Continuous Pain Intensity Estimation From Facial Expressions

Sebastian Kaltwang, Ognjen Rudovic and Maja Pantic
{*sebastian.kaltwang08, o.rudovic, m.pantic*}*@imperial.ac.uk*

Department of Computing, Imperial College London, UK

Abstract. Automatic pain recognition is an evolving research area with promising applications in health care. In this paper, we propose the first fully automatic approach to continuous pain intensity estimation from facial images. We first learn a set of independent regression functions for continuous pain intensity estimation using different shape (facial landmarks) and appearance (DCT and LBP) features, and then perform their late fusion. We show on the recently published UNBC-MacMaster Shoulder Pain Expression Archive Database that late fusion of the afore-mentioned features leads to better pain intensity estimation compared to feature-specific pain intensity estimation.

1 Introduction

Automatic pain recognition has received increased attention in the recent years mostly because of its applications in health care, ranging from monitoring patients in intensive-care units to assessment of chronic lower back pain [1]. Current research on automatic pain detection is based on automatic analysis of facial expressions, since it has been shown that facial cues are very informative for pain detection [2].

To date, there are only few works that have addressed the problem of automatic pain detection [3–7]. Brahnam et al. [3] used Principal Component Analysis, Linear Discriminant Analysis and Support Vector Machines (SVMs) for binary classification of pain images (i.e., pain vs. no pain). Gholami et al. [4] used intensities from facial images to train a Relevance Vector Machine (RVM) classifier for pain detection. Littlewort et al. [5] proposed a two-layer SVM-based approach for the classification of image sequences in terms of real pain and posed pain. In their approach, the presence of Facial Action Units (AUs) (see [8] for AU description) per frame is detected with a set of AU-specific SVM classifiers based on Gabor features. The outputs of the AU-specific SVMs are then temporally filtered and used as an input to the SVM classifier. The work by Lucey et al. [6] also addresses AU and pain detection based on SVMs. They detect pain either directly using image features or by applying a two-step approach, where first AUs are detected and then this output is fused by Logistical Linear Regression in order to detect pain. In their more recent work in [7], the authors train separate SVM classifiers for three-level pain intensity estimation.

Except from the approach proposed in [7], the rest of the aforementioned methods have been proposed for pain detection only (i.e., pain vs. no pain). In this paper, we propose a three-step approach to continuous pain intensity estimation per video frame (in contrast to [7], which estimates pain for a whole video sequence only). The outline of the proposed approach is depicted in Fig. 1. In the first step, we extract shape-based features (i.e, locations of characteristic facial points) and appearance-based features (Local Binary Patterns (LBPs) [9] and Discrete Cosine Transform (DCT) [10]) from facial images of subjects displaying different intensities of pain. The pain intensity was annotated by the database creators using sixteen discrete values (0 to 15), with 0 meaning no pain and 15 meaning its peak. In the second step, for each set of features we train separate regression models (in this paper, we employ Relevance Vector Regression (RVR) [11]) for prediction of the pain intensity levels. Note that although the regressor training is performed using discrete outputs (i.e., intensity labels from 0 to 15), during inference the regressors give a continuous estimation of the pain intensity. Finally, the outputs of the regressors trained using different feature sets are combined in two ways: (i) by computing the mean estimate of the regressors, and (ii) by using the outputs of separate regressors as an input to another RVR, which gives a single estimate for the pain intensity. In contrast to the aforementioned methods which deal with pain detection only (i.e., pain vs. no pain), the proposed approach is the first one that performs continuous pain intensity estimation. Furthermore, we show that the proposed feature-fusion scheme outperforms the separately trained RVRs on different feature sets, whereby the combination of appearance features (DCT and LBP) performs best. We also demonstrate the performance of the proposed approach in the task of continuous intensity estimation of the facial AUs.

