Computer Vision and Image Understanding 147 (2016) 77–86

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Multiview Distance Metric Learning on facial feature descriptors for automatic pain intensity detection

Neeru Rathee[a], Dinesh Ganotra[b,*]

[a] MSIT, Guru Gobind Singh Indraprastha University, Dwarka, New Delhi, India
[b] Indira Gandhi Delhi Technical University for Women, Kashmeri Gate, New Delhi, India

ARTICLE INFO

Article history:
Received 26 March 2015
Accepted 14 December 2015

Keywords:
Multiview Distance Metric Learning
Pain detection
Pain intensity level detection
Support vector machine
Facial feature descriptors

ABSTRACT

In this study, a novel approach for automatic pain intensity detection is presented that capitalizes the complimentary information from various facial feature descriptors. We extract facial features by using the popular feature descriptors, which include Gabor features, Histogram of Orientation Gradient features and Local Binary Pattern features. A Multiview Distance Metric Learning (MDML) method is applied on these features to seek a common distance metric such that the features of the frames belonging to the same pain intensity are pulled as close as possible and that belonging to the different intensity levels are pushed as far as possible, simultaneously. Moreover, MDML extracts complementary information from various feature descriptors. The feature vector so obtained, are applied to Support Vector Machine for pain detection and pain intensity detection. We assess our algorithm on UNBC-McMaster Shoulder Pain Expression Archive Database. Experimental results represent that the efficiency of the proposed approach is 90% for pain detection and 75% for four level pain detection, which represents the potential of the proposed approach. A comparison of the proposed MDML approach is presented with other popular dimension reduction methods to prove its feature extraction capability.

1. Introduction

Faithful pain evaluation is the key concern of clinical practitioners since last many years. The most widely used technique is the patient self-report as it is convenient and does not require any special skill or advanced technology. Other methods which have been successfully used for pain assessment include a visual analog scale (VAS) and clinical interviews. The pain VAS is a unidimensional measure of pain intensity [1]. For pain intensity, the scale is most commonly anchored by “no pain” (score of 0) and “pain as bad as it could be” or “worst imaginable pain” (score of 100 [100-mm scale]) [2–4].

The self-reporting methods are not so reliable and have several drawbacks, which include reactivity to suggestion, efforts at impression management, deception, difference between patients and the clinician conceptualization of pain [5]. Moreover, self-report measures cannot be used with young children, with individuals with certain type of neurological impairment and dementia, with many patients in post-operative care or transient states of consciousness, and those with severe disorders requiring assisted breathing among other conditions [6].

All the above mentioned situations are handled with the help of an observer rating, where an observer chooses the face on the “faces of pain” scale, which resembles the facial expressions of the patient [7]. But this is highly impractical and inefficient if an onlooker is needed for a lengthy period of time, which might be a case of a person in an Intensive care unit [8]. In addition, the methods of observer measure and self-report are highly subjective. The output is measured when the patient is at their emotional apex, rather than continuously over time.

The other methods of pain assessment such as analysis of tissue pathology, neurological “signatures”, imaging procedures, testing of muscle strength are accomplished with difficulty because they are often inconsistent with other evidences of pain [9]. Moreover, they are highly invasive and need binding to the patients. Another important issue posed in pain measurement is the ability to differentiate real pain from faked pain. Some studies suggest that malingering rates are as high as 10% in chronic patients [10] and much higher in litigation contexts [11]. Another important issue raised by Hadjistavropoulos et al. [12] is that naive human subjects are marginally capable of differentiating real pain from fake pain by observing the facial expression.

* Corresponding author.
E-mail address: dinesh_ganotra@hotmail.com (D. Ganotra).

http://dx.doi.org/10.1016/j.cviu.2015.12.004
1077-3142/© 2016 Published by Elsevier Inc.

Table 1
Brief summary of the recent methods proposed for pain detection.

Sr. no.	Author name	Feature descriptor	Pain levels	Measures of pain intensity	Classifier	Performance measure	Accuracy
1	Ashraf et al. [6]	S-PTS, S-APP, C-APP+S-PTS	2	OPI PSPI	SVM (linear)	Leave One Subject Out CV	82%
2	Lucey et al. [16]	PTS, APP, PTS+APP	2	PSPI	SVM (linear)	Leave One Subject Out CV	78.37%
3	Lucey et al. [41]	PTS, APP	2	PSPI	SVM (linear)	Leave One Subject Out CV	78.03.%
4	Lucey et al. [8,15]	SAPP SPTS CAPP SAPP+SPTS+CAPP	2	PSPI	SVM (linear)+LLR	Leave One Subject Out CV	75.10% 76.90% 80.90% 84.70%
5	Hammal and Cohn [19]	CAPP	4	PSPI	SVM (linear)	Leave One Subject Out CV	73%
6	Kaltwang et al. [20]	DCT,LBP,PTS, DCT + LBP	16	PSPI	RVR	Leave One Subject Out CV	MSE=1.386, PCC=0.590
7	Lucey et al. [13]	AAM+SVM	2	PSPI OPI	SVM (linear)	Leave One Subject Out CV	83.9%
8	Khan et al. [21]	PLBP, PHOG	2	PSPI	SVM, 2NN, Random Forests, Decision tree	10-fold CV	96.4% 96.9%
9	Florea et al. [42]	Histogram of topographical features	16	PSPI	SVR (linear)	Leave One Subject Out CV	MSE=1.18, PCC = 0.55
10	Sikka et al. [25]	MS-MIL	2	PSPI	MILBoost based classification	Leave One Subject Out CV	83.70%
11	Rathee et al. [43]	TPS	16	PSPI	SVM	Leave One Sequence Out CV	96%

S-PTS: Similarity Normalized Shape, S-APP: Normalized Appearance, C-APP: Canonical Appearance, PTS: Normalized Shape, APP: Appearance, DCT: Discrete Cosine Transform, LBP: Local Binary Pattern, PLBP: Pyramid LBP, PHOG: Pyramid Histogram of Orientation Gradients, SVM: Support Vector Machine, PCC: Pearson Correlation Coefficient, RVR: Relevance Vector Regression, NN: Nearest Neighbor, MSE: Mean square Error, LLR: linear Logistic Regression, PSPI: Prkachin and Solomon Pain Intensity, CV: cross validation, AAM: Active Appearance Model, OPI: Observer Pain Intensity, MS-MIL: Multiple Segments Multiple Instance Learning, SVR: Support Vector Regression, RBF: Radial Basis Function.

A potential solution proposed by researchers is to use facial expressions. For the past couple of decades, efforts have been made to isolate such facial actions [13]. A measure of pain based on the facial action coding system (FACS) is applied to a frame by frame basis by Prkachin and Solomon [14] and results in a reliable measure of pain. The information obtained from such systems can be used for training a real-time automatic system that provides a significant advantage in sufferer's care and cost reduction [6,15,16]. Another system is proposed by Werner et al. [17] for pain detection by using facial expression and head pose information as features.

Several groups are working hard and designed the systems that can automatically distinguish pain and no-pain. [6,15,18]. With the advancement of automatic systems and their clinical and real time usages, researchers are oriented to measure the intensities of pain [19]. Recently, the authors in [13] trained extended SVM classifiers for three-level pain intensity estimation. Kaltwang et al. [20] propose a three-step approach to continuous pain intensity estimation per video frame. The shape and appearance based features (LBPs and DCTs) are extracted from facial images and fed to Relevance Vector Regression (RVR) for pain intensity level detection. The outputs of the regressors are combined by computing the mean estimate of regressors and by using the outputs of separate regressors as an input to another RVR, which produces a single estimate for pain intensity. Hammal and Cohn [19] used Log normal filters to identify 4 levels of pain.

