PAIN DETECTION THROUGH SHAPE AND APPEARANCE FEATURES

Rizwan Ahmed Khan[1,2], *Alexandre Meyer*[1,2], *Hubert Konik*[1,3], *Saida Bouakaz*[1,2]

[1] Université de Lyon, CNRS
[2] Université Lyon 1, LIRIS, UMR5205, F-69622, France
[3] Université Jean Monnet, Laboratoire Hubert Curien, UMR5516, 42000 Saint-Etienne, France

{Rizwan-Ahmed.Khan, Alexandre.Meyer, Saida.Bouakaz}@liris.cnrs.fr
Hubert.Konik@univ-st-etienne.fr

ABSTRACT

In this paper we are proposing a novel computer vision system that can recognize expression of pain in videos by analyzing facial features. Usually pain is reported and recorded manually and thus carry lot of subjectivity. Manual monitoring of pain makes difficult for the medical practitioners to respond quickly in critical situations. Thus, it is desirable to design such a system that can automate this task. With our proposed model pain monitoring can be done in real-time without any human intervention. We propose to extract shape information using pyramid histogram of orientation gradients (PHOG) and appearance information using pyramid local binary pattern (PLBP) in order to get discriminative representation of face. We tested our proposed model on UNBC-McMaster Shoulder Pain Expression Archive Database and recorded results that exceeds state-of-the-art.

***Index Terms*—** pain, classification, PHOG, PLBP

1. INTRODUCTION

Most of the models for facial expressions recognition [1, 2, 3, 4, 5, 6] function only for six universal facial expressions. There exist very few computational models that can recognize very subtle facial expressions i.e. pain [7, 8, 9] and fatigue [10, 11]. Pain monitoring of patients (in a clinical scenario) is a very complicated, subjective but as well as very important task. Usually pain is self reported and according to Hadjistavropoulos et al. [12] it (self report) has many limitations. Thus, it is desirable to design such a system that can automate this task. Generally, manual monitoring of pain has following problems: first, pain cannot be recorded continuously. Second, the problem of subjectivity i.e. different patients have different perception of pain and can under report or over report the pain. Lastly, the person recording the pain has to make judgment of pain level, which could vary from person to person. An automatic computer vision system can solve all of the above mentioned problems.

In this paper we are proposing a framework that can recognize pain by analyzing face. This work can be considered as the first of its kind, as the previous algorithms that recognize pain [7, 8, 9] utilize facial action coding system (FACS) [13] or employ some kind of face registration, cropping or alignment [14, 15, 16] as a preprocessing. Our proposed framework is neither based on FACS nor it requires face alignment. FACS describes the facial expressions in terms of 46 component movements or action units (AUs), which roughly correspond to the individual facial muscle movements. The problem with using FACS is the time required to code every frame of the video. FACS was envisioned for manual coding by FACS human experts. It takes over 100 hours of training to become proficient in FACS, and it takes approximately 2 hours for human experts to code each minute of video [7]. Second limitation of existing algorithms for pain detection [7, 8, 9] is the problem of face registration. The algorithms proposed in [8, 9] are based on Active Appearance Models (AAMs) [17] and it is known that AAM fails to register face when either the initial shape estimate of face is too far off and /or the appearance model fails to direct search toward a good match. Another limitation of AAMs is the computational complexity associated with the training phase of it [18]. Our proposed algorithm rectify these problems as it does not require FACS coding and face image to be registered.

We are proposing a model that can recognize pain by analyzing shape and appearance information of the face. We have used shape and appearance features for detecting pain as according to Lucey et al. [8] both shape (i.e. contour) and appearance (i.e. texture) are important for pain detection. In our proposed model we have extracted shape information using pyramid histogram of orientation gradients (PHOG) [19] and appearance information using pyramid local binary pattern (PLBP).

