Automatically Detecting Pain Using Facial Actions

Patrick Lucey,
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213 USA

Jeffrey Cohn,
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213 USA

Simon Lucey,
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213 USA

Iain Matthews,
Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213 USA

Sridha Sridharan, and
SAIVT Laboratory, Queensland University of Technology, Brisbane, QLD, 4000, Australia

Kenneth M. Prkachin
Department of Psychology, University of Northern British Columbia, Prince George, BC, V2N4Z9, Canada

Patrick Lucey: patlucey@andrew.cmu.edu; Jeffrey Cohn: jeffcohn@cs.cmu.edu; Simon Lucey: slucey@cs.cmu.edu; Iain Matthews: iainm@cmu.edu; Sridha Sridharan: s.sridharan@qut.edu.au; Kenneth M. Prkachin: kmprk@unbc.ca

Abstract

Pain is generally measured by patient self-report, normally via verbal communication. However, if the patient is a child or has limited ability to communicate (i.e. the mute, mentally impaired, or patients having assisted breathing) self-report may not be a viable measurement. In addition, these self-report measures only relate to the maximum pain level experienced during a sequence so a frame-by-frame measure is currently not obtainable. Using image data from patients with rotator-cuff injuries, in this paper we describe an AAM-based automatic system which can detect pain on a frame-by-frame level. We do this two ways: directly (straight from the facial features); and indirectly (through the fusion of individual AU detectors). From our results, we show that the latter method achieves the optimal results as most discriminant features from each AU detector (i.e. shape or appearance) are used.

1. Introduction

In clinical settings pain is difficult to assess and manage. There is a long history of researchers pursuing the goal of obtaining an objective measure of pain through analyzes of tissue pathology, neurological "signatures," imaging procedures testing of muscle strength and so on. These approaches have been fraught with difficulty because they are often inconsistent with other evidence of pain [19], in addition to being highly invasive and constraining to the patient. Because it is convenient, does not require advanced technology

or special skills and can provide information, patient self-report has become the most widely used technique to measure pain.

While useful, self-report measures have significant limitations [6, 9]. These include inconsistent metric properties across scale dimensions, reactivity to suggestion, efforts at impression management or deception, and differences between clinician's and sufferers' conceptualization of pain [20]. Moreover, self report cannot be used in important populations, such as young children, patients who have limited abilities to communicate, the mentally impaired, and patients who require assisted breathing. In addition to this, self-report measures only coincide in the instant when the patient is at their emotional apex (e.g. highest pain intensity in this case) and do not provide information on the patient's emotional state other than these peak periods. In short recordings of duration of a couple of seconds this information is ideal, however, if such a system is to be used for monitoring purposes which could extend to minutes, hours or even days (e.g. monitoring a patient in the ICU ward, or a person sitting at a computer desk), such self-reporting measures are of little use.

Significant efforts have been made to identify reliable and valid facial indicators of pain [7]. Most of these efforts have involved methods that require manual labelling of facial action units (AUs) defined by the Facial Action Coding System (FACS) [8], which must be performed offline by highly trained observers [4, 8] making them ill-suited for real-time application in clinical settings. However, a major benefit of using facial expressions is that a measure of pain can be gained at each time step (i.e. each video frame). And by training up a classifier to recognize the facial expressions associated with pain, a computer vision system can be developed to carry out this task in an automatic and real-time (or close to) manner.

In terms of FACS, pain has been associated with many AUs. Prkachin [15] found that four actions - brow lowering (AU4), orbital tightening (AU6 and AU7), levator contraction (AU9 and AU10) and eye closure (AU43) - carried the bulk of information about pain. In a recent follow up to this work, Prkachin and Solomon [16] found that these four "core" accounted for most of the variance in pain expression. They proposed summing the intensity of these facial actions to define pain intensity (see Section 2.1 for overview). In this paper we used this FACS pain intensity scale to define pain at each frame.

1.1. Related Work

There have been many recent attempts to detect emotions directly from the face, mostly using FACS [8]. Comprehensive reviews can be found in [17, 18]. In terms of automatically detecting pain, there have only been a few recent works. Littlewort et al. [11] used their AU recognizer based on Gabor filters, Adaboost and support vector machines (SVMs) to differentiate between genuine and fake pain. In this work no actual pain/no-pain detection was performed as differentiation between genuine, fake and baseline sequences was done via analyzing the various detected AUs. The classification for this work was done at the sequence level.