The detection of pain or pain intensity is accomplished in two major steps: feature extraction and classification. Feature extraction methods include model based methods and pixel based methods. The most popular model used for modeling the faces for pain feature extraction is an Active Appearance Model (AAM). Model based methods have an inherent advantage of illumination and scaling invariance, but suffers from the problem of exhaustive computations due to the iterative fitting of models. Various pixel based methods proposed in the literature include Discrete Cosine Transform (DCT) [20], Local Binary Pattern (LBP) [20] and Histogram of Orientation Gradients (HOG) [21].

The main issue associated with the pixel based features is their large dimension. To reduce the dimension of data, Khan et al. [21] propose Pyramid Local Binary Pattern (PLBP) and Pyramid Histogram of Orientation Gradients (PHOG) methods for feature extraction. The classification problem of pain detection is solved by binary classification of pain images (i.e. pain and no pain) and techniques used include PCA, LDA, SVM [22] and the Relevance Vector Machine (RVM) [23]. Along with pain detection, researchers have tried to differentiate between real and posed pain [24]. Recently, Sikka et al. [25] proposed multiple segments-multiple instance learning (MSMIL) for continuous pain intensity estimation. Another approach that can be used for pain detection is by computing the intensity of action units. An approach for estimation of facial action units based on contextual analysis is represented by Rudovic et al. [26]. A brief of recent methods proposed for pain detection or pain intensity detection is presented in Table 1.

We have adopted pixel based feature descriptors (Gabor, LBP and HOG) in the proposed work for a faithful representation of facial images of different pain levels. The motive was to extract complimentary information from various descriptors. This aim has been achieved in past by semisupervised multiview distance metric learning (SSMDML) for cartoon synthesis [27], and by Neighborhood Repulsive Metric Learning (NRML) for kinship verification [28]. In MNRML the pairwise constrains are specified in the form of the data while SSMDML is graph based distance metric learning, which learns the distance metric to classify the cartoon characters by measuring the dissimilarity between them.

Since the facial features for images with different pain levels are very similar and various facial feature descriptors contain complimentary information, we propose MDML. MDML proposed in the presented approach is fully supervised and performs two tasks simultaneously: mapping the data to discriminative space and extracting complimentary features from various descriptors. In MDML, the distance metric is learnt in such a way that the intraclass samples (with the same pain intensity levels) are pulled as close as possible and interclass samples (with different pain intensity levels) are pushed as far as possible, simultaneously. The feature vector obtained by applying MDML is applied to Support Vector Machine (SVM) for assessing their pain representation capability.

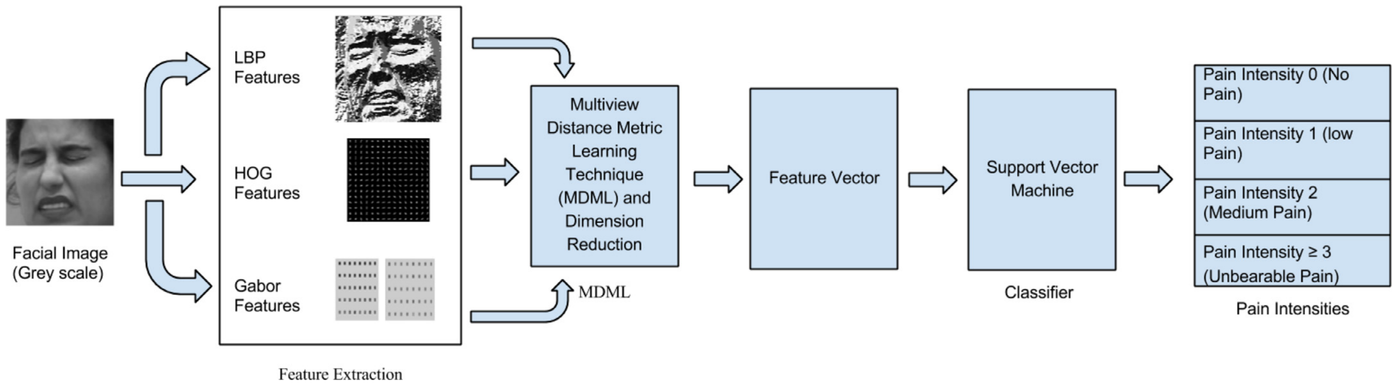

Fig. 1. Block diagram representation of pain intensity detection algorithm.

The proposed MDML is used to combine various facial feature descriptors, and is solved in an iterative manner using the basics of EM-based optimization. The optimization adopted to combine the features is completely different from those of SSMDML. The SS-MDML approach proposed in the literature has been used for classification purpose. They have adopted alternating optimization and gradient descent optimization for updating three different parameters: weight matrix corresponding to different views of features, distance metric between various cartoon characters and labels of the cartoon characters. The updated labels were used for classification of cartoon characters. This clearly shows the difference between the two approaches.

To prove the novelty and efficiency of the proposed approach, comparison of computation cost of the optimization method adopted by both the approaches is done in Section 3. The reason behind selecting computation cost rather than accuracy as a criteria of comparison is the application of SSMDML for classification and of the proposed approach for dimension reduction and feature fusion.

The rest of the paper is organized as follows. Section 2 describes the basics of feature descriptors adopted for representation of pain. In Section 3, basic mathematics of MDML is presented. The classifier for pain intensity detection, SVM, is explained in Section 4. The database used for evaluating the proposed method is well introduced in Section 5. Section 6 explains the experimental results. Finally, conclusion and future perspective are mentioned in Section 7. The pain detection approach adopted in this paper is presented well in Fig. 1.

2. Feature extraction

Feature extraction performs two basic tasks: transformation of input image into a feature vector and its dimension reduction for efficient classification. A well-defined feature extraction algorithm ensures more robust and efficient classification or recognition. Feature extraction from facial images is highly dependent on face localization. The requirement of real time applications further adds stringent constraints on the efficiency of this task.

2.1. Face registration

Image registration is the primary step for feature extraction which is popularly used for aligning similar facial images. For alignment of facial images, we have used 66 landmark points around facial components provided with UNBC McMaster database [8]. The reference landmark points are obtained by taking the average of all the landmark points over the entire training data. A 2D similarity transformation is utilized to transform the new image of the new reference coordinate system [29]. The registered facial images are then masked to extract facial images. The masked facial images are then resized to the size 128 × 108 by using the bilinear interpolation technique.

2.2. Facial feature descriptors

The resized images computed in the previous step are processed for feature extraction. The descriptors used in the proposed approach are described in brief in this section.

2.2.1. Gabor features

Gabor wavelet is one of the most famous technique for representing texture information of an object. A Gabor filter is defined with a Gaussian kernel that is modulated with a sinusoidal plane. To extract the texture information, filter bank consisting of Gabor filters with various scales and rotations is created. A detail of facial feature extraction using Gabor filters is presented by Tian et al. [30]. Recently, Li et al. [29] have applied Gabor features around landmark points for continuous FACS intensity estimation. In the proposed approach, we have implemented 40 Gabor filters (5 scales and 8 orientations) on the facial images so as to extract a feature vector of size 2820.

2.2.2. HOG

HOG features, which were firstly applied for human detection [31], are computed by counting the occurrences of gradient orientation in localized portions of an image. HOG features are based on the fact that the appearance and shape of the facial features can be described by the distribution of intensity gradients. The features so obtained are highly discriminative and represent a facial expression faithfully [29,32]. Motivated by the significant results for emotion recognition, we have adopted HOG features for pain representation. To extract the features from facial images, a cell of size 18 × 16 was built and horizontal gradient filter with 59 orientation bins is applied. To create the feature vector, HOG features for all the cells were computed and concatenated together so as to result in a vector of size 2832 (48 × 59).