2. UNBC-MCMASTER SHOULDER PAIN EXPRESSION ARCHIVE DATABASE

We have used UNBC-McMaster Shoulder Pain Expression Archive Database [20] to test the performance of the proposed model as done in [8, 9]. In the distributed database

Fig. 1. Examples from UNBC-McMaster Shoulder Pain Expression Archive Database. Considerable head movement occurs during the sequence.

archive there are 200 sequences across 25 subjects, which totals 48,398 images. All of the subjects were having a problem with shoulder pain. Spontaneous expression of pain from patients is recorded using digital cameras in a laboratory room as they underwent eight standard range-of-motion tests. Fig. 1 shows an example frames from the database. It is observable that there is a considerable head movement that occurs during sequences as the patient experiences pain.

All frames in the distribution are FACS coded and the PSPI (Prkachin and Solomon Pain Intensity Scale) pain score [21, 22] is also provided for every frame. PSPI defines pain with the help of FACS action units. According to PSPI pain is the sum of intensities of action units related to eye brow lowering, cheek raiser, nose wrinkles, lip raiser and eye closure. We used PSPI score to divide the database in two parts. First part contained frames with pain score of 0 (no pain), while the other part contained frames that show patients with pain (PSPI > 0). In this way 40,029 frames were categorized in first part (82.7%) and 8,369 frames were categorized in second part (17.3%).

3. EXTRACTED FEATURES

As mentioned earlier, we extracted shape and appearance features from the face as according to Lucey et al. [8] both shape (i.e. contour) and appearance (i.e. texture) are important for detecting pain. In our proposed framework for pain detection, we extracted shape information using pyramid histogram of orientation gradients (PHOG) [19]. For extracting appearance information we are proposing a new descriptor called pyramid local binary pattern (PLBP). As the name suggests it is based on local binary pattern (LBP) descriptor [23].

3.1. Pyramid local binary pattern (PLBP)

Pyramid local binary pattern (PLBP) is a *pyramidal-based spatial* representation of local binary pattern (LBP) descriptor. PLBP represents stimuli by their local texture (LBP) and the spatial layout of the texture. The spatial layout is acquired by tiling the image into regions at multiple resolutions. The idea is illustrated in Fig. 2. If only the coarsest level is used, then the descriptor reduces to a global LBP histogram. Comparing to the multi-resolution LBP of Ojala et al.[24] , our descriptor selects samples in a more uniformly distributed manner. Whereas Ojala's LBP takes samples centered around a point leading to missing some information in the case of face (which is different than a repetitive texture).

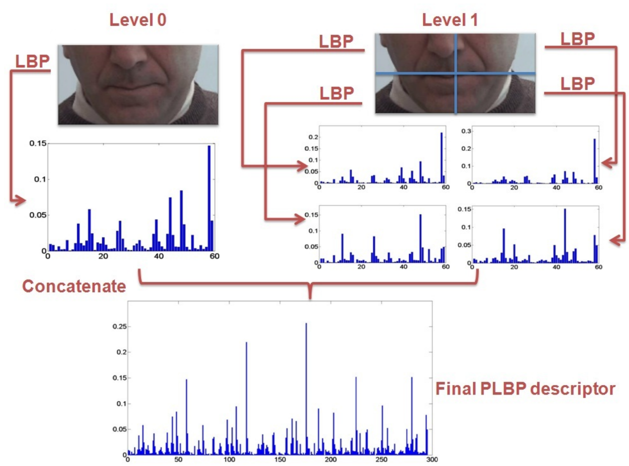

Fig. 2. Pyramid of Local Binary Pattern. First row: stimuli at two different pyramid levels, second row: histograms of LBP at two respective levels, third row: final descriptor.

LBP features were initially proposed for texture analysis [23], but recently they have been successfully used for facial expression analysis [2, 25]. The most important property of LBP features are their tolerance against illumination changes and their computational simplicity [23, 24]. The operator labels the pixels of an image by thresholding the 3 x 3 neighborhood of each pixel with the center value and considering the result as a binary number. Then the histogram of the labels can be used as a texture descriptor. Formally, LBP operator takes the form:

$$LBP(x_c, y_c) = \sum_{n=0}^{7} s(i_n - i_c)2^n \tag{1}$$

where in this case n runs over the 8 neighbors of the central pixel c, i_c and i_n are the gray level values at c and n and $s(u)$ is 1 if $u \geq 0$ or 0 otherwise.

