Selective Deep Features for Micro-Expression Recognition

Devangini Patel, Xiaopeng Hong, and Guoying Zhao
Center for Machine Vision and Signal Analysis, University of Oulu, Finland
Emails: devangini27study@gmail.com, {xhong, gyzhao}@ee.oulu.fi

***Abstract*—Micro-expression recognition is a challenging task in computer vision field due to the repressed facial appearance and short duration. Previous work for micro-expression recognition have used hand-crafted features like LBP-TOP, Gabor filter and optical flow. This paper is the first work to explore the possible use of deep learning for micro-expression recognition task. Due to the lack of data for micro-expression, training a CNN model from micro-expression data is not feasible. Instead, transfer learning from objects and facial expressions based CNN models are used. The aim is to use feature selection to remove the irrelevant deep features for our task. This work extends evolutionary algorithms to search an optimal set of deep features so that it does not overfit the training data and generalizes well for the test data. Promising results are presented for various micro-expression datasets.**

I. Introduction

Micro-expressions are brief involuntary facial expressions created when a person is trying to hide their true feelings. It is possible to pose facial expressions to deceive other people, but micro-expressions can not be posed. Thus, micro-expressions can be used for true emotion understanding. There are many potential applications for micro-expressions especially for business negotiation, finding lies during court sessions, evaluating the emotional state and effectiveness of treatment [1]. Even though micro-expressions are visually similar to facial expressions, micro-expressions are short and repressed which makes micro-expression recognition more challenging than facial expression recognition. In a psychological experiment [2] conducted using METT micro-expression training dataset [3], the average micro-expression recognition rate was 50%. So, there is a need for an automatic software system to accurately recognize micro-expressions.

Most work on micro-expressions in the field of Computer Vision has mainly focused on micro-expression recognition [4] [5] [6] [7] [8]. Micro-expression recognition task is defined as recognizing the emotional label of well-segmented video containing micro-expression from start to end. Initial research work on micro-expression recognition has been conducted on posed micro-expressions. Wu et al. [6] extracted features using Gabor filters and used Support Vector Machine (SVM) to recognize them. Polikovsky et al. [5] developed a 3D gradient descriptor and used k-means algorithm to classify onset, apex and offset frames. Later, recognition research focused on spontaneous micro-expression. Pfister et al. [4] used Temporal Interpolation Model (TIM) to increase the frames and Local Binary Pattern-Three Orthogonal Planes (LBP-TOP) features to classify. Huang et. al [8] proposed SpatioTemporal Completed Local Quantization Patterns (STCLQ) feature for recognition. Liu et. al [7] proposed Main Directional Mean Optical Flow feature and used SVM for classification.

Deep Learning is a group of machine learning methods biologically inspired from the structure of the brain. Convolutional Neural Networks (CNN) learn hierarchical features from multiple labelled images. It has been used for various applications including recognition of objects [9], actions [10], places [11] and face verification. Deep Learning requires a lot of data but for micro-expression recognition there is not enough data: there are three spontaneous micro-expression datasets which altogether contain 748 videos. Qi et al. [12] took inspiration from video coding methods to develop deep statistics using transferred spatiotemporal deep CNN features to recognize textures and dynamic scenes. For that work, the ImageNet based CNN network was quite suitable for transferring features. Ng et. al [13] have used cascaded fine tuning of CNN trained on ImageNet dataset with more labelled and low resolution data for recognition of facial expression recognition in the wild. Similarly, the current work has tried to transfer features from CNN models trained on ImageNet and facial expression datasets for micro-expression recognition. This is the first work that uses deep learning for micro-expression recognition. Various experiments are performed by using some available feature selection techniques on these deep feature statistics, which show how feature selection can improve performance. Evolutionary search is adapted to search different combinations of these deep statistics. Different strategies to get the optimal set of features have been explored.

The structure of this paper is as follows: Sec. II discusses the database used, Sec. III explains the motivation behind using feature selection of deep features for this task, Sec. IV explains the recognition algorithm in detail, Sec. V presents the results and Sec. VI discusses the results and concludes the paper.