2.2.3. LBP

LBP features are found to be a powerful method for texture feature extraction and have been popularly accepted for facial feature representation [33,34]. The most important properties of LBP are computation simplicity and illumination invariance. Moore and Bowden [35] investigate many variants of LBP for multi-view facial expression recognition to investigate the importance of multi-resolution and orientation analysis for feature representation. We have used extended LBP, which is rotation invariant. Rotation invariant LBP uses fewer bins as compared to regular LBP and reduces the number of components used for binary pattern representation.

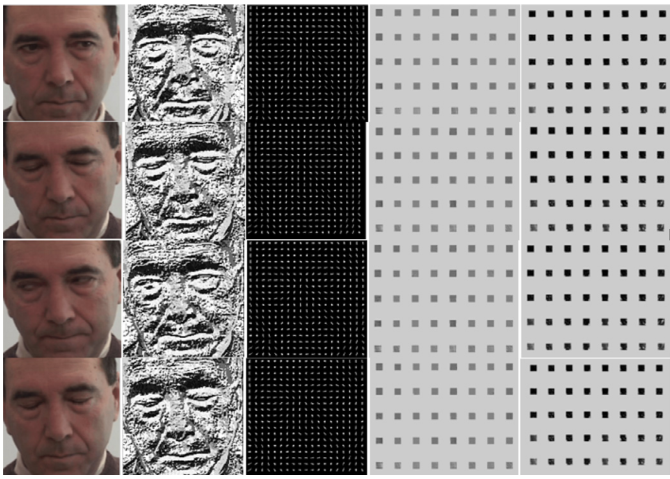

Fig. 2. Facial feature descriptors for frames with various pain intensities. The first column represents various pain intensities, the second column represents LBP features, third column represents HOG features and last two columns represents Gabor features.

Feature extraction is implemented as follows: first, the face image is divided into several nonoverlapping blocks. Then, LBP histograms are calculated for each block. Finally, the block LBP histograms are concatenated into a single vector. As a result, the facial expressions are represented by the LBP and the shape of the face is recovered by the concatenation of different local histograms.

The results of feature extraction from facial images are represented well in Fig. 2.

3. Multiview Distance Metric Learning

In literature, to combine information from various descriptors, they are simply concatenated. We have proposed MDML for combining the information from various descriptors, which extract complimentary information from the above mentioned feature descriptors. In MDML, we pull sample frames of a pain intensity close to their mean template (of similar pain intensity) and push them far from mean templates of other pain levels (Fig. 3).

3.1. Basics behind MDML

Let there be K feature descriptors of face and they are represented by $S^p = \{x_{it}^p \;|i = 1, 2.....L;\; t = 1, 2, ...N_i\}$ is the pth descriptor of tth image of ith level of pain images, $\overline{x_j^p}$ is the mean template of jth pain intensity level of pth descriptor, where $p = 1, 2, ..., K$. We aim to find a common distance metric such that the distance between x_{it}^p and $\overline{x_j^p}(i = j)$ is as small as possible and distance between x_{it}^p and $\overline{x_j^p}(j = i \pm 1)$ and between $\overline{x_i^p}$ and $\overline{x_j^p}(j = i \pm 1)$ is as large as possible, simultaneously. The distance metric is given by

$$d(\overline{x_j}, x_{it}) = \sqrt{(\overline{x_j} - x_{it})^T A(\overline{x_j} - x_{it})} \tag{1}$$

where A is a square matrix of size $n \times n$, and n represents the dimension of feature descriptor (the same for all the descriptors). The distance metric d should possess the properties of symmetry, non-negativity, and triangle inequality. Hence matrix A is symmetric and positive semidefinite. We formulate the DML algorithm as the following optimization problem:

$$\begin{aligned}
\max_A H(A) &= H_1(A) + H_2(A) + H_3(A) + H_4(A) - H_5(A) \\
&= \frac{1}{N_1}\sum_{i=1}^{L-1}\sum_{t=1}^{N_i} d^2(\overline{x_{i+1}}, x_{it}) + \frac{1}{N_2}\sum_{i=2}^{L}\sum_{t=1}^{N_i} d^2(\overline{x_{i-1}}, x_{it}) \\
&\quad + \frac{1}{L-1}\sum_{i=1}^{L-1} d^2(\overline{x_{i+1}}, \overline{x_i}) + \frac{1}{L-1}\sum_{i=2}^{L} d^2(\overline{x_{i-1}}, \overline{x_i}) \\
&\quad - \frac{1}{N_3}\sum_{i=1}^{L}\sum_{t=1}^{N_i} d^2(\overline{x_i}, x_{it}) \\
&= \frac{1}{N_1}\sum_{i=1}^{L-1}\sum_{t=1}^{N_i}(\overline{x_{i+1}} - x_{it})^T A(\overline{x_{i+1}} - x_{it}) \\
&\quad + \frac{1}{N_2}\sum_{i=2}^{L}\sum_{t=1}^{N_i}(\overline{x_{i-1}} - x_{it})^T A(\overline{x_{i-1}} - x_{it}) \\
&\quad + \frac{1}{L-1}\sum_{i=1}^{L-1}(\overline{x_{i+1}} - \overline{x_i})^T A(\overline{x_{i+1}} - \overline{x_i}) \\
&\quad + \frac{1}{L-1}\sum_{i=2}^{L}(\overline{x_{i-1}} - \overline{x_i})^T A(\overline{x_{i-1}} - \overline{x_i}) \\
&\quad - \frac{1}{N_3}\sum_{i=1}^{L}\sum_{t=1}^{N_i}(\overline{x_i} - x_{it})^T A(\overline{x_i} - x_{it})
\end{aligned} \tag{2}$$

where $N_1 = \sum_{i=1}^{L-1} N_i$, $N_2 = \sum_{i=2}^{L} N_i$, $N_3 = \sum_{i=1}^{L} N_i$, N_i is the number of image frames in ith pain level, and $\overline{x_i}$ represents the mean of feature descriptor of all the image frames of ith pain intensity, x_{it} represent the feature vector of tth image of ith pain intensity. The aim of H_1 and H_2 in Eq. (2) is to push image frames feature from mean feature as far as possible if they belong to the different pain level, with mean feature vector in the learned metric space. Similarly, aim of H_3 and H_4 is to ensure that the distance between mean feature vectors of different pain levels is maximized in the learned metric space. At the same time, H_5 ensures that image frame features (x_{it}) are pulled as close as possible to the mean feature template if they belong to the same pain intensity.

The optimization problem mentioned in Eq. (2) is solved iteratively as there is no closed form solution for such an optimization problem. The matrix A is symmetric and positive semidefinite and can be represented as

$$A = WW^T \tag{3}$$

where W is a nonsquare matrix of size $n \times m$ such that $m \le n$. Then Eq. (1) can be rewritten as

$$\begin{aligned}
d(\overline{x_j}, x_{it}) &= \sqrt{(\overline{x_j} - x_{it})^T A(\overline{x_j} - x_{it})} \\
&= \sqrt{(\overline{x_j} - x_{it})^T WW^T(\overline{x_j} - x_{it})} \\
&= \sqrt{(\overline{u_j} - v_{it})^T(\overline{u_j} - v_{it})}
\end{aligned} \tag{4}$$

where $u_j = W^T x_j$ and $v_{it} = W^T x_{it}$. Eqs. (2) and (4) can be combined to solve the value of H_1 as

$$\begin{aligned}
H_1(A) &= \frac{1}{N_1}\sum_{i=1}^{L-1}\sum_{t=1}^{N_i}(\overline{x_{i+1}} - x_{it})^T WW^T(\overline{x_{i+1}} - x_{it}) \\
&= tr\left(W^T \frac{1}{N_2}\sum_{i=1}^{L-1}\sum_{t=1}^{N_i}(\overline{x_{i+1}} - x_{it})^T(\overline{x_{i+1}} - x_{it})W\right) \\
&= tr(W^T P_1 W)
\end{aligned} \tag{5}$$

where $P_1 \triangleq \frac{1}{N_1}\sum_{i=1}^{L-1}\sum_{t=1}^{N_i}(\overline{x_{i+1}} - x_{it})(\overline{x_{i+1}} - x_{it})^T$.