In terms of pain/no-pain detection, Ashraf et al. [1] used the UNBC-McMaster Shoulder Pain Archive to classify video sequences as pain/no-pain. Active appearance models (AAMs) were used to decouple the face into rigid and non-rigid shape and appearance

parameters and a SVM was used to classify the video sequences. In this work, they found that by decoupling the face into separate non-rigid shape and appearance components significant performance improvement can be gained over just normalizing the rigid variation in the appearance. Ashraf et al. [2] then extended this work to the frame-level to see how much benefit would be gained at labeling at the frame-level over the sequence-level. Even though they found that it was advantageous to have the pain data labeled at the frame-level, they proposed that this benefit would be largely diminished when encountering large amounts of training data.

In the works performed by Ashraf et al. [1, 2], the pain detection system was 'direct' (i.e. using the extracted features to train up the pain/no-pain SVM classifier). This approach has the disadvantage of using all the facial information, a lot of it not being related to pain. A more prudent strategy maybe just to use the information associated with the four core actions related to pain [15, 16]. A means of achieving this could be via an "indirect" approach (i.e. detecting pain as a series of recognized AUs) as only the AUs related to pain are used in detection. This indirect approach has the added advantage of using particular feature sets (i.e. non-rigid shape or appearance) which best detect a certain facial action which was shown in Lucey et al. [12]. For example, brow lowering (AU4) is best recognized by the shape features as these features produce strong changes, whilst the appearance changes around this area is hard to decipher as facial furrows and wrinkles vary across individuals.

Using this as our motivation, in this paper we describe an AAM/SVM based automatic system which can detect pain indirectly from a fused set of individual AU detectors which are known to correlate with pain at a frame-by-frame level. We show that this indirect system achieves better performance over a direct system, as we use an optimal feature set which better describe each facial action, which greatly differs from the approach of Ashraf et al. [1, 2].

The rest of the paper is organized as follows. In the next section we define a frame-level pain intensity metric derived by Prkachin and Solomon [16], as well as a description of the database that we used for our work. We then describe our automatic system which we use to directly detect pain and individual AUs followed by the associated results. We then describe our indirect automatic pain detection system which fuses individual AU detectors through output score fusion. We conclude with these results as well as some concluding remarks.

2. Detecting Pain from Facial Actions

2.1. Defining Pain in terms of AU

Much is known about how humans facially express pain from studies in behavioral science. Most of these studies encode pain from the movement of facial muscles into a series of AUs, based on the FACS. An example of the facial actions of a person in pain is given in Figure 1.

In 1992, Prkachin[15] conducted a study on facial expressions and found that four actions - brow lowering (AU4), orbital tightening (AU6 and AU7), levator contraction (AU9 and AU10) and eye closure (AU43) - carried the bulk of information about pain. In a recent

follow up to this work, Prkachin and Solomon [16] indeed found that these four "core" actions contained the majority of pain information As such they defined pain as the sum of intensities of brow lowering, orbital tightening, levator contraction and eye closure. The Prkachin and Solomon pain scale is defined as:

$$\mathrm{Pain}=\mathrm{AU4}+(\mathrm{AU6}\|\mathrm{AU7})+(\mathrm{AU9}\|\mathrm{AU10})+\mathrm{AU43} \quad (1)$$

That is, the sum of AU4, AU6 or AU7(whichever is higher), AU9 or AU10(whichever is higher) and AU43 to yield a 16-point scale [1]. Frames that have an intensity of 1 and higher are defined as pain. For the example in Figure 2, which has been coded as AU4B + AU6E + AU7E + AU9E + AU10D + AU12D + AU25E + AU43A, the resulting pain intensity would be 2 + 5 + 5 + 1 = 13. This is because AU4 has an intensity of 2, AU6 and AU7 are both of intensity 5 so just the maximum is taken, AU9 is of intensity 5 and AU10 is of intensity 4 so again the maximum is taken which is 5, and AU43 is of intensity 1 (eyes are shut).

2.2. UNBC-McMaster Shoulder Pain Expression Archive Database

The UNBC-McMaster Shoulder Pain Expression Archive database was used for this work. It contains video of the faces of adult subjects (129 subjects - 63 male, 66 female) with rotator cuff and other shoulder injuries. Subjects were recorded during movement of their affected and unaffected shoulder during active and passive conditions. In the active condition, subjects initiated shoulder rotation on their own. In the passive condition, a physiotherapist was responsible for the movement. In the experiments conducted in this paper, only the active condition was used. Within the active condition, tests were performed on both the affected and the unaffected shoulder to provide a within subject control. The camera angle for these tests were approximately frontal to start. Moderate head motion was common. Video of each trial was rated offline by a FACS certified coder. To assess inter-observer agreement, 1738 frames selected from one affected-side trial and one unaffected-side trial of 20 participants were randomly sampled and independently coded. Intercoder percent agreement as calculated by the Ekman-Friesen formula [8] was 95%, which compares favorably with other research in the FACS literature. For more information on the database, please refer to [16].