II. Dataset

There are three publicly available spontaneous micro-expression databases for recognition task: spontaneous micro-expressions dataset (SMIC) [14], the Chinese Academy of Sciences Micro-expression (CASME) [15] and CASME II [16]. These datasets have recorded micro-expression faces in frontal view. The full version of SMIC contains three datasets: (i) a high-speed (HS) dataset recorded by a high-speed camera at 100 fps, (ii) a visible (VIS) dataset recorded by a normal color camera at 25 fps; and (iii) a near infrared (NIR) dataset

recorded by a near infrared camera also at 25 fps. The CASME database contains 195 MEs elicited from 19 participants. The CASME II database [16] is collected to provide more samples with enhanced video quality (higher resolution and frame rate). It includes 247 ME samples from 26 subjects. In this paper, the experiments are carried out using SMIC (VIS and HS) and CASME II databases; their details are given in Tab. I.

Database	samples	subjects	fps	image size	emotions labelled
SMIC-VIS (visible)	71	8	25	640x480px	3 (positive, negative and surprise)
SMIC-HS (high speed)	164	16	100	640x480px	3 (positive, negative and surprise)
CASME II	247	26	200	280x340px	5 (happiness, disgust, surprise, repression, others)

TABLE I
SPONTANEOUS MICRO-EXPRESSION DATASETS WITH THEIR DETAILS

III. MOTIVATION

Recently, deep features transferred from the ImageNet dataset [17] are used for tasks like action recognition [10], scenes and dynamic scenes [12]. Similarly, we aim to transfer features. These deep features are very different from our case and we do not have an enough large dataset to fine-tune the network closer towards our datasets. Not all features would be discriminative for recognizing micro-expressions. Feature selection method is a step in machine learning used to reduce the dimensionality of the data by picking out important features of the dataset while ignoring others. Using feature selection improves the machine learning performance e.g. PCA-HOG features selected by Stepwise Forward Selection (SFS) [18] have better performance than the baseline method for pedestrian detection [19].

Tab. II shows the results of SVM accuracy for various deep features extracted from ImageNet-Vgg-f [20] at layer pool5 by passing SMIC-VIS dataset and Tab. III shows when Linear Discriminant Analysis (LDA) [21] is used on the statistics extracted by [12] on the previous features. Linear SVM provided by LibLinear library [22] for different C values in $\{0.1, 1, 10, 100, 1000, 10000, 100000, 1000000\}$ are used (remaining C values in range 10^{-7} to 10^{7} had no change in performance) and the best one is reported. This shows that applying a basic feature selection like LDA can improve the results. Further, performing LDA on the both training and test data (global LDA) prior to SVM gives higher accuracy. The aim of this work is to develop a feature selection method which can get these global features. Further, it shows that the temporal standard deviation is a good statistic.

There are two main types of feature selection methods: (1) Filter and (2) Wrapper. In the filter method, the features are selected on the basis of the feature scores. The drawbacks are: (1) the feature scoring method might not estimate the desirability of features appropriately, (2) it assesses the features independent of each other but together they might not perform better and (3) there is no feedback which going back to selector about the performance of updated set and how to modify it to improve the results. In wrapper approach, the potential sets of features are evaluated using machine learning classifiers and combinations of the features are searched in exhaustive, forward or hybrid search. In the case of deep features, exhaustive search would not be ideal because of the large number of combinations. In hybrid search, the features are searched in the order of their rank and selected if they improve the classification accuracy. Tab. IV shows the results of certain feature selection methods on unnormalized temporal standard deviation features. Local greedy forward search is a modification of Stable SFS [18] in which the training data is split into pretraining and validation and the minimum validation accuracy is taken instead of the mean validation error. The importance of the wrapper based approach can be found from comparing Tab. IV with rows 1-4 of Tab. III.

Features	Accuracy (%)
Normalized spatial features	28
Normalized temporal features	32
Normalized spatiotemporal features	22
Unnormalized spatial features	22
Unnormalized temporal features	32
Unnormalized spatiotemporal features	21

TABLE II
SVM CLASSIFICATION ACCURACY FOR VARIOUS DEEP STATISTICS OF VIS DATASET TAKEN FROM LAYER POOL5 OF IMAGENET-VGG-F CNN MODEL.

Features	Accuracy (%)
LDA on unnormalized spatiotemporal features	38
LDA on unnormalized temporal features	47
LDA on unnormalized temporal mean	49
LDA on unnormalized temporal standard deviation	47
Global LDA on unnormalized spatiotemporal features	78
Global LDA on unnormalized temporal features	77
Global LDA on unnormalized temporal mean	73
Global LDA on unnormalized temporal variance	77
Global LDA on unnormalized temporal standard deviation	84

TABLE III
SVM CLASSIFICATION ACCURACY WHEN FEATURE SELECTION IS PERFORMED ON FEATURES MENTIONED IN TAB. II.