Similarly, $H_2(A)$, $H_3(A)$, $H_4(A)$ and $H_5(A)$ can be simplified as

$$\begin{aligned}
H_2(A) &= tr\left(W^T \frac{1}{N_2}\sum_{i=2}^{L}\sum_{t=1}^{N_i}(\overline{x_{i-1}} - x_{it})^T(\overline{x_{i-1}} - x_{it})W\right) \\
&= tr(W^T P_2 W)
\end{aligned} \tag{6}$$

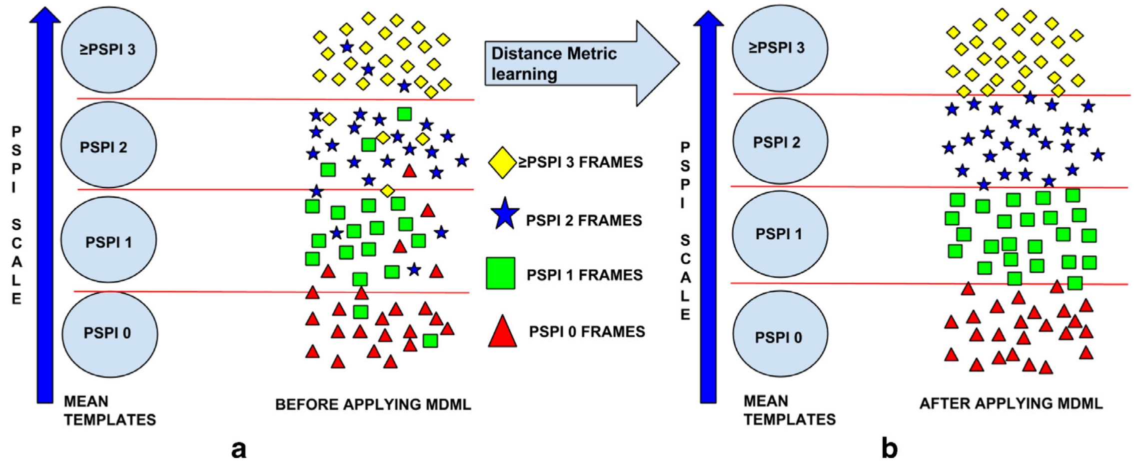

Fig. 3. Illustration showing the basics of DML. The distance between features and their mean get reduced if they belong to the same pain intensity, and get increased if they belong to the different pain intensity after DML.

$$H_3(A) = tr\left(W^T \frac{1}{L-1}\sum_{i=1}^{L-1}(\overline{x_{i+1}} - \overline{x_i})^T(\overline{x_{i+1}} - \overline{x_i})W\right) = tr(W^T P_3 W) \tag{7}$$

$$H_4(A) = tr\left(W^T \frac{1}{L-1}\sum_{i=1}^{L-1}(\overline{x_i} - \overline{x_{i-1}})^T(\overline{x_i} - \overline{x_{i-1}})W\right) = tr(W^T P_4 W) \tag{8}$$

$$H_5(A) = tr\left(W^T \frac{1}{N_3}\sum_{i=1}^{L}\sum_{t=1}^{N_i}(\overline{x_i} - x_{it})^T(\overline{x_i} - x_{it})W\right) = tr(W^T P_5 W) \tag{9}$$

where $P_2 \triangleq \frac{1}{N_2}\sum_{i=2}^{L}\sum_{t=1}^{N_i}(\overline{x_{i-1}} - x_{it})(\overline{x_{i-1}} - x_{it})^T$, $P_3 \triangleq \frac{1}{L-1}\sum_{i=1}^{L-1}(\overline{x_{i+1}} - \overline{x_i})(\overline{x_{i+1}} - \overline{x_i})^T$, $P_4 \triangleq \frac{1}{L-1}\sum_{i=1}^{L-1}(\overline{x_i} - \overline{x_{i-1}})(\overline{x_i} - \overline{x_{i-1}})^T$ and $P_5 \triangleq \frac{1}{N_3}\sum_{i=1}^{L}\sum_{t=1}^{N_i}(\overline{x_i} - x_{it})(\overline{x_i} - x_{it})^T$.

The DML algorithm can be formulated as

$$\max_{w} H(W) = tr[W^T(P_1 + P_2 + P_3 + P_4 - P_5)W] \tag{10}$$

subjected to $W^T W = I$, which is a constraint so as to restrict the scale of W so that optimization can be carried out with respect to w. The optimization problem is solved by using the eigenvalue method of solving DML. The eigenvalue problem can be formulated as

$$(P_1 + P_2 + P_3 + P_4 - P_5)w = \lambda w \tag{11}$$

Solving Eq. (11), we obtained $n \times m$ transformation matrix $W = [w_1, w_2, ...w_m]$, which are m eigenvectors corresponding to m largest eigenvalues ordered according to $\lambda_1 \geq \lambda_2 \geq ..... \geq \lambda_m$. The transformation matrix obtained in the proposed approach is a square matrix as we aim to map the data to higher discriminative space rather than dimension reduction. The original feature descriptors x_{it} are mapped to v_{it} as follows:

$$v_{it} = w^T x_{it} \tag{12}$$

To extract the complimentary information from the various feature descriptors, we formulate our algorithm for multiple feature selection as

$$\max_{W,\beta}\sum_{p=1}^{K}\beta_p^q\ tr[W^T(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W] \tag{13}$$

subjected to $W^T W = I, \sum_{p=1}^{K}\beta_p^q = 1,\ \beta_p \geq 0,\ q \geq 0$

where β_p^q is the non-negative weighted vector representing the contribution of various descriptors. The higher value of β_p^q indicates higher contribution of that descriptor.

Eq. (13) is a non-convex optimization problem with non-linear constraints. As mentioned in [28], there is no closed form solution and value of W and β_p^q is calculated by performing iterations. We keep W fixed after applying the DML described above and β_p^q is updated by constructing a Lagrange function.

$$F(\beta_p^q, \zeta) = \sum_{p=1}^{K}\beta_p^q tr[W^T(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W] - \zeta\left(\sum_{p=1}^{K}\beta_p^q - 1\right) \tag{14}$$

If we consider $\frac{\partial F(\beta_p^q,\zeta)}{\partial \beta_p^q} = 0$ and $\frac{\partial F(\beta_p^q,\zeta)}{\partial \zeta} = 0$, then Eq. (14) can be rewritten as

$$q\beta_p^{q-1} tr[W^T(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W] - \zeta = 0 \tag{15}$$

$$\sum_{p=1}^{K}\beta_p^q - 1 = 0 \tag{16}$$

Eqs. (3) and (4) can be combined to compute the value of β_p as follows:

$$\beta_p^q = \frac{1/tr[W^T(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W]^{1/(q-1)}}{\sum_{p=1}^{K} 1/tr[W^T(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W]^{1/(q-1)}} \tag{17}$$

After computing the value of β_p, W is updated using the following equation:

$$\max_{W}\sum_{p=1}^{K} W^T tr\left[\beta_p^q(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)W\right] \tag{18}$$

Subject to $W^T W = I$

The value of W is computed by solving the following eigenvalue equation:

$$\sum_{p=1}^{K}\beta_p^q(P_1^p + P_2^p + P_3^p + P_4^p - P_5^p)w = \lambda w \tag{19}$$

3.2. Computation complexity of MDML

For the proposed MDML method, each iteration involves calculating contribution factor (β) and solving a standard eigen value equation resulting in $O((K+n)N^2)$ complexity for first part and

$O(n^2)$ for second part, where K is the number of views, n represents dimensionality of the kth feature descriptor (which is the same for all the feature descriptors i.e. for all the views) and N represents the number of samples. The overall complexity of the proposed MDML is $O(((K+n)N^2)+n^2)T)$, where T is the number of Iterations.