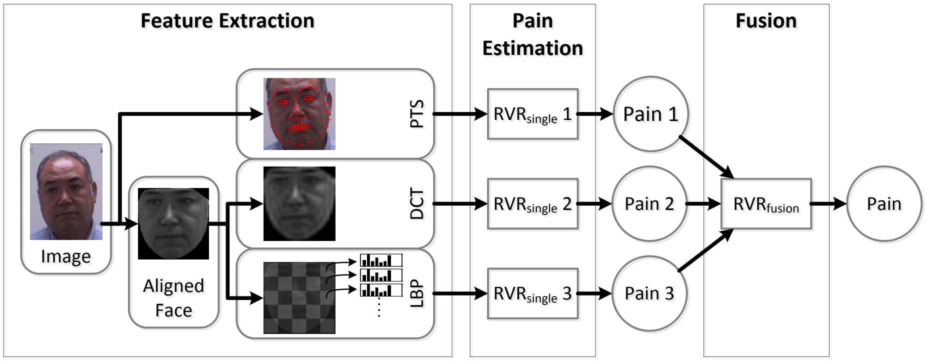

Fig. 1. Overview of the proposed approach to continuous pain intensity estimation. We first extract three feature sets from a face image: facial landmarks (PTS), Discrete Cosine Transform coefficients (DCT) and Local Binary Patterns (LBP). We then use Relevance Vector Regression (RVR) to learn the feature-specific functions, which independently estimate the pain intensity from each feature set. In the final step, we use a second layer RVR to perform the fusion of the pain intensity estimations obtained by the feature-specific functions

The rest of the paper is organized as follows. Section 2 describes the employed database. Feature extraction is detailed in Section 3. The regression-based approach to continuous pain intensity estimation is presented in Section 4. Section 5 shows the experiments and discusses the results. Section 6 concludes the paper.

2 Database

We use the publicly available UNBC-MacMaster Shoulder Pain Expression Archive Database [6] for our experiments. It contains face videos of patients suffering from shoulder pain while performing range-of-motion tests of their arms. Two different movements are recorded: (1) the subject moves the arm himself, and (2) the subject's arm is moved by a physiotherapist. Only one of the arms is affected by pain, but movements of the other arm are recorded as well as a control set. 200 sequences of 25 subjects were recorded (in total 48,398 frames). For each frame, AU intensities are provided for the AUs 4, 6, 7, 9, 10, 12, 20, 25, 26 and 27 on a 6 level scale (0-5) and for AU 43 on a 2 level scale (present or not). The number of frames available per AU intensity level is shown in Table 1.

Table 1. Frame distribution over AU intensity levels

Intensity	0	1	2	3	4	5
AU4	47324	202	509	225	74	64
AU6	42841	1776	1663	1327	681	110
AU7	45034	1360	991	608	305	100
AU9	47975	93	151	68	76	35
AU10	47873	171	208	63	61	22
AU12	41511	2145	1799	2158	736	49
AU20	47692	286	282	118	0	20
AU25	45992	766	803	611	138	88
AU26	46306	430	918	265	478	1
AU27	48380	6	3	3	6	0
AU43	45964	2434	-	-	-	-

According to [12], the pain intensity is quantified as having 16 discrete levels (0 to 15) based on the AUs as:

$$Pain = AU4 + max(AU6, AU7) + max(AU9, AU10) + AU43 \tag{1}$$

This score for the pain intensity is provided by the database creators, and is used in this work as the ground-truth for the pain intensity estimation. The distribution of the pain intensity levels in the database is shown in Fig. 2.

