Hierarchical Committee of Deep CNNs with Exponentially-Weighted Decision Fusion for Static Facial Expression Recognition

Bo-Kyeong Kim, Hwaran Lee, Jihyeon Roh, Soo-Young Lee
Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST)
Daejeon, Republic of Korea
bokyeong1015@gmail.com, {hwaran.lee, rohleejh, sylee}@kaist.ac.kr

ABSTRACT

We present a pattern recognition framework to improve committee machines of deep convolutional neural networks (deep CNNs) and its application to static facial expression recognition in the wild (SFEW). In order to generate enough diversity of decisions, we trained multiple deep CNNs by varying network architectures, input normalization, and weight initialization as well as by adopting several learning strategies to use large external databases. Moreover, with these deep models, we formed hierarchical committees using the validation-accuracy-based exponentially-weighted average (VA-Expo-WA) rule. Through extensive experiments, the great strengths of our committee machines were demonstrated in both structural and decisional ways. On the SFEW2.0 dataset released for the 3rd Emotion Recognition in the Wild (EmotiW) sub-challenge, a test accuracy of 57.3% was obtained from the best single deep CNN, while the single-level committees yielded 58.3% and 60.5% with the simple average rule and with the VA-Expo-WA rule, respectively. Our final submission based on the 3-level hierarchy using the VA-Expo-WA achieved 61.6%, significantly higher than the SFEW baseline of 39.1%.

Categories and Subject Descriptors

I.4.9 [**Image Processing and Computer Vision**]: Applications

Keywords

Hierarchical Committee; Exponentially-Weighted Decision Fusion; Deep Convolutional Neural Network

1. INTRODUCTION

Committee machines (also known as classifier ensembles) generally yield a better performance than a single classifier [1, 2], and thus have extensively applied in various research fields including vision [3, 4], speech [5, 6], text [7], and bio-data [8]. From previous studies on designing a good committee to outperform its individual members, *generating diverse decisions from various individuals* has been shown to be crucial in *providing complementary information about input data* [9, 10].

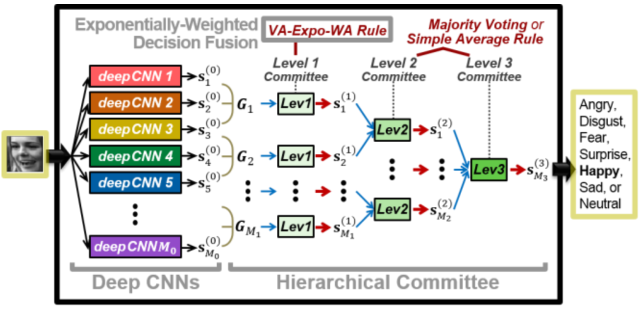

Figure 1. The overall system for facial expression recognition

With recent advances in deep learning and parallel computing, forming a committee of multiple deep neural networks was presented in [11, 12], has attained impressive successes [13-15], and now becomes a widely used approach [16-18]. Particularly, in this paper, we investigate the multi-column deep neural network (MCDNN) [14] for static facial expression recognition in the wild (SFEW). The standard MCDNN is a committee of deep convolutional neural networks (deep CNNs) with a simple averaging decision rule in a single structure level. Since the MCDNN has already proved its superiority in many visual classifications, we expect that its excellence in recognition could also be demonstrated in SFEW.

More importantly, we present 2 simple yet effective ways to improve the MCDNN: '*training more diverse individuals*' and '*forming a better committee in both decisional and structural aspects*'. The former is achieved by designing various network architectures in addition to applying the commonly-used schemes (e.g., different input normalizations and different random weight initialization). Furthermore, we adopt several strategies for using external data in training deep CNNs in order to pursue more diverse decisions and errors. The latter is achieved with a better ensemble rule based on an exponentially-weighted decision fusion. Moreover, we build a hierarchical committee which can make more reliable decisions. As structural levels in the committee become higher, the consensus of multiple sub-groups could be formed and thus enhance the reliability of decisions. The overall proposed system is shown in Figure 1.

Our framework based on the improved committee machines of deep CNNs is tested on the SFEW 2.0 database [19], released for a sub-competition in the 3rd Emotion Recognition in the Wild 2015 (EmotiW2015) challenge. The remainder of this paper describes the proposed approach in detail, experimental results (including our test-label submissions for this SFEW competition), and conclusions.

ICMI '15, November 09-13, 2015, Seattle, WA, USA
© 2015 ACM. ISBN 978-1-4503-3912-4/15/11…$15.00
DOI: http://dx.doi.org/10.1145/2818346.2830590

2. PROPOSED APPROACH

2.1 Deep CNNs as Individual Members

A deep CNN consists of several feature extraction stages (with alternating convolutional and pooling layers), followed by a recognition stage (with fully-connected layers) [15, 20]. Because of its excellent classification ability as well as hierarchical feature development mimicking the human visual system, we selected the deep CNN for the base member of a committee as in a standard MCDNN.

To build diverse deep CNNs in standard MCDNNs, being trained with 'different training data sets' was mainly focused rather than using 'different classifiers'. The effect of 'different training data' was achieved by several preprocessing methods on the original data such as deformation and normalization. For the effect of 'different classifiers', the MCDNNs applied multiple random seeds for weight initialization, but the identical network architectures were used for all individual members. We believed that *various network architectures* also largely contributed to obtaining different classifiers and thus to increasing diversity of decisions in forming a committee. Therefore, we applied various architectures for deep CNNs as well as differently preprocessed data and different weight initialization. Furthermore, we explored several training *strategies for making use of external data* along with the SFEW data in order to pursue more diverse errors. The experimental details about how to build the individual deep CNNs and their recognition accuracies are presented in Section 4.

2.2 Exponentially-Weighted Decision Fusion

When forming a committee, how to combine decisions from individual members has been extensively investigated. In this paper, we first explored 3 widely-used rules for decision fusion [1, 2]: the majority voting, median rule, and simple average rule. Then, we introduced an effective combination rule based on exponential weighting to give more weights on well-performed individuals.

The 'majority voting' directly uses the predicted class labels to select a class with the largest number of votes. On the other hands, instead of using the labels, the 'median rule' and 'simple average rule' use the class-related continuous confidences or scores. In our experiments, the median/simple average rule decided a class with the highest median/average of posterior class probabilities yielded from deep CNNs. For these 3 rules, the individuals have equal rights for participation so that any reliability or importance on each of their decision is not considered.