For SSMDML, each iteration perform updating of confidence of feature vector with its label by using alternating optimization and the second part is for updating the distance metric between cartoon characters by using gradient descent approach. The cost of complexity of first part is $O(N^3)$ and that of second part is $O(SN^2n_k^2)$, where S represents the computation time of gradient descent method, n_k represents dimensionality of the kth view and N represents the number of samples . Overall complexity is $O(T(N^3+SN^2\sum_{k=1}^{K}n_k^2))$, where T is the iteration time of the alternating optimization procedure.

From the above description, it is clear that for first part the computation of first part is higher for SSMDML as long as number of samples is more than the dimension of feature vector (which is always the case for efficient classification). The computation cost of the second part of MDML is always less than SSMDML. From this discussion, we may conclude that the proposed MDML is efficient as compared to SSMDML in terms of computation complexity.

4. Support Vector Machine

Pain intensity detection is a closed set recognition problem in which an unknown pain intensity is classified as belonging to one of the four registered pain intensities. In pain intensity detection, a reject scenario is not defined. SVM introduced by Vapnik and Vapnik [36], solves the classification problem directly and avoids the problem of intermediate density estimation as in HMM and GMM. SVM is a unique discriminating approach that models the boundary between classes of training data in higher dimensional space [37]. The data are classified by finding a hyperplane with a maximum margin between the classes. The data set of training images with L labels is classified by

$$s(v) = sign\left[\sum_{i=1}^{L}\alpha_i y_i K(v_i, v) + b\right] \tag{20}$$

where $v_i \in R^n$, $y_i \in -1, 1$, α_i is the Lagrange multiplier of the dual optimization problem, b is bias and $K(v_i, v_j)$ is a kernel, which gives variability to the classifier to classify the data in different modes. We have adopted linear kernel, which can be given by

$$K_{Linear}(v_i, v_j) = v_i^T v_j \tag{21}$$

where v_i is the training vector. The main advantage of linear kernel is that there is no need of parameterization and it results in hyperplane with maximum margin.

SVM is a binary classifier, but it can be extended for more than two classes by using *one against one* and *one against all* technique [38]. In *one against one* technique, one SVM is used in every pair of classes. So, L-pain classification problem is solved by constructing $L(L-1)/2$ SVM classifiers. On the other hand, in *one against all* techniques, one SVM is used for each class which classify that class against the rest of the class. The number of SVM classifiers is equal to the number of classes (L) in this technique. We have adopted *one against one* technique of LIBSVM toolkit [39] for detection of 4 different levels of pain.

5. Database availability

The assessment of the proposed model for the recognition of pain expression intensities is made on UNBC-McMaster Shoulder Pain expression archive. It contains face videos of patients (66 females and 63 males) suffering from shoulder pain while performing range-of-motion tests of their arms. Movements of the arms were recorded in two different modes: (1) active mode where the subject moves the arm himself (with camera orientation frontal initially) and (2) passive mode where the subject's arm is moved by a physiotherapist (with camera orientation about 70 degrees from the frontal). Only one of the arms is affected by pain, but the movements of the other arm are recorded as well, as a control set. In both conditions, moderate changes in pose were common as participants contorted in pain [8].

Videos of 25 subjects (200 sequences, 48,398 frames) are available to the research community. These videos were recorded at a resolution of 320 × 240 pixels, out of which the face area spanned an average approximately of 140 × 200 pixels. For each frame, Prkachin and Solomon pain intensity (PSPI) [14] score was computed to quantify pain intensity in 16 discrete levels (0–15) based on AUs as:

$$Pain = AU4 + max(AU6, AU7) + \max(AU9, AU10) + AU43 \tag{22}$$

The PSPI ratings provided with the dataset were chosen as a measuring tool for the analysis of pain intensity detection. The PSPI ratings were selected as these are least dependent on self-reporting (like VAS) and observer dependent (Like OPI) and were calculated from Facial Action Units (FACs). The PSPI scale of 0–16 is used to represent different pain levels for the proposed pain intensity detection, and distribution for the same is listed in Table 2.

6. Experimental results

In the proposed method, we carried out experiments on UNBC-McMaster dataset for pain intensity detection. The features extracted using various facial feature descriptors mentioned in Section 2 are combined using weight matrix, W and contribution factor, β obtained through MDML training described in Section 3. The features are then applied to SVM for classification of pain levels.

To have an impartial comparison of the proposed approach with the approaches mentioned in the literature, we have conducted experiments in two parts: 1. we have detected pain by considering only two levels of pain: pain or no pain, 2. we have used SVM for detecting four different intensity levels of pain by following the distribution mentioned by Hammal and Cohn [19].

6.1. Experiments for pain detection

The pain detection experiments are done by considering it a binary classification problem. The two classes include pain (frames with PSPI $\geq$ 1) and no pain (frames with PSPI 0). The number of total frames with no pain is 40,029 and with pain is 8369. To avoid this nonuniform distribution, we limited the frames of PSPI 0 level to $\frac{1}{5}$ of the total frames of each subject. The extracted features are used to train the classifier by using leave one subject out cross validation. In leave one subject CV, The features extracted from one subject are used for testing and others for training. The UNBC database consists of videos of 25 subjects, so 25 iterations were performed to compute the average CV accuracy.

To train the SVM classifier at high speed, we adopted the method provided in budgetsvm toolbox [40]. The only kernel available with Pegasos SVM is linear kernel. To compare the performance of the various kernels of SVM, we adopted Low-rank Linearization SVM (LLSVM) with RBF kernel and polynomial kernel. The confusion matrix for all the kernels is represented in Table 3. The pain detection accuracy, using linear and RBF kernel is comparable and better than those obtained using the polynomial kernel.

Table 2
Distribution of pain levels based on PSPI score among various subjects of UNBC McMaster shoulder pain archive dataset.

	Sub1	Sub2	Sub3	Sub4	Sub5	Sub6	Sub7	Sub8	Sub9	Sub10	Sub11	Sub12	Sub13
PSPI 0	1895	1028	1544	810	2228	2503	640	1452	1962	896	1031	311	2180
PSPI 1	84	27	0	84	317	37	64	18	6	553	0	92	29
PSPI 2	56	29	0	0	155	34	18	6	418	459	355	45	31
≥ PSPI 3	99	36	64	0	52	35	51	136	88	130	116	361	121
	Sub14	Sub15	Sub16	Sub17	Sub18	Sub19	Sub20	Sub21	Sub22	Sub23	Sub24	Sub25	
PSPI 0	3212	2819	2758	2281	1599	2453	1726	1212	1490	478	822	699	
PSPI 1	0	0	89	386	243	158	83	35	56	0	135	413	
PSPI 2	43	0	121	47	37	73	32	29	0	19	17	327	
≥ PSPI 3	105	0	624	84	162	224	64	51	20	21	209	256	

Table 3
Confusion matrix for pain detection using various kernels of SVM.

	Linear		RBF		Polynomial	
	Pain	No pain	Pain	No pain	Pain	No pain
Pain	97	3	98	2	95	5
No pain	18	82	20	80	23	77

Table 4
Confusion matrix of the proposed approach for 4-level pain intensity classification by leave one subject out cross validation using a linear kernel of SVM.

	PSPI 0	PSPI 1	PSPI 2	≥ PSPI 3
PSPI 0	93.33	3.92	2.75	0.00
PSPI 1	6.98	67.48	20.18	5.36
PSPI 2	2.34	18.03	69.08	10.55
≥ PSPI 3	2.99	13.86	13.12	70.02

To prove the efficiency of the proposed approach, we compared the results with the methods mentioned in literature. The comparative graph is shown in Fig. 4.

