Ghasemi Ghaleh Bahmani, Afsaneh, Baktashmotlagh, Mahsa, Denman, Simon, Sridharan, Sridha, Nguyen Tien, Dung, & Fookes, Clinton
(2017)
Deep discovery of facial motions using a shallow embedding layer.
In Luo, J, Zeng, W, & Zhang, Y J (Eds.) *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP).*
Institute of Electrical and Electronics Engineers Inc., United States of America, pp. 1567-1571.

https://doi.org/10.1109/ICIP.2017.8296545

DEEP DISCOVERY OF FACIAL MOTIONS USING A SHALLOW EMBEDDING LAYER

Afsaneh Ghasemi *Mahsa Baktashmotlagh* *Simon Denman* *Sridha Sridharan*
Dung Nguyen Tien *Clinton Fookes*

Speech, Audio, Image and Video Technology (SAIVT) Laboratory
Queensland University of Technology, Australia
{afsaneh.ghasemighalehbahmani, m.baktashmotlagh, s.denman,s.sridharan,
d.nguyentien, c.fookes}@qut.edu.au

ABSTRACT

Unique encoding of the dynamics of facial actions has potential to provide a spontaneous facial expression recognition system. The most promising existing approaches rely on deep learning of facial actions. However, current approaches are often computationally intensive and require a great deal of memory/processing time, and typically the temporal aspect of facial actions are often ignored, despite the potential wealth of information available from the spatial dynamic movements and their temporal evolution over time from neutral state to apex state. To tackle aforementioned challenges, we propose a deep learning framework by using the 3D convolutional filters to extract spatio-temporal features, followed by the LSTM network which is able to integrate the dynamic evolution of short-duration of spatio-temporal features as an emotion progresses from the neutral state to the apex state. In order to reduce the redundancy of parameters and accelerate the learning of the recurrent neural network, we propose a shallow embedding layer to reduce the number of parameters in the LSTM by up to 98% without sacrificing recognition accuracy. As the fully connected layer approximately contains 95% of the parameters in the network, we decrease the number of parameters in this layer before passing features to the LSTM network, which significantly improves training speed and enables the possibility of deploying a state of the art deep network on real-time applications. We evaluate our proposed framework on the DISFA and UNBC-McMaster Shoulder pain datasets.

1. INTRODUCTION

Facial expression, as a unique universal language, is the most natural means for humans to express their internal emotional states in any face-to-face communication. The wealth of information present in facial motions alongside the recent availability of massive amounts of data has increased research interest in developing new approaches to model and facilitate human-computer interaction. Recent approaches for automatic facial expression recognition have been disrupted by the use of Deep Neural Networks (DNN) [2, 9, 12, 11]. These networks have demonstrated impressive performance on a wide variety of audio and vision data; however training a complex model with a huge amount of data results in large computational costs due to the embedding of a chain of convolutional [29] and fully connected layers to overcome a highly non-convex optimization, often requiring the utilization of highly optimized CPU or GPU architectures. As such, the recognition task is typically constrained by the trade-off between an over parameterized network and the performance accuracy, which makes for a very slow training due to a huge number of redundant parameters compared to other machine learning approaches such as GMMs [28] and HMMs [1].

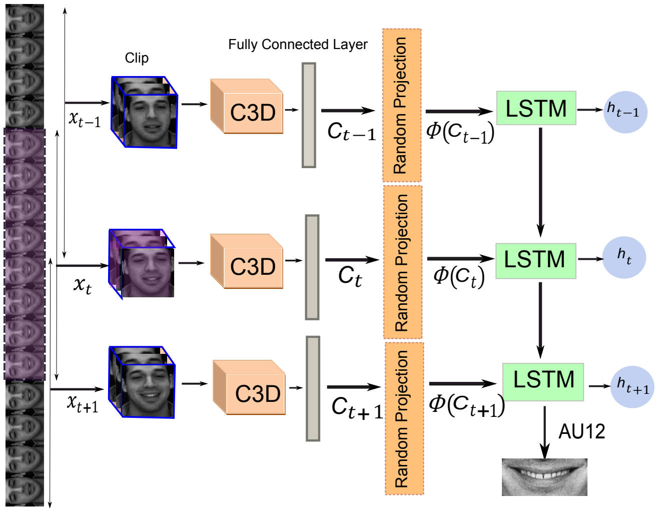

Fig. 1. We propose a framework for facial action detection using 3D DCNNs and LSTMs. Spatio-temporal features are extracted from overlapping windows to integrate the dynamic temporal aspects of the actions. We then apply a LSTM network to discover the transitions of actions from a neutral state to an apex state. We also propose a shallow embedding layer before passing the last fully connected layer output to the LSTM, resulting in much more efficient and faster learning by the LSTM.