A straightforward way to regard the importance of members' decisions is to compute a weighted mean of class scores with assigning the weights as validation performances. We denoted it as the 'VA-Simp-WA' rule, short for the validation-accuracy-based simple weighted average. However, when the committee members yield the similar accuracies and thus almost equal weights are used, the VA-Simp-WA does not differ from the 'simple average rule'. Our exponentially-weighted decision fusion has been motivated by considering the aforementioned case. In determining the weights, we adopted *an exponential function which influences on the differences between numbers* (e.g. '3^1-2^1=1' < '3^2-2^2=5'). We expected that *this characteristic of exponent can give more weights on the members with (even slightly) higher accuracies*.

Let us denote our method as the '*VA-Expo-WA*' rule, short for the validation-accuracy-based exponentially-weighted average, and continue our discussion with mathematical notations. Suppose a member model m (=1,...,M) with its validation accuracy of z_m provides a posteriori class probability vector $\mathbf{s}_m$ for an input pattern. Then, the final ensemble of M models' decisions in the VA-Expo-WA becomes

$$\mathbf{s}_{final} = \frac{\sum_{m=1}^{M}(z_m)^q\,\mathbf{s}_m}{\sum_{m=1}^{M}(z_m)^q} = \sum_{m=1}^{M} d_m \mathbf{s}_m \tag{1}$$

where a decision weight d_m reflects the normalized significance of the model m's decision ($0 \leq d_m \leq 1$) and an exponent q is a hyper-parameter to determine how much the qualified members are emphasized (q>1) or de-emphasized ($q \leq 1$). Finally, a class with the highest value in exponentially-weighted class probabilities is chosen. Here, the value of q is found by a simple uniform search: scanned over [-50:0.1:150] and selected to provide the maximum performance on validation data after the fusion. The scanning procedure and the corresponding decision weights for the selected q are illustrated in Figure 2. We confirmed that this searching method required little additional computation, while it found a proper q which can improve the generalization on validation data. Note that the VA-Expo-WA rules with q=0 and q=1 are identical to the simple average rule and the VA-Simp-WA, respectively. The superior performance of our VA-Expo-WA rule compared to the commonly-used decision fusion rules are shown in Section 5.

2.3 Hierarchical Committee

The existing literatures using hierarchical architectures of committees aimed to divide a hard problem into the easier ones (based on the divide-and-conquer strategy) with a statistical framework [21, 22] and/or to efficiently combine the outputs of different classifiers trained on heterogeneous features [3, 23]. Meanwhile, we constructed hierarchical committees of deep CNNs with the following procedure according to the 2 expected merits.

1) Organize the M_0 individual members into the 1st level sub-groups, $\{\boldsymbol{G}_1, \ldots, \boldsymbol{G}_{M_1}\}$ having some overlapped members. After that, make a decision for each group according to the 1st level decision fusion rule.
2) Collect all sub-groups' decisions in the lth level, $\{\mathbf{s}_{m_l}^{(l)},\ m_l = 1, \ldots, M_l\}$ where l (= 1, …, L-1) and m_l are indices for the level and the sub-group, respectively. Then, re-organize them into the (l+1)th level groups, and make a decision for each group according to the (l+1)th level decision fusion rule.
3) Repeat '2)' until reaching to get a final decision at the last Lth level.

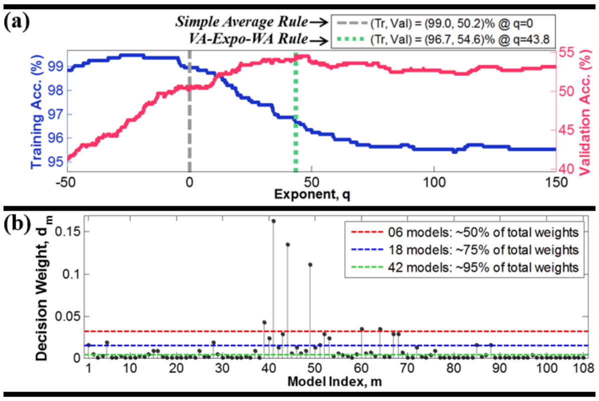

Figure 2. (a) The training and validation accuracies (%) as an exponent '*q*' is scanned in the exponentially-weighted decision fusion and (b) the corresponding decision weights for the selected *q*. This result was obtained when 108 models formed a committee with the VA-Expo-WA rule (see Section 5).

The first merit is that, *more reliable decisions* could come from the *strong consensus of multiple sub-groups* in *higher structural levels*. Second, the *increased diversity of errors* could be obtained by *setting some members to be overlapped in certain sub-groups*. Then, depending on other members in these groups, the *overlapping members differently contribute to the next level decision*. Since the former is quite intuitive, let us explain the latter with a toy example. Suppose a 2-class problem and 5 member classifiers of (a, b, c, d, e) who claim the class label for an input sample as (1, 1, 1, 2, 2), respectively. When they are divided into 2 sub-groups, $\boldsymbol{G}_1$: {a, b, c} and $\boldsymbol{G}_2$: {c, d, e}, with an overlapping member 'c' and the majority voting is applied, the 'c' differently contributes to the final decision. More specifically, without grouping, there is no doubt of the selection of class 1 by 3 votes from a, b, and c among 5 members. However, with grouping, both classes get the equal number of mid-level decisions (the class 1 from $\boldsymbol{G}_1$ and the class 2 from $\boldsymbol{G}_2$) due to the different impacts of c's claim on both sub-groups, so the final decision depends on the mean class probabilities. We expected that these groups with overlapping members could lead to more various decisions in the low structural levels, finally serving as diverse errors in the last level. See Section 6 for the experimental details and favorable classification results of our hierarchical committees, which outperform the standard single-level committees.

3. FACE REGISTRATION

3.1 SFEW 2.0 Database

The SFEW 2.0 database [19] was created by extracting frames from emotional movie clips in the AFEW data corpus [24]. The task was to assign 7 expression labels (angry, disgust, fear, happy, sad, surprise, and neutral) to these frames in close-to-real world conditions. For the training, validation, and test set, the SFEW database contains 958, 436, and 372 images, respectively.