3 Feature Extraction

In the first step of our approach, we perform extraction of three different sets of features. The first set, denoted as Set 1, contains the locations of 66 facial landmark points (PTS)

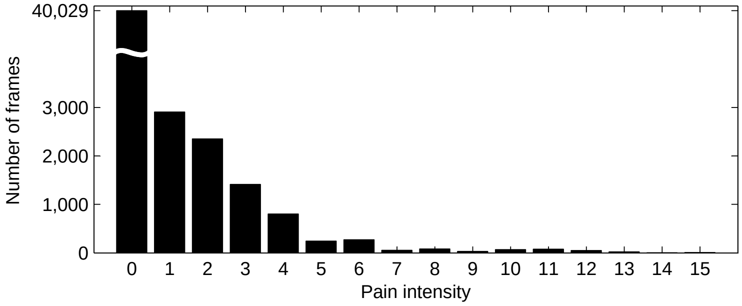

Fig. 2. Frame distribution over pain intensity levels

(see Fig.1) that are extracted by the database creators using the Active Appearance Model (AAM) [13]. As a preprocessing step, these points were aligned by applying Procrustes analysis.

The second set, denoted as Set 2, contains features obtained by applying the Discrete Cosine Transform (DCT) [10] to the aligned facial images. Specifically, the faces were first aligned to a base shape using the points from the triangulated mesh of the AAM (see [6] for details). Later, the 2-dimensional DCT was applied to the aligned images, and the first 500 coefficients were used as features, which were selected based on the *zig-zag* scheme [14]. We employ the PTS and DCT features in this paper since they have been previously proposed for pain detection (i.e., pain vs. no pain) in [6] .

The third set, denoted as Set 3, contains Local Binary Pattern (LBP) [9] features extracted from the above-mentioned shape-aligned images. We use these features since they have been shown to be effective for facial expression recognition [15]. An aligned face image was divided into patches and LBP histograms were extracted from each patch separately. After an initial parameter search, we chose uniform LBPs with 8 radial points on a radius of 2 pixel. The image was divided into 9x9 equally sized non-overlapping patches with a resolution of 14x13 pixel. The LBP histogram extracted from a patch resulted in a 59-D feature vector. The final LBP feature vector was the 4779-D concatenation of all 81 histograms.

We employ these three sets of features because they contain different types of information. PTS are geometric features, and are robust to illumination changes. However, they cannot accurately capture subtle facial movements (e.g., the eye wrinkles). This can be well described by the appearance features (i.e., DCT and LBP) that are derived from pixel intensities of an image. Compared to PTS, DCT and LBP are much more sensitive to skin color variation, and texture variation due to the illumination changes. Note, however, that DCT and LBP capture different characteristics of the texture changes. Specifically, DCT features describe image appearance on a large scale, which can be seen from a DCT reconstructed image: The overall image structure is still preserved, but sharp edges are lost (see Fig. 1). Conversely, LBP features are local descriptors that model statistics of the gradient orientations within a small pixel neighborhood, i.e. they describe the edges. For the aforementioned reasons, we hope that by fusion of these three types of features we can improve the overall accuracy of the continuous pain intensity estimation, as proposed in this paper.

4 Continuous Pain Intensity Estimation

To perform continuous pain intensity estimation from a single feature set, we learn a regression function that maps the features to the corresponding (discrete) pain intensity levels. This function is learned by means of the Relevance Vector Regression (RVR) model [11]. It models the target function by selecting representative cases, the so called 'Relevance Vectors', which are used in the model during inference of a query image. We use RVR instead of the popular Support Vector Regression in the target task because it usually results in a more sparse model, i.e., less relevance vectors are selected than support vectors for the same task [11]. In our case, this is important since we deal with image sequences. Formally, for each feature set we model the outputs (y) of the target function as:

$$y(x;\mathbf{w},\gamma,\delta) = \sum_{n=1}^{N} w_n K(x,x_n) + \varepsilon, \tag{2}$$

where x is the input feature vector, $\{x_1,...,x_N\}$ are N training inputs and $\mathbf{w} = (w_1,...,w_N)$ are the weight parameters. Here, the sparsity of the model comes from the fact that most of the weights parameters tend to go to zero, thus, the corresponding training samples are not used for inference. As the kernel function, we use the standard Radial Basis Function (RBF) kernel with the length scale parameter γ. The noise on the outputs is modeled as a Gaussian with zero mean and the variance δ.

