Dominant Channel Fusion Architectures - An Intelligent Late Fusion Approach

Peter Bellmann
Inst. of Neural Information Processing
Ulm University
Ulm, Germany
peter.bellmann@uni-ulm.de

Patrick Thiam
Inst. of Medical Systems Biology
Inst. of Neural Information Processing
Ulm University
Ulm, Germany
patrick.thiam@uni-ulm.de

Friedhelm Schwenker
Inst. of Neural Information Processing
Ulm University
Ulm, Germany
friedhelm.schwenker@uni-ulm.de

***Abstract*—Multi-modal classification tasks (MMCTs) constitute an important part of pattern recognition. In MMCTs the data is described through different channels (input streams), which are defined by different recording sensors or types of features (e.g. categorical or numerical). The different channels can be combined by different information fusion (IF) techniques. Popular IF approaches are late fusion architectures. In the common late fusion approach, one trains a classification model (CM) for each of the distinct (i.e. non-overlapping) channels and combines the outputs of the CMs by some aggregating rule. In this study, we propose to add an initial evaluation step to determine the best performing channel, which we define as the dominant channel. We use the dominant channel to design a modified late fusion architecture with overlapping channels by including combinations of non-dominant channels with the dominant one. This idea has the following two main advantages. First, this approach is straightforward. No additional parameters and hence optimisation techniques are required. Second, besides its simplicity our outcomes show that this approach is effective, since it significantly outperforms the common late fusion approach (including non-overlapping, distinct input streams). Moreover, it reaches and even outperforms state-of-the-art results based on fusion approaches that are more complex.**

***Index Terms*—Late Fusion, Multi-modal Classification, Multiple Classifier Systems, Pain Intensity Recognition**

I. Introduction

COMMONLY, in real-world applications, pattern recognition tasks are multi-modal. In general, this means that the data is recorded by more than one sensor. As a daily example, imagine you want to decide (i.e. classify) whether your cookies, which are already on a baking tray in the oven, have already the *perfect* condition. Then, besides measuring the baking time, you should inspect the smell, the colour and the consistency of the cookies.

In classification tasks, an appropriate choice of a classification model (CM) is crucial to obtain satisfying results. Additionally, in multi-modal classification tasks, one can apply different information fusion techniques. Two of the main fusion approaches are the *early* and the *late* fusion [1]. Early fusion operates at feature level. Thereby, for each data sample, the features that are extracted from all sensor recordings/input streams, are combined to one single feature vector, and are then processed by one CM. Late fusion operates at the CMs' output levels. Thereby, data recordings specific to each sensor/feature type are used to train CMs separately. The outputs of each CM are then combined by a fixed or trainable rule to obtain the classification architecture's final decision [2].

In the current study, we propose to include an initial evaluation step to determine the best performing channel, which we denote as the *dominant* channel. The dominant channel is used to form a modified late fusion architecture with overlapping input streams. Therefore, we extend the non-dominant channels by the dominant one. This approach is both, straightforward and effective. Our experiments show that this simple idea leads to a significant improvement of the common late fusion approach.

The remainder of this work is organised as follows. In Section II, we provide some related work. We motivate and define our proposed dominant channel fusion architectures in Section III. In Section IV, we shortly describe the data sets, which we use for our experimental analyses. We summarise all of our experimental settings in Section V. Section VI includes initial experimental evaluations, which support the motivation for our proposed late fusion approach. In Section VII, we evaluate our proposed approach in comparison to the common late fusion approach and to state-of-the-art outcomes. We discuss the experimental outcomes and the properties of our approach in Section VIII. Finally, in Section IX we conclude this study.

II. Related Work

There exist many approaches, which were proposed by different researches to improve the common late fusion approach. In the following, we shortly summarise a couple of the proposed approaches to show the variety of ideas in the field of information fusion.

In [3], Ye et al. introduce their robust late fusion, which is based on rank minimization of so-called *comparative relationship matrices* (CRMs). The authors implement one classification model for each of the available input streams (type of features) and use the models' score vector outputs for the computation of the CRMs. They then solve a matrix decomposition problem, which leads to the ensemble's final decision.

During the research, Peter Bellmann was supported by a scholarship of the Landesgraduiertenförderung Baden-Württemberg at Ulm University.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

In [4], Liu et al. propose a sample-specific late fusion (SSLF) approach. In the SSLF method, a weight is learned for each labelled (training) sample. Then, for each unlabelled (test) sample, the SSLF method propagates the fusion weights in a transductive manner, based on a convex objective function. In [5], Zheng et al. introduce the query-adaptive late fusion (QALF) approach. Similarly to the approaches from [3] and [4], the QALF architecture is based on the classification models' scores. In contrast to the SSLF method, the QALF method estimates the effectiveness of each input stream. Therefore, the effects of *weak performing* features are reduced, in an unsupervised and query-adaptive manner.

In [6], Glodek et al. propose a Kalman Filter (KF) [7] based fusion approach for the classification of time series. The KF constitutes a linear dynamical system, which is based on Markov Models. The authors use base classifiers that provide outputs with corresponding confidence values, and hence whose outputs can be rejected if the confidence is *low*. The classifiers' outputs are used as inputs, which are fed to an additional KF layer. Thereby, the authors make use of the fact that the KF can estimate missing values of a time series by combining the currently available values with the information that is provided by previous states.

