Automatic Estimation of Self-Reported Pain by Interpretable Representations of Motion Dynamics

Benjamin Szczapa, Mohamed Daoudi, Stefano Berretti, Pietro Pala, Alberto Del Bimbo, Zakia Hammal

Automatic Estimation of Self-Reported Pain by Interpretable Representations of Motion Dynamics

Benjamin Szczapa*‡, Mohamed Daoudi†, Stefano Berretti‡, Pietro Pala‡, Alberto Del Bimbo‡ and Zakia Hammal§
*Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France
†IMT Lille Douai, Univ. Lille, CNRS, UMR 9189 CRIStAL, F-59000 Lille, France
‡Department of Information Engineering, University of Florence, Italy
§Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA

***Abstract*—We propose an automatic method for pain intensity measurement from video. For each video, pain intensity was measured using the dynamics of facial movement using 66 facial points. Gram matrices formulation was used for facial points trajectory representations on the Riemannian manifold of symmetric positive semi-definite matrices of fixed rank. Curve fitting and temporal alignment were then used to smooth the extracted trajectories. A Support Vector Regression model was then trained to encode the extracted trajectories into ten pain intensity levels consistent with the Visual Analogue Scale for pain intensity measurement. The proposed approach was evaluated using the UNBC McMaster Shoulder Pain Archive and was compared to the state-of-the-art on the same data. Using both 5-fold cross-validation and leave-one-subject-out cross-validation, our results are competitive with respect to state-of-the-art methods.**

I. Introduction

Pain is an unpleasant sensory and emotional experience associated with actual or potential tissue damage and caused by illness or injury [1]. The assessment of pain is accomplished primarily through subjective self-report using the Visual Analog Scale (VAS) or the Numerical Rating Scale (NRS) [2]. The most commonly used scale in clinical assessment is the VAS [3], [4], [5], [6]. However, while useful, self-reported pain is difficult to interpret and may be impaired or, in some circumstances, not possible to obtain (*e.g.*, for children or patients requiring breathing assistance).

Significant efforts have been made in human behavioral studies to identify reliable and valid facial indicators of pain [7], [8], [9], [10]. In these studies, pain expression and intensity were reliably characterized at the frame level by the activation of a set of anatomical facial actions using the manual Facial Action Coding System (FACS) [11]. However, manual FACS based pain assessment requires over a hundred hours of training for FACS certification, and approximately an hour or more to manually annotate a minute of video. The intensive time required to annotate videos using the FACS makes it ill suited for real-time application and clinical use. A powerful alternative to manual annotation is the automatic and objective assessment of pain from facial expression [12].

The last decade has witnessed an increasing effort to address the need for an automatic, objective, and efficient measurement of pain from video. Most previous efforts in automatic assessment of pain have focused on pain detection or pain intensity estimation at the frame-level (see [12] and [13] for a detailed review of previous efforts on the topic).

A few recent exceptions [14], [15], have investigated video based pain intensity measurement consistent with self-reported VAS. The VAS is a self-reported pain scale that indicates pain experience on a 0 to 10 scale (where 0 is for "no pain" and 10 is for "worse possible pain"). For instance, using the UNBC-McMaster Shoulder Pain Archive database [16], Martinez *et al.* [14] proposed a two step learning approach to estimate pain consistent with the VAS. The authors employed a Recurrent Neural Network (RNN) to first estimate pain score at frame level. The estimated scores were then fed into a personalized Hidden Conditional Random Fields (HCRF) to estimate pain score at the video level consistent with the VAS. Using the same pain database, Liu *et al.* [15] proposed a two-stage personalized model, named DeepFaceLIFT, for automatic estimation of the self-reported VAS score. The authors used a Neural Network and Gaussian process regression model and combined facial expression and a set of hand-crafted personal features for pain score measurement at the video level.