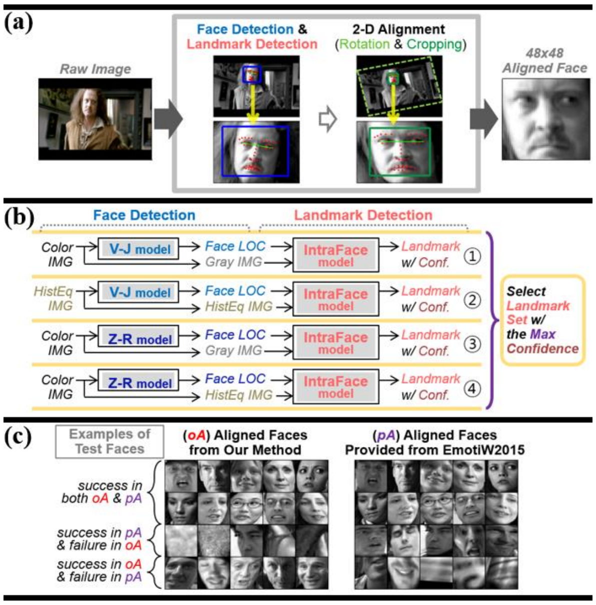

Figure 3. (a) Face registration based on a 2-D alignment, (b) our multi-pipeline-based alignment, and (c) examples of aligned faces for test data. In (b), the terms 'IMG', 'HistEq', 'LOC', and 'Conf' are short for 'image', 'histogram equalized', 'location', and 'confidence', respectively. In (c), faces in the corresponding positions between oA and pA are processed from the same test image.

3.2 Face Registration

For face registration, we conducted a conventional 2-D alignment based on eye locations as illustrated in Figure 3(a). To improve robustness in face/landmark detection, multiple detection pipelines were designed to produce different landmark estimations, and the best estimation among them was finally used for the 2-D alignment. We used the Viola-Jones (V-J) model [25] and the Zhu-Ramanan (Z-R) model [26] for face detection along with the IntraFace model [27] for landmark detection. As shown in Figure 3(b), we considered 4 single pipelines based on the following observations: i) some faces, failed by the V-J, could be detected by the Z-R, and vice versa, ii) depending on face locations from the V-J and Z-R, the landmark estimation of the IntraFace became different, iii) the V-J and IntraFace sometimes yielded complementary outputs when histogram-equalized images were used as input. Among 4 possible landmark sets from those pipelines, the landmark set with the highest confidence provided from the IntraFace was eventually selected for alignment.

In Table 1, we compared the performances between our multi-pipeline-based alignment and single-pipeline-based ones. For each data type, we computed the ratio of successful alignments to the whole number of samples as well as the ratio of cases where only face detection succeeded. Our 4-pipeline-based alignment performed better than the single-pipeline ones, implying that complementary detection results were obtained from 4 single pipelines. Therefore, combining them can lead to the robust face registration in real-world conditions. Note that, for training and validation data, the erroneous and failed alignments were semi-automatically processed by hands for a later usage in training deep models. However, *for testing data*, any human intervention was not applied in the context of *a fully-automatic system*.

Figure 3(c) depicts examples for test faces processed by our alignment method and provided from EmotiW2015. As shown in the 1st and 2nd rows of the figure, the faces processed from our method (oA) were more similarly aligned each other compared to the provided alignments (pA). It could lead to superior accuracies of oA as denoted in Table 2. However, in addition to oA, we also used pA in training deep models for the following reasons: for giving deformation effects (such as translation and rotation) to ours and for providing complementary information when either oA or pA failed (as shown in the 3rd and 4th rows of the figure).

Table 1. Alignment-success rate (%) of our alignments

Alignment Method		**Success Rate (%)**					
		Alignment (Both Face & Landmark Detection)			**Only Face Detection**		
		Train	Valid	Test	Train	Valid	Test
Single-Pipeline-Based	①	70.5	70.4	73.1	1.4	1.1	2.4
	②	71.6	74.1	77.2	2.6	1.8	3.5
	③	58.6	60.8	57.5	25.5	29.8	30.9
	④	56.6	56.4	53.5	27.5	34.2	34.9
Multi-Pipeline-Based		**81.8**	**83.5**	**90.1**	9.5	6.0	4.0

Table 2. Validation (& testing, if available) accuracy (%) of our alignments (oA) and provided alignments (pA)

Classification Method	**oA**	**pA**
{LPQ-pHOG} + rbfSVM: **baseline** [19]	-	36.0 (39.1)
A single deep CNN[a]: ***PREP***iNor – {***CNN***L – ***FC***3072}***R***1	52.5 (57.3)	46.8
A committee of 108 deep CNNs[b] with the VA-Expo-WA rule	54.6 (60.5)	52.2 (56.7)

[a, b] For the detailed information, see Section 4.3 and 5, respectively.

4. DESIGNING AND TRAINING INDIVIDUAL COMMITTEE MEMBERS

In this section, we first described how to design multiple deep CNNs to pursue diverse errors in forming a good committee. Next, for training these models, how to make use of external data along with the SFEW data was presented with other learning details. Finally, classification results of individual models were examined.

4.1 Designing Individual Deep CNNs

With the aim of getting diversity of decisions and errors from individual members, we designed 216 deep CNNs using different input preprocessing methods, random weight initializations, and network architectures. A single deep CNN is denoted as (2) and the detailed explanations for sub-notations are followed.

$$\boldsymbol{PREP}\alpha_1,\alpha_2 - \{\,\boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma\,\}_{\boldsymbol{R}\delta} \tag{2}$$

$$\text{where}\begin{cases} \boldsymbol{PREP}\alpha_1,\alpha_2\text{: preprocessing type of data,} \\ \qquad \alpha_1 = \begin{cases} \text{raw} & \mathit{Raw} \\ \text{iNor} & \mathit{Illumination\ Normalization} \\ \text{cEnh} & \mathit{Contrast\ Enhancement} \end{cases} \\ \qquad \alpha_2 = \begin{cases} \text{oA} & \mathit{Our\ Aliged\ Faces} \\ \text{pA} & \mathit{Provided\ Aligned\ Faces} \end{cases} \\ \boldsymbol{CNN}\beta\text{: size of CNN receptive field,} \\ \qquad \beta = \begin{cases} \text{S} & \mathit{Small} \\ \text{M} & \mathit{Medium} \\ \text{L} & \mathit{Large} \end{cases} \\ \boldsymbol{FC}\gamma\text{: number of neurons in a fully-connected hidden layer,} \\ \qquad \gamma = \{3072, 2048, 1024, 512\} \\ \boldsymbol{R}\delta\text{: random seed number for weight initialization,} \\ \qquad \delta = \{1, 2, 3\} \end{cases}$$

4.1.1 Preprocessing type of data

We considered various normalization techniques on differently aligned faces. Each raw image was rescaled from 0 to 1 via a min-max normalization. To reduce illumination variation in images, as used in [28], the isotropic diffusion based normalization [29] from INface toolbox [30] was applied with the default parameter setting. Moreover, to enhance contrast for each image, as used in [14], we applied the histogram equalization implemented in MATLAB. The examples of normalized images are shown in Figure 4(a).