Later, the LBP operator is extended to use neighborhood of different sizes [24] as the original operator uses 3 x 3 neighborhood. Using circular neighborhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighborhood. The LBP operator with P sampling points on a circular neighborhood of radius R is given by:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \tag{2}$$

where, g_c is the gray value of the central pixel, g_p is the value of its neighbors, P is the total number of involved neighbors and R is the radius of the neighborhood.

Another extension to the original operator is the definition of *uniform patterns*, which can be used to reduce the length of the feature vector and implement a simple rotation-invariant descriptor. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. Accumulating the patterns which have more than 2 transitions into a single bin yields an LBP operator, denoted $LBP^{u2}_{P,R}$. patterns. These binary patterns can be used to represent texture primitives such as spot, flat area, edge and corner.

We extend LBP operator so that the stimuli can be represented by its local texture and the spatial layout of the texture. We call this extended LBP operator as pyramid of local binary pattern or PLBP. PLBP creates the spatial pyramid by dividing the stimuli into finer spatial sub-regions by iteratively doubling the number of divisions in each dimension. It can be observed from the Fig. 2 that the pyramid at level l has 2^l sub-regions along each dimension ($R_0, \ldots R_{m-1}$). Histograms of LBP features at the same levels are concatenated. Then, their concatenation at different pyramid levels gives final PLBP descriptor (as shown in Fig. 2). It can be defined as:

$$H_{i,j} = \sum_{l} \sum_{xy} I\{f_l(x,y) = i\} I\{(x,y) \in R_l\} \tag{3}$$

where $l = 0 \ldots m-1$, $i = 0 \ldots n-1$. n is the number of different labels produced by the LBP operator and

$$I(A) = \begin{cases} 1 & \text{if A is true ,} \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

While, the dimensionality of the descriptor can be calculated by $N \sum_l 4^l$

The framework (see Section 4) extracts 59 LBP features from one sub-regions using $LBP^{u2}_{8,2}$ operator, which denotes a uniform LBP operator with 8 sampling pixels in a local neighborhood region of radius 2. This pattern reduces the histogram from 256 to 59 bins. In our experiment for pyramid of level 1, *l*=1 and *N*= 59, we obtained 590 dimensional feature vector for the complete face image (295 dimensions for upper part of the face and 295 dimensions for lower face part, see Section 4 for the discussion on face parts). Fig. 2 also illustrates extraction of PLBP features for level 1 pyramid from one face portion.

3.1.1. *Novelty of the proposed descriptor*

There exist some methods in literature that uses pyramid of LBP for different applications and they look similar to our proposed descriptor i.e. [26, 27, 28]. Our proposition is novel and there exist differences in the methodology that creates differences in the extracted information. Method for face recognition proposed in [26] create pyramids before applying LBP operator by down sampling original image i.e. scale-space representation, whereas we propose to create the spatial pyramid by dividing the stimuli into finer spatial sub-regions by iteratively doubling the number of divisions in each dimension. Secondly, our approach reduces memory consumption (do not requires to store same image in different resolutions) and is computationally more efficient. Guo et al. [27] proposed approach for face and palmprint recognition based on multiscale LBP. Their proposed method seems similar to our method for expression recognition but how multiscale analysis is achieved deviates our approach. Approach proposed in [27] achieves multiscale analysis using different values of P and R, where $LBP(P, R)$ denotes a neighborhood of P equally spaced sampling points on a circle of radius R (discussed earlier). Same approach has been applied by Moore et al. [28] for facial features analysis. Generally the drawback of using such approach is that it increases the size of the feature histogram and increases the computational cost. [28] reports dimensionality of feature vector as high as 30,208 for multiscale face expression analysis as compared to our proposition which creates 590 dimensional feature vector (see table 2) for the same task. We achieve the task of multiscale analysis much more efficiently than any other earlier proposed methods.