Out of the database, we used 203 sequences from 25 different subjects. Of these 203 sequences: 111 were on the affected shoulder and 92 on the unaffected shoulder; 101 contained pain (at least one frame in the video sequence had frames which had a pain intensity of 1 or greater) and 102 without pain. An example of a patient in pain is given in Figure 2(a). As can be seen in this example, considerable head movement occurs during the sequence. This highlights the challenge of this problem, as registering the patient's face can become difficult due to these added variabilities. The data is also different from a lot of currently available datasets as the video sequences have various durations, with sequences

---

[1]Action units are scored on a 6-point intensity scale that ranges from 0 (absent) to 5 (maximum intensity). Eye closing (AU43) binary (0 = absent, 1 = present). In FACS terminology, ordinal intensity is denoted by letters rather than numeric weights, i.e., 1 = A, 2 = B, … 5 = E.

last from 90 to 700 frames. Within these sequences, the patient may display various expressions multiple times.

3. Description of Pain Detection Modules

Our automatic pain detection system consists of firstly tracking a patient's face throughout a sequence. We do this using active appearance models (AAMs). Once the face is tracked, we use the information from the AAMs to extract both shape and appearance features. We then use these features as an input into a support vector machine (SVM), which we use for classification. We explain each of these modules in the following subsections.

3.1. Active Appearance Models (AAMs)

Active appearance models (AAMs) have been shown to be an accurate method of aligning a pre-defined linear shape modle that also has linear appearance variation, to a previously unseen source image containing the object of interest. In general, AAMs fit their shape and appearance components through a gradient-descent search, although other optimization methods have been employed with similar results [5].

The shape $\mathbf{s}$ of an AAM [5] is described by a 2D triangulated mesh. In particular, the coordinates of the mesh vertices define the shape s. These vertex locations correspond to a source appearance image, from which the shape was aligned. Since AAMs allow linear shape variation, the shape s can be expressed as a base shape $\mathbf{s}_0$ plus a linear combination of m shape vectors $\mathbf{s}_i$:

$$\mathbf{s}=\mathbf{s}_0+\sum_{i=1}^{m}p_i\mathbf{s}_i \quad (2)$$

where the coefficients $\mathbf{p} = (p_1, \ldots, p_m)^T$ are the shape parameters. These shape parameters can typically be divided into similarity parameters $\mathbf{p}_s$ and object specific parameters $\mathbf{p}_o$, such that $\mathbf{P}^T=[\mathbf{P}_s^T, \mathbf{P}_o^T]$. Similarity parameters are associated with geometric similarity transform (i.e. translation, rotation and scale). The object-specific parameters, are the residual parameters representing geometric variations associated with the actual object shape (e.g., mouth opening, eyes shutting, etc.). Procrustes alignment [5] is employed to estimate the base shape $\mathbf{s}_0$.

Keyframes within each video sequence were manually labelled, while the remaining frames were automatically aligned using a gradient descent AAM fit described in [14]. Figure 2(a) and (b) shows the AAM in action, with the 68 point mesh being fitted to the patient's face in every frame.

3.2. Feature Extraction

Once we have tracked the patient's face using the AAM by estimating the base shape and appearance parameters, we can use this information to derive features from the face. From the initial work conducted in [1, 13], we extracted the following features:

- **PTS:** similarity normalized shape, $\mathbf{s}_n$, refers to the 68 vertex points in s_n for both the x- and y- coordinates, resulting in a raw 136 dimensional feature vector. These points are the vertex locations after all the rigid geometric variation (translation, rotation and scale), relative to the base shape, has been removed. The similarity normalized shape $\mathbf{s}_n$ can be obtained by synthesizing a shape instance of $\mathbf{s}$, using Equation 2, that ignores the similarity parameters $\mathbf{p}$. An example of the normalized shape features, PTS, is given in Figure 2(c).

- **APP:** canonical normalized appearance $\mathbf{a}_0$ refers to where all the non-rigid shape variation has been normalized with respect to the base shape $\mathbf{s}_0$. This is accomplished by applying a piece-wise affine warp on each triangle patch appearance in the source image so that it aligns with the base face shape. If we can remove all shape variation from an appearance, we'll get a representation that can be called as shape normalized appearance, $\mathbf{a}_0$. The canonical normalized appearance $\mathbf{a}_0$ is different to the similarity normalized appearance $\mathbf{a}_n$ as it removes the non-rigid shape variation and not the rigid shape variation. The resulting features yield an approximately 27,000 dimensional raw feature vector. A mask is applied to each image so that the same amount of pixels are used. To reduce the dimensionality of the features, we used a 2D discrete cosine transform (DCT). Lucey et al. [12] found that using $M = 500$ gave the best results. Examples of the reconstructed images with $M = 500$ is shown in Figures 2(d). It is worth noting that regardless of the head pose and orientation, the appearance features are projected back onto the normalized base shape, so as to make these features more robust to these visual variabilities.