Once the feature-set-specific target functions are learned, we perform late fusion of their outputs. This fusion is performed in two ways: (i) mean fusion and (ii) RVR fusion. In the mean fusion approach, we calculate the mean of the outputs, obtained by the feature-set-specific target functions $\{y_1,...,y_L\}$, as $y_f = \frac{1}{L}\sum_{l=1}^{L} y_l$, where y_f is the mean fusion output and L is the number of the feature sets. RVR fusion is performed by learning another RVR model that uses the outputs of the feature-set-specific target functions as an input, i.e., $\hat{y} = (y_1,...,y_L)$, which are continuous estimates of the pain level intensities, and the (discrete) pain level intensities as outputs. This fusion function is given by

$$y_f(\hat{y};\mathbf{w^f},\gamma^f,\delta^f) = \sum_{m=1}^{M} w_m^f K^f(\hat{y},\hat{y}_m) + \varepsilon^f, \tag{3}$$

where $\{\hat{y}_1,...,\hat{y}_M\}$ are M training inputs, obtained from the first-layer outputs, and $\mathbf{w^f} = (w_1^f,...,w_M^f)$ are the weight parameters, γ^f is the length scale of the Radial Basis Function kernel K^f and ε^f is the noise, as defined above. Note that the training samples used to learn the feature-set-specific target functions may differ from the samples used to learn the fusion function.

5 Experiments and Results

We performed two sets of experiments. In the first set of experiments we evaluated the performance of the proposed approach in the task of continuous AU intensity estimation. In the second set, we evaluated the performance in the task of continuous pain intensity estimation. In all our experiments we applied a leave-one-out cross-validation procedure. Specifically, we used facial images of 24 subjects for training and one subject for testing. The feature-specific target functions were trained using the same training data as for the fusion functions. Note that in terms of generalization performance, the performance of the proposed 2-layer approach is expected to be better if the feature-specific target functions and the fusion function are trained using data corresponding to different subjects. However, we found that this strategy results in worse performance than using the same training data to train both layers. This could be due to the limited number of available subjects: if the subjects are split between the first and the second layer, then each of the layers is trained on less subjects, and hence the performance decreases. Note also that AU27 was left out, since only few examples with intensities greater than zero are present in the dataset (see Table 1). We measured the performance of the proposed approach using the mean squared error (MSE) and the Pearson correlation coefficient (CORR). The MSE and CORR were computed on the differences between the predicted pain/AU intensities and the relevant ground truth. Furthermore, MSE and CORR were computed per subject and per sequence, and then correspondingly weighted by the number of frames in each sequence, in order to obtain an average value for each measure.

Table 2 shows the results for the feature-specific target function learned in the task of continuous pain/AU intensity estimation. In the case of pain intensity estimation, the ground truth contains 16 discrete intensity levels, while in the case of AUs there are 6 discrete intensity levels. In addition, we show the results of two methods for pain intensity estimation: Pain (I) is directly estimated from the features as described in Section 4, where Pain (II) is calculated from the estimated AU intensities by using Eq. 1. As can be seen, the results obtained by the latter method are in some cases outperformed by the former method, where the pain intensity is estimated directly from the training data. This is a consequence of the error propagation in the AU estimation, since for some AUs only few positive examples (i.e., with intensity level greater than zero) were available during training. Since the Pain (II) is computed by using a deterministic formula, the inaccuracies in the estimation of each AU are added in the final estimate of the pain intensity. Note also from Table 2 that for AU intensity estimation, LBP features outperform PTS and DCT features. This is because the LBPs are local descriptors and are able to better capture appearance variation caused by changes in AU intensities, since different AUs are located in different regions of a facial image. The accuracy in AU intensity estimation attained by using LBPs directly translates into the accuracy attained by the Pain (II) approach, which outperforms Pain (I) in the case of the LBPs. On the other hand, in the case of DCT features, which capture global changes in appearance, the Pain (I) is more accurate than the Pain (II) approach. This again shows that estimating the pain intensity level from AU intensities is sensitive to the errors in AU intensity estimation. We also observe that in the case of predicting AU20 with LBP features, the MSE can be misleading: the result of 0.103 seems better than the MSE of other AUs, but CORR is only 0.092. This is due to the imbalanced data used for training (see Table 1 and Fig. 2) where the vast majority of the frames have the zero intensity. Overall, LBP features perform best in terms of the MSE measure, while in the case of the CORR measure, the difference is not that apparent, though LBPs are still the best in most cases. On the other hand, DCT features perform best in the task of pain intensity estimation. Overall, appearance features (DCT and LBP) work better than shape features (PTS). However, the poor performance of shape features might be caused by registration errors, because the Procrustes alignment cannot cope properly with out-of-plane rotations. A better registration will likely improve the single shape and the fusion results, therefore we would not suggest to rely on appearance features alone.