The increase use of deep learning approaches for real time recognition tasks makes speed increases and efficiency gains in the training, and particularly the evaluation stages of large scale networks, without compromising performance, highly desirable. Furthermore, a real time facial action detector requires a highly sophisticated model to efficiently capture the non-linearity of salient motions based on the Facial Action Coding System (FACS) [10], which has been developed to describe facial activities in terms of visually observable facial muscle actions. The necessity of a complex model for such a task heightens the need for efficiency gains within the model.

Our main contributions are as follows: we present a framework which uses 3D convolutional deep networks to encapsulate appearance and motion information simultaneously. We also detect facial actions by incorporating salient short temporal segments into the memory units of the LSTM, which allows the network to learn when to forget and when to update hidden states. In addition, we utilize kernel tricks with a random projection to reduce the number of parameters of the LSTM network, enabling the LSTM network to learn the discriminative features of each class of action with less computational complexity for training, and obtain similar or better testing performance.

2. RELATED WORKS

Automatic recognition of action units is still challenging due to pose, illumination, shape, and texture variations. In addition, there is no quantitative definition of how action units can appear in various complex combinations to form facial actions [31]. Action units are typically detected by extracting hand-crafted features including Gabor wavelet coefficients [35, 37, 32], Local Binary Patterns (LBP) [17, 38, 27], Histograms of Oriented Gradients (HOG) [30, 21], scale-invariant feature transform (SIFT) descriptors [5] and active shape and appearance features [23, 3]; followed by training a model using a SVM classifier. However, it is problematic since no matter how many iterations are used to train a model, the extracted hand-crafted features are always fixed. Very recently, deep learning approaches have obtained promising results for the recognition of action units. A deep 2D convolutional framework proposed in [11] shows the capability of deep learning in providing an abstract representation of spatial information from low-level features to high level feature for facial action unit detection. In addition, [26] proposed a deep region network, which allows learning the activated regions of the face independently during the training of the network. However, the dynamic temporal aspects of facial actions have been ignored in recent works, despite them providing a rich source of information as an action evolves over time [33].

In addition, several recent works [13, 8] demonstrate the significant redundancy of parameters in deep learning models. In [16], the number of parameters in the final weight layer of the network reduced by up to 40% using a low-rank factorization approach, which results in a drop off in training time, without a significant loss in final recognition accuracy, in comparison with a full-rank representation. In another study [13], only 10% of the network parameters are learned, while the rest of the weight values are predicted at each iteration based on a few learned parameters. [6, 4] proposed a hashing approach to speed up the learning of the parameters of the fully-connected layers of a Neural Network, which significantly reduces the computational complexity of both forward and backward propagation in the network. [7] also proposed an approach that scales up the kernel tricks to be applied on extremely large data sets using Doubly Stochastic Gradients. The Network in Network architecture of [19] achieves state of the art results on several deep learning benchmarks by replacing the fully connected layers with global average pooling over feature maps, which prevents the over-fitting problem of traditional fully connected layers. In deep fried convolutional network [36] a 2D convolutional neural networks and Adaptive Fastfood transform [18] are utilized to reduce the number of parameters in the fully connected layers. All of aforementioned approaches demonstrate that DCNNs contain significant redundancy, and can be simplified without compromising accuracy.

3. METHODOLOGY

We use a convolutional neural network classifier for facial action unit recognition. We first present the data preparation in Section 3.1, and we then describe the proposed network architecture and methodology in Section 3.2.

3.1. Pre-Processing

Prior to processing facial components, images need to be normalized against variations including translation, scale, and rotation. First, we rotate all face images to horizontal using eye corners identified from manually labelled landmarks corresponding facial components such as the eyes, nose, and lips. Then, all images are cropped by creating a bounding box around the set of predefined landmarks. This region is then expanded by 15 pixels to improve the subsequent descriptor extraction, and we then resize all final facial images into 128×96 pixels. Following the normalization, we use the approach of [14] to centre the data by subtracting the average image over the whole training set from each image.