For different input deformations (e.g., translation and rotation), we used the aligned faces provided from EmotiW2015 (pA) as well as our aligned faces (oA), as previously introduced in Section 3.2. Furthermore, some faces erroneously aligned by our method can be compensated by the provided alignments, thus leading to more diverse and complementary information about input data.

4.1.2 Architectures of deep CNNs

As the baseline architecture, we referred to Tang's deep CNN [31], the winner model of ICMLW2013's facial expression recognition challenge [32]. It consisted of the 1 input-transform and 3 convolution+pooling stages, followed by fully-connected hidden and output layers. In the input-transform stage for image mirroring and translating, data were augmented by extracting 42x42 patches from the 48x48 faces. The subsequent layers corresponded to a configuration of {***CNN***M - ***FC***3072} in our notation (see Table 3), except for average-pooling in the 2nd and 3rd stages of Tang's model. To get more specific settings, see his implementation at [33].

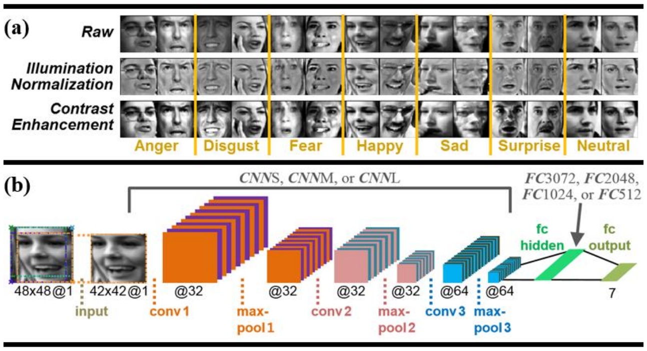

Figure 4. (a) Various normalizations on the aligned validation faces and (b) a deep CNN architecture

Based on Tang's architecture, we designed diverse deep CNNs by changing the sizes of filters for various receptive fields and by changing the number of neurons in a fully-connected hidden layer as denoted in Table 3. The ***CNN***M had a medium-size receptive field (with 5x5, 4x4, and 5x5 filters for each conv layer, respectively), the ***CNN***L had a relatively large receptive field (with 7x7 filters for all conv layers), and the ***CNN***S had a relatively small one (with 3x3 filters for all conv layers). For all ***CNN*** types, the strides and pads were properly set to ensure the same sizes of output maps (5x5 @64) in the max-pool 3 layer. Moreover, for each ***CNN*** type, 4 kinds of fully-connected hidden layer (***FC***) were used. In this ***FC*** layer, the dropout [34] was applied to reduce over-fitting in training deep models. Notice that, from Tang's model, we modified the pooling layers in the 2nd and 3rd stages from average-pooling to max-pooling, since it provided better classification results in our preliminary experiments. For the nonlinearity, Rectified Linear Unit (ReLU) activations were applied for all conv and penultimate layers, and a softmax activation was for the output layer.

Table 3. Configuration of deep CNNs

Layer[a]	***CNN***S		***CNN***M		***CNN***L	
	maps[b]	kernel[c]	maps	kernel	maps	kernel
input	42x42 @1	-	42x42 @1	-	42x42 @1	-
conv 1	42x42 @32	**3x3**, (1,1)	42x42 @32	**5x5**, (1, 2)	42x42 @32	**7x7**, (1, 3)
max -pool 1	21x21 @32	2x2, (2, 0)	21x21 @32	3x3, (2, 1*)	21x21 @32	2x2, (2, 0)
conv 2	19x19 @32	**3x3**, (1, 0)	20x20 @32	**4x4**, (1, 1)	19x19 @32	**7x7**, (1, 2)
max -pool 2	10x10 @32	2x2, (2, 1*)	10x10 @32	3x3, (2, 1*)	10x10 @32	2x2, (2, 1*)
conv 3	10x10 @64	**3x3**, (1, 1)	10x10 @64	**5x5**, (1, 2)	10x10 @64	**7x7**, (1, 3)
max -pool 3	5x5 @64	2x2, (2, 0)	5x5 @64	3x3, (2, 1*)	5x5 @64	2x2, (2, 0)
fc hidden	***FC***3072: 3072 neurons with a dropout probability = 0.8, ***FC***2048: 2048 neurons with a dropout probability = 0.5, ***FC***1024: 1024 neurons with a dropout probability = 0.5, or ***FC***512: 512 neurons with a dropout probability = 0.5					
fc output	7 neurons (one per class)					

[a] conv, max-pool, fc: convolutional, max-pooling, fully-connected.

[b] maps: *the size of output maps @ the number of output maps*.

[c] kernel: *the size of kernels*, (*stride*, *pad*) where '*stride*' refers to spacing size of kernels, '*pad* without an asterisk*' refers to zero-padding to all 4 spatial directions (top, bottom, left and right directions) of input maps, and '*pad* with an asterisk*' refers to zero-padding to the top and left.

4.2 Training Individual Deep CNNs

4.2.1 *Usage of external data*

The size of SFEW data is quite small to train deep CNNs. Inspired by [28], we also decided to use 2 external databases along with the SFEW data for training models: the Facial Expression Recognition 2013 database (FER-2013 DB) [35] and the Toronto Face Dataset (TFD) [36]. The FER-2013 DB, released for ICMLW2013's sub-challenge [32], was created using the Google image search API. Since realistic facial expressions were collected from the internet, large variations reflecting real-world conditions existed in the FER-2013 DB. From this dataset, 28,698 training faces (after removing 11 non-number-filled images from original training data) and 3,589 private testing faces were used for our experiment. The TFD was constructed by merging together 30 pre-existing face datasets. The faces in TFD were strictly aligned and almost all of them were fully-frontal. From the TFD, 4,178 labelled faces were used. Notice that both datasets contained 48x48 gray-scale faces labelled with the identical 7 expression categories used in the SFEW data.

After determining the external databases to be used, we explored how to use them together with the SFEW data for training models. The following 3 strategies were considered:

i. Random initialization ⇒ In learning, using data as follows:
{FER-2013 DB + TFD} for 'training'
{*SFEW Train* + *SFEW Valid*} for 'validation'

ii. Random initialization ⇒ In learning, using data as follows:
{FER-2013 DB + TFD + *SFEW Train*} for 'training'
{*SFEW Valid*} for 'validation'

iii. Initialization from *a pre-trained model* constructed by using
{FER-2013 DB Train + TFD} for 'training'
{FER-2013 DB Test} for 'validation'
⇒ In learning, using data as follows:
{*SFEW Train*} for 'training'
{*SFEW Valid*} for 'validation'

In [28], the strategies 'i' and 'ii' were discussed to train a deep CNN which yielded per-frame predictions for video-based emotion recognition. In their experiment, the strategy 'i' was finally selected based on a better validation performance. In addition to the 'i' and 'ii', we also investigated one type of *transfer learning* scheme as denoted in the strategy 'iii' [37].