Another example, for classification schemes that are using the base classifiers' outputs as inputs for a second classification model layer, is the pseudo inverse (PI) [8] based fusion approach [9]. The PI provides an optimal least squares solution for linear systems (i.e. systems of the form $Ax = b$, with a *singular* coefficient matrix A). In a PI based fusion architecture, the objective is to determine the optimal least squares solution between the classifiers' outputs and the desired label outputs, by computing the corresponding PI.

III. Dominant Channel Architectures

This section provides the idea of our proposed dominant channel (DC) fusion architectures, which we simply call DC architectures.

A. Motivation: Using Expert Knowledge

Let us assume that the given data is described through different channels. When a late fusion approach is applied, usually one classification model is trained for each of the channels. Then, the outputs of the models are combined for the final decision. In general, one of the channels is assumed to lead to the best classification performance among all available channels. We call this channel *dominant* channel. We propose to apply a late fusion approach with modified data channels. Those include combinations of the originally given channels extended by the dominant channel. Let us assume that we have data from three different channels. Moreover, let channel 1 be the dominant channel. Then, we could modify the late fusion approach by using three classification models as follows. One model is trained on the channels 1 and 2, one on the channels 1 and 3, and the last model is trained solely on the dominant channel (channel 1). We call this specific kind of DC architectures *binary* DC fusion architectures.

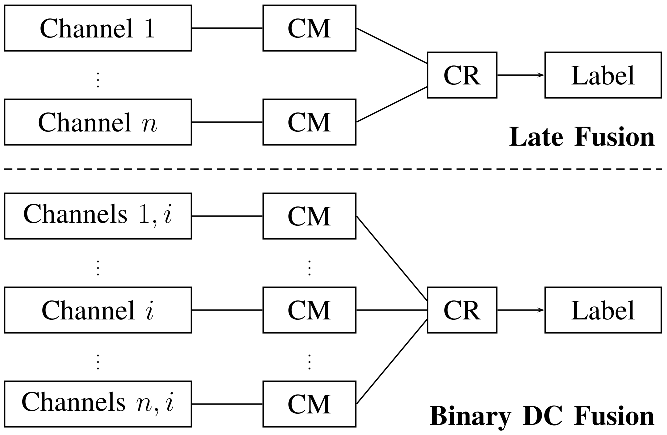

Fig. 1: **Late and Binary DC Fusion Architectures**. CM stands for classification model. CR stands for combining rule. Here, the dominant channel is denoted by the index i.

The common late fusion and the binary DC fusion approaches are depicted in Figure 1. We will denote the binary DC architecture by DC_2. Moreover, for $i > 2$, we define DC_i architectures analogously to DC_2. Thus, one modified channel of a DC_i architecture consists of i original channels ($i-1$ non-dominant channels and the dominant one). For example, let us consider the channel set $\{1, 2, 3, 4\}$ with the dominant channel 2. A ternary (DC_3) architecture consists of the following (modified) channels: $\{(1, 3, 2), (1, 4, 2), (3, 4, 2), 2\}$.

A model based on a specific channel can perform poorly, but in combination with another channel, the performance can be improved. In Sec. VI, we will experimentally analyse this assumption and show that each channel performs best, when it is combined with the dominant one, in general.

B. Why do DC Architectures work?

Adding the dominant channel to some or even all of the other channels decreases the overall diversity at first sight. Therefore, we propose using unstable base classifiers, such as decision trees [10]. A classifier is called unstable if *small* changes in the training data lead to significant changes of the classifier's output [2]. For comparison, in the decision tree based bagging approach [11], each base classifier is trained on the same feature space. The differences in the base classifiers' individual training subsets lie in the random choice of training samples. Since the combination of non-dominant channels with the dominant one leads to different new channels, our DC architectures lead to overlapping, however still different feature subspaces, similar to the random forest method [12].

Moreover, a decrease in diversity, resulting from the use of our constructed new channels, does not imply a decrease in performance. It has been shown, that in general, highly diverse classification architectures are not the best performing ones [2], [13]–[15].

C. Dominant Channel Determination

One can define different ways to determine the dominant channel. In the following, we propose three approaches for dominant channel determination.

Hold-Out Method. Divide the training data into a training and a validation set. Define the channel specific to the best validation set performance as the dominant one.

Cross Validation. Apply a k-fold cross validation, $k \in \mathbb{N}_{>1}$, on the training data. Define the channel specific to the best averaged performance across the k folds as the dominant channel.

Cross Validation with Votes. Apply a k-fold cross validation, $k \in \mathbb{N}_{>1}$, on the training data. For each fold, the channel with the best performance (winner) gets a vote. In case of a draw, each winner receives a vote for this fold. Define the channel with the maximum amount of votes (maximum k) as the dominant one. In case of a draw, take the averaged performance across the k folds.

For the unlikely case that two or more channels are defined as dominant channels, one can define the combination of those channels as the dominant one. It is not recommended defining the dominant channel based on the training samples accuracy (also known as resubstitution accuracy), since it tends to be too optimistic.

IV. Data Sets

In this section, we describe the data sets, which we are using for this study. The first part includes the BioVid Heat Pain Database[1] as well as the SenseEmotion Database, whereas the second part focuses on the so-called mfeat data set. We are using part A of the BioVid Heat Pain Database in this study.