- **PTS+APP:** combination of shape and appearance features $\mathbf{s}_n + \mathbf{a}_0$ refers to the shape features being concatenated to the appearance features.

3.3. Support Vector Machine Classification

SVMs have been proven useful in a number of pattern recognition tasks including face and facial action recognition. Because they are binary classifiers they are well suited to the task of pain and AU recognition (if you consider the problem of pain/AU vs no pain/no AU). SVMs attempt to find the hyperplane that maximizes the margin between positive and negative observations for a specified class. A linear SVM classification decision is made for an unlabelled test observation $\mathbf{x}^*$ by,

$$\mathbf{w}^T\mathbf{x}^* \quad \begin{matrix} >^{true} \\ <_{false} \end{matrix} \quad b \qquad (3)$$

where $\mathbf{w}$ is the vector normal to the separating hyperplane and b is the bias. Both $\mathbf{w}$ and b are estimated so that they minimize the structural risk of a train-set, thus avoiding the possibility of overfitting to the training data. Typically, $\mathbf{w}$ is not defined explicitly, but through a linear sum of support vectors. As a result SVMs offer additional appeal as they allow for the employment of non-linear combination functions through the use of kernel functions, such as the radial basis function (RBF), polynomial and sigmoid kernels. A linear kernel was used in our experiments due to its ability to generalize well to unseen data in

many pattern recognition tasks [10]. Please refer to [10] for additional information on SVM estimation and kernel selection.

4. Direct Vs Indirect Automatic Pain Detection

In comparing the direct vs indirect automatic pain detection systems, we devised two sets of experiments. In the first experiment, we developed the direct pain detection system which detected for each frame whether or not the patient was in pain or not (i.e. detect frames with pain intensity of one or greater).

In the second set of experiments we developed the indirect pain detection system, which fused the outputs of each AU. Prior to this though, we wanted to see how conducive each AU was to the automatic detection of pain. This experiment was designed to show how discriminant each AU was in pain detection. Even though we are working on the assumption that the 4 core actions, or six AUs (AUs 4, 6, 7, 9, 10 & 43) carry the bulk of the information relating to pain - some may suffer from poor accuracy in terms of detection. This could be due possibly due to lack of training data or lack of discriminating features. Based on this heuristic evidence we will then fuse together the output scores of the various AUs to detect pain.

For each experiment, we provide results for each particular feature set (see Section 3.2), to determine if there was any benefit in using one particular set over another. To check for subject generalization, a leave-one-subject-out strategy was used and each SVM classifier was trained using positive examples which consisted of the frames that were labeled as pain. The negative examples consisted of all the other frames that were labeled as no-pain.

In order to predict whether or not a video frame contained pain, the output score from the SVM was used. As there are many more frames labeled as no-pain then pain, the overall agreement between correctly classified frames can skew the results somewhat. As such we used the receiver-operator characteristic (ROC) curve, which is a more reliable performance measure. This curve is obtained by plotting the hit-rate (true positives) against the false alarm rate (false positives) as the decision threshold varies. From the ROC curves, we used the area under the ROC curve (A'), to assess the performance which has been used in similar studies [11]. The A' metric ranges from 50 (pure chance) to 100 (ideal classification)[2].

4.1. Direct Automatic Pain Detection

The classification performances across the various feature sets for the direct automatic pain detector are given in Table 1. As can be seen by these results, there is little difference in performance between the shape, appearance and combined features, with the combined features yielding the best performance with an A' rating of 75.08. This results suggest that when incorporating all the various of parts the face directly, there is little discrepancy between the two feature sets, with little complimentary information existing. However, as shown in Lucey et al. [12] this is not the case for AU detection, as there is a large

---

[2]In literature, the A' metric varies from 0.5 to 1, but for this work we have multiplied the metric by 100 for improved readability of results

discrepancy between the feature sets for some AUs. The next set of experiments tries to exploit this by fusing the optimal feature sets together.

4.2. Indirect Automatic Pain Detection

4.2.1 Using AUs to Detect Pain—To see how conducive each AU was in detecting pain, we trained a SVM for each AU using positive examples which consisted of the frames that the FACS coder labelled containing that particular AU (regardless of intensity, i.e. A-E). The negative examples consisted of all the other frames that were not labelled with that particular AU. We then tested this model by comparing its output to the frames which were labelled by the FACS coder to have a pain intensity greater or equal than one. The results of this analysis are shown in Table 2.