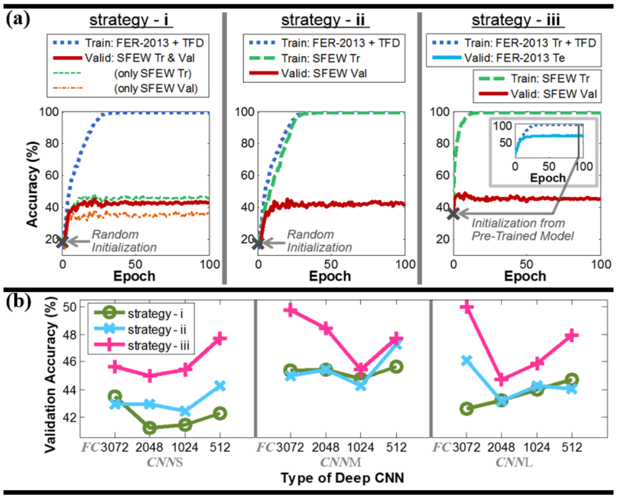

Figure 5. Comparison of 3 strategies for using external data along with SFEW data in training: (a) learning curves from a deep CNN and (b) validation accuracies for 12 deep CNNs

To figure out the most proper usage of external data for this SFEW competition, we evaluated the performances of several models trained differently with the aforementioned 3 strategies. Specifically, the following 12 deep CNNs having various architectures were used:

$$\boldsymbol{PREP}\text{raw, oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R1}} \quad \text{for } \forall\beta, \forall\gamma \tag{3}$$

Figure 5(a) depicts learning curves during training a deep CNN of {***CNN***M - ***FC***2048} with 3 strategies. The strategy 'iii' provided superior performances not only at the initial epoch but also at the convergence. Figure 5(b) shows validation accuracies for all examined training strategies and network architectures. Regardless of architectures, the 'iii' outperformed the other two. It indicates that the transfer learning scheme is effective because of representing more similar and suitable feature distributions between training and validation data. Therefore, we eventually decided to use the strategy 'iii'. We first pre-trained 108 deep CNNs using two external data of FER-2013 DB and TFD. Then, 216 models were fine-tuned using the SFEW data: 108 models fine-tuned using oA + 108 using pA. Note that, for the last two submissions, in addition to these 216 models trained with the 'iii', we also incorporated the 24 models with the 'i' and 'ii' to form a committee for a better handling of various facial expressions.

4.2.2 *Other training details*

We used the MatConvNet toolbox [38] on NVIDIA GeForce GTX 690 GPUs. Each deep CNN was trained using the stochastic gradient descent with a batch size of 200 and momentum of 0.9. Except for the last fully-connected layer with weight decay of 0.002, weight decay of 0.0001 was applied for all other layers. Moreover, the learning rate was equal for all layers, while its value started from 0.004 and became half at every 25 epoch. During total 100 epochs, we selected a model yielding the max validation performance.

To avoid over-fitting, the dropout and data augmentation were applied. A dropout probability of 0.8 was used for deep CNNs with ***FC***3072, while 0.5 was used for ***FC***2048, ***FC***1024, and ***FC***512. The training data were augmented by 10 times, through using 5 crops of size 42x42 (1 from resizing an original 48x48 face and 4 from extracting its 4 corners) and their horizontal flopping. At the test phase, to maintain consistency with the training, 10 patches extracted from each face were fed to the model and the corresponding 10 predictions were averaged to produce a final prediction.

4.3 Classification Performance of Deep CNNs

For 216 deep CNNs trained with the strategy 'iii' (a transfer learning scheme), their classification rates to validation data were reported in Table 4. Our best single model with the highest validation accuracy of 52.5% was ***PREP***iNor,oA – {***CNN***L – ***FC***3072}$_{\boldsymbol{R1}}$. This model became the 1st submission for the SFEW competition, yielding a test accuracy of 57.3%.

We also analyzed general tendencies in performances of deep CNNs. In the aspect of face alignments, the 108 models trained using oA showed a better mean accuracy than those using pA. Furthermore, to examine the trends according to preprocessing types and CNN architectures, we computed the mean accuracy of 36 models for each preprocessing (**per-*PREP* group**) and for each CNN (**per-*CNN* group**). The illumination normalization was superior to other preprocessing types, and the ***CNN***M with the medium size of receptive field performed better than other CNN architectures.

Table 4. Validation accuracy (%) of individual deep CNNs. 108 models (top) trained using our aligned faces and 108 (bottom) using provided alignments from EmotiW2015. The highest accuracy for a given architecture (each column) is written in bold. The asterisk* denotes the single best model.

***Aligned Faces From Our Method* (oA)**		*CNN*S				*CNN*M				*CNN*L				**Mean (Std) of per-*PREP* group**: 36 models / group
		FC 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	***FC*** 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	***FC*** 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	
PREP raw, oA	***R*1**	45.6	45.0	45.4	47.7	49.8	48.4	45.4	47.7	50.0	44.7	45.9	47.9	*46.9* *(1.7)*
	***R*2**	46.3	45.0	46.8	45.0	47.5	48.4	49.1	49.1	47.5	46.1	45.0	46.6	
	***R*3**	45.0	43.6	47.0	45.6	49.1	47.0	47.3	50.0	48.4	45.9	47.3	46.1	
PREP iNor, oA	***R*1**	48.6	**49.5**	**48.6**	49.1	49.1	46.3	**50.9**	50.2	**52.5***	**49.5**	**50.5**	**52.3**	*49.2* *(1.6)*
	***R*2**	47.9	46.3	48.4	**50.7**	**52.1**	**49.5**	49.8	50.5	50.2	47.9	48.6	48.2	
	***R*3**	**48.9**	46.3	46.6	49.5	48.6	48.9	47.9	**50.7**	47.5	47.0	**50.5**	50.5	
PREP cEnh, oA	***R*1**	45.9	44.5	45.0	45.9	47.5	45.2	48.2	49.1	42.9	41.3	43.4	43.1	*45.1* *(2.4)*
	***R*2**	45.2	44.7	43.8	44.0	49.8	47.5	48.2	49.8	45.2	42.9	39.7	41.3	
	***R*3**	43.4	43.4	46.3	44.7	48.2	44.7	45.9	45.2	45.6	42.2	43.8	45.2	
Mean (Std) of per-*CNN* group: 36 models / group		*46.3* *(1.9)*				*48.4* *(1.7)*				*46.5* *(3.2)*				**Total 108 models** *47.0 (2.5)*