6.2. Experiments for pain intensity detection

Experiments are performed to classify four intensity pain levels. We have adopted the data distribution as mentioned by Hammal and Cohn [19]. The pain levels are classified using budgetsvmtrain. The performance of the proposed SVM classifiers for the four intensity levels is evaluated. Recall, precision and F1 are used to quantify the performance of each classifier in comparison with ground truth, the PSPI. We have performed 5-fold cross validation and leave one subject out cross validation to prove the efficacy of the proposed algorithm.

In 5-fold cross validation, the frames from each pain level were divided into five sets and five iterations were performed to compute the cross validation accuracy. In each iteration, four sets were used for training and one for testing. The average accuracy of all the five iterations was calculated to compute 5-fold cross validation accuracy. It was observed that due to non-uniform data distribution for various pain levels and availability of a large number of frames for pain 0 (or PSPI 0 or no pain) the SVM was over fitted to PSPI 0 level, thus classifying most of the frames as PSPI 0 resulting in a high F1 score for PSPI 0 and high classification accuracy of the system.

To avoid this biasing, we carried out experiments by limiting the number of frames for PSPI 0 level of each subject to 500. The motivation behind using 500 frames is a limited number of frames (which are very less) in other pain levels of all the subjects. Moreover, hit and trial experiments done with a wide range of pain 0 frames (300–1000 with an increment of 50 frames in each set) for obtaining unbiased results. It was observed that when the number of PSPI 0 frames for each subject was larger in test set than the F1 score for pain 0 and system classification accuracy was high leaving F1 score of other pain levels very low.

In the above mentioned approach, the frames of the same person were present during training and testing set so the results are not so reliable. To compute the reliable results, leave one subject out cross validation was computed, where the training set consists of the data from all the subjects leaving one subject, which is used for testing. As the database is having data for 25 subjects so 25 iterations were performed. In each iteration, frames of 24 subjects were used for training and the frames of remaining subject were

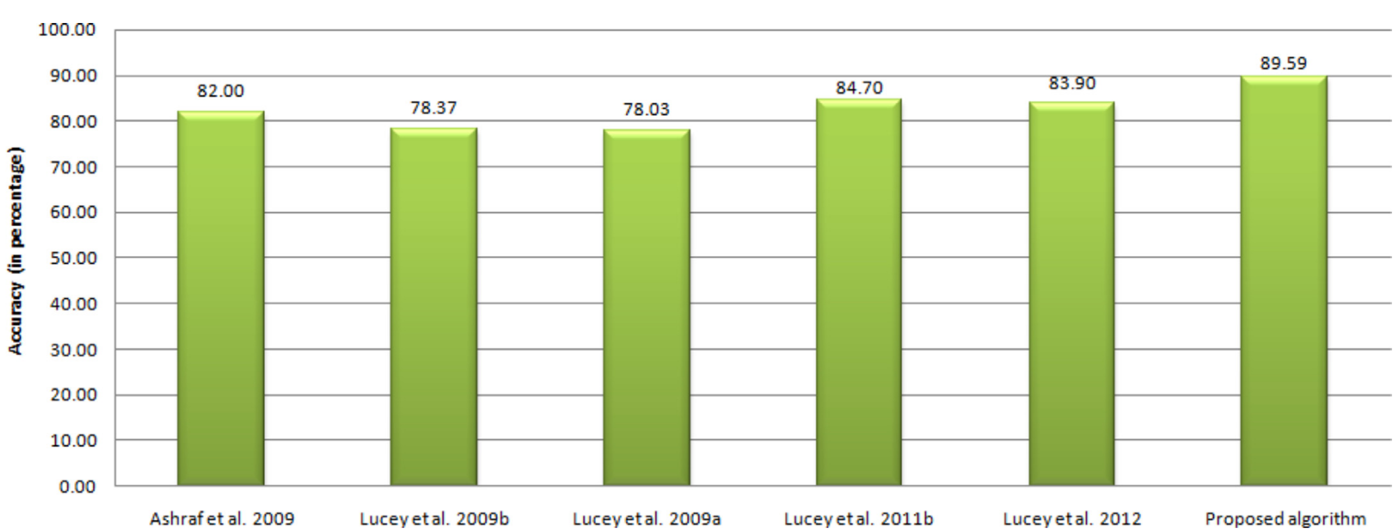

Fig. 4. Comparison of pain detection results with other methods proposed in the literature.

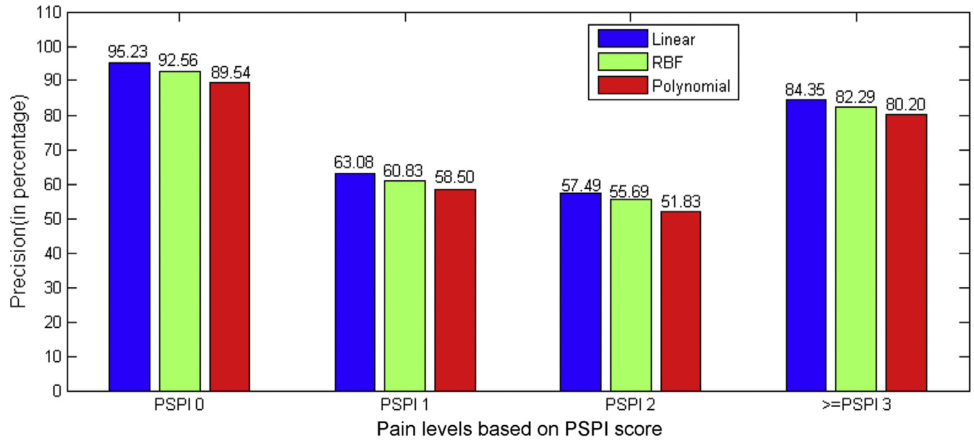

Fig. 5. Comparison of precision rate using different SVM kernels for 4-level pain intensity detection.

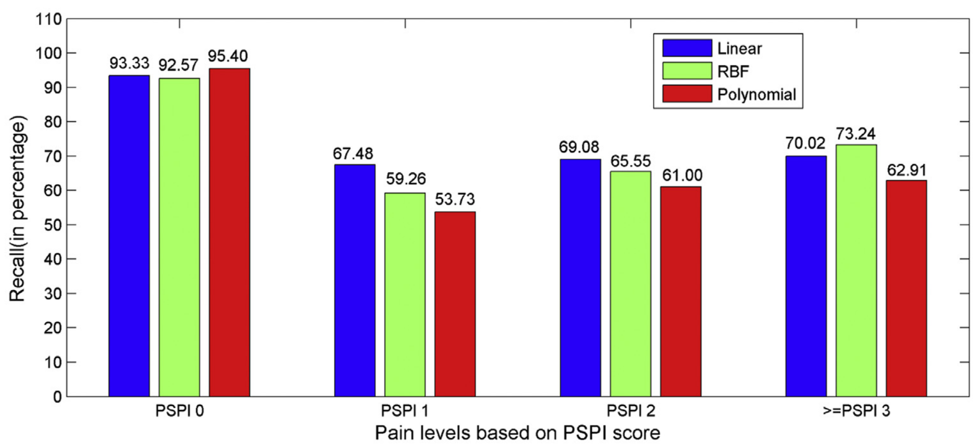

Fig. 6. Comparison of recall rate using different SVM kernels for 4-level pain intensity detection.