In terms of detecting pain, AUs 6 and 43 are clearly the most reliable with AU6 achieving an $A^{'}$ rating of 74.64 for the combined features and AU43 getting a rating of 73.56 for the PTS features. These AUs are almost as discriminating as using the entire face as shown by the direct pain detection result $A^{'} = 75.08$. AU12 was the next best with a rating of 70.91 for the combined features. AUs 7 and 9 were slightly behind with ratings of around 65. AUs 20, 25 and 26 all poorly corresponded to pain, whilst AU 4 and 10 were not much better.

Considering that AUs 4, 6, 7, 9, 10 and 43 are used in defining the pain metric, it would be intuitive that the automatic recognition of these AUs would correspond highly with pain as this metric has been heavily motivated by the work in literature. However, the results suggest otherwise with only AUs 6, 7, 9 and 43 having reasonably high correspondence, with AU4 and 10 performing quite poorly. This result may advocate the consideration that these AUs not be used in the pain metric when training the system for automatic pain recognition. It may also be prudent to include AU12 into this metric, as it corresponded quite well to recognizing pain. The experiments in the next subsection explore various combinations of AUs in an effort to maximize the pain recognition results.

4.2.2 Indirect Pain Detection through AU Output Score Fusion—For these experiments we combined the various AUs through logistical linear regression (LLR) fusion [3]. LLR fusion is a method which simultaneously fuses and calibrates scores of multiple sub-systems together. It is a linear fusion technique, which like the SVM is trained on the scores with positive labels being those which coincide with the frames that are labelled as pain greater than two and the others being labelled as negative. Like the previous experiment, we then tested this fusion model by comparing its output to the frames which were labelled by the FACS coder to have a pain intensity greater or equal than one. We again compared these results to the direct pain detector.

The results for the indirect pain detector experiments are shown in Table 3. In this table, it is worth noting that the last column is denoted as the PTS+APP* features. This is different from just the PTS+APP features as the '*' denotes the optimal combination of the various AUs, rather than the combination of all features sets. For example, the best performing features for AU6 are the APP features and the PTS features for AU7. So the PTS+APP* features for the orbital tightening (AU6 &7) are the LLR fusion of the output scores for the APP features for AU6 and the output scores for the PTS features for AU7. This is in contrast

to just the PTS+APP features which would be just the LLR fusion of the PTS and APP for both AU6 and AU7.

The pain detection results for the individual four core facial actions (i.e. brow lowering, orbital tightening, levator contraction and eye closure) are given in the first four rows of results. This was designed again to show how these certain actions corresponded to the task of pain detection. As noted in the previous subsection, the brow lowering results are quite poor, whilst the eye closure action achieves a much better result. The fusion of AU6 and AU7 for the orbital tightening action seems to have diminished the performance of the AU6, with the performance dropping slightly (74.94 down to 74.28), suggesting that AU7 is of little benefit in classifying pain. The same could be said of the levator contraction action, with AU10 being of little benefit to this task.

The next three rows of results were designed to test our hypothesis that using the best performing AUs to perform indirect pain detection would yield better performance than the direct approach. Our results suggest backs up these assertions with the indirect pain detection system consisting of fusing the best performing together AUs outperforming direct approach 78.37 to 75.08, using the PTS+APP* features. This result suggests that by using AU detectors to indirectly detect pain is an more effective method, especially if pain intensity is defined summation of AU intensities. Interestingly however, this approach was also better than the indirect system which fused all the AUs together which form the pain metric (78.37 vs 77.26). This approach was also better than fusing all the AUs together (78.37 vs 77.26).

An example of the output scores from the direct and indirect pain detectors are given in Figure 2. At the top of this example are the frames which coincide with the actions of interest. Below these frames are the 3 plots which coincide direct pain detector described in Section 4.1. Below this is the indirect pain detector with the LLR output scores from fusing the best AUs together for the various feature sets. In these curves, the horizontal line denotes the threshold which the scores must be above for the patient to be deemed to be in pain. The bottom plot is the actual transcribed pain intensity. As can be seen from this example, the direction of all these curves correspond with the transcription, particularly with the output scores coinciding with the pain intensity peaks (i.e. frame 100 to 130). The major benefit of the indirect method showing with the pain being correctly detected in the frames 260–280.

5. Conclusions and Future Work

In this paper, we have looked at automatically detecting pain at a frame-by-frame level via their facial movements. For this work we used the UNBC-McMaster database which contains patients with shoulder injuries portraying real or spontaneous emotion. Specifically, we compared two methods of detecting pain: directly (i.e. using extracted features to directly detect pain/no-pain); and indirectly (i.e. detecting pain as a series of detected AUs). As pain can be defined as a combination of AUs, we thought that using the indirect approach would be an efficient way of detecting pain as only pain related information would be utilized in the detection process via the detected AUs. From our 2 set of experiments we were able to show this.