Method	Accuracy (%)
Ranked forward search with LDA	43
Ranked forward search with Fisher	52
Ranked forward search with Gini	47
Rank forward search using t-Test	50
Stable SFS [18]	40
Local greedy forward search	70

TABLE IV
SVM CLASSIFICATION ACCURACY WHEN DIFFERENT FEATURE SELECTION METHODS ARE PERFORMED ON VIS DATASET UNNORMALIZED TEMPORAL STANDARD DEVIATION FEATURES FROM TAB. II.

IV. RECOGNITION ALGORITHM

The overall architecture is shown in Fig. 1. The architecture is composed of three stages: (1) video processing, (2) feature extraction, (3) evolutionary feature selection.

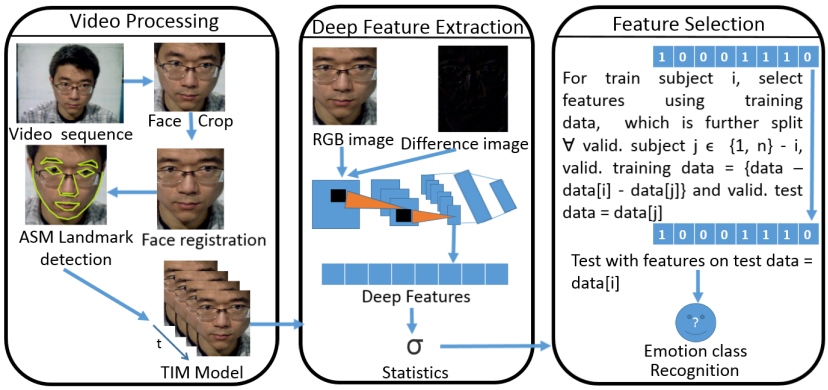

Fig. 1. Overall architecture of proposed method

A. *Video processing*

This step is to make the input video sequence appropriate for input to the CNN network. Firstly, we want the images to contain only the face region so the face can be compared spatially and secondly we want the videos to contain an equal amount of images, so that the videos are comparable in time domain. Faces are detected using Haar face detector [23], cropped and registered to a face model by using the 68 landmarks from Active Shape Model (ASM) [24]. Time Interpolation Model (TIM) [25] is used to upsample or downsample the video to make the lengths of all the videos the same (20 frames).

B. *Feature extraction*

For this, ImageNet-VGG-f CNN [20] pretrained on ImageNet database [17] is used. In addition, a CNN is trained on facial expression datasets is used. Matconvnet library [26] is used for transfer learning and training.

1) Training facial expression datasets: CK+ [27] and SPOS [28] facial expression datasets are used for training. These datasets contain frontal face spontaneous and posed facial expressions. The first 30% frames of CK+ and first and last 20% of SPOS are ignored. TIM [25] is used to adjust the number of frames to 170 frames to subsequently create more data. Especially for posed facial expressions, using TIM can create the intermediate frames. The faces are detected by Haar Cascade [23] and cropped. The mean subject face is subtracted from each subject video and the average frame is subtracted from all frames. Shuffling the frames further improves the training of CNN. The network is loosely based on [29] and is shown in Fig. 2. The network is trained for 100 epochs with the learning rate as 0.005 for first 50 epochs and 0.001 for the remaining. The CNN network is trained with a batch size of 100. The parameters are decided using trial and error. The images corresponding to 2 random chosen subjects (#images = #videos of 2 subjects * 170) is used to validate the training performance. The training and validation error curves through the training process are shown in Fig. 3.

2) Feature Extraction from CNN: Since Imagenet dataset is different from micro-expressions, the features are extracted from the layer directly under all the full connected layers, pool5. Whereas for the facial expressions dataset which is similar to micro-expressions, the features are taken from the layer before the last full connected layer.

3) Spatiotemporal feature computation: Method in [12] is used to extract spatio-temporal deep features. Instead of variance, the standard deviation is used as it has the same power as mean. Referring to Tab. II, the unnormalized temporal standard deviation is a good enough feature.