3.2. Network Architecture

An input video is divided into overlapping windows of length 16 frames, with the windows overlapping by 5 frames. Then the network takes inputs in the form of a volume of frames (i.e. a short video clip) and predicts facial action units. Our proposed network consists of three 3D convolution layers, each of which is followed by a max-pooling layer. The input dimensions to the first layer is $16 \times 128 \times 96$. The numbers of filters for three convolution layers are $64, 128, 256$

respectively with a filter size of 5×5. Then the output of the last 3D convolutional layer is passed to a fully-connected layer with 1024 neurons. After training the 3D convolutional framework, the feature map is fed into our proposed shallow embedding layer, and then as input to the LSTM with 32 hidden states, to learn discriminative spatio-temporal features for estimating the class of AUs.

3.2.1. Convolutional 3D (C3D)

Architectures with volumetric convolutions have been successfully used in video analysis [34]. Compared to 2D convolutional networks, a 3D convolutional network has the ability to model the temporal aspect of actions. In a 3D Convolutional network, convolution and pooling operations are performed spatio-temporally while only spatial information is extracted in a 2D convolutional network.

3.2.2. LSTM

Long Short Term Memory networks (LSTMs) [15] is one of the best sequence learners for time-series data and shows promising results for a large variety of problems such as speech recognition, multiple sequence behavior recognition and action recognition from videos [**?**]. LSTMs are a type of RNN, which are able to learn long-term dependencies of temporal information and avoids the problem that events lying far back in time tend to be forgotten. LSTMs contain a set of memory blocks, which each contain one or more memory cells. The salient temporal information can be remembered over arbitrarily long periods of time through the memory cells of the LSTM structure. The LSTM architecture consists of three layers: an input layer, a hidden layer, and an output layer. The number of input layer units corresponds to the dimension of the feature vector. In our experiments, we use LSTMs of 32 memory blocks and an output layer of one unit, corresponding to the class of facial action.

3.2.3. Random Features Approximations

The random projection is a fast, simple approach that can be applied to reduce the dimensionality of features in large scale datasets. As shown in [25], the random projection converges to the Gaussian RBF kernel, with a cost of $O(nd)$ for operations and $O(nd)$ for storage, where n is the number of samples and d is the number of dimensions in random space. As such, the random projection prevents an increase in the cost of computing a non-liner decision function as the dataset grows.

More specifically, the Kernel methods [22] guarantee that kernels can be expressed as an inner product in the Hilbert space:

Theorem 1 *Any kernel* $k : \boldsymbol{x} \times \boldsymbol{x} \rightarrow R$ *satisfying* $\int k(\boldsymbol{x}, \boldsymbol{x}^T) f(\boldsymbol{x}) f(\boldsymbol{x}^T) d\boldsymbol{x} d\boldsymbol{x}^T \geq 0$ *for all* $L_2(\boldsymbol{x})$ *measurable functions* f *can be expanded into,*

$$k(\boldsymbol{x}, \boldsymbol{x}^T) = \sum_j \lambda_j \Phi_j(\boldsymbol{x}) \Phi_j(\boldsymbol{x}^T) \tag{1}$$

where $\lambda_j > 0$ *and the* Φ_j *are orthonormal on* $L_2(\boldsymbol{x})$*.*

According to [25], the continues shift-invariant kernel expressed in Eq. 1 can be approximated by randomly sampling the λ_j from a data-independent distribution $p(\lambda)$ and generating a lower dimensional random features. Therefore, Eq. (1) can be expressed as

$$k(\boldsymbol{x}, \boldsymbol{x}^T) \approx \frac{\sum_j \lambda}{n} \sum_1^n \Phi_{\lambda_i}(\boldsymbol{x}) \Phi_{\lambda_i}(\boldsymbol{x}^T) \tag{2}$$

where

$$\lambda_j \sim p(\lambda) \ where \ p(\lambda i) \propto \lambda_i. \tag{3}$$

and

$$\Phi_{\lambda_j}(\boldsymbol{x}) = \frac{1}{\sqrt{n}} exp([\boldsymbol{Z}_x]_j) \tag{4}$$