In addition to this finding, another interesting result we found was that even though it is relatively well established in psychological research that AUs 4, 6, 7, 9, 10 and 43 are the core AUs associated with pain, two of these AUs (4 and 10) were poor in detecting pain, whilst AU12 was quite good. To back this up, in our best performing indirect system we found that omitting AU4 and AU10 and including AU12 yielded the best performance. We endeavor to examine this aspect as well as investigating the problem of pain intensity recognition in our future work. However, incorporating AU12 introduces other complexities. In [16], Pkachin and Solomon found that AU12 distinguishes pain from nonpain, but loads on a different factor in principal components analysis. This could mean that, although the action is correlated with pain, it represents a different kind of process which could be smiling as AU12 is the principle movement in this facial action. From preliminary experiments, it is believed that people smile during or after a pain reaction. From observation it often seems that they are reacting in amusement to their own pain response. If that's true, then incorporating it into some kind of pain metric runs the risk of confusing signals. This is to be investigated further.

In our future research we also plan to examine the dyadic relationship between head movement and facial actions for pain recognition. We believe that incorporating such a relationship into the pain detector, as well as other modes such as eye gaze, audio (moaning and groaning) and body movement (guarding) will improve pain detection performance. We believe establishing such a framework would be advantageous in detecting other emotions such as depression.
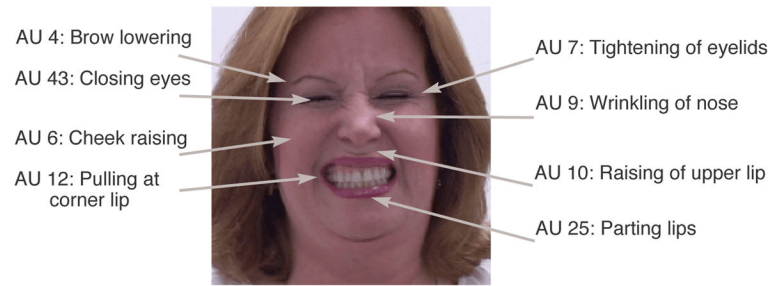
Acknowledgments

References

1. Ashraf, A.; Lucey, S.; Cohn, J.; Chen, T.; Ambadar, Z.; Prkachin, K.; Solomon, P.; Theobald, B-J. The painful face: pain expression recognition using active appearance models. Proceedings of the 9th international conference on Multimodal interfaces; Nagoya, Aichi: ACM; 2007. p. 9-14.

2. Ashraf A, Lucey S, Cohn J, Chen T, Prkachin K, Solomon P. The painful face: Pain expression recognition using active appearance models. to appear in Image and Vision Computing. 2009

3. Brummer N, du Preez J. Application-independent evaluation of speaker detection. Computer Speech and Language. 2006; 20:230–275.

4. Cohn, J.; Ambadar, Z.; Ekman, P. Observer-based measurement of facial measurement of facial expression with the facial action coding system. In: Coan, A.; Allen, J., editors. The handbook of emotion elicitation and assessment. Oxford University Press; New York, USA: p. 203-221.

5. Cootes T, Edwards G, Taylor C. Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2001; 23(6):681–685.

6. Cornelius, R. The Science of Emotion. Prentice Hall; Upper Saddler River: 1996.

7. Craig, K.; Prkachin, K.; Grunau, R. Handbook of pain assessment. The facial expression of pain.

8. Ekman, P.; Friesen, W.; Hager, J. Facial Action Coding System: Research Nexus. Network Research Information; Salt Lake City, UT, USA: 2002.

9. Hadjistavropoulos, T.; Craig, K. Pain: Psychological perspectives. Erlbaum; New York, USA: 2004. Social influences and the communication of pain.

10. Hsu, C.; Chang, CC.; Lin, CJ. Technical report. 2005. A practival guide to support vector classification.

11. Littlewort, GC.; Bartlett, MS.; Lee, K. Faces of pain: automated measurement of spontaneousallfacial expressions of genuine and posed pain. Proceedings of the International Conference on Multimodal Interfaces; New York, NY, USA: ACM; 2007. p. 15-21.

12. Lucey, P.; Cohn, J.; Lucey, S.; Sridharan, S.; Prkachin, K. Automatically detecting action units from faces of pain: Comparing shape and appearance features. Proceedings of the IEEE Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB); 2009.

13. Lucey, S.; Ashraf, A.; Cohn, J. Investigating spontaneous facial action recognition through aam representations of the face. In: Kurihara, K., editor. Face Recognition Book. Pro Literatur Verlag; 2007.