A. BioVid Heat Pain & SenseEmotion Databases

The BioVid Heat Pain Database (BVDB) [16], as well as the SenseEmotion Database (SEDB) [17] have been collected at Ulm University. Both data sets were recorded for research purposes in the field of automatic pain (intensity) and emotion recognition, respectively. In this study, we focus solely on the pain intensity recognition task. For part A of the BVDB, the recordings of 87 participants (43 female and 44 male) are available, whereas data specific to 40 subjects (20 female and 20 male) is available for the pain intensity recognition related part of the SEDB.

In both data sets, the participants were healthy. Pain was induced in form of heat using a Medoc thermode[2], which was attached to one of the participants' forearms. An individual calibration phase led to four and three equidistant subject specific pain levels, for the BVDB and the SEDB respectively. The participants were stimulated 20 (BVDB) and 30 (SEDB) times with each of the corresponding temperature levels, in randomised order with a fixed duration of four seconds. Each pain level related stimulus was followed by a stimulus with $32°C$, which was defined as the neutral level for each participant (see Fig. 2). For the SEDB, the experiments were conducted twice. Once, the thermode was attached to the participant's left forearm, and once it was attached to the participant's right forearm. Thus, the SEDB constitutes two data sets, which we will simply call SEDB Left (or SEDB-L) and SEDB Right (or SEDB-R).

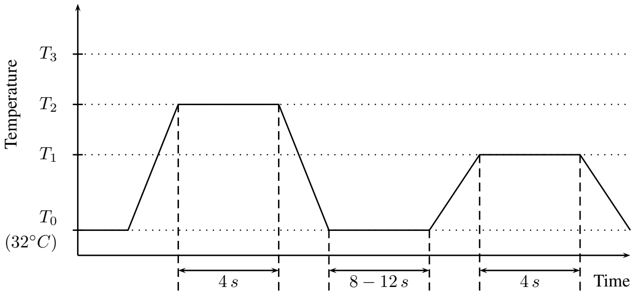

Fig. 2: **Example for stimulation and recovering phases**. This sketch depicts a sequence for a participant from the SEDB. In the BVDB, there are five temperature levels.

For the BVDB as well as the SEDB, the recorded modalities include electrocardiogram (ECG), electrodermal activity (EDA), electromyogram (EMG) and videos (VID) from three different angles. In addition, audio (AUD) and respiration (RSP) signals were recorded for the SEDB. ECG and EDA signals measure heart activity and skin conductance, respectively. EMG is a measure for muscle activity. EMG data were recorded from the trapezius muscle (in part A), which is located at the back, in the shoulder area of a human torso. EDA signals were recorded from the participants' ring and index fingers. RSP data was collected by an elastic belt system. We refer the readers to [16] and [17] for a full data set description, including data acquisition experiments for the emotion recognition tasks. Table I summarises the properties of the BVDB and the SEDB.

In this study, we use hand-crafted features, which were extracted from windows, with a length of 4.5 seconds for the biopotentials and audio signals from the SEDB, 5.5 seconds for the biopotentials of the BVDB, and 6.5 seconds for videos from the SEDB. Feature extraction is not part of our current contribution. Therefore, we refer the reader, for detailed analyses on feature extraction and normalisation, to [18] or [19] for the BVDB, and to [20] for the SEDB. Moreover, we refer the reader to some of our latest studies, for the analyses of different deep models for different pain recognition tasks, based on physiological signals [21], as well as video sequences [22]. Table II summarises the recorded modalities and the corresponding numbers of extracted features for both, the BVDB and the SEDB. The video features are divided into three feature based channels, i.e. geometric features (GEO), head pose (HPO), and local binary patterns from three orthogonal planes (LBP-TOP) [23]. We will denote LBP-TOP simply by LBP, in all figures and tables, in the interest of legibility.

[1] Online available at http://www.iikt.ovgu.de/BioVid.html

[2] https://medoc-web.com/products/pathway/

TABLE I: **Characteristics of the BioVid Heat Pain Database (BVDB) and the SenseEmotion Database (SEDB)**. The values for the SEDB are equal for both parts, i.e. for SEDB-L and SEDB-R.

	BVDB [16]	SEDB [17]
# participants	87 (43 f, 44 m)	40 (20 f, 20 m)
# classes	5 ($T_0, \ldots, T_4$)	4 ($T_0, \ldots, T_3$)
# samples per class	20 (× 87)	30 (× 40)
# samples in total	8700	4800
ECG/EDA/EMG/VID	✓	✓
AUD/RSP	–	✓

TABLE II: **Number of extracted features for the BioVid Heat Pain Database (BVDB) and the SenseEmotion Database (SEDB)**. n.u.: not used in the experiments.

Modality	BVDB	SEDB-L/SEDB-R
ECG	68	115
EMG	56	61
EDA	70	72
RSP	–	59
AUD	–	980
GEO+HPO+LBP	n.u.	$714 + 252 + 2160$
Total number	194	4413

Kessler et al. proposed including remote Photoplethysmography features from video channels [24], [25]. In one of our latest studies based on the SEDB [26], we showed that a simple quartile-based data transformation significantly improves the accuracy of nearest neighbour classifiers [27].

B. Mfeat Data Set

The publicly available mfeat (multiple features) data set [28] consists of 2000 samples of handwritten digits. Thus, there are ten classes (digits $0, \ldots, 9$). The dimensionality of the feature space is 649. The features are divided into six channels, i.e. Fourier coefficients of the character shapes, profile correlations, Karhunen-Love coefficients, pixel averages in 2×3 windows, Zernike moments and morphological features. Table III states the different features with corresponding dimensions.