To generate an approximation of a Gaussian RBF kernel, $k(\boldsymbol{x}, \boldsymbol{x}^T) = exp(-||\boldsymbol{x} - \boldsymbol{x}^T||^2 / 2\sigma^2)$, each sample, $\boldsymbol{z}_i \in \mathbb{R}^d$, is generated from a normal distribution $N(0, \sigma^2)$ and each feature ϕ_j is a $\sin$ or $\cos$ wave varying along the direction given by one row of Z, with varying periods. We can approximate the above expectation, and hence approximate the kernel $k(x, x')$ with an inner product of stacked $\sin$ and $\cos$ features. Therefore, we select n samples from $p(w)$ to generate the weight vector, $W = (w_1, w_2, ...w_n)^T$. Then, the inner-product of the following random features can approximate the kernel,

$$\phi_{rbf} = \frac{\sqrt{a}}{n} (\cos(\boldsymbol{W}\boldsymbol{x}) \sin(\boldsymbol{W}\boldsymbol{x}))^T \tag{5}$$

In practice, ϕ_{rbf} is the output of our proposed shallow embedding layer, consisting of a linear layer, $\boldsymbol{W}\boldsymbol{x}$, and non-liner operations (sine and cosine), that correspond to a particular implicit kernel function. In fact, we can implicitly approximate a squared exponential kernel by drawing the rows of W from a Gaussian distribution and use Eq. (5) to implement the proposed shallow embedding layer.

The random dimension for our approach was set to 14 and 8 for DISFA and UNBC-McMaster pain datasets respectively as we could not observe any improvements by decreasing the embedding dimensions further. Moreover, as noted in [24], the randomised embedding dimension should not be too much smaller than the original dimension D to prevent a point in the set from being mapped to the origin.

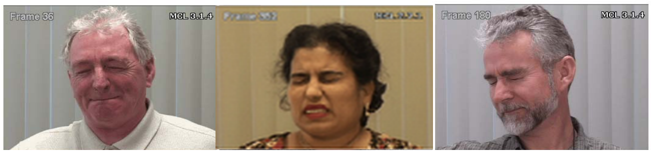

Fig. 2. Examples images UNBC-McMaster. Each frame has a pain intensity level equal to 5 (high pain).

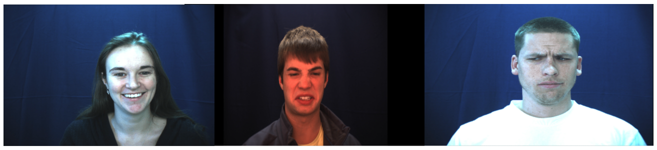

Fig. 3. Samples images DISFA. The intensities of 12 AUs are encoded framewise in a 0 to 5 scale.

4. EXPERIMENTAL SETUP

4.1. Datasets

UNBC-McMaster Shoulder Pain: The UNBC-McMaster Shoulder Pain dataset [20] contains videos of 129 patients suffering from shoulder pain in one shoulder in two test conditions. In the first, patient's move their shoulder by themselves with the camera approximately frontal to start. In the second, the shoulder is moved by a physiotherapist with the camera approximately 70 degrees to start. In this paper, we used 200 sequences from 25 subjects, all from the first condition where moderate head motion was common. AUs related to pain are annotated for every frame by trained FACS coders. Example images from the database are shown in Figure 2. **DISFA: Denver Intensity of Spontaneous Facial Actions:** The Denver Intensity of Spontaneous Facial Actions (DISFA) database [21] (see Figure 3) includes spontaneous AUs for 27 adult subjects in which each subject has been recorded by a stereo camera while they viewed video clips intended to elicit spontaneous emotional expression. FACS codes and 66 points landmarks corresponding to the key-points on the face are also provided for every frame.

		Accuracy% on Disfa Dataset				
AU	N	C_{1024}	$\phi(C_{32})$	$\phi(C_{16})$	$\phi(C_{14})$	$\phi(C_{10})$
1	1715	**0.93**	0.84	0.90	0.91	0.58
2	1436	**0.94**	0.89	0.92	0.91	0.54
4	4782	**0.81**	0.73	0.76	0.78	0.43
5	393	**0.98**	0.94	**0.98**	**0.98**	0.59
6	3729	**0.85**	0.71	0.77	0.80	0.50
9	1410	**0.94**	0.90	0.94	0.88	0.56
12	5902	0.55	0.59	0.63	**0.73**	0.53
15	1557	**0.94**	0.87	0.91	0.90	0.60
17	2557	**0.90**	0.83	0.88	**0.90**	0.57
20	895	**0.96**	0.93	**0.96**	0.82	0.52
25	8995	0.53	0.52	0.53	**0.62**	0.45
26	4799	**0.81**	0.64	0.73	0.74	0.58
Avg		0.74	0.68	0.72	**0.76**	0.54

Table 1. Average AU detection accuracy of proposed shallow embedding layer with Different Random Projection Sizes on DISFA dataset.