14. Matthews I, Baker S. Active appearance models revisited. International Journal of Computer Vision. 2004; 60(2):135–164.

15. Prkachin K. The consistency of facial expressions of pain: a comparison across modalities. Pain. 1992; 51:297–306. [PubMed: 1491857]

16. Prkachin K, Solomon P. The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. Pain. 2008; 139:267–274. [PubMed: 18502049]

17. Tian, Y.; Cohn, J.; Kanade, T. Facial expression analysis. In: Li, S.; Jain, A., editors. The handbook of emotion elicitation and assessment. Springer; New York, NY, USA: p. 247-276.

18. Tong Y, Liao W, Ji Q. Facial action unit recognition by exploiting their dynamic and semantic relationships. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2007; 29(10): 1683–1699. [PubMed: 17699916]

19. Turk, D.; Melzack, R. Handbook of pain assessment. The measurement of pain and the assessment of people experiencing pain.

20. Williams A, Davies H, Chadury Y. Simple pain rating scales hide complex idiosyncratic meanings. Pain. 2000; 85:457–463. [PubMed: 10781919]

AU 4: Brow lowering

AU 43: Closing eyes

AU 6: Cheek raising

AU 12: Pulling at corner lip

AU 7: Tightening of eyelids

AU 9: Wrinkling of nose

AU 10: Raising of upper lip

AU 25: Parting lips

Figure 1.
An example of facial actions associated when a person in is pain. In this example, AU's 4, 6, 7, 9, 10, 12, 25 and 43.
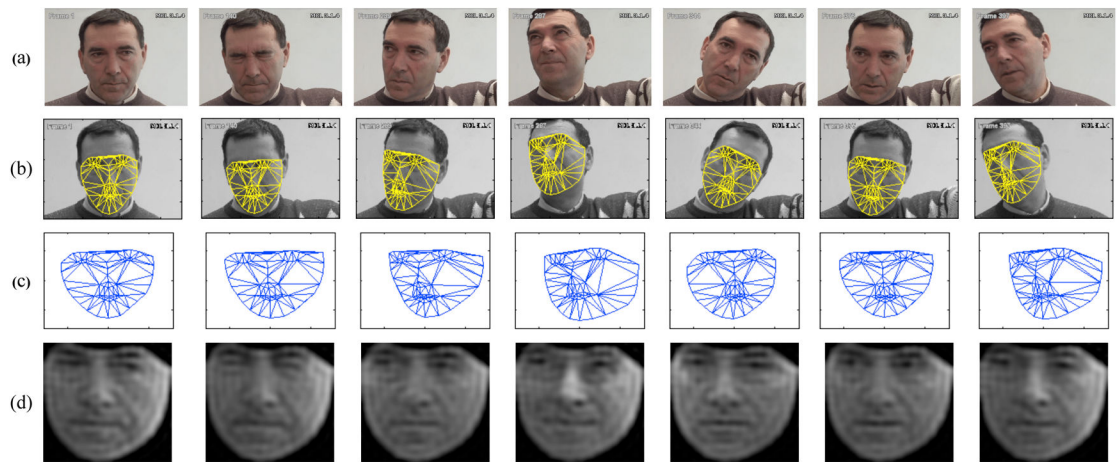
Figure 2.
Example of the output of the AAM tracking and the associated shape and appearance features: (a) the original sequence, (b) the AAM tracked sequence, (c) the normalized shape features (PTS), and (d) normalized appearance using 500 DCT coefficients (APP500).
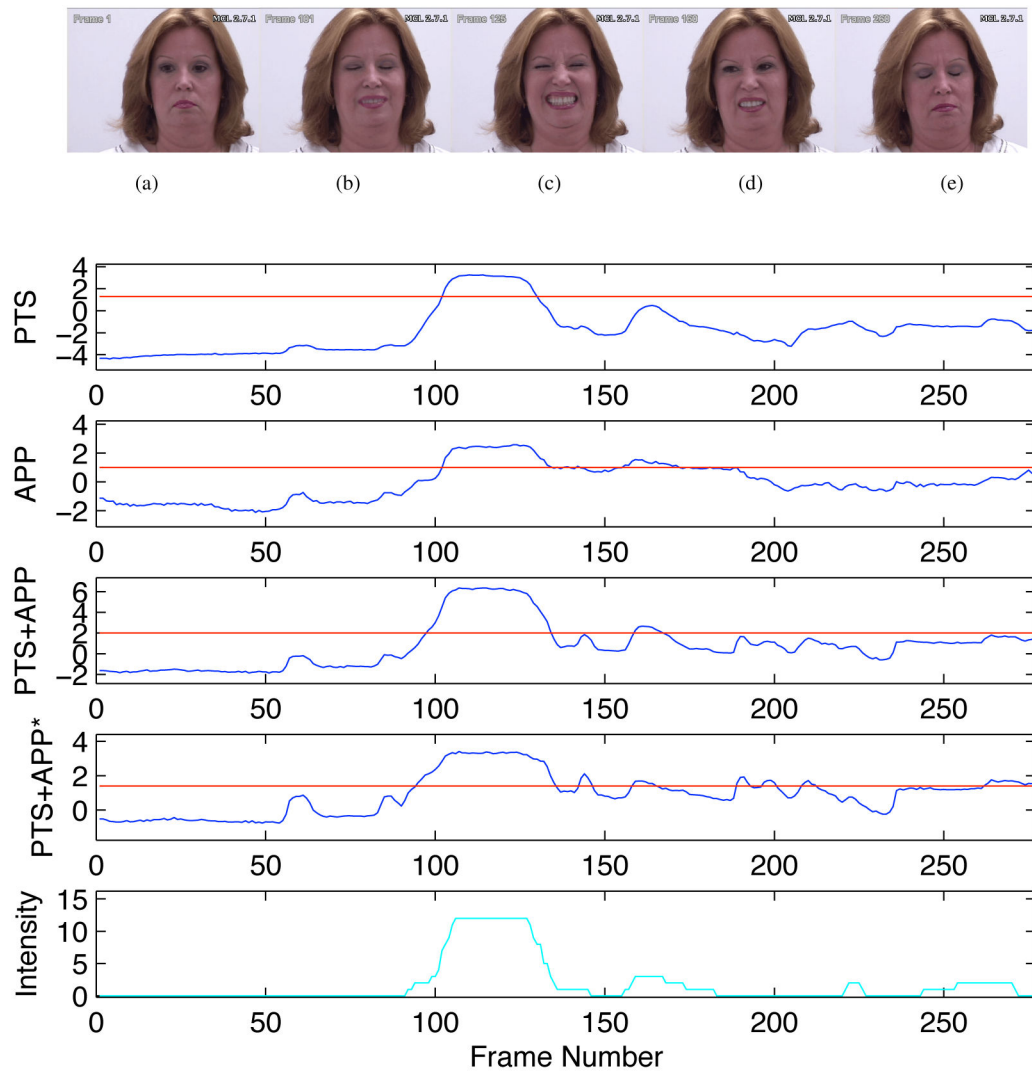
Figure 3.
The output scores from the SVM for the various features used. In these curves, the horizontal red lines denotes the threshold which the scores have to be above for the patient to be deemed to be in pain. The bottom curve is the actual transcribed pain intensity (as described in Section 2.1). Above the curves are the frames which coincide with actions of interest, namely: (a) frame 1, (b) frame 101, (c) frame 125, (d) frame 160 and (e) frame 260.