3.2. Pyramid histogram of orientation gradients (PHOG)

Pyramid histogram of orientation gradients (PHOG) [19] features are selected to extract shape information as they have proven to be highly discriminative for facial expression recognition task [29, 30, 3]. PHOG is a spatial shape descriptor. It first extracts Edge contours of the given stimuli using the Canny edge detector. Then, the image is divided into finer spatial grids by iteratively doubling the number of divisions in each dimension. The grid at level l has 2^l cells along each dimension. Afterwards, a histogram of orientation gradients (HOG) are calculated using 3 x 3 Sobel mask and the contribution of each edge is weighted according to its magnitude. Within each cell, histogram is quantized into *N* bins. Each bin represents the accumulation of number of edge orientations within a certain angular range. To obtain the final PHOG descriptor, HOG at the same levels are concatenated. The final PHOG descriptor is a concatenation of HOG at different pyramid levels. Generally, the dimensionality of the PHOG descriptor can be calculated by: $N \sum_l 4^l$.

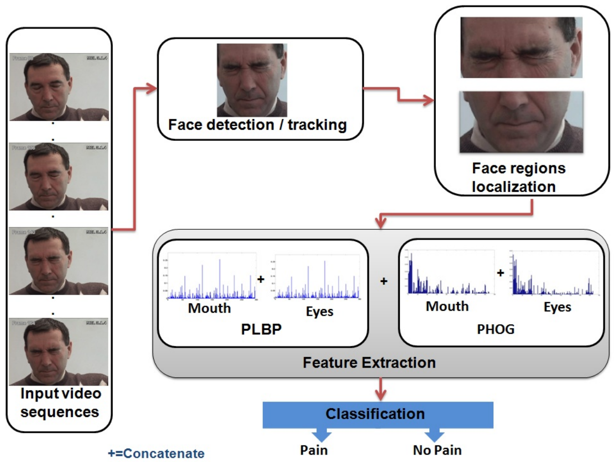

Fig. 3. Overview of the framework.

4. FRAMEWORK FOR PAIN DETECTION

The schematic overview of the proposed framework is illustrated in Fig. 3 and its steps are discussed below:

1. The first step of the framework is to detect the face from the input image sequence. The framework uses Viola-Jones object detection algorithm [31] to detect/track face in the video.

2. Then, the framework divides the detected face image into two equal parts. The upper face part contain regions of eyes and wrinkles on the upper portion of nose, while the lower part contain the regions of mouth and lower portion of the nose (see Fig. 3 for illustration). This is done as according to Ashraf et al. [9], regions around the eyes, eyebrows, and lips contribute significantly towards pain vs. no pain detection and these regions can be roughly localized by dividing the face image into two parts. The purpose of dividing the face image into two is to give equal importance to the upper and lower portion of the face. Thus, the extracted features will contain localized as well as holistic information of the face as the final feature vector is the concatenation of features from different regions and different pyramid levels.

3. Afterwards, the framework extracts PHOG and PLBP features from the upper and lower face portions and concatenates them to make final feature vector. In Fig. 3 the upper face portion is annotated as "eyes", while the lower face portion is annotated as "mouth".

4. Then, the concatenated feature vector is fed to the classifier for the final classification of the sequence.

5. EXPERIMENT

The performance of the framework was evaluated for four different classifiers (mentioned below) and up to three pyramid levels (for PHOG and PLBP).