Previous efforts for video based pain assessment used artificial neural networks to first estimate pain score at the frame level before combining them to estimate pain score at the video level. We propose to extend previous work in video based assessment of pain intensity by estimating VAS score directly from video using a geometry based approach. To capture changes in the dynamics of facial movement relevant to pain expression, we propose an original framework based on Gram matrix computation and trajectory modeling on the Riemannian manifold of symmetric positive-semidefinite (PSD) matrices [17]. With this representation, pain estimation is modeled as a problem of computing similarity between trajectories on the manifold using Support Vector Regression [18].

II. Face Representation

We propose a video based measurement of pain intensity scores using the dynamics of facial movement. Figure 1 shows an overview of the proposed approach. Given a set of n_{seq} sequences, we first build the trajectories on the manifold $\mathcal{S}^+(d,m)$ from the Gram matrices of each frame of each sequence using the landmark configurations (and their

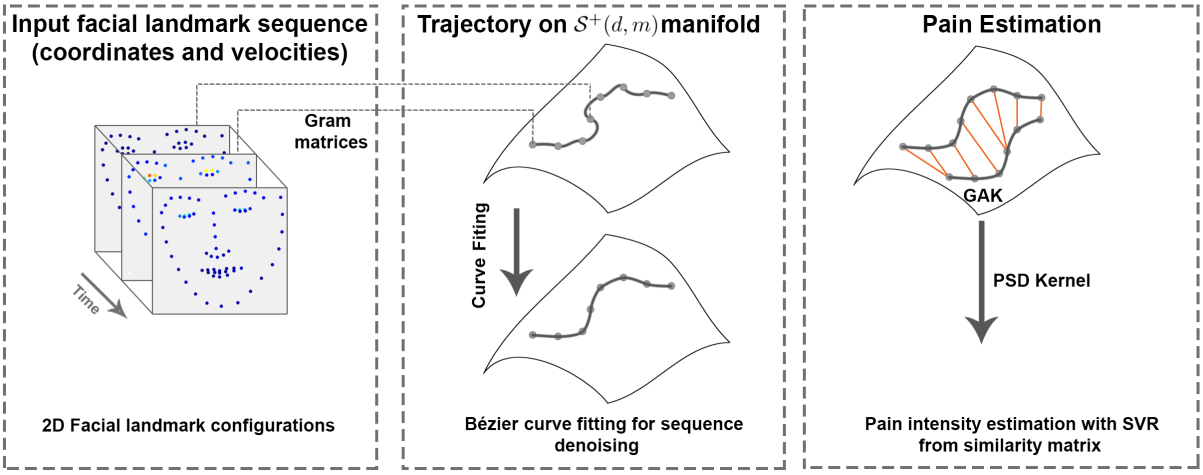

Fig. 1: Overview of the proposed approach: (left plot) First, facial landmarks are detected using Active Appearance Model (AAM) on each video frame and velocities are computed as the displacement of the coordinates between two consecutive frames. Then Gram matrices are computed from the combination of the landmark coordinates and velocities. These matrices delineate a trajectory on the $\mathcal{S}^+(d,m)$ manifold; (middle plot) We apply a curve fitting algorithm to the trajectory for smoothing and noise reduction; (right plot) The Global Alignment Kernel (GAK) is then used to align the trajectories on the manifold, which results in a similarity score between the trajectories. Finally, we use the kernel generated from GAK with SVR to estimate the pain intensity.

velocities) as input features. We then compute the distances between all the trajectories and build a kernel K that contains all the similarity scores after aligning the trajectories with the Global Alignment Kernel (GAK). Finally, we estimate pain intensity score based on the similarity matrix.