***Aligned Faces Provided From EmotiW2015* (pA)**		*CNN*S				*CNN*M				*CNN*L				**Mean (Std) of per-*PREP* group**: 36 models / group
		FC 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	***FC*** 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	***FC*** 3072	***FC*** 2048	***FC*** 1024	***FC*** 512	
PREP raw, pA	***R*1**	43.8	43.6	40.8	45.7	48.0	45.2	46.1	45.4	44.5	44.3	45.7	46.8	*44.5* *(2.0)*
	***R*2**	43.1	41.9	42.9	40.5	45.9	45.4	46.8	46.4	45.4	43.8	42.9	45.4	
	***R*3**	43.8	39.6	42.9	41.7	47.1	45.7	46.1	44.5	45.0	44.3	45.0	47.3	
PREP iNor, pA	***R*1**	46.4	43.8	**45.4**	42.6	**47.5**	44.7	**48.7**	45.4	46.8	44.0	47.1	45.0	*46.1* *(1.9)*
	***R*2**	**47.5**	44.3	43.6	**46.4**	46.8	**46.6**	**48.7**	45.4	47.3	**45.9**	**49.0**	**47.8**	
	***R*3**	46.6	**45.2**	43.6	46.1	46.8	45.7	45.4	**52.2**	**47.8**	45.7	43.3	46.1	
PREP cEnh, pA	***R*1**	44.0	39.1	42.2	41.2	46.4	46.1	46.6	47.8	42.2	42.4	42.2	43.8	*43.5* *(2.4)*
	***R*2**	43.8	41.9	44.3	43.6	**47.5**	45.4	44.7	47.8	41.7	42.2	43.6	43.1	
	***R*3**	43.3	40.8	42.4	40.5	46.4	41.0	44.0	46.8	40.3	39.3	41.9	44.3	
Mean (Std) of per-*CNN* group: 36 models / group		*43.3* *(2.0)*				*46.3* *(1.8)*				*44.5* *(2.2)*				**Total 108 models** *44.7 (2.4)*

5. EXPONENTIALLY-WEIGHTED DECISION FUSION

Before moving on examining a hierarchical committee, we demonstrated the superiority of our VA-Expo-WA rule in a conventional single-level committee, by comparing it to the widely-used rules for decision fusion. Here, we formed the 3 committees consisting of 108 models trained using our aligned faces (oA; corresponding to the case of Figure 2), 108 models using provided alignments (pA), and 216 using both alignments of oA and pA. As shown in Table 5, regardless of the committee types, our VA-Expo-WA rule outperformed all other rules. In addition, as we expected, the VA-Simp-WA did not much differ from the simple average rule since individual models produced even and similar validation accuracies as denoted in Table 4.

Table 5. Validation (& testing, if available) accuracy (%) of single-level committees with various decision fusion rules

Decision Fusion Rule	**Single-Level Committee**		
	108 models from oA	**108 models from pA**	**216 models from both**
Majority Voting	51.6	48.2	50.7
Median	50.7	47.3	49.8
Simple Ave. (q=0)	50.2 (58.3)	47.8	49.8
VA-Simp-WA (q=1)	50.7	47.8	49.8
VA-Expo-WA	**54.6 (60.5)**[a]	52.2 (56.7)[b]	**56.4 (60.0)**[c]

[a, b, c] For single-level committees with the VA-Expo-WA rule, the values of exponent q were selected as 43.8, 58.1, and 60.5, respectively.

Meanwhile, it is worth noting that even though the committee of 216 models with the VA-Expo-WA provided the highest classification rate on validation data, its test rate was not the best. The best test accuracy of 60.5% was achieved by the committee of 108 models using oA. It may imply that, at some level yielding a maximum validation performance, adding other models to the committee did not work properly with the VA-Expo-WA. Rather, it seemed to harm the generalization on test data because too many models were participated to improve the final validation accuracy of the committee.

We further investigated the strength of our VA-Expo-WA rule that can reveal the importance or contribution of each individual on a final ensemble. Figure 2(b) shows the decision weights of 108 models using oA, and we computed the threshold above which a certain portion of the total weights was covered. The 6, 18, and 42 models were in charge of about 50%, 75% and 95% of the total, respectively. These information could be used for model selection and pruning.

6. HIERARCHICAL COMMITTEE

To construct hierarchical committees, we organized 216 deep CNNs (obtained with the training strategy 'iii') into 12 subgroups for the 1st level, having some overlapping members:

$$\boldsymbol{G_1}: \boldsymbol{PREP}\text{raw, oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$
$$\boldsymbol{G_2}: \boldsymbol{PREP}\text{iNor, oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$
$$\boldsymbol{G_3}: \boldsymbol{PREP}\text{cEnh, oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$

$$\boldsymbol{G_4}: \boldsymbol{PREP}\text{raw, pA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$
$$\boldsymbol{G_5}: \boldsymbol{PREP}\text{iNor, pA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$
$$\boldsymbol{G_6}: \boldsymbol{PREP}\text{cEnh, pA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \ \text{ for } \forall\beta, \forall\gamma, \forall\delta$$

$$\boldsymbol{G}_7\text{: } \boldsymbol{PREP}\alpha_1, \text{oA} - \{\, \boldsymbol{CNN}\text{S} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$
$$\boldsymbol{G}_8\text{: } \boldsymbol{PREP}\alpha_1, \text{oA} - \{\, \boldsymbol{CNN}\text{M} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$
$$\boldsymbol{G}_9\text{: } \boldsymbol{PREP}\alpha_1, \text{oA} - \{\, \boldsymbol{CNN}\text{L} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$

$$\boldsymbol{G}_{10}\text{: } \boldsymbol{PREP}\alpha_1, \text{pA} - \{\, \boldsymbol{CNN}\text{S} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$
$$\boldsymbol{G}_{11}\text{: } \boldsymbol{PREP}\alpha_1, \text{pA} - \{\, \boldsymbol{CNN}\text{M} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$
$$\boldsymbol{G}_{12}\text{: } \boldsymbol{PREP}\alpha_1, \text{pA} - \{\, \boldsymbol{CNN}\text{L} - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}\delta} \text{ for } \forall\alpha_1, \forall\gamma, \forall\delta$$

Each sub-group consisted of 36 deep CNNs. The 3 **per-*PREP* group**s (***G***$_1$-***G***$_3$) and 3 **per-*CNN* group**s (***G***$_7$-***G***$_9$) were formed from 108 models trained using our aligned faces (oA), and similarly the 3 **per-*PREP*** (***G***$_4$-***G***$_6$) and 3 **per-*CNN*** (***G***$_{10}$-***G***$_{12}$) **group**s were from 108 models using provided alignments (pA).