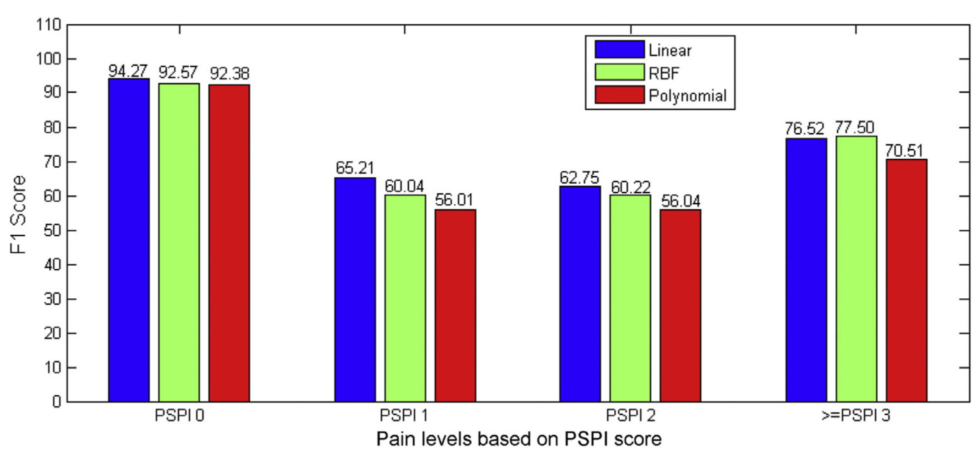

Fig. 7. Comparison of F1 score using different SVM kernels for 4-level pain intensity detection.

used for testing. The average of all the iterations is calculated to compute leave one subject out cross validation accuracy.

We have adopted linear kernel of SVM for classification of 4 levels of pain. The confusion matrix for the same is presented in Table 4.

The confusion matrix infers that the classifier is capable to classify pain with PSPI 0 and PSPI $\geq$ 3 with less confusion as compared to the pain with PSPI 2 and PSPI 3. To compare the linear kernel with other kernels like RBF and polynomial, we performed experiments with these kernels also. To measure the performance the proposed approach with different kernels, we computed precision, recall and F1 score for all the three kernels by using the following equations:

$$Precision = \frac{TP}{TP+FP}, \quad Recall = \frac{TP}{TP+FN} \tag{23}$$

$$F1\ score = 2\,\frac{Precision \times Recall}{Precision + Recall} \tag{24}$$

where *TP* represents true positive, *FP* represents False positive, *TN* is True Negative and *FN* refers to False Negative. The comparative graph is presented in Figs. 5–7.

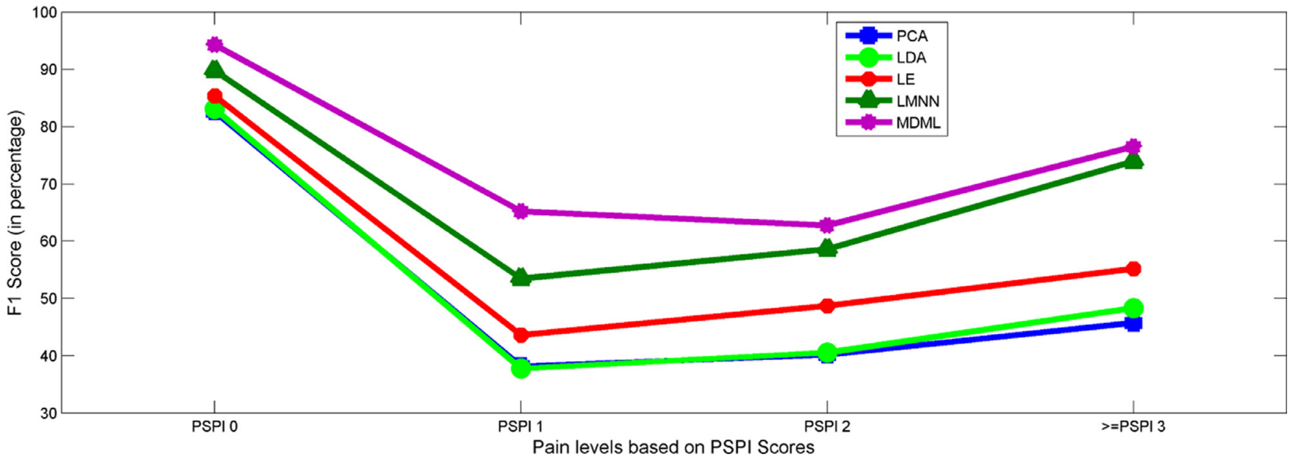

Fig. 8. Comparison of dimension reduction methods with the proposed MDML for 4-level pain intensity detection.

Table 5
Performance comparison of the proposed approach with another approach mentioned in literature.

	Precision		F1 score	
	Proposed approach	Hammal and Cohn [19]	Proposed approach	Hammal and Cohn [19]
PSPI 0	95	65	94	57
PSPI 1	63	37	65	67
PSPI 2	57	35	63	40
PSPI 3	84	70	77	60

To prove the efficacy of the proposed MDML method, we have compared the performance of the proposed approach with other popular dimension reduction methods like principal component analysis (PCA), Linear Discriminant Analysis (LDA), Laplacian Eigenmaps (LE) and Large Margin Nearest Neighborhood metric learning (LMNN). The performance of the above mentioned methods is compared to MDML in terms of F1 score and is presented in Fig. 8. The performance of the proposed MDML is better than another dimension reduction methods depicting its capability for feature extraction.

To further prove the efficiency of the proposed method over the method proposed in literature, it is compared with the method proposed by Hammal and Cohn [19]. The performance parameters are compared in Table 5.

7. Conclusion and perspectives

We have proposed a novel approach to solve the problem of pain and pain intensity detection. The facial features were extracted using popular facial feature descriptors, LBP, HOG and Gabor features. To obtain complementary information from feature descriptors, we introduced MDML algorithm. The pain detection results are obtained using three different kernels: linear, RBF and polynomial. The linear kernel results in better detection of pain as compared with other kernels. To have an impartial comparison with the methods proposed in literature, we have presented the results for pain detection. To finally prove the efficacy of the proposed MDML method, we have compared the results with other popular dimension reduction methods. The proposed approach results in 90% accuracy which is comparable to the recent approaches mentioned in Table 1.

In the present work, we employed pixel based descriptors to obtain the localized feature set. In the future, we are interested in applying the proposed method to combine geometry, and pixel based features so as to further improve the efficiency of the system. Moreover, our method depends on static information from video frames and thus leaves timing cues of pain and head motion untouched. Utilizing the dynamics associated with pain sequences along with present static approach may lead to a more effective system in the detection or estimation of 16 levels of pain intensity by doing regression analysis. Another interesting work that may be done is to use deep learning for feature extraction for pain classification.

Acknowledgment

The authors would like to acknowledge the financial support of UGC (grant no. SR:32-17/2006(SR)) and DST (grant no. SR/FTP/S-61/2005) Government of India.