A. Facial Shape Representation

Given an image sequence Is, we represent the dynamics of facial movements with a time series formed by the coordinates (x, y) of n tracked facial landmarks. At a generic time (frame) f, facial expression is represented by a configuration $Z \in \mathrm{R}^{n\times 2}$ composed of n tracked facial landmarks $p_i = (x_i, y_i)$, where $i \in \{1, \ldots, n\}$. Thus, an image sequence is represented by a sequence of configuration matrices $Is = \{Z_1, \ldots, Z_f, \ldots Z_\tau\}$ with f denoting the frame number and τ the number of frames of the sequence Is. In addition to landmark coordinates, we compute for each landmark p_i its velocity as the magnitude of the displacement between two consecutive landmark configurations Z_f and Z_{f+1}. We denote the velocity matrix at frame F as $V_F = Z_{f+1} - Z_f \in \mathrm{R}^{n\times 2}$, with $F \in \{1, \ldots, \tau - 1\}$. The final facial representation R is the concatenation of the configuration matrix Z and the velocity matrix V, where $R = [Z; V] \in \mathrm{R}^{2n\times 2}$.

We aim to measure the dynamic changes of the curves made of landmark configurations, while remaining invariant to rigid transformations like rotations and translations. Invariance to rigid transformation within each frame is obtained by computing coordinates of landmarks (and their velocities) as offsets with respect to the center of the face that is measured as the arithmetic mean of the landmarks:

$$(\bar{x_i}, \bar{y_i}) = \frac{1}{n}\sum_{i=1}^{n}(x_i, y_i)\ . \tag{1}$$

We denote A the normalized facial configuration of matrix R. Similarly to [17], [19], this representation is further refined by extracting the Gram matrix G, which is the inner product of each facial configuration matrix as:

$$G = AA^T = \langle p_i, p_j \rangle\ , \quad 1 \leq i, j \leq 2n\ . \tag{2}$$

In the following, we denote $m = 2n$ the size of the facial configuration matrix for simplicity.

B. Riemannian Geometry of Gram Matrix

Given that each Gram matrix represents the landmarks configuration at the frame level, we propose (1) a geometry of space to model the dynamic changes of landmarks during a video sequence, and (2) a metric that allows to compute the distance between consecutive Gram matrices. In the following, we present a general metric that works for both 2D or 3D data and an optimized metric for 2D data.

Gram matrices are $m \times m$ positive-semidefinite (PSD) matrices of rank smaller than or equal to d (in our case the rank is always equal to d). In this representation, d is the dimensionality of the space where each landmark lies (i.e., $d = 2$ for 2D landmarks and $d = 3$ for 3D landmarks). We consider here the Riemannian geometry of the space $\mathcal{S}^+(d,m)$ of $m \times m$ positive-semidefinite matrices of rank d. This Riemannian geometry has been studied in [20], [21], [22], [23], [24], [25] and used in [26], [27], [28], [29]. In order to develop algorithms on the manifold, we resort to first order

local approximations on the manifold. These approximations are called the *tangent spaces*. This requires two fundamental tools: the Riemannian *logarithm*, that maps points from the manifold to the tangent space, and the Riemannian *exponential* that allows us to map tangent vectors from the tangent space to the manifold.

We consider here the manifold of $\mathcal{S}^+(d,m)$ as the quotient manifold $\mathbb{R}_*^{m\times d}/\mathcal{O}_d$, where $\mathbb{R}_*^{m\times d}$ is the set of full-rank $m \times d$ matrices and $\mathcal{O}_d$ is the orthogonal group in dimension d. The identification of $\mathcal{S}^+(d,m)$ with the quotient $\mathbb{R}_*^{m\times d}/\mathcal{O}_d$ comes from the following observation: Any PSD matrix $G \in \mathcal{S}^+(d,m)$ can be factorized as $G = AA^T$, with $A \in \mathbb{R}_*^{m\times d}$. However, this factorization is not unique, as any matrix $\tilde{A} := AQ$, with $Q \in \mathcal{O}_d$, satisfies $\tilde{A}\tilde{A}^T = AQQ^TA^T = G$. The two points A and $\tilde{A}$ are thus *equivalent* with respect to this factorization, and the set of equivalent points:

$$A\mathcal{O}_d := \{AQ | Q \in \mathcal{O}_d\} ,$$

is called the equivalence class associated to G. The quotient manifold $\mathbb{R}_*^{m\times d}/\mathcal{O}_d$ is defined as the set of equivalence classes. The mapping $\pi : \mathbb{R}_*^{m\times d} \to \mathbb{R}_*^{m\times d}/\mathcal{O}_d$, between points and their equivalence class, induces a Riemannian metric on the quotient manifold from the Euclidean metric in $\mathbb{R}_*^{m\times d}$. This metric results in the following distance between PSD matrices [22]:

$$d(G_i, G_j) = \mathrm{tr}(G_i) + \mathrm{tr}(G_j) - 2\mathrm{tr}\left(\left(G_i^{\frac{1}{2}} G_j G_i^{\frac{1}{2}}\right)^{\frac{1}{2}}\right) . \quad (3)$$

This distance can be expressed in terms of the facial configurations $A_i, A_j \in \mathbb{R}_*^{m\times d}$ as follows:

$$d(G_i, G_j) = \min_{Q\in\mathcal{O}_d} \|A_jQ - A_i\|_F , \quad (4)$$

where $\|.\|_F$ is the Frobenius norm. The optimal solution is $Q^* := VU^T$, where $A_i^TA_j = U\Sigma V^T$ is a singular value decomposition.

In the specific case of 2D landmarks, when $d = 2$, the distance can be reformulated. Considering $G_i, G_j \in \mathcal{S}^+(2,m)$ to be two Gram matrices obtained from facial configurations $A_i, A_j \in \mathbb{R}^{m\times 2}$, the Riemannian distance (3) can be expressed as:

$$d(G_i, G_j) = \mathrm{tr}(G_i) + \mathrm{tr}(G_j) - 2\sqrt{(a+d)^2 + (c-b)^2} , \quad (5)$$

where $A_i^TA_j = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$. The interested readers can find the proof of this expression in [17, §9].

III. Representation of Face Dynamics

A. Trajectory Modeling

The dynamic changes of facial landmarks movement originate trajectories on the Riemannian manifold of positive-semidefinite matrices of fixed rank. More specifically, we fit a curve β_G to a sequence of facial configurations $\{A_0, \ldots, A_\tau\}$ represented by their corresponding Gram matrices $\{G_0, \ldots, G_\tau\}$ in $\mathcal{S}^+(d,m)$. This curve enables us to model the spatio-temporal evolution of the elements on $\mathcal{S}^+(d,m)$. Modeling a sequence of landmarks as a piecewise-geodesic curve on $\mathcal{S}^+(d,m)$ showed very promising results when the data are well acquired, *i.e.*, without tracking errors or missing data. To account for both missing data and tracking errors, we rely on a more recent curve fitting algorithm: fitting by composite cubic blended curves [30, §5]. Specifically, given a set of points $\{G_0, \ldots, G_\tau\} \in \mathcal{S}^+(d,m)$ associated to times $\{t_0, \ldots, t_\tau\}$, with $t_i := i$, the curve β_G, defined on the interval $[0, \tau]$, is defined as:

$$\beta_G(t) := \gamma_i(t - i), \qquad t \in [i, i+1] , \quad (6)$$

where each curve γ_i is obtained by blending together fitting cubic Bézier curves computed on the tangent spaces of the data points d_i and d_{i+1} (represented by Gram matrices on the manifold).

These fitting cubic Bézier curves depend on a parameter λ, allowing us to balance two objectives: (1) proximity to the data points at the associated time instants, and (2) regularity of the curve (measured in terms of mean square acceleration). A high value of λ results in a curve with possibly high acceleration that almost interpolates the data, while taking $\lambda \to 0$ results in a smooth function approximating the original trajectory.