There were several ways to build the hierarchy in '*structural*' aspects (regarding *the number of hierarchical levels* and *the structure for re-combination of higher-level decisions*) and in '*decisional*' aspects (regarding *the decision fusion rule* for each level). At an earlier phase of experiments, we submitted the predicted test labels obtained from adopting the VA-Expo-WA rule for all structural levels in the hierarchy. From these submissions, we got some unexpected results; validation accuracies were higher than our previous submissions, but testing accuracies dropped. However, we learned an empirical lesson; applying the VA-Expo-WA rule for all levels of a hierarchy may have a negative impact on generalization for test data, since the exponent selection was optimized for the validation performance. Therefore, we decided to use the VA-Expo-WA rule only for the 1st level. For decision fusions in higher levels, the majority voting and simple average rules were used for a better generalization.

We first considered a simple 2-level hierarchical structure as illustrated in Figure 6(a). With a fixed VA-Expo-WA rule of the 1st level, we varied decision fusion rules of the 2nd level as the majority voting or simple average rules. Moreover, as mentioned in Section 4.2.1, for handling more various face expressions and pursuing more diverse errors, we additionally examined 24 models from the training strategies 'i' and 'ii'. These models were also formed into the 1st level groups, having the 12 models each:

$$\boldsymbol{g}_1\text{: } \boldsymbol{PREP}\text{raw}, \text{oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}1} \text{ for } \forall\beta, \forall\gamma,$$
trained with the ***strategy 'i'***

$$\boldsymbol{g}_2\text{: } \boldsymbol{PREP}\text{raw}, \text{oA} - \{\, \boldsymbol{CNN}\beta - \boldsymbol{FC}\gamma \,\}_{\boldsymbol{R}1} \text{ for } \forall\beta, \forall\gamma,$$
trained with the ***strategy 'ii'***

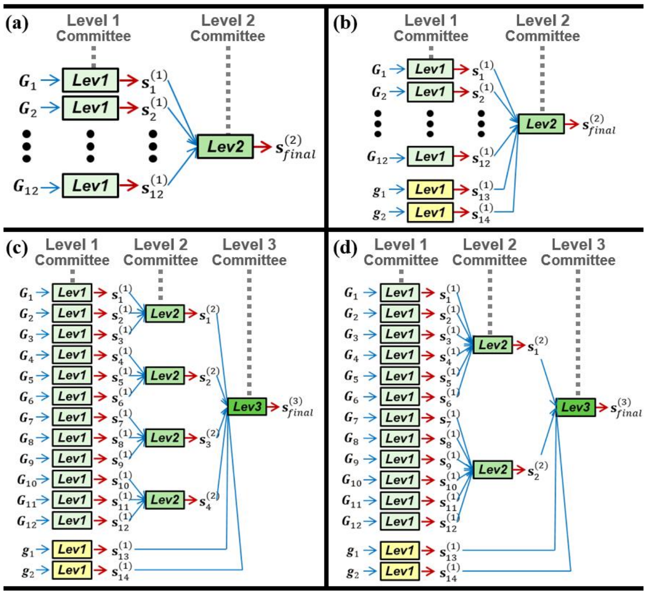

Figure 6. Diagrams of 2- and 3-level hierarchical committees

Table 6. Validation (& testing, if available) accuracy (%) of hierarchical committees with using the VA-Expo-WA rule as the 1st level decision fusion

2-Level Hierarchical Committee	**Decision Fusion in the 2nd Level**	
	Simple Ave. Rule	Majority Voting
(a)	53.4	56.2
(b)	53.9	56.2 (60.2)

3-Level Hierarchical Committee	**Decision Fusion in the 2nd & 3rd Levels**	
	Simple Ave. Rule & Majority Voting	Majority Voting & Majority Voting
(c)	**53.9 (61.6)**	56.2
(d)	52.5	**52.8 (61.6)**

By adding these 2 groups' decisions to the 1st level, we built another 2-level hierarchy as shown in Figure 6(b). For clarity in the subsequent discussion, we shall to name each hierarchy as each index with a parenthesis in Figure 6. The top part of Table 6 denotes classification performances of the 2-level hierarchies. For both (a) and (b), the majority voting in the 2nd level performed better than the simple average rule. However, the hierarchy (b) with a great validation accuracy of 56.2% did not yield a better test accuracy compared to previous submissions. We suggested the following reason; the added 2 groups, ***g***$_1$-***g***$_2$, to the 1st level were expected to produce more diverse decisions based on different training strategies, but their impacts on the 2nd level were quite small due to competing with 12 decisions from ***G***$_1$-***G***$_{12}$.

Hence, we decided to reduce the influence of decisions from ***G***$_1$-***G***$_{12}$ on the final prediction by forming the 3-level hierarchical committee as shown in Figure 6(c) and 6(d). Note that these 3 levels on the side of ***G***$_1$-***G***$_{12}$ not only reduced the number of decisions (from 12 in the hierarchy (b) to 4 in the (c) or to 2 in the (d)) but also made compact and reliable decisions passing through multiple levels. Here, 2 types of decision fusions in the 2nd and 3rd levels were considered, as demonstrated in the bottom part of Table 6. Applying majority voting for both 2nd and 3rd levels showed better validation accuracies than using the simple average rule for the 2nd level with the majority voting for the 3rd level. More importantly, in the test performances, these 2 types of 3-level hierarchies did not differ from each other but were superior to the 2-level hierarchies. It indicates that, as forming a hierarchical committee with higher levels, the structural consideration is more important than the decision fusion method to give enough diversity in decisions.

7. CONCLUSION

In this paper, we present a framework based on committee machines of deep CNNs. To generate diverse errors for a better committee, we first constructed multiple deep CNNs as individual committee members. Here, deep models were trained by applying various network architectures, several strategies to use external data, and different input preprocessing and random initialization. With these individuals, we formed hierarchical committees which adopted the valid-accuracy-based exponentially-weighted average. This exponentially-weighted decision fusion was superior to other commonly-used ensemble methods by increasing a generalization capability. Furthermore, the hierarchical structure indeed made more reliable decisions with the consensus of various sub-groups.