TABLE III: **Extracted features and feature dimensions of the mfeat (multiple features) data set.** This data set consists of 2000 handwritten digits.

Features	Acronym	Dimension
Profile Correlations	Fac	216
Fourier Coefficients	Fou	76
Karhunen-Love Coefficients	Kar	64
Morphological Features	Mor	6
Pixel Averages	Pix	240
Zernike Moments	Zer	47

V. Experimental Settings

This section provides an overview of all experimental settings, which we will use throughout the study. We will call the common late fusion approach without additional modifications simply late fusion, when the context is clear.

Combination Rule. As the combination rule, we choose to apply a fixed rule to ensure that the combination of the different classification models (channels and modified channels) is independent from any random parameters, such as initialisation values. Moreover, we focus on the mean rule, since it has shown to be more robust against estimation errors in comparison to other fixed rules, such as the product, max or majority vote rule [29], [30].

Note that in general, a decision tree provides continuous outputs consisting of the scores for each class. The scores correspond to the proportion of the classes, in the final leaf nodes. In case that the decision trees provide simple label outputs, an ensemble of decision trees can use the proportion of the predicted labels to compute the corresponding scores.

Equal Choice of Training Samples. We use bagged decision trees for each (extended) channel as classification model. Thus, each of the decision trees is trained on a different set of training samples. For the common late fusion approach, we pull the training samples for each base classifier at random. For the DC fusion approach, we take the same training samples to ensure a fair comparison.

Evaluation Approaches. For the BVDB and the SEDB, we apply leave-one-participant-out (LOPO) cross validations. This means, that we apply k-fold cross validations with k being the number of participants. Thus, in iteration i, the data specific to the ith participant is used as the test set, and the rest of the data is used as the training set. For the mfeat data set, we apply a straightforward 10-fold cross validation, dividing the data randomly into 10 folds, according to equal distribution.

Performance Measure. As the performance measure, we take the unweighted accuracy, i.e. the number of correctly classified test samples divided by the total number of test samples:

$$accuracy = \frac{|\{y \in Y : CM(y) = l(y)\}|}{|\{Y\}|},$$

whereby $Y \subset \mathbb{R}^d$ denotes the test set, and $l(y)$ denotes the true label of y. The unweighted accuracy is an appropriate choice, since all of the data sets that are included in our study constitute balanced multi-class tasks.

Significance Tests. In Sec. VII, we will apply the two-sided Wilcoxon signed-rank test [31] at a significance level of 5%, to test for significant improvement of accuracy.

Illustrations on the SEDB. Since the data subsets SEDB Left and SEDB Right are very similar, we will focus solely on SEDB Left in our figures and tables to avoid the repetition of explanations and discussions.

VI. Analysis of Binary Channel Extensions

In this section, we analyse the effects of combining each of the non-dominant channels with the dominant one. Figure 3 depicts cross validation accuracy results for each of the given channels, according to the settings from the previous section. Figure 4 illustrates the accuracy results for each of the given channels in combination with the corresponding dominant channel. We denote binary channel combinations, where one channel is extended by the dominant one, with an additional plus sign, e.g. ECG+:= (ECG, EDA), if EDA is defined as the dominant channel.

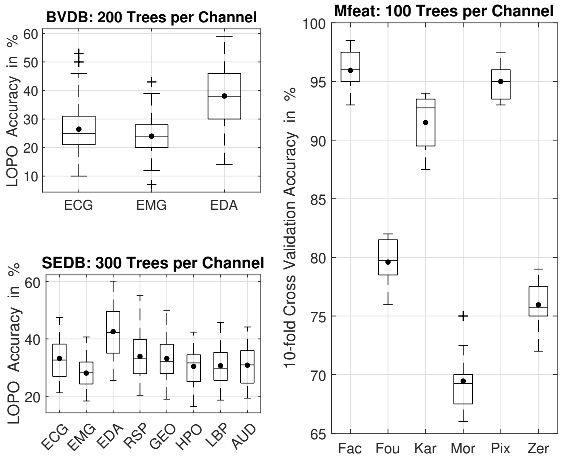

Fig. 3: **Single channel evaluation**. **Left**: The EDA channel is the dominant channel for the BVDB and the SEDB, respectively. **Right**: The Fac channel is the dominant channel for the mfeat dataset. LOPO stands for leave-one-participant-out. The features of the mfeat data set are defined in Table III. The median and mean values are defined by a horizontal line and a dot, respectively.

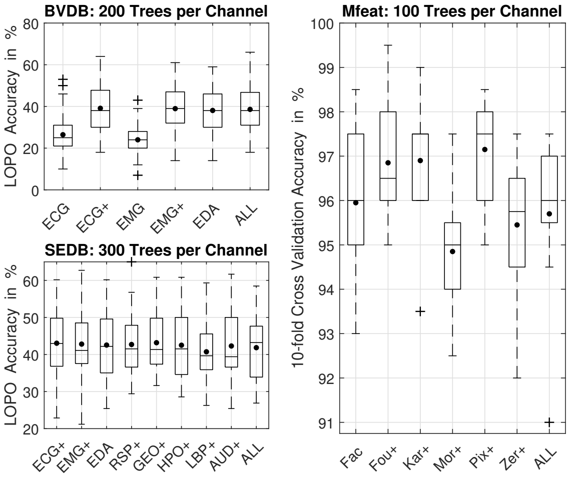

Fig. 4: **Dominant channel extension**. Channels extended by the dominant channel are denoted by an additional plus sign (+). ALL stands for the combination of all available channels (early fusion). LOPO stands for leave-one-participant-out. The features of the mfeat data set are defined in Table III. The median and mean values are defined by a horizontal line and a dot, respectively.