B. Global Alignment

As explained in the previous section, we represent a sequence as a trajectory of Gram matrices in $\mathcal{S}^+(d, mn)$. Because videos could be of different duration (i.e., in our case video sequences of pain), the length of corresponding trajectories represented in this manifold can be different. A commonly used method to compute the similarity between trajectories with different length is Dynamic Time Warping (DTW). However, DTW does not define a proper metric and cannot be used to derive a valid positive-definite kernel. This would hamper the use of many approaches (including Support Vector Regression) to learn the mapping between trajectories in $\mathcal{S}^+(d,m)$ and pain intensity. Cuturi *et al.* [31] proposed the Global Alignment Kernel (GAK) to address non-positive definite kernel defined by DTW. GAK allows to derive a valid positive-definite kernel when aligning two time series. As opposed to the DTW, the GAK generated kernel, that is the similarity matrix between all the sequences, can be used directly with Support Vector Regression. In fact, the kernels built with DTW do not show favorable positive definiteness properties as they rely on the computation of an optimum rather than the construction of a feature map. In terms of complexity, similar to naive implementation of DTW, the computational complexity of the GAK kernels is quadratic.

Let us now consider $G^1 = \{G_0^1, \cdots, G_{\tau_1}^1\}$ and $G^2 = \{G_0^2, \cdots, G_{\tau_2}^2\}$, two trajectories of Gram matrices. Given a metric to compute the distance between two elements of each sequence, we propose to compute the matrix D of size $\tau_1 \times \tau_2$, where each $D(i,j)$ is the distance between two elements of the sequences, with $1 \le i \le \tau_1$ and $1 \le j \le \tau_2$:

$$D(i,j) = d(G_i^1, G_j^2) . \quad (7)$$

The kernel $\tilde{k}$ can now be computed using the halved Gaussian Kernel on this same matrix D. Therefore, the kernel $\tilde{k}$ can be defined as:

$$\tilde{k}(i,j) = \frac{1}{2} * exp\left(-\frac{D(i,j)}{\sigma^2}\right) . \quad (8)$$

As reported in [31], we can redefine our kernel as:

$$k(i,j) = \frac{\tilde{k}(i,j)}{(1-\tilde{k}(i,j))} . \quad (9)$$

This strategy guarantees that the kernel is positive-semidefinite and can be used in its own. Finally, we can compute the similarity score between the two trajectories G^1 and G^2. This computation is performed in quadratic complexity, like DTW. To do so, we define a new matrix M that contains the path to the similarity between our two sequences. We define M as a zeros matrix of size $(\tau_1+1) \times (\tau_2+1)$ and $M_{0,0} = 1$. Computing the terms of M is done using Theorem 2 in [31, §2.3]:

$$M_{i,j} = (M_{i,j-1} + M_{i-1,j-1} + M_{i-1,j}) * k(i,j) . \quad (10)$$

The similarity score between the trajectories G^1 and G^2 is given by the value at $M_{(\tau_1+1),(\tau_2+1)}$.

IV. Pain Estimation with Support Vector Regression

We build a new matrix K of size $n_{seq} \times n_{seq}$, where n_{seq} is the number of sequences in the dataset used to test our method. This symmetric matrix contains all the similarity scores between all the sequences of the dataset. This matrix is built with values computed from positive-semidefinite kernel, meaning that it is a positive-semidefinite matrix itself. Now that we have a valid and positive-semidefinite kernel K, as demonstrated by Cuturi *et al.* [31], we can use it directly as a valid kernel for classification. To estimate pain intensity score (i.e., self-reported VAS scores), we use a Support Vector Regression (SVR) model. To train our SVR model, we give as input a training set that is a part of our kernel K containing the similarity scores between all training trajectories. This part of the kernel, containing the training set, is also positive-semidefinite by definition. We also give a vector containing the labels for the trajectories in our training kernel. Because pain scores are continuous, to test the performance of our method, we compute the Mean Absolute Error (MAE) between the estimated pain scores and the ground truth (i.e., self-reported VAS pain scores). The MAE is computed as follows:

$$\text{MAE} = \frac{1}{n_{seq}} \sum_{i=1}^{n_{seq}} |y_i - x_i| , \quad (11)$$

where n_{seq} is the number of sequences in the dataset, y_i is the ground truth (i.e., self-reported VAS pain score), and x_i is the predicted pain score.