Table 2. Single feature results for Action Unit (AU) and pain intensity estimation, measured by the mean squared error (MSE) and the Pearson correlation coefficient (CORR). Pain (I) is estimated directly from the features and Pain (II) is calculated from the estimated AU intensities using Eq. 1. The best result for each target and each measure is printed in bold letters

Measure	MSE			CORR		
Features	PTS	DCT	LBP	PTS	DCT	LBP
AU4	0.341	0.254	**0.204**	.096	**.140**	.133
AU6	0.906	0.592	**0.590**	.385	**.528**	.527
AU7	0.806	0.504	**0.379**	.120	.303	**.342**
AU9	0.119	**0.119**	0.113	**.246**	.224	.190
AU10	0.084	**0.079**	0.097	.171	**.203**	.169
AU12	1.010	0.717	**0.600**	.330	.484	**.548**
AU20	0.505	0.158	**0.103**	.012	**.092**	**.092**
AU25	0.707	0.579	**0.486**	.130	.104	**.204**
AU26	0.896	0.834	**0.475**	.013	.016	**.111**
AU43	0.300	0.273	**0.176**	.240	.291	**.465**
Pain (I)	2.592	**1.712**	1.812	.363	**.528**	.483
Pain (II)	2.532	1.716	**1.484**	.348	.480	**.518**

The results for the mean-fusion approach are shown in Table 3. In most cases, MSE and CORR improves over the results obtained with single features only. This shows that the employed features contain complementary information. Based on the CORR results, the DCT+LBP fusion gives the best results in most cases. This is not surprising, because DCT and LBP, although both being appearance-based features, capture different information: DCT captures global, while LBP captures local appearance variation.

The results for the RVR feature fusion are shown in Table 4. The results are similar to those obtained by the mean fusion in the sense that almost all values improve over the single feature results, as expected. However, the improved performance of DCT+LBP features is even more pronounced in the case of the RVR fusion approach, giving the best CORR results overall. Although we would expect the RVR fusion to perform at least as good as the mean fusion in all tasks, this does not seem to be the case. A reason for this could be the fact that both layers in the proposed approach are trained on the same data (because of the limited training data), which could have led the 2nd-layer RVR to over-fit the data. We plan to address this in our future work.

Table 3. Mean feature fusion results for Action Unit (AU) and pain intensity estimation, measured by the mean squared error (MSE) and the Pearson correlation coefficient (CORR). The best results are given in bold letters

Measure	MSE				CORR			
Features	PTS+DCT	PTS+LBP	DCT+LBP	all	PTS+DCT	PTS+LBP	DCT+LBP	all
AU4	0.224	0.201	0.206	**0.191**	.205	.260	.294	**.295**
AU6	0.543	0.544	0.496	**0.472**	.500	.508	.526	**.543**
AU7	0.479	0.429	**0.361**	0.376	.276	.276	**.376**	.343
AU9	0.087	0.091	0.096	**0.083**	.370	.339	.323	**.382**
AU10	0.064	0.070	0.075	**0.064**	.371	.312	.334	**.370**
AU12	0.656	0.625	0.568	**0.563**	.529	.545	.582	**.588**
AU20	0.177	0.179	**0.103**	0.119	.103	.095	**.133**	.129
AU25	0.474	0.449	0.455	**0.415**	.212	.213	**.264**	.252
AU26	0.622	**0.482**	0.557	0.493	.090	.118	.090	**.120**
AU43	0.232	**0.184**	0.191	0.187	.360	.396	**.462**	.439
Pain (I)	1.469	1.642	1.508	**1.373**	.489	.481	**.554**	.547
Pain (II)	1.928	1.850	**1.368**	1.480	.395	.403	**.529**	.494