1. Support vector machine (SVM) with χ^2 kernel

2. C4.5 decision tree (DT) with reduced-error pruning

3. Random forest (RF) of 10 trees

4. *2* Nearest neighbor (*2*NN) based on Euclidean distance

The framework [1] is evaluated on the complete database [20] (40,029 frames for no-pain examples and 8,369 frames for pain examples). The obtained results are presented in Fig. 4 and Table 1. These values are calculated using *10*-fold cross validation. The data presented in the figure not only shows the result for the proposed framework (last column in all four graphs, annotated as "combined") but also shows the result if PLBP or PHOG descriptors are used separately (with the same framework as discussed in Section 4, the only difference will occur in step 3 of the framework, where instead of extracting both the features only one feature will be extracted). The result proves that the recognition accuracy of the proposed framework for pain detection (i.e pain vs no pain) increases by combining two features. The results of proposed framework are tabulated in Table 1.

Table 1. Proposed framework recognition rate (%) for three pyramid levels.

	Pyramid Level		
	Level 0	Level 1	Level 2
SVM	90.7	96.1	96.4
2NN	96.1	96.9	96.9
Decision tree	87.3	87.5	90.2
Random Forest	92.7	92	95.9

One of the most interesting aspects of our approach is that it gives excellent results for a simple 2NN classifier which is a non-parametric method. This points to the fact that framework do not need computationally expensive methods such as SVM, Random forests or decision trees to obtain good results. In general, the proposed framework achieved high expression recognition accuracies irrespective of the classifiers, proves the descriptive strength of the features.

[1] visit www.youtube.com/watch?v=O_AIde58ZEo to watch demo video of the proposed framework

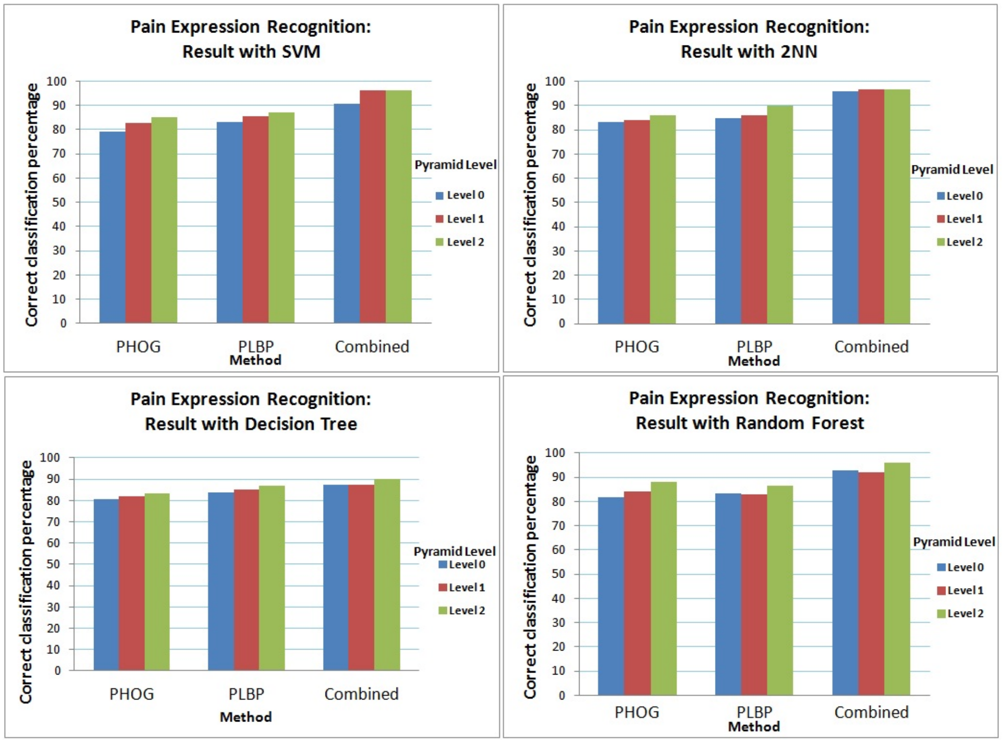

Fig. 4. Results obtained with the proposed framework. Results are presented for four different classifiers, with the first row showing results for "SVM" and "2NN" while the second row showing results for "decision tree" and "random forest".