V. Experimental Results

The UNBC-McMaster Shoulder Pain Archive [16] was used to evaluate the reliability of the proposed approach for pain intensity measurement from the dynamics of facial landmark sequences. We used MatLab for the code and the Manopt library [32].

A. The UNBC-McMaster Shoulder Pain Archive

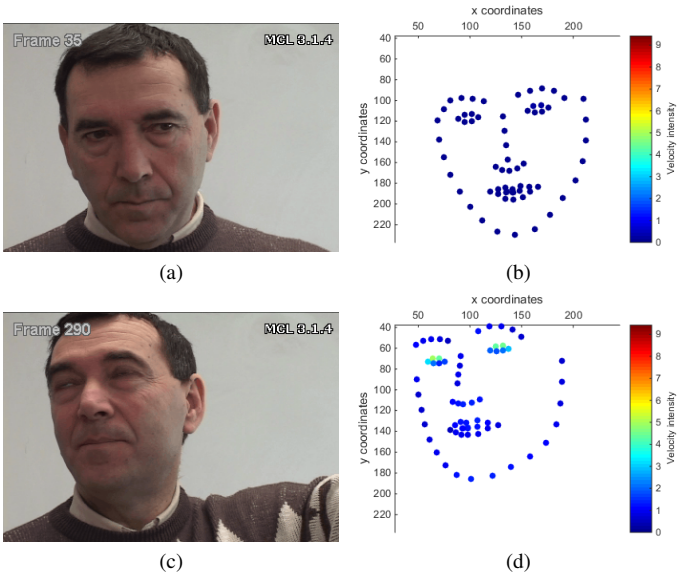

Fig. 2: Example images from the UNBC-McMaster Shoulder Pain Archive in (a) and (c). In (b) and (d) their corresponding landmark coordinates and velocities, respectively (best viewed in color) [16].

The UNBC-McMaster Shoulder Pain Archive dataset [16] is a widely used dataset for pain expression recognition and intensity estimation. The dataset contains 200 facial videos of 25 different subjects performing a series of active and passive range-of-motion of their affected and unaffected shoulders. Each video sequence is annotated for pain intensity score using three self-reported scales (including the VAS) and an Observer Pain Rating scale. The sequences are also annotated at the frame-level using the manual FACS (Facial Action Coding System). Figure 2 shows two images from a sequence of the dataset with their corresponding facial landmark representations and velocities. Our goal is to estimate pain intensity scores consistent with the VAS. Table I shows the distribution of the VAS scores across the dataset. We can observe that the number of sequences are not the same for all the VAS scores.

Figure 3 shows the number of sequences per subject. We can observe some disparity between the subjects that may represent a challenge for training as the number of sequences used will not be consistent across the dataset.

B. Evaluation protocols

We used three different protocols to evaluate the proposed method: Leave-One-Sequence-Out cross validation, Leave-One-Subject-Out cross validation, and 5-fold cross validation.

TABLE I: Distribution of the VAS pain scores in the UNBC-McMaster Shoulder Pain Archive

VAS Score	Number of Sequences
0	35
1	42
2	24
3	20
4	21
5	11
6	11
7	6
8	18
9	10
10	2

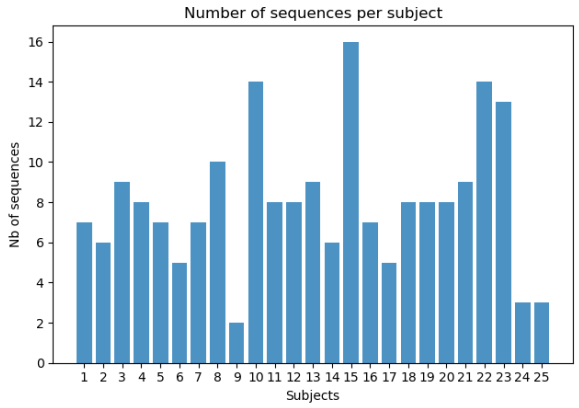

Fig. 3: Number of sequences per subjects in the UNBC-McMaster Shoulder pain archive dataset.