Table 4. RVR feature fusion results for Action Unit (AU) and pain intensity estimation, measured by the mean squared error (MSE) and the Pearson correlation coefficient (CORR). The best results are given in bold letters

Measure	MSE				CORR			
Features	PTS+DCT	PTS+LBP	DCT+LBP	all	PTS+DCT	PTS+LBP	DCT+LBP	all
AU4	0.264	0.248	**0.242**	0.274	.209	.199	**.243**	.177
AU6	0.539	0.550	**0.480**	0.549	.487	.514	**.533**	.502
AU7	0.423	0.428	**0.343**	0.400	.248	.321	**.402**	.314
AU9	0.132	0.233	**0.120**	0.201	.401	.326	**.479**	.414
AU10	0.087	0.074	0.071	**0.070**	.080	.243	**.424**	.294
AU12	0.782	0.713	**0.617**	0.657	.507	.542	**.576**	.545
AU20	0.140	**0.088**	0.109	0.147	.049	.059	**.086**	.049
AU25	0.669	**0.538**	0.572	0.762	.106	.199	**.235**	.090
AU26	0.604	**0.414**	0.490	0.582	.005	.060	**.090**	.015
AU43	0.243	**0.158**	0.179	0.182	.352	.512	**.516**	.437
Pain (I)	1.801	1.567	**1.386**	1.804	.489	.485	**.590**	.502
Pain (II)	1.867	1.899	**1.633**	1.770	.342	.345	**.471**	.369

Fig. 3 shows an example of the pain intensity estimation from one test image sequence. The estimation is based on our best model, i.e., DCT+LBP RVR fusion (the Pain (I) approach). In most cases, the continuous pain intensity estimation is close to the ground-truth. Note, however, the peaks around the frames 95, 120 and 336, which are all caused by the eye blinks. This is a consequence of the fact that the proposed approach is static (i.e., it is trained per frame), and therefore, it cannot differentiate between an eye blink (short time) and eye closure (long time). During the training stage, the model has learned that the closed eyes are related to pain, and that is why the eye blinks result in sudden peaks in the estimated pain intensity, as shown in Fig. 3.