From Fig. 4, we can make the following observations for the BVDB. Extending the weak performing non-dominant channels (ECG and EMG) by the dominant channel (EDA) outperforms the original channels significantly. Moreover, the extended channels seem to outperform the dominant channel, and even the combination of all three channels (early fusion).

TABLE IV: **Averaged leave-one-participant-out cross validation accuracies of pairwise combined channels for the BVDB in %**. The figures in row i and column j depict the accuracy resulting from combining channel i with channel j. For each row, the best combining channel is depicted in bold. The best overall performance is underlined. The result of combining all features (early fusion) is stated in the upper left table cell. Chance level accuracy: 20%.

38.59	ECG	EMG	EDA
ECG	26.44	26.46	**38.87**
EMG	26.46	24.00	**37.76**
EDA	**38.87**	37.76	38.07

TABLE V: **Averaged leave-one-participant-out cross validation accuracies of pairwise combined channels for the SEDB Left in %**. The figures in row i and column j depict the accuracy resulting from combining channel i with channel j. For each row, the best combining channel is depicted in bold. The best overall performance is underlined. The result of combining all features (early fusion) is stated in the upper left table cell. Chance level accuracy: 25%.

41.85	ECG	EDA	RSP	GEO	LBP	AUD
ECG	33.22	**42.16**	35.46	34.77	33.90	33.39
EMG	32.13	**40.83**	33.25	32.13	30.65	30.32
EDA	42.16	42.55	42.29	42.81	**43.17**	42.32
RSP	35.46	**42.29**	33.87	35.72	34.26	34.08
GEO	34.77	**42.81**	35.72	33.16	32.55	32.99
HPO	34.00	**42.45**	35.02	33.04	31.63	31.48
LBP	33.90	**43.17**	34.26	32.55	30.62	31.30
AUD	33.39	**42.32**	34.08	32.99	31.30	30.83

TABLE VI: **Averaged 10-fold cross validation accuracies of pairwise combined channels for the mfeat data set in %**. The figures in row i and column j depict the accuracy resulting from combining channel i with channel j. For each row, the best combining channel is depicted in bold. The best overall performance is underlined. The result of combining all features (early fusion) is stated in the upper left table cell.

95.70	Fac	Fou	Kar	Mor	Pix	Zer
Fac	95.95	96.85	96.90	94.85	**97.15**	95.45
Fou	**96.85**	79.60	94.55	78.50	96.45	81.55
Kar	**96.90**	94.55	91.50	91.00	96.20	91.10
Mor	**94.85**	78.50	91.00	69.45	94.80	79.10
Pix	**97.15**	96.45	96.20	94.80	95.00	95.50
Zer	**95.45**	81.55	91.10	79.10	95.50	75.95

For the SEDB, comparing Fig. 3 to Fig. 4 (in both figures, the results for the left subset are depicted), we can make the following observations. The extended channels first, improve significantly over the original channels, second, can outperform the best channel and third, are also able to outperform the early fusion approach.

For the mfeat data set, also by comparing Figures 3 and 4, we get the same observations. The extended channels mostly outperform the original channels and the dominant one, as well as the early fusion approach.

In Tables IV, V and VI, the classification performances for each binary combination are stated for the BVDB, SEDB-L and the mfeat data set, respectively (The combination of one channel with itself represents the result achieved by this channel without any combination). In Table V, we removed the columns corresponding to the worst performing channels, i.e. EMG and HPO (for reasons of space). Thus, Table V is not symmetric. The missing accuracies are 24.00% and 30.42% for the EMG and HPO channels, respectively. The combination of both channels, which is also missing in Table V, led to an accuracy value of 29.94%.

From Tables IV, V and VI, we can observe that combining each channel with the dominant one leads to the best results, also outperforming the early fusion approach. A comparison to the late fusion approach is undertaken in Sec. VII.

VII. DC Architectures Experiments

This section provides the results for our proposed DC architectures with comparisons to the common late fusion approach, as well as to state-of-the-art results.

In this section, we define an architecture by its set of input channels. For example, the common late fusion approach for the BVDB, which has only three channels is denoted by the set {ECG, EDA, EMG}.

A. Dominant Channel Determination

In each testing cross validation (CV) step, we applied an additional stratified 10-fold cross validation based on the training data. For each fold, we designed a bagged decision tree ensemble with 50 base classifiers for each of the channels, separately. The channel with the highest accuracy (winner) was noted for each fold. Finally, the channel with the most votes (maximum 10) was defined as the dominant one.

For the BVDB, as well as the SEDB, the EDA channel was voted at least 9 times as the winner for each test subject during the 10-fold cross validation on the corresponding training data. Therefore, the EDA channel was always defined as the dominant channel for the BVBD and the SEDB, respectively. For the mfeat data set, channel Pix was defined as the dominant channel for each CV. By contrast, in Sec. VI, channel Fac was defined as the dominant channel. In Sec. VI, we calculated each channel's accuracy based on the corresponding test sets. In the current section, we determined the dominant channel, according to a real-world scenario, without any knowledge of the (current) test data. Moreover, for the determination of the dominant channel, we designed ensembles with 50 base classifiers. In Sec. VI, we used ensembles consisting of 100 base classifiers, where the Pix channel was almost as good as the Fac channel (see Fig. 3). For the mfeat data set, Table VII summarises how often each channel won in each of the 10-fold cross validations, conducted on the training data.