a) Leave-One-Sequence-Out cross validation protocol: In this protocol, training and testing are performed on different sequences. For each round, we use all sequences of the dataset but one for training. and the remaining sequence for testing. That is, data from the same subject can be used during the training and the testing phase as there are at least two sequences per subject in the dataset. Therefore, this protocol is sequence-independent, but not subject-independent. We use this protocol as a baseline for our approach.

b) Leave-One-Subject-Out cross validation protocol: In this second protocol, for each round, we use all the sequences from all subjects but one for training, and the remaining subject for testing (no overlap between the training subjects and the test subject). We perform this operation for all the subjects (*i.e.*, 25 rounds) in the dataset, so that each subject is used for testing once.

c) 5-fold cross validation protocol: This third protocol is similar to the Leave-One-Subject-Out cross validation protocol, but instead of taking only the sequences of one subject at a time for testing, we take all the sequences of five subjects for testing and the remaining sequences for the training. To choose the five subjects for testing, we choose the five first subjects in the dataset, then the five next subjects and so on until all the subjects are used for testing.

The advantage of using cross validation is to prevent from having performance results that are due to the chance (all data will be used to train and test the proposed method). The average across all folds is more representative of the whole dataset.

C. Pain estimation from landmark coordinates and velocities

Our goal is to estimate the VAS pain score for each sequence of the dataset. We test our method with the three protocols described above and report the results in Table II. For each protocol, we fix the value of the curve fitting parameter lambda to 1000 and the Gaussian kernel in the sequence alignment sigma to 0.8 (see Table II). *Protocol* indicates the protocol used for training and testing our method; *% of frames* indicates the percentage of frames used from each sequence for training and testing; *MAE* indicates the Mean Absolute Error and *RMSE* the Root Mean Square Error of our estimation (see Table II).

TABLE II: Results of our method with the three different protocols.

Protocol	% of frames	MAE	RMSE
Leave-One-Sequence-Out	25%	**2.3166**	3.1459
	100%	2.5291	3.3263
Leave-One-Subject-Out cross validation	25%	**2.523**	3.2692
	100%	2.9176	3.5133
5-fold cross validation	25%	**2.4365**	3.147
	100%	2.7944	3.5088

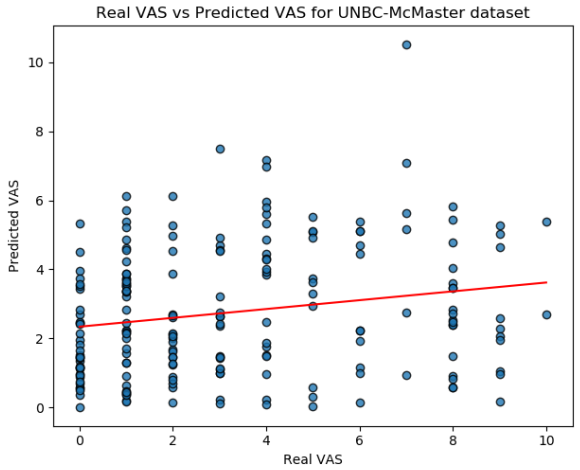

Fig. 4: Distribution of the predicted VAS values compared with the real VAS using the 5-fold cross validation protocol with 25% of frames. The red line is a least-square fitting of the predicted values.

From Table II, we notice that in every cases, the MAE is lower when we down-sample 1 frame each 4 frames, leading to 25% of the frames available for pain assessment. This is due to the high amount of non-pain frames that are present in the dataset. We also notice that the best MAE we obtained is 2.3166 with the Leave-One-Sequence-Out protocol. This result is expected as this protocol is not subject-independent and sequences of the same subject can be used for both training and

testing. The second best MAE we obtained is 2.4365, using the 5-fold cross validation protocol. We report the RMSE as a second measure of the error of our estimation. Results show the same trend as the MAE with the best RMSE observed for the Leave-One-Sequence-Out protocol.