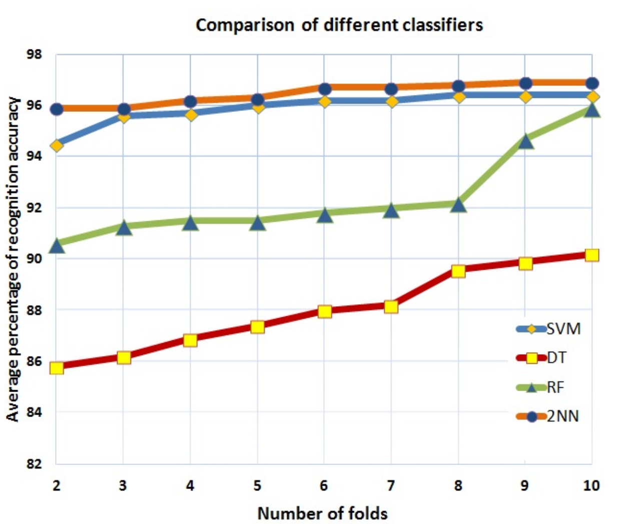

Fig. 5. Evolution of the achieved average recognition accuracy for the expression of pain with the increasing number of folds for the *k*-fold cross validation technique.

Fig. 5 shows the influence of the size of the training set on the performance of the four classifiers used in the experiment. For all the classifiers we have computed the average recognition accuracy using different number of folds (*k*'s) for the *k*-fold cross validation technique and plotted them in the Fig. 5. Fig. 5 also shows the supremacy of 2NN classifier in terms of achieved recognition rate. It is also observable from the figure that 2NN classifier achieved highest recognition rate among the four classifiers even with relatively small training set (i.e. 2-folds). This indicates how well our novel feature space was clustered.

Table 2. Feature vector dimensionality for different descriptors. Values presented here are obtained after concatenation of histograms for upper and lower face images.

	PHOG [19]	PLBP	Combined (proposed descriptor)
Level 0	16	118	134
Level 1	80	590	670
Level 2	336	2468	2814

Another significant contribution of the proposed framework is the computational simplicity. State-of-the-art algorithms [8, 9] achieves hit rate of 84.7% and 81.2 % respectively by using $\sim$ 27,000 dimensional feature vector. While the proposed framework is able to produce results better than state-of-the-art algorithms and it utilizes relatively significantly smaller feature vector (in terms of dimensions). Feature vector dimensionality for different pyramid levels are presented in Table 2.

6. CONCLUSION

In this paper we presented a novel framework for automatic recognition of pain. With the proposed framework high recognition accuracy, reduction in feature vector dimensionality and reduction in computational time for feature extraction is achieved. Our proposed framework can be used for real-time applications since its unoptimized Matlab implementation run at 8 frames/second (on windows 7 machine, with i7-2760QM processor and 6 GB RAM). In future research we plan to examine relationship between head movement and facial expression of pain. Incorporating such a relationship into the pain detection framework can be very useful. Secondly, we plan to extend this framework so that it can recognize various pain intensity levels.

7. ACKNOWLEDGMENT

This project is supported by the Région Rhône-Alpes, France.

8. REFERENCES

[1] G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of facial expression extracted automatically from video," *Image and Vision Computing*, vol. 24, pp. 615–625, 2006.

[2] G. Zhao and M. Pietikäinen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 915–928, 2007.

[3] R. A. Khan, A. Meyer, H. Konik, and S. Bouakaz, "Human vision inspired framework for facial expressions recognition," in *IEEE International Conference on Image Processing*, 2012.

[4] M. Pantic and I. Patras, "Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 36, pp. 433–449, 2006.

[5] M.F. Valstar, I. Patras, and M. Pantic, "Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data," in *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2005, pp. 76–84.

[6] R. A. Khan, M. Alexandre, H. Konik, and S. Bouakaz, "Framework for reliable, real-time facial expression recognition for low resolution images," *Pattern Recognition Letters*, vol. 34, pp. 1159–1168, 2013.