B. Results

TABLE VII: **Mfeat: Dominant Channel Determination**. Channel Pix was determined as the dominant channel for each cross validation (CV). The figures denote the number of votes (wins) for each 10-fold CV based on the training data.

CV	1	2	3	4	5	6	7	8	9	10
Fac	4	2	2	2	4	3	4	1	3	1
Fou	0	0	0	0	0	0	0	0	0	0
Kar	0	0	1	0	0	0	1	1	0	1
Mor	0	0	0	0	0	0	0	0	0	0
Pix	**6**	**8**	**7**	**8**	**6**	**7**	**5**	**8**	**7**	**8**
Zer	0	0	0	0	0	0	0	0	0	0

TABLE VIII: **Implemented DC architectures**. Size denotes the number of bagged base classifiers (decision trees) for each channel.

Data Set	Size	Implemented DC architectures
BVDB	200	DC_2, $DC_2 \cup \{ECG, EMG\}$
SEDB	300	DC_2, $DC_2 \backslash \{EDA\}$
Mfeat	100	DC_2, DC_3, DC_4, DC_5

In this section, we compare the results obtained by different DC architectures to state-of-the-art outcomes reported on the considered data sets. Table VIII includes different DC architectures, which we implemented in our experiments. Table IX includes the results for all data sets for the common late fusion approach, as well as the state-of-the-art outcomes and the best performing dominant channel architecture. We applied the same cross validations as in the state-of-the-art literature. For each of the data sets, we implemented four different DC architectures. For the BVDB, we stated the best achieved result from [13], which is one of the latest studies conducted solely on the biopotentials of the BVDB in the literature. The result from [13] was obtained by using an early fusion approach with the bagging method designing an ensemble of 200 decision trees as base classifiers (It is shown in [13] that the early fusion approach outperforms the common late fusion in combination with the mean rule since ECG and EMG channels both perform significantly worse than the EDA channel and hence alleviate the overall performance). For this data set, we applied the DC architectures from Table VIII, i.e. DC_2 = {ECG+, EMG+, EDA}, $DC_2 \cup$ {ECG, EMG} = {ECG, EMG, EDA, ECG+, EMG+}, as well as both aforementioned sets extended by the combination of all three channels. The best performance, which is depicted in

TABLE IX: **Mean accuracies and standard deviations in %**. For the BVDB and the SEDB, we applied the leave-one-participant-out cross validation. For the mfeat data set, we applied a 10-fold cross validation. SEDB-L and SEDB-R correspond to the left and right data set, respectively. LF: **L**ate **F**usion with mean rule. SotA: **S**tate-**o**f-**t**he-**A**rt. DC: **D**ominant **C**hannel architecture with mean rule. The DC architecture outperforms the LF approach significantly, according to the two-sided Wilcoxon signed-rank test with a significance level of 5% for the BVDB and SEDB data sets.

Data Set	LF	DC	SotA
BVDB	38.16 ± 11.4	$\mathbf{40.31 \pm 10.9}$	39.34 ± 10.2 [13]
SEDB-L	40.32 ± 7.78	$\mathbf{43.61 \pm 7.74}$	42.48 ± 8.35 [20]
SEDB-R	41.80 ± 8.04	$\mathbf{43.91 \pm 8.33}$	43.11 ± 7.93 [20]
Mfeat	98.00 ± 1.18	$\mathbf{98.60 \pm 1.02}$	$98.40 \pm$ n.a. [32]

Table IX, was achieved by the architecture defined by the set $\{$ECG, EMG, EDA, ECG+, EMG+$\}$.

For both of the SEDB subsets (SEDB-L and SEDB-R), we compare our results to the results of one of our latest studies [20], where we combined each of the channels with the common late fusion approach, however by using the pseudo inverse, which is a trainable combination rule. Applying the pseudo inverse outperforms the mean rule significantly (see Table IX). Again, we tested four different DC architectures, i.e. both of the DC architectures from Table VIII, as well as both of the architectures extended by the combination of all eight channels. The best results, which are stated in Table IX, were achieved by the DC architecture defined by the set $DC_2\backslash\{\text{EDA}\} = \{$ECG+, EMG+, RSP+, GEO+, HPO+, LBP+, AUD+$\}$.

For the mfeat data set, we compare our results to the outcomes of [32]. The authors in [32] used support vector machines with radial basis function kernels in combination with a bi-objective genetic algorithm feature selection. For this data set, we also implemented four different DC architectures, a binary, ternary, quaternary and quinary DC architecture, respectively. The best result, which is depicted in Table IX, was achieved by the quinary DC architecture, in which each modified channel consists of four original channels and the dominant channel (5 modified channels in total, since we have only 6 original channels, including the dominant one).

VIII. Discussion

In this section, we discuss the following aspects. First, we take the misidentification of dominant channels into account. Second, we discuss the possible size for a fully-established DC architecture. Subsequently, we explain why our choice of data sets seems to cover three very different classification tasks in regard to our proposed DC architectures, even though the BVDB and the SEDB seem to constitute similar data sets.