References

[1] H.M. McCormack, D.J. de L Horne, S. Sheather, Clinical applications of visual analogue scales: a critical review, Psychol. Med. 18 (4) (1988) 1007–1019.
[2] M.P. Jensen, P. Karoly, S. Braver, The measurement of clinical pain intensity: a comparison of six methods, Pain 27 (1) (1986) 117–126.
[3] C.S. Burckhardt, K.D. Jones, Adult measures of pain: the McGill Pain Questionnaire (MPQ), Rheumatoid Arthritis Pain Scale (RAPS), Short-Form McGill Pain Questionnaire (SF-MPQ), Verbal Descriptive Scale (VDS), Visual Analog Scale (VAS), and West Haven-Yale Multidisciplinary Pain Inventory (WHYMPI), Arthritis Care Res. 49 (S5) (2003) S96–S104.
[4] M.B. Ferraz, M. Quaresma, L. Aquino, E. Atra, P. Tugwell, C. Goldsmith, Reliability of pain scales in the assessment of literate and illiterate patients with rheumatoid arthritis., J. Rheumatol. 17 (8) (1990) 1022–1024.
[5] A.C.d. C. Williams, H.T.O. Davies, Y. Chadury, Simple pain rating scales hide complex idiosyncratic meanings, Pain 85 (3) (2000) 457–463.
[6] A.B. Ashraf, S. Lucey, J.F. Cohn, T. Chen, Z. Ambadar, K.M. Prkachin, P.E. Solomon, The painful face–pain expression recognition using active appearance models, Image Vis. Comput. 27 (12) (2009) 1788–1796.
[7] D.L. Wong, C. Baker, Pain in children: comparison of assessment scales, Pediatr. Nurs. 14 (1) (1988) 9–17.
[8] P. Lucey, J.F. Cohn, K.M. Prkachin, P.E. Solomon, I. Matthews, Painful data: the UNBC-McMaster shoulder pain expression archive database, in: Proceedings of the 2011 IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011), IEEE, 2011, pp. 57–64.
[9] D.C. Turk, R. Melzack, The measurement of pain and the assessment of people experiencing pain. (2001).
[10] G. Littlewort, M.S. Bartlett, I.R. Fasel, J. Susskind, J.R. Movellan, Dynamics of facial expression extracted automatically from video, Image Vis. Comput. 24 (2006) 615–625, doi:10.1016/j.imavis.2005.09.011.
[11] B. Schmand, J. Lindeboom, S. Schagen, R. Heijt, T. Koene, H. Hamburger, Cognitive complaints in patients after whiplash injury: the impact of malingering, J. Neurol. Neurosurg. Psychiatry 64 (3) (1998) 339–343.
[12] H.D. Hadjistavropoulos, K.D. Craig, T. Hadjistavropoulos, G.D. Poole, Subjective judgments of deception in pain expression: accuracy and errors, Pain 65 (2) (1996) 251–258.
[13] P. Lucey, J.F. Cohn, K.M. Prkachin, P.E. Solomon, S. Chew, I. Matthews, Painful monitoring automatic pain monitoring using the UNBC-McMaster shoulder pain expression archive database, Image Vis. Comput. 30 (3) (2012) 197–205.

[14] K.M. Prkachin, P.E. Solomon, The structure, reliability and validity of pain expression: evidence from patients with shoulder pain, Pain 139 (2) (2008) 267–274.
[15] P. Lucey, J.F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, K.M. Prkachin, Automatically detecting pain in video through facial action units, IEEE Trans. Syst. Man Cybern. Part B: Cybern. 41 (3) (2011) 664–674.
[16] P. Lucey, J. Cohn, S. Lucey, I. Matthews, S. Sridharan, K.M. Prkachin, Automatically detecting pain using facial actions, in: Proceedings of the 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, ACII 2009, IEEE, 2009, pp. 1–8.
[17] P. Werner, A. Al-Hamadi, R. Niese, S. Walter, S. Gruss, H.C. Traue, Towards pain monitoring: facial expression, head pose, a new database, an automatic system and remaining challenges, in: Proceedings of the British Machine Vision Conference, 2013.
[18] Z. Hammal, M. Kunz, Pain monitoring: a dynamic and context-sensitive system, Pattern Recognit. 45 (4) (2012) 1265–1280.
[19] Z. Hammal, J.F. Cohn, Automatic detection of pain intensity, in: Proceedings of the 14th ACM International Conference on Multimodal Interaction, ACM, 2012, pp. 47–52.
[20] S. Kaltwang, O. Rudovic, M. Pantic, Continuous pain intensity estimation from facial expressions, in: Advances in Visual Computing, Springer, 2012, pp. 368–377.
[21] R.A. Khan, A. Meyer, H. Konik, S. Bouakaz, Pain detection through shape and appearance features, in: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), 2013, IEEE, 2013, pp. 1–6.
[22] S. Brahnam, C.-F. Chuang, F.Y. Shih, M.R. Slack, Machine recognition and representation of neonatal facial displays of acute pain, Artif. Intell. Med. 36 (3) (2006) 211–222.
[23] B. Gholami, M. Haddad, A.R. Tannenbaum, Agitation and pain assessment using digital imaging, in: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2009, IEEE, 2009, pp. 2176–2179.
[24] G.C. Littlewort, M.S. Bartlett, K. Lee, Automatic coding of facial expressions displayed during posed and genuine pain, Image Vis. Comput. 27 (12) (2009) 1797–1803.
[25] K. Sikka, A. Dhall, M. Bartlett, Weakly supervised pain localization and classification with multiple segment learning, Image Vis. Comput. 32 (10) (2014) 659–670.
[26] O. Rudovic, V. Pavlovic, M. Pantic, Context-sensitive dynamic ordinal regression for intensity estimation of facial action units, IEEE Trans. Pattern Anal. Mach. Intell. 37 (5) (2015) 944–958, doi:10.1109/TPAMI.2014.2356192.
[27] J. Yu, M. Wang, D. Tao, Semisupervised multiview distance metric learning for cartoon synthesis, IEEE Trans. Image Process. 21 (11) (2012) 4636–4648.
[28] J. Lu, X. Zhou, Y.-P. Tan, Y. Shang, J. Zhou, Neighborhood repulsed metric learning for kinship verification, IEEE Trans. Pattern Anal. Mach. Intell. 36 (2) (2014) 331–345.
[29] Y. Li, S.M. Mavadati, M.H. Mahoor, Y. Zhao, Q. Ji, Measuring the intensity of spontaneous facial action units with dynamic bayesian network, Pattern Recognit. 48 (11) (2015) 3417–3427, doi:10.1016/j.patcog.2015.04.022.
[30] Y.-L. Tian, T. Kanade, J.F. Cohn, Evaluation of Gabor-wavelet-based facial action unit recognition in image sequences of increasing complexity, in: Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on, IEEE, 2002, pp. 229–234.
[31] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, vol. 1, 2005, pp. 886–893, doi:10.1109/CVPR.2005.177.
[32] S. Mavadati, M. Mahoor, K. Bartlett, P. Trinh, J. Cohn, DISFA: a spontaneous facial action intensity database, IEEE Trans. Affect. Comput. 4 (2) (2013) 151–160, doi:10.1109/T-AFFC.2013.4.
[33] X. Feng, M. Pietikainen, A. Hadid, Facial expression recognition with local binary patterns and linear programming, Pattern Recognit. Image Anal. C/C Raspozn. Obraz. I Anal. Izobr. 15 (2) (2005) 546.
[34] C. Shan, S. Gong, P.W. McOwan, Facial expression recognition based on local binary patterns: a comprehensive study, Image Vis. Comput. 27 (6) (2009) 803–816.
[35] S. Moore, R. Bowden, Local binary patterns for multi-view facial expression recognition, Comput. Vis. Image Underst. 115 (4) (2011) 541–558.
[36] V.N. Vapnik, V. Vapnik, Statistical Learning Theory, vol. 2, Wiley, New York, 1998.
[37] C.J.C. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discov. 2 (1998) 121–167.
[38] M.-I. Faraj, J. Bign, Synergy of lip-motion and acoustic features in biometric speech and speaker recognition, IEEE Trans. Comput. 56 (2007) 1169–1175, doi:10.1109/TC.2007.1074.
[39] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 27:1–27:27.Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm
[40] N. Djuric, L. Lan, S. Vucetic, Z. Wang, BudgetedSVM: a toolbox for scalable SVM approximations, J. Mach. Learn. Res. 14 (1) (2013) 3813–3817.
[41] P. Lucey, J. Cohn, S. Lucey, S. Sridharan, K.M. Prkachin, Automatically detecting action units from faces of pain: Comparing shape and appearance features, in: Proceeding of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops 2009 CVPR Workshops, 2009, pp. 12–18.
[42] C. Florea, L. Florea, C. Vertan, Learning pain from emotion: transferred hot data representation for pain intensity estimation, in: ECCV workshop on ACVR, vol. 8927, 2014, pp. 778–790.
[43] N. Rathee, D. Ganotra, A novel approach for pain intensity detection based on facial feature deformations, J. Vis. Comun. Image Represent 33 (C) (2015) 247–254.