C. *Evolutionary Feature Selection*

The search methods used in Tab. IV added one feature at a time based on some heuristic value. Whereas evolutionary search can perform parallel search of feature sets. Just like how biological evolution works, these feature sets are crossed-over using roulette wheel and mutated to bring a better population of feature sets. It has been used previously for feature selection [30], [31]. Random migrants are introduced in the population at each generation to create diversity and hall of fame is implemented to maintain the best individuals in the population. In order to implement evolutionary algorithm for such a high dimensional dataset, parallel processing library MPI [32] and OpenMP [33] were used. Different subsets of the population were produced and evaluated on different processors, and their individuals and fitness values are shared at every generation. The main contributions in this algorithm are explained below.

1) Initialization: A population consists of a set of individuals; individuals consist of a binary array where 1 indicates that the feature is selected and 0 if not and C parameter of the linear SVM. C value is selected from $\{0.01, 0.1, 1, 10, 100\}$ (smaller set of C values is selected to reduce the complexity and run time of the algorithm). Some evolutionary methods [30], [31] for feature selection use a random probability of selecting a feature when initializing the chromosomes. This limits the number of features selected at any generation. When the number of features is large, the range explored is small, as shown in lines gene-min and gene-max in Fig. 4. In order to solve this problem, the count of features selected should be random at the initialization step. It is done by (1) generating a random number n in [1, number of features] and (2) set n random features to be selected. Fig. 4 shows the range of

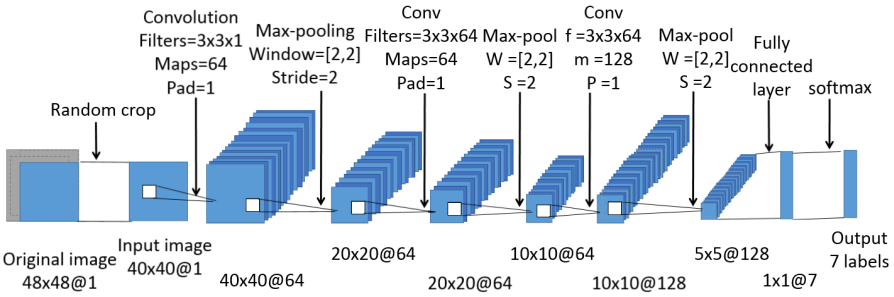

Fig. 2. The convolutional neural network used for facial expressions

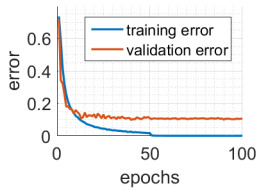

Fig. 3. The training and validation errors as the network is trained

features selected across generations for first 500 features of transferred Imagenet-vgg-f spatiotemporal standard deviation features taken from pool5 using VIS dataset; when the gene or count is selected randomly.

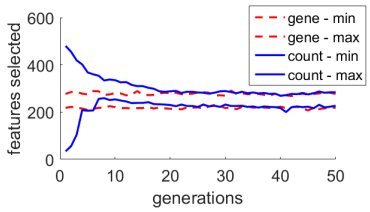

Fig. 4. The range of search space explored for different initialization schemes: (1) random features and (2) random count of features.

2) Fitness Evaluation: In SFS method [34], the training dataset is divided into pretraining and validation sets for each validation subject where validation set contains data corresponding to data for the validation subject and pretraining data contains the remaining data. The test data size is equal to the number of videos for the training subject and training data size is equal to the remaining videos count. The labels are the emotions for each dataset as mentioned in Tab. I. The validation error is calculated for each subject and the mean error is taken. The problem with the mean statistic is that it is biased by extreme values. Tab. V shows the validation error values of different validation sets of along with certain statistics. It can be seen that the means of datasets 3, 4, 5 are quite similar whereas the values are quite different. The idea is to punish the means if the values are quite spread out, i.e. if the values are more spread out then the mean should be decreased more than a mean for a set of values which are less spread out; this ensures that the stability of the performance. Here, statistics like standard deviation or variation can be used to capture this spread of values. One standard deviation below mean statistic already does this for normal distributions but when this value is computed for the dataset values as seen in the 7th column of Tab. V, it is not a fair statistic. When comparing datasets 2 and 4, $\mu_4 = \mu_2$ but since $\sigma_4 > \sigma_2$, dataset 4 becomes less favourable. The aim is to find the weight of standard deviation in the new heuristic. Instead, the validation errors can be converted to decimal from [0, 100] range to [0, 1] range and then statistic $\mu - \sigma$ with $\mu - \sigma^2$ can be computed as shown in Tab. VI. Comparing column 9 for datasets 2 and 4 and for others too, it can be seen that the $\mu - \sigma^2$ statistic is a better heuristic than $\mu - \sigma$. The statistic $\mu - \sigma^2$ varying with two different validation set accuracies is shown in Fig. 5.