Our proposed approach was demonstrated on the SFEW competition data released for the EmotiW 2015 sub-challenge. To sum up our submissions, the test accuracy of the best single deep CNN was 57.3%, while the single-level committees of 108 models trained using our aligned faces yielded 58.3% with the simple average rule and 60.5% with the exponentially-weighted decision fusion. Furthermore, the last two submissions based on 3-level hierarchical committees of total 240 deep CNNs achieved 61.6%, greatly outperforming the SFEW baseline of 39.1%. We believe that the superiority of our committee machines could further be drawn in other pattern recognition problems as well as SFEW.

In our future works, we will design various and good objective functions in training individual deep CNNs in order to get more diverse decisions. Moreover, how to determine the structure of hierarchical committees will be intensively studied in both academic and engineering manners.

8. ACKNOWLEDGMENT

The authors thank Suh-Yeon Dong for her valuable discussions and comments on this work (suhyeon.dong@gmail.com). This work was supported by the Industrial Strategic Technology Development Program (10044009), funded by the Ministry of Knowledge Economy (MKE, Korea).

9. REFERENCES

[1] Kittler, J., *et al*. 1998. On combining classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(3), 226-239.

[2] Polikar, R. 2006. Ensemble based systems in decision making. *IEEE Circuits Syst. Mag.*, 6(3), 21-45.

[3] Su, Y., *et al*. 2009. Hierarchical ensemble of global and local classifiers for face recognition. *IEEE Trans. Image Process.,* 18(8), 1885-1896.

[4] Pajares, G., *et al*. 2010. A Hopfield Neural Network for combining classifiers applied to textured images. *Neural Networks,* 23(1), 144-153.

[5] Rodríguez-Liñares, L., *et al*. 2003. On combining classifiers for speaker authentication. *Pattern Recognit.,* 36(2), 347-359.

[6] Wu, C. H., & Liang, W. B. 2011. Emotion recognition of affective speech based on multiple classifiers using acoustic-prosodic information and semantic labels. *IEEE Trans. Affective Comput.,* 2(1), 10-21.

[7] Bell, D., *et al*. 2005, On combining classifier mass functions for text categorization. *IEEE Trans. Knowl. Data Eng.,* 17(10), 1307-1319.

[8] Boulesteix, A. L., *et al*. 2008. Microarray-based classification and clinical predictors: on combined classifiers and additional predictive value. *Bioinformatics*, 24(15), 1698-1706.

[9] Shipp, C. A., & Kuncheva, L. I. 2002. Relationships between combination methods and measures of diversity in combining classifiers. *Inform. Fusion*, 3(2), 135-148.

[10] Aksela, M., & Laaksonen, J. 2006. Using diversity of errors for selecting members of a committee classifier. *Pattern Recognit.*, 39(4), 608-623.

[11] Ciresan, D. C., *et al*. 2010. Deep, big, simple neural nets for handwritten digit recognition. *Neural Comput.*, 22(12), 3207-3220.

[12] Cireşan, D. C., *et al*. 2011. Convolutional neural network committees for handwritten character classification. In *ICDAR 2011*, 1135-1139.

[13] Cireşan, D., *et al*. 2012. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32, 333-338.

[14] Ciresan, D., *et al*. 2012. Multi-column deep neural networks for image classification. In *CVPR 2012,* 3642-3649.

[15] Krizhevsky, A., *et al*. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS 2012,* 1097-1105.

[16] Agostinelli, F., *et al*. 2013. Adaptive multi-column deep neural networks with application to robust image denoising. In *NIPS 2013,* 1493-1501.

[17] Sun, Y., *et al*. 2014. Deep learning face representation from predicting 10,000 classes. In *CVPR 2014*, 1891-1898.

[18] Wu, D., & Shao, L. 2014. Deep dynamic neural networks for gesture segmentation and recognition. In *ECCV 2014 Workshops*, 552-571.

[19] Dhall, A., *et al*. 2015. Video and Image based Emotion Recognition Challenges in the Wild: EmotiW 2015, In *ICMI 2015*, in press.

[20] LeCun, Y., *et al*. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.

[21] Jordan, M. I., & Jacobs, R. A. 1994. Hierarchical mixtures of experts and the EM algorithm. *Neural Comput.*, 6(2), 181-214.

[22] Titsias, M. K., & Likas, A. 2002. Mixture of experts classification using a hierarchical mixture model. *Neural Comput.*, 14(9), 2221-2244.

[23] Liu, M., *et al*. 2012. Hierarchical ensemble of multi-level classifiers for diagnosis of alzheimer's disease. In *MLMI 2012*, LNCS 7588, 27–35.

[24] Dhall, A., *et al*. 2012. Collecting large, richly annotated facial-expression databases from movies. *IEEE Multimedia*, 19(3), 34-41.

[25] Viola, P., & Jones, M. J. 2004. Robust real-time face detection. *Int. J. Comput. Vision*. 57(2), 137-154

[26] Zhu, X., & Ramanan, D. 2012. Face detection, pose estimation, and landmark localization in the wild. In *CVPR 2012*, 2879-2886.

[27] Xiong, X., & De la Torre, F. 2013. Supervised descent method and its applications to face alignment. In *CVPR 2013*, 532-539

[28] Kahou, S. E., *et al*. 2013. Combining modality specific deep neural networks for emotion recognition in video. In *ICMI 2013*, 543-550.

[29] Gross, R., & Brajovic, V. 2003. An image preprocessing algorithm for illumination invariant face recognition. In *AVBPA 2003*, 10-18.

[30] http://luks.fe.uni-lj.si/sl/osebje/vitomir/face_tools/INFace/

[31] Tang, Y. 2013. Deep Learning with Linear Support Vector Machines, In *ICML 2013 Workshop on Representational Learning*, Atlanta, USA.

[32] Goodfellow, I. J., *et al*. 2015. Challenges in representation learning: A report on three machine learning contests, *Neural Networks*, 64, 59-63.

[33] https://code.google.com/p/deep-learning-faces/

[34] Srivastava, N., *et al*. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1), 1929-1958.

[35] http://www.kaggle.com/c/challenges-in-representation-learning-facialexpression-recognition-challenge.

[36] Susskind, J. M., *et al*. 2010. The Toronto face database, *Tech. Rep.*, Department of Computer Science, University of Toronto, Toronto, Canada, TR-2010-001.

[37] Pan, S. J., & Yang, Q. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.,* 22(10), 1345-1359.

[38] Vedaldi, A., & Lenc, K. 2014. Matconvnet – convolutional neural networks for matlab. *CoRR.*, abs/1412.4564.