4.2. Experimental Results

We explore how well our features can approximate the exact kernel computation. We decrease the parameters of LSTM network by 98% using a Random Kitchen Sink projection from 1024 dimensions into 32, 16, 14, and 10 dimensions for DISFA; and into 64, 32, 16, 8, and 7 dimensions for UNBC-McMaster. We find that by decreasing the dimensions to less than 14 for DISFA and 8 for UNBC-McMaster, performance begins to degrade (see Table 1). However, we find that we can achieve the best result of around 76% and 94% for AU detection on DISFA and UNBC-McMaster respectively by projecting the feature map to 14 and 8 dimensions for DISFA and UNBC-McMaster respectively. The average results are shown in Table 1 and Table 2 over the number of videos available per action.

		Accuracy% on UNBC Dataset					
AU	N	C_{1024}	$\phi(C_{64})$	$\phi(C_{32})$	$\phi(C_{16})$	$\phi(C_{8})$	$\phi(C_{7})$
4	1074	**0.95**	0.84	0.90	0.91	0.94	0.94
6	5557	**0.95**	0.83	0.90	0.92	0.94	0.94
7	3366	**0.97**	0.85	0.90	0.94	0.96	0.96
9	423	**0.94**	0.83	0.89	0.91	0.93	0.93
10	525	**0.92**	0.82	0.88	0.91	**0.92**	**0.92**
12	6956	**0.93**	0.82	0.88	0.90	0.92	0.92
20	706	**0.93**	0.81	0.87	0.90	0.92	0.92
25	2433	**0.93**	0.82	0.86	0.91	**0.93**	**0.93**
26	2199	**0.96**	0.84	0.88	0.93	0.95	0.95
43	2343	0.89	0.80	0.84	0.88	**0.90**	**0.90**
Avg		**0.94**	0.83	0.88	0.91	**0.94**	0.93

Table 2. Average AU detection accuracy of proposed shallow embedding layer with Different Random Projection Sizes on UNBC-McMaster dataset.

We also explore the processing time of the LSTM network in the test/train phase when we apply the random feature approximation. We train the model with 24,150 samples for DISFA, and 9,190 samples for UNBC McMaster. The average processing time of a batch of 32 images is shown in Table 3, and indicates the ability of random projection to decrease the computational complexity of the LSTM network, enabling it to perform on embedded devices.

	Processing Time of LSTM			
	C_{1024}	$\phi(C_{32})$	$\phi(C_{16})$	$\phi(C_{14})$
Train (sec)	262.26	36.47	35.61	**34.26**
Test (sec)	0.37	0.03	0.03	**0.02**

Table 3. The processing time of LSTM in training and test phase with Different Random Projection Sizes.

5. CONCLUSIONS

In this paper, we propose the use of a shadow embedding layer in a deep learning framework which is able to provide spatio-temporal deep features to a Long Short Temporal Memory (LSTM) network, resulting a super-fast temporal action unit classification approach with a high level of accuracy. The proposed shallow embedding layer achieves a substantial reduction in the number of parameters without sacrificing predictive performance on the DISFA dataset and the UNBC McMaster pain archive.

6. REFERENCES

[1] P. S. Aleksic and A. K. Katsaggelos. Automatic facial expression recognition using facial animation parameters and multistream hmms. *IEEE Transactions on Information Forensics and Security*, 1(1):3–11, March 2006.

[2] N. Anand and P. Verma. Convoluted feelings convolutional and recurrent nets for detecting emotion from audio data.

[3] A. B. Ashraf, S. Lucey, J. F. Cohn, T. Chen, Z. Ambadar, K. M. Prkachin, and P. E. Solomon. The painful face–pain expression recognition using active appearance models. *Image and vision computing*, 27(12):1788–1796, 2009.

[4] A. H. Bakhtiary, A. Lapedriza, and D. Masip. Speeding up neural networks for large scale classification using wta hashing. *arXiv preprint arXiv:1504.07488*, 2015.