Dataset no.	validation errors			μ	σ	$\mu - \sigma$
1	10	20	30	20	8.2	11.8
2	20	30	80	43.3	26.2	17.1
3	**40**	**50**	**60**	**50**	8.2	**41.8**
4	**20**	**30**	**100**	**50**	35.6	**14.4**
5	**50**	**50**	**50**	**50**	0	**50**
6	70	80	90	80	8.2	71.8

TABLE V
DATASET AND THEIR VALIDATION ERRORS IN THE RANGE [0, 100]

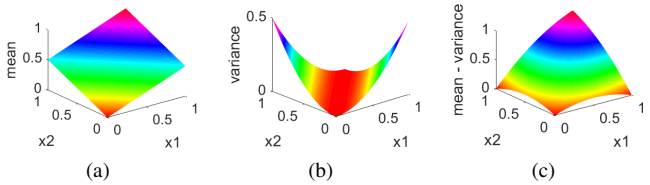

Fig. 5. Different statistics plotted against two values of accuracy: (a) mean, (b) variance, (c) proposed heuristic. Here x1 and x2 represent two accuracies and these figures show how these statistics vary with x1 and x2.

3) Evolution Termination: Generally, evolutionary methods set a tolerable error or maximum limit of generations as the termination condition of the evolutionary algorithm. In this case, the optimal accuracy is not known before hand, and the performance might degrade after some generations. In order to address this issue, the relationship between three heuristics: (1) best Hall of Fame (HOF) fitness function, (2) training accuracy and (3) test accuracy are observed as shown in Fig. 6. If both

No.	values			μ	σ	σ^2	$\mu-\sigma$	$\mu-\sigma^2$
1	0.1	0.2	0.3	0.2	0.08	0.007	0.11	0.19
2	0.2	0.3	0.8	0.43	0.26	0.069	0.17	0.36
3	**0.3**	**0.4**	**0.5**	0.4	0.08	0.007	0.41	**0.49**
4	**0.2**	**0.3**	**1**	0.5	0.35	0.13	0.14	**0.37**
5	**0.5**	**0.5**	**0.5**	0.5	0	0	0.5	**0.5**
6	0.7	0.8	0.9	0.8	0.08	0.007	0.71	0.79

TABLE VI
PERCENTILE VALUES OF THE DATASET AND THEIR STATISTICS

differences between maximum and minimum absolute training and HOF and training accuracies is less than 0.01; then the first feature generation set is selected as shown in Fig. 6a. Else, the accuracies (HOF/train) with the maximum difference between maximum and minimum values is selected as potential values. From these potential feature sets, if the maximum derivative of the dominating accuracy is greater than 0.005 then that particular generation is used as shown in Fig. 6b else the second best is used as shown in Fig. 6c. These decisions were selected from analyzing the graphs.

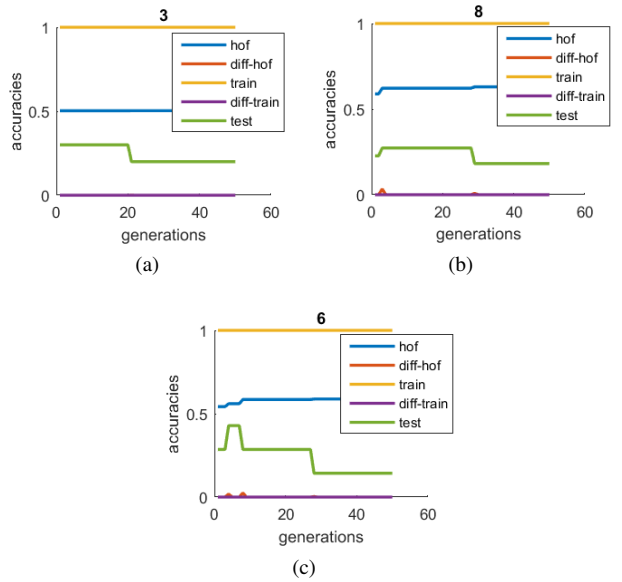

Fig. 6. The relationship between training, test accuracies and best fitness of hall of fame across generations along with the derivative of HOF and training accuracies for 3 different cases.