[7] G. Littlewort, M. S. Bartlett, and K. Lee, "Faces of pain: automated measurement of spontaneous facial expressions of genuine and posed pain," in *9th international conference on Multimodal interfaces*, 2007.

[8] P. Lucey, J. F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, and K.M. Prkachin, "Automatically detecting pain in video through facial action units," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 3, pp. 664 –674, june 2011.

[9] 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*, vol. 27, no. 12, pp. 1788 – 1796, 2009.

[10] Z. Zhang and J. Zhang, "Driver fatigue detection based intelligent vehicle control," in *18th International Conference on Pattern Recognition*, 2006, vol. 2, pp. 1262 –1265.

[11] X. Fan, B. Yin, and Y. Sun, "Yawning detection for monitoring driver fatigue," in *International Conference on Machine Learning and Cybernetics*, aug. 2007, vol. 2, pp. 664 –668.

[12] T. Hadjistavropoulos, K. D. Craig, and S.K. Lacelle, *Pain: Psychological Perspectives*, chapter Social influences and the communication of pain, pp. 87–112, 2004.

[13] P. Ekman and W. Friesen, "The facial action coding system: A technique for the measurement of facial movements," *Consulting Psychologist*, 1978.

[14] M. Monwar and S. Rezaei, "Video analysis for view-based painful expression recognition," in *IEEE International Joint Conference on Neural Networks*, 2008, pp. 3619–3626.

[15] M. M. Monwar and S. Rezaei, "Appearance-based pain recognition from video sequences," in *International Joint Conference on Neural Networks*, 2006, pp. 2429–2434.

[16] J. Chen, X. Liu, P. Tu, and A. Aragones, "Person-specific expression recognition with transfer learning," in *IEEE International Conference on Image Processing (ICIP)*, 2012.

[17] T. F Cootes, G. J. Edwards, and Taylor C. J., "Active appearance models," in *European Conference on Computer Vision*, 1998.

[18] Y. M. Lui, J.R. Beveridge, A.E. Howe, and L.D. Whitley, "Evolution strategies for matching active appearance models to human faces," in *International Conference on Biometrics: Theory, Applications, and Systems*, 2007.

[19] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *6th ACM International Conference on Image and Video Retrieval*, 2007, pp. 401–408.

[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," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2011, pp. 57 –64.

[21] K. Prkachin, "The consistency of facial expressions of pain: a comparison across modalities," *Pain*, vol. 51, pp. 297–306, 1992.

[22] K. Prkachin and P. E. Solomon, "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain," *Pain*, vol. 139, pp. 267–274, 2008.

[23] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distribution," *Pattern Recognition*, vol. 29, pp. 51–59, 1996.

[24] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 971–987, 2002.

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

[26] Wei Wang, Weimin Chen, and Dongxia Xu, "Pyramid-based multi-scale lbp features for face recognition," in *International Conference on Multimedia and Signal Processing (CMSP)*, 2011, vol. 1, pp. 151 –155.

[27] Zhenhua Guo, Lei Zhang, D. Zhang, and Xuanqin Mou, "Hierarchical multiscale lbp for face and palmprint recognition," in *IEEE International Conference on Image Processing*, sept. 2010, pp. 4521–4524.

[28] S. Moore and R. Bowden, "Local binary patterns for multi-view facial expression recognition," *Computer Vision and Image Understanding*, vol. 115, no. 4, pp. 541 – 558, 2011.

[29] Y. Bai, L. Guo, L. Jin, and Q. Huang, "A novel feature extraction method using pyramid histogram of orientation gradients for smile recognition," in *International Conference on Image Processing*, 2009.

[30] A. Dhall, A. Asthana, R. Goecke, and T. Gedeon, "Emotion recognition using PHOG and LPQ features," in *IEEE Automatic Face and Gesture Recognition Conference FG2011, Workshop on Facial Expression Recognition and Analysis Challenge FERA*, 2011.

[31] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.