Table 1

Direct pain detection: Area underneath the ROC curve where each feature set is used to directly classify whether a frame is pain or no-pain. Results for shape, appearance and combination features are given.

PTS	APP	PTS+APP
74.25	73.20	**75.08**

Table 2

Indirect pain detection results by individual AU: Area underneath the ROC curve whereby separate AU detectors were used to detect pain for shape, appearance and combination features. These results are compared to the direct approach given in the last row.

AU	PTS	APP	PTS+APP
4	57.35	60.32	57.93
6	69.91	73.09	**74.64**
7	64.11	61.47	65.49
9	65.23	58.86	64.48
10	55.05	56.19	56.30
12	68.92	69.31	70.91
20	50.50	50.25	51.34
25	59.13	55.64	57.88
26	55.27	50.16	55.05
43	**73.56**	64.74	73.28
Direct	74.25	73.20	**75.08**

Table 3

Indirect pain detection results through AU fusion: Area underneath the ROC curve whereby separate AU detectors were fused together using LLR to detect pain. These results are compared to the direct approach given in the last row.

AU Combination	PTS	APP	PTS+ APP	PTS+ APP*
Eyebrow Lowering (4)	57.35	60.32	57.93	—
Orbital Tightening (6 & 7)	69.51	72.95	73.68	74.28
Levator Contraction (9 & 10)	63.92	59.61	64.05	65.68
Eye Closure (43)	73.56	62.98	73.28	—
Best performing AUs (6, 7, 9, 12 & 43)	75.65	74.29	76.44	**78.37**
Pain Metric (4, 6 &7, 9&10, 43)	74.69	73.09	73.56	77.28
All Aus	73.78	73.02	74.72	77.26
Direct (from Table 1)	74.25	73.20	75.08	—