A. Identifying Sub-dominant Channels

For the BVDB, as well as for both SEDB subsets, the EDA channel leads to the best test accuracy values (see Sec. VI). Moreover, the EDA channel was clearly defined as the dominant channel in the real-world scenario, based on the corresponding evaluations of the training data (see Sec. VII). By contrast, we showed for the mfeat data set, that the Fac channel leads to the best test accuracy values (see Sec. VI). However, in the real-world scenario, channel Pix was defined as the dominant channel, based on the corresponding evaluations of the training data (see Sec. VII). Using Pix as the dominant channel led to a performance of 98.60 ± 1.02. We repeated the experiments with Fac as the dominant channel. This led to a performance of $98.65{\pm}0.82$, which is just slightly better than the result stated in Table IX. This shows that choosing a *sub-dominant* channel does not affect the quality of the DC architectures significantly. The risk of choosing a bad performing channel as the dominant one is relatively low. It is important that the dominant channel is determined based on an adequate evaluation. Thus, one has to apply a cross validation on the training data, or choose a representative subset of the training data as validation set.

TABLE X: **Sizes of complete DC architectures**. n: number of given channels. $N(n) := n + 2^{n-1} - 1$: number of channels in a complete DC architecture.

n	3	4	5	6	7	8	9	10
$N(n)$	6	11	20	37	70	135	264	521

B. Complete DC Architecture

Considering n data channels, $n \in \mathbb{N}_{>1}$, there exist $2^{n-1} - 1$ possibilities to form artificially fused channels including the dominant channel. Thus, a fully-established DC architecture consists of $n + 2^{n-1} - 1$ channels, including the n original channels (and the combination of all available channels, i.e. the early fusion, included in the 2^{n-1} term). We call the fully-established unique DC architecture *complete* DC architecture. Table X states the sizes of complete DC architectures according to different numbers of channels.

For example, for the BVDB, we have three channels (ECG, EMG, EDA). According to Table X, the complete DC architecture consists of six channels, i.e. $\{$(ECG, EMG, EDA), ECG, EMG, EDA, ECG+, EMG+$\}$.

Thus, our proposed DC architectures constitute a family of late fusion architectures with a maximum channel size of the unique complete DC architecture, and a minimum channel size of one (solely the dominant channel).

C. Adequate Choice of Data Sets in this Study

The BVDB and the SEDB are similar data sets, based on similar settings for data augmentation. For the BVDB, we used solely the physiological signals, whereas for the SEDB, we used all available recordings. For both data sets, EDA was identified as the dominant channel. However, for the BVDB, the EDA channel has a dimensionality of 70, out of 194 (see Table II), which makes $70/194 \approx 36\%$ of the whole feature space. On the other hand, for the SEDB, the EDA channel has a dimensionality of 72, out of 4413 (see Table II), which makes $72/4413 \approx 1.63\%$ of the whole feature space. Therefore, we showed that our proposed DC architectures are able to improve the overall performance, independently from the relative size of the dominant channel. In contrast to both of the data sets, we included the mfeat data set, which constitutes a *well-posed* classification task. By a well-posed classification task, we denote a classification task, in which high accuracies (significantly above 90%) can be reached easily.

IX. Conclusion

In this study, we proposed a new family of fusion approaches based on the common late fusion architecture, which we call dominant channel (DC) fusion architectures, or simply DC architectures. Given a data set, which is defined by at least three data sources/types of features (channels), the first step of our approach, is to determine the dominant channel for the current classification task. The dominant channel is defined as the best performing channel, according to the current performance evaluation measure (e.g. unweighted accuracy). Thus, the first step is to apply a k-fold cross validation ($k \in \mathbb{N}_{>1}$) for each of the given channels separately (or to define a validation set for this purpose). Step two is to design a late fusion architecture, which includes different combinations of the original channels with the dominant one. In our study, we used bagged decision tree ensembles for each of the resulting new channels in combination with the simple mean rule, for a fair comparison.

This study provides three important outcomes. First, DC architectures improve the common late fusion approach. Second, designing DC architectures with the simple mean rule is able to reach and even outperform state-of-the-art results arising from classification models that are more complex. And third, DC architectures lead to good results for cases where the dominant channel constitutes a relatively small fraction, and also where the dominant channel constitutes a *normal* fraction ($\approx 1/n$, whereby n denotes the number of channels) of the feature space. Moreover, DC architectures lead to good results on both, well-posed and *complex* classification tasks.

Acknowledgment

The research leading to these results has received fundings from the Federal Ministry of Education and Research (BMBF project: SenseEmotion). The work of Peter Bellmann and Friedhelm Schwenker is supported by the project *Multimodal recognition of affect over the course of a tutorial learning experiment* (SCHW623/7-1) funded by the German Research Foundation (DFG). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

References

[1] C. Snoek, M. Worring, and A. W. M. Smeulders, “Early versus late fusion in semantic video analysis,” in *ACM Multimedia*. ACM, 2005, pp. 399–402.

[2] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Wiley, 2004.

[3] G. Ye, D. Liu, I. Jhuo, and S. Chang, “Robust late fusion with rank minimization,” in *CVPR*. IEEE Computer Society, 2012, pp. 3021–3028.

[4] D. Liu, K. Lai, G. Ye, M. Chen, and S. Chang, “Sample-specific late fusion for visual category recognition,” in *CVPR*. IEEE Computer Society, 2013, pp. 803–810.