V. RESULTS

A. Time

The proposed approach is not real time. The features were computed on a university server with Nvidia K40 graphics processor and took upto 1 hour to compute. The feature selection code was more time and resource consuming. It was run on CSC supercomputer cluster called sisu. It took approximately 2 hours to execute for SMIC features, 12 hours for HS and 2 days for CASME II.

B. Experimental analysis

The proposed technique with different fitness function and databases are evaluated as given in Tab. VII. The test is done by using Leave One Subject Out (LOSO) protocol in which if one subject is tested then the remaining subject's data is used for feature selection. The evolutionary algorithm has 50 individuals with ten randomly generated individuals and is run for 50 generations. The performance will increase if population size and generation count parameters are increased but on the other hand it will take more time. The crossover rate should be high enough so that the desirable features are carried forward to future generations; it is kept as 0.8. And the mutation rate should be low so that the desirable features are not easily removed, it is set as 0.05.

1) Fitness function: The two fitness functions μ and $\mu-\sigma^2$ for different datasets are compared in Tab. VII. For most cases, the fitness function $\mu-\sigma^2$ is a better choice.

	Database		
Fitness Function	VIS	HS	Casme II (4-fold)
μ	56.3	49.3	46.1
$\mu-\sigma^2$	56.3	53.6	47.30

TABLE VII
RECOGNITION RATES FOR DIFFERENT DATABASES

2) Different CNN models: The two CNN models based on objects i.e. pretrained Imagenet-vgg-f and the trained facial expressions CNN are compared for different datasets using the fitness function $\mu-\sigma^2$ in Tab. VIII. Features extracted from facial expressions dataset performs better as it is more closer to micro-expressions as compared to Imagenet with ME.

	CNN	
Database	Object	FE
VIS	29.5	56.3
HS	-	53.6

TABLE VIII
RECOGNITION RATES FOR DIFFERENT CNN MODELS

C. Comparison with other methods

This approach has been compared with the state of art method [8] and baseline approach [14] in Tab. IX. It performs better than the baseline approach but is less than state of art method.

	Database		
Approach	VIS	HS	Casme II
Our approach	56.3	53.6	47.3
LBP-TOP (baseline) [14]	52	48	38
STCLQ [8]	-	64	59

TABLE IX
COMPARISON WITH DIFFERENT METHODS

VI. DISCUSSION AND CONCLUSION

This paper proposes a feature selection framework of deep learning for micro-expression recognition task. We showed that the proposed method has a suitable use for subject wise diverse cases like micro-expression recognition.

The reason why it could not beat the state-of-art method was that the fitness function evaluation heuristic was such that the training accuracy would overfit the data as seen in Fig. 6. Mechanisms to avoid such things would need to be explored.

In addition, the parameters of this method e.g. hyperparameters of CNN, thresholds in generation termination, etc. could be calculated based on the data to improve performance. Further, approaches to make the architecture end-to-end and real time need to be explored so that it can be used in real time applications.

A larger population and more generations could be used in the evolutionary feature selection step which would effectively increase the performance. A broader range of C values for the SVM should also be investigated. The facial expressions based CNN can be fine-tuned with the micro-expressions dataset along with data augmentation to improve the performance. Rather than just using the appearance and motion at each frame, the whole time sequence dynamics can be used to train a Long Short Term Memory Network (LSTM) [35].

ACKNOWLEDGMENT

The authors would like to thank CSC Finland for providing computing resources. This work was sponsored by the Academy of Finland, Infotech Oulu and Tekes Fidipro program.

REFERENCES

[1] P. Ekman, *Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage (Revised Edition)*. WW Norton & Company, 2009.

[2] J. Endres and A. Laidlaw, "Micro-expression recognition training in medical students: a pilot study," *BMC medical education*, vol. 9, no. 1, p. 47, 2009.

[3] P. Ekman, "Mett. micro expression training tool," *CD-ROM. Oakland*, 2003.

[4] T. Pfister, X. Li, G. Zhao, and M. Pietikäinen, "Recognising spontaneous facial micro-expressions," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1449–1456.

[5] S. Polikovsky, Y. Kameda, and Y. Ohta, "Facial micro-expressions recognition using high speed camera and 3d-gradient descriptor," in *Crime Detection and Prevention (ICDP 2009), 3rd International Conference on*. IET, 2009, pp. 1–6.