[5] S. Berretti, A. d. Bimbo, P. Pala, B. B. Amor, and M. Daoudi. A set of selected sift features for 3d facial expression recognition. In *ICPR*, pages 4125–4128. IEEE, 2010.

[6] W. Chen, J. T. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. Compressing neural networks with the hashing trick. *CoRR, abs/1504.04788*, 2015.

[7] B. Dai, B. Xie, N. He, Y. Liang, A. Raj, M.-F. F. Balcan, and L. Song. Scalable kernel methods via doubly stochastic gradients. In *Advances in Neural Information Processing Systems*, pages 3041–3049, 2014.

[8] E. L. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, pages 1269–1277, 2014.

[9] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal. Recurrent neural networks for emotion recognition in video. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 467–474. ACM, 2015.

[10] P. Ekman and W. Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Consulting Psychologists Press, 1978.

[11] A. Ghasemi, S. Denman, S. Sridharan, and C. Fookes. Discovery of facial motions using deep machine perception. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–7, March 2016.

[12] A. Gudi, H. E. Tasli, T. M. den Uyl, and A. Maroulis. Deep learning based facs action unit occurrence and intensity estimation. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 06, pages 1–5, May 2015.

[13] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[14] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[15] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[16] M. Jaderberg, A. Vedaldi, and A. Zisserman. Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866*, 2014.

[17] B. Jiang, M. F. Valstar, and M. Pantic. Action unit detection using sparse appearance descriptors in space-time video volumes. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 314–321. IEEE, 2011.

[18] Q. Le, T. Sarlós, and A. Smola. Fastfood-approximating kernel expansions in loglinear time. In *Proceedings of the international conference on machine learning*, 2013.

[19] M. Lin, Q. Chen, and S. Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.

[20] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews. Painful data: The unbc-mcmaster shoulder pain expression archive database. *Image, Vision, and Computing Journal*, pages 197–205, 2012.

[21] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn. Disfa: A spontaneous facial action intensity database. *Affective Computing, IEEE Transactions on*, 4(2):151–160, 2013.

[22] J. Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 209:415–446, 1909.

[23] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *ECCV*, pages 504–513. Springer, 2008.

[24] S. Oymak and J. A. Tropp. Universality laws for randomized dimension reduction, with applications. *arXiv preprint arXiv:1511.09433*, 2015.

[25] A. Rahimi and B. Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in neural information processing systems*, pages 1313–1320, 2009.

[26] T. Senechal, D. McDuff, and R. Kaliouby. Facial action unit detection using active learning and an efficient non-linear kernel approximation. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 10–18, 2015.

[27] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009.

[28] U. Tariq, J. Yang, and T. S. Huang. Maximum margin gmm learning for facial expression recognition. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–6. IEEE, 2013.

[29] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In *ECCV*, ECCV'10, pages 140–153, Berlin, Heidelberg, 2010. Springer-Verlag.

[30] Y. Tian, R. Sukthankar, and M. Shah. Spatiotemporal deformable part models for action detection. In *CVPR*, pages 2642–2649. IEEE, 2013.

[31] Y.-I. Tian, T. Kanade, and J. F. Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 23(2):97–115, 2001.

[32] Y.-l. Tian, T. Kanade, and J. F. Cohn. Eye-state action unit detection by gabor wavelets. In *Proceedings of the Third International Conference on Advances in Multimodal Interfaces*, ICMI '00, pages 143–150, London, UK, UK, 2000. Springer-Verlag.

[33] Y. Tong, W. Liao, and Q. Ji. Facial action unit recognition by exploiting their dynamic and semantic relationships. *IEEE transactions on pattern analysis and machine intelligence*, 29(10):1683–1699, 2007.

[34] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497. IEEE, 2015.

[35] M. Valstar and M. Pantic. Fully automatic facial action unit detection and temporal analysis. In *CVPRW*, pages 149–149, June 2006.

[36] Z. Yang, M. Moczulski, M. Denil, N. de Freitas, A. Smola, L. Song, and Z. Wang. Deep fried convnets. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1476–1483, 2015.

[37] J. Yu and B. Bhanu. Evolutionary feature synthesis for facial expression recognition. *Pattern Recognition Letters*, 27(11):1289–1298, 2006.

[38] G. Zhao and M. Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):915–928, 2007.