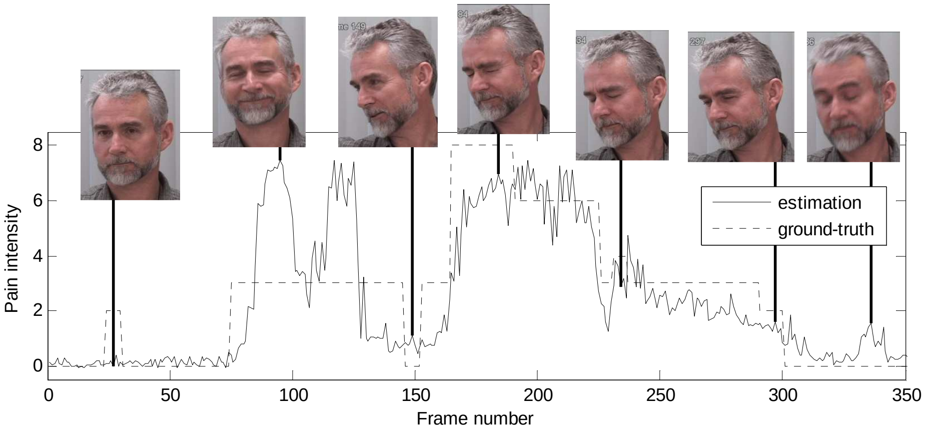

Fig. 3. Example pain estimation sequence for DCT+LBP RVR fusion

6 Conclusion

We have proposed a three-step approach to continuous pain intensity estimation based on Relevance Vector Regression. We have shown that for the task of continuous pain and AU intensity estimation, the proposed approach achieves better results when trained using appearance-based features (either DCT or LBP) than with the shape features (PTS). Also, when used as single input features, LBPs worked best in most cases. Furthermore, we showed that the fusion of DCT and LBP features gives the best performance in the target task. However, we believe that by a proper alignment of the shape-based features (e.g. by using [16]), the overall performance attained by the fusion of these three feature sets should improve. We also showed that direct pain estimation can be more accurate than calculation from the the AUs, which is probably due to the inaccuracies in AU intensity estimation. The approach presented in this paper estimates the AU intensities independently and does not exploit information about their co-occurrences. Furthermore, the current approach is static, and it cannot distinguish between eye blinks and eye closures, which are important cues for pain intensity estimation. These limitations of the proposed approach are the focus of our future research.

References

1. Prkachin, K.M., Hughes, E., Schultz, I., Joy, P., Hunt, D.: Real-time assessment of pain behavior during clinical assessment of low back pain patients. Pain **95** (2002) 23–30
2. Williams, A.C.d.C.: Facial expression of pain: An evolutionary account. Behavioral and brain sciences **25** (2002) 439–455
3. Brahnam, S., Chuang, C.F., Shih, F.Y., Slack, M.R.: Machine recognition and representation of neonatal facial displays of acute pain. Artificial intelligence in medicine **36** (2006) 211–22
4. Gholami, B., Haddad, W.M., Tannenbaum, A.R.: Agitation and pain assessment using digital imaging. In: Int'l Conf. of the Engineering in Medicine and Biology Society, IEEE (2009) 2176–2179
5. Littlewort, G.C., Bartlett, M.S., Lee, K.: Automatic coding of facial expressions displayed during posed and genuine pain. Image and Vision Computing **27** (2009) 1797–1803
6. Lucey, P., Cohn, J., Prkachin, K., Solomon, P., Matthews, I.: Painful data: The UNBC-McMaster shoulder pain expression archive database. In: Int'l Conf. on Automatic Face & Gesture Recognition and Workshops, IEEE (2011) 57–64
7. Lucey, P., Cohn, J.F., Prkachin, K.M., Solomon, P.E., Chew, S., Matthews, I.: Painful monitoring: Automatic pain monitoring using the UNBC-McMaster shoulder pain expression archive database. Image and Vision Computing **30** (2012) 197–205
8. Ekman, P., Friesen, W.V.: Facial action coding system: A technique for the measurement of facial movement (1978)
9. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. on Pattern Analysis and Machine Intelligence **24** (2002) 971–987
10. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete Cosine Transform. IEEE Trans. on Computers **C-23** (1974) 90–93
11. Tipping, M.E.: Sparse Bayesian learning and the relevance vector machine. The Journal of Machine Learning Research **1** (2001) 211–244
12. Prkachin, K., Solomon, P.: The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. Pain **139** (2008) 267–274
13. Cootes, T., Edwards, G., Taylor, C.: Active appearance models. IEEE Trans. on Pattern Analysis and Machine Intelligence **23** (2001) 681–685
14. Wallace, G.K.: The JPEG still picture compression standard. Communications of the ACM **34** (1991) 30–44
15. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on Local Binary Patterns: A comprehensive study. Image and Vision Computing **27** (2009) 803–816
16. Rudovic, O., Pantic, M.: Shape-constrained Gaussian Process Regression for Facial-point-based Head-pose Normalization. In: Int'l Conf. on Computer Vision, IEEE (2011) 1495 – 1502