[6] Q. Wu, X. Shen, and X. Fu, "The machine knows what you are hiding: an automatic micro-expression recognition system," in *Affective Computing and Intelligent Interaction*. Springer, 2011, pp. 152–162.

[7] Y. J. Liu, J. K. Zhang, W. J. Yan, S. J. Wang, G. Zhao, and X. Fu, "A main directional mean optical flow feature for spontaneous micro-expression recognition," *IEEE Transactions on Affective Computing*, vol. PP, no. 99, pp. 1–1, 2015.

[8] G. Z. X. Huang, X. Hong, and W. Z. . M. Pietikäinen, "Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns." *Neurocomputing*, vol. 175, pp. 564–578, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215015726

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[10] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[11] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *Advances in neural information processing systems*, 2014, pp. 487–495.

[12] X. Qi, C.-G. Li, G. Zhao, X. Hong, and M. Pietikäinen, "Dynamic texture and scene classification by transferring deep image features," *arXiv preprint arXiv:1502.00303*, 2015.

[13] H.-W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, "Deep learning for emotion recognition on small datasets using transfer learning," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ser. ICMI '15. New York, NY, USA: ACM, 2015, pp. 443–449.

[14] X. Li, T. Pfister, X. Huang, G. Zhao, and M. Pietikäinen, "A spontaneous micro-expression database: Inducement, collection and baseline," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 2013, pp. 1–6.

[15] W.-J. Yan, Q. Wu, Y.-J. Liu, S.-J. Wang, and X. Fu, "Casme database: a dataset of spontaneous micro-expressions collected from neutralized faces," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 2013, pp. 1–7.

[16] W.-J. Yan, X. Li, S.-J. Wang, G. Zhao, Y.-J. Liu, Y.-H. Chen, and X. Fu, "Casme ii: An improved spontaneous micro-expression database and the baseline evaluation," *PloS one*, vol. 9, no. 1, p. e86041, 2014.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[18] L. I. Kuncheva, "A stability index for feature selection." in *Artificial intelligence and applications*, 2007, pp. 421–427.

[19] T. Kobayashi, A. Hidaka, and T. Kurita, "Selection of histograms of oriented gradients features for pedestrian detection," in *Neural Information Processing*. Springer, 2008, pp. 598–607.

[20] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[21] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[22] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[23] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. I–511.

[24] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer vision and image understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[25] G. Z. . M. P. Z. Zhou, "Towards a practical lipreading system." in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, 2011, pp. 137–144.

[26] A. Vedaldi and K. Lenc, "Matconvnet – convolutional neural networks for matlab," in *Proceeding of the ACM Int. Conf. on Multimedia*, 2015.

[27] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*. IEEE, 2010, pp. 94–101.

[28] T. Pfister, X. Li, G. Zhao, and M. Pietikäinen, "Differentiating spontaneous from posed facial expressions within a generic facial expression recognition framework," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 868–875.

[29] S. E. Kahou, C. Pal, X. Bouthillier, P. Froumenty, Ç. Gülçehre, R. Memisevic, P. Vincent, A. Courville, Y. Bengio, R. C. Ferrari *et al.*, "Combining modality specific deep neural networks for emotion recognition in video," in *Proceedings of the 15th ACM on International conference on multimodal interaction*. ACM, 2013, pp. 543–550.

[30] J. Van Hulse, T. M. Khoshgoftaar, A. Napolitano, and R. Wald, "Feature selection with high-dimensional imbalanced data," in *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*. IEEE, 2009, pp. 507–514.

[31] K. C. Tan, E. J. Teoh, Q. Yu, and K. Goh, "A hybrid evolutionary algorithm for attribute selection in data mining," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8616–8630, 2009.

[32] M. P. Forum, "Mpi: A message-passing interface standard," University of Tennessee, Knoxville, TN, USA, Tech. Rep., 1994.

[33] OpenMP Architecture Review Board, "OpenMP application program interface version 3.0," May 2008. [Online]. Available: http://www.openmp.org/mp-documents/spec30.pdf

[34] A. W. Whitney, "A direct method of nonparametric measurement selection," *Computers, IEEE Transactions on*, vol. 100, no. 9, pp. 1100–1103, 1971.

[35] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, no. 10, pp. 2451–2471, 2000.