[5] L. Zheng, S. Wang, L. Tian, F. He, Z. Liu, and Q. Tian, “Query-adaptive late fusion for image search and person re-identification,” in *CVPR*. IEEE Computer Society, 2015, pp. 1741–1750.

[6] M. Glodek, S. Reuter, M. Schels, K. Dietmayer, and F. Schwenker, “Kalman filter based classifier fusion for affective state recognition,” in *MCS*, ser. Lecture Notes in Computer Science, vol. 7872. Springer, 2013, pp. 85–94.

[7] R. E. Kalman, “A new approach to linear filtering and prediction problems,” 1960.

[8] R. Penrose, “A generalized inverse for matrices,” in *Proceedings of the Cambridge Philosophical Society*, vol. 51, 1955, pp. 406–413.

[9] F. Schwenker, C. Dietrich, C. Thiel, and G. Palm, “Learning of decision fusion mappings for pattern recognition,” *International Journal on Artificial Intelligence and Machine Learning (AIML)*, vol. 6, pp. 17–21, 2006.

[10] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Wadsworth, 1984.

[11] L. Breiman, “Bagging predictors,” *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[12] ——, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[13] P. Bellmann, P. Thiam, and F. Schwenker, *Multi-classifier-Systems: Architectures, Algorithms and Applications*. Cham: Springer International Publishing, 2018, pp. 83–113.

[14] G. Brown and L. I. Kuncheva, “”good” and ”bad” diversity in majority vote ensembles,” in *MCS*, ser. Lecture Notes in Computer Science, vol. 5997. Springer, 2010, pp. 124–133.

[15] L. I. Kuncheva and C. J. Whitaker, “Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy,” *Machine Learning*, vol. 51, no. 2, pp. 181–207, 2003.

[16] S. Walter, S. Gruss, H. Ehleiter, J. Tan, H. C. Traue, S. C. Crawcour, P. Werner, A. Al-Hamadi, and A. O. Andrade, “The biovid heat pain database data for the advancement and systematic validation of an automated pain recognition system,” in *CYBCONF*. IEEE, 2013, pp. 128–131.

[17] M. Velana, S. Gruss, G. Layher, P. Thiam, Y. Zhang, D. Schork, V. Kessler, S. Meudt, H. Neumann, J. Kim, F. Schwenker, E. André, H. C. Traue, and S. Walter, “The senseemotion database: A multimodal database for the development and systematic validation of an automatic pain- and emotion-recognition system,” in *MPRSS*, ser. Lecture Notes in Computer Science, vol. 10183. Springer, 2016, pp. 127–139.

[18] M. Kächele, P. Thiam, M. Amirian, F. Schwenker, and G. Palm, “Methods for person-centered continuous pain intensity assessment from bio-physiological channels,” *J. Sel. Topics Signal Processing*, vol. 10, no. 5, pp. 854–864, 2016.

[19] M. Kächele, M. Amirian, P. Thiam, P. Werner, S. Walter, G. Palm, and F. Schwenker, “Adaptive confidence learning for the personalization of pain intensity estimation systems,” *Evolving Systems*, vol. 8, no. 1, pp. 71–83, 2017.

[20] P. Thiam, V. Kessler, M. Amirian, P. Bellmann, G. Layher, Y. Zhang, M. Velana, S. Gruss, S. Walter, H. C. Traue, J. Kim, D. Schork, E. Andre, H. Neumann, and F. Schwenker, “Multi-modal pain intensity recognition based on the senseemotion database,” *IEEE Transactions on Affective Computing*, pp. 1–1, 2019.

[21] P. Thiam, P. Bellmann, H. A. Kestler, and F. Schwenker, “Exploring deep physiological models for nociceptive pain recognition,” *Sensors*, vol. 19, no. 20, p. 4503, 2019.

[22] P. Thiam, H. A. Kestler, and F. Schwenker, “Two-stream attention network for pain recognition from video sequences,” *Sensors*, vol. 20, no. 3, p. 839, 2020.

[23] G. Zhao and M. Pietikaeinen, “Dynamic texture recognition using local binary patterns with an application to facial expressions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, 2007.

[24] V. Kessler, P. Thiam, M. Amirian, and F. Schwenker, “Pain recognition with camera photoplethysmography,” in *IPTA*. IEEE, 2017, pp. 1–5.

[25] ——, “Multimodal fusion including camera photoplethysmography for pain recognition,” in *ICCT*. IEEE, 2017, pp. 1–4.

[26] P. Bellmann, P. Thiam, and F. Schwenker, “Using a quartile-based data transformation for pain intensity classification based on the senseemotion database,” in *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, Sep. 2019, pp. 310–316.

[27] T. M. Cover and P. E. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[28] D. Dua and C. Graff, “UCI machine learning repository,” 2017. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Multiple+Features

[29] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, “On combining classifiers,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, 1998.

[30] D. M. J. Tax, M. van Breukelen, R. P. W. Duin, and J. Kittler, “Combining multiple classifiers by averaging or by multiplying?” *Pattern Recognition*, vol. 33, no. 9, pp. 1475–1485, 2000.

[31] F. Wilcoxon, “Individual comparisons by ranking methods,” *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[32] A. K. Das, S. Das, and A. Ghosh, “Ensemble feature selection using bi-objective genetic algorithm,” *Knowl.-Based Syst.*, vol. 123, pp. 116–127, 2017.