Overview

Data Set	# Classes	# Samples	# Features	# Channels
BioVid	2	3480	194	3
Mfeat	10	2000	649	6
Arrhythmia	2	452	274	4
Fisher Iris	3	150	4	4

Experimental Settings

- **Classifier Choice** Bagged Decision Tree (DT) Ensemble
- **Classification Models**
  - BioVid 200 DTs per Channel
  - Remaining Sets 100 DTs per Channel
- **Weighting Models** 100 DTs per Channel
- **Measure** Accuracy

Results – Accuracy Performance

TABLE V: **Mean accuracy and standard deviation values in %.** The best performing method is depicted in bold. CV: Cross Validation. LOPO: Leave-One-Participant-Out. The early fusion corresponds to Breiman's bagging approach.

Data Set	CV	Early Fusion	Late Mean	Our Approach
BVDB	LOPO	81.90 ± 15.2	82.93 ± 16.0	$\mathbf{83.94 \pm 15.3}$
Mfeat	20-fold	96.02 ± 1.64	97.60 ± 1.47	$\mathbf{98.00 \pm 1.34}$
ARR	20-fold	74.62 ± 7.86	75.15 ± 10.6	$\mathbf{76.48 \pm 8.93}$
Iris	10-fold	94.39 ± 4.10	95.33 ± 5.49	$\mathbf{96.67 \pm 4.71}$

Results – Operational Cost

TABLE VI: **Averaged training and testing time in s**. The faster performing method is depicted in bold.

Approach	BVDB	Mfeat	ARR	Iris
Late Mean	21.8 ± 0.3	$\mathbf{12.5 \pm 0.3}$	$\mathbf{2.53 \pm 0.2}$	$\mathbf{1.45 \pm 0.1}$
Our	$\mathbf{20.2 \pm 0.3}$	13.9 ± 0.3	4.29 ± 0.3	3.00 ± 0.2

Conclusion & Future Work

- The proposed approach is a valid alternative for trainable LFs
- This approach can be formulated as a plain ensemble method

- In Future work we aim to analyse the effectiveness of confidence

$$\tilde{s}_i(x) = \begin{cases} s_i^{(1)}(x), & \text{if } s_i^{(1)}(x) \geq \theta, \\ 0, & \text{otherwise.} \end{cases}$$

$$\tilde{s}_i(x) = \begin{cases} +s_i^{(1)}(x), & \text{if } s_i^{(1)}(x) \geq \theta_2, \\ 0, & \text{if } s_i^{(1)}(x) \in (\theta_1, \theta_2), \\ -s_i^{(1)}(x), & \text{if } s_i^{(1)}(x) \leq \theta_1. \end{cases}$$

THANK YOU VERY MUCH FOR LOOKING AT THE SLIDES ☺

PLEASE FEEL FREE TO POST YOUR QUESTIONS!