Similar to [15], we present in Fig. 4 the distribution of the predicted VAS score against the true reported VAS. This allows us to observe that our approach is capable of predicting many low VAS scores and can have difficulty in estimating higher values.

D. Comparison with state-of-the-art

We compared our approach to the two state-of-the-art methods for VAS pain intensity measurement from video (see Table III). Here, we report the best results for DeepFaceLIFT [15] that only uses the VAS as training labels as the authors also present results while combining VAS and OPR labels. They obtained a MAE of 2.30 using a 5-fold cross validation protocol. Our results are close to theirs, while only using a geometry based formulation of facial landmark dynamics (meaning that our method is less expensive as we do not have to train a neural network). Our results are comparable to RNN-HCRF [14] results, as they obtain a MAE of 2.46, though using a different protocol. In fact, in the results for RNN-HCRF, data have been randomly split by taking the sequences of 15 subjects for training and the sequences of 10 subjects for testing. It is also important to highlight that in RNN-HCRF the face appearance is also used, while our method only considers the shape of the face.

One of the advantage of our method over the two approaches presented here is the explainability of the results. As our method is based on facial landmarks and modeling of their dynamics as a trajectory on the manifold, it is possible to interpret the predicted VAS score for a new observation based on distances of this observation to train trajectories. This makes it possible to support the explanation of results on a much more solid base than would be by using alternative models for prediction, such as those based on deep neural networks. Interpretability is also very important in a day-to-day use by practitioners as they can better estimate the pain from the different parts of the face.

TABLE III: Comparison of our method with state-of-the-art results

Method	**Protocol**	**Labels for training**	**MAE**
DeepFaceLift [15]	5-fold cross validation	VAS	2.30
RNN-HCRF [14]	random split	VAS & PSPI	2.46
Ours	5-fold cross validation	VAS	2.4365

VI. Conclusion

We proposed a method based on facial landmarks dynamics to estimate pain intensity from video. Our approach shows competitive results with respect to state-of-the-art methods on the UNBC-McMaster Shoulder Pain Archive, while only considering the shape of the face. Future work will focus on the combination of facial shape and appearance as well as the inclusion of other pain scales such as the observer pain rating scale to further improve the pain scores estimation.

Acknowledgment

Zakia Hammal's effort was supported by the National Institute of Nursing Research of the National Institutes of Health under Awards Number R21NR016510 and R01NR018451. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health. The proposed work was also partially supported by the French State, managed by the National Agency for Research (ANR) under the Investments for the future program with reference ANR-16-IDEX-0004 ULNE. We thank Prof. J-C. Alvarez Paiva from University of Lille for fruitful discussions on the formulation of the distance between $n \times 2$ landmark configurations in Eq. (5). We also thank Dr. Pierre-Yves Gousenbourger from Université Catholique de Louvain for providing us the curve fitting Matlab code.

References

[1] H. Merskey and et al., "Pain terms: a list with definitions and notes on usage," *Pain*, vol. 6, no. 3, 1979.

[2] J. Younger, R. McCue, and S. Mackey, "Pain outcomes: A brief review of instruments and techniques," *Current Pain and Headache Reports*, vol. 13, pp. 39–43, 2009.

[3] B. Aicher, H. Peil, B. Peil, and H.-C. Diener, "Pain measurement: Visual analogue scale (vas) and verbal rating scale (vrs) in clinical trials with otc analgesics in headache," *Cephalalgia*, vol. 32, no. 3, pp. 185–197, 2012.

[4] J. T. Farrar, R. K. Portenoy, J. A. Berlin, J. L. Kinman, and B. L. Strom, "Defining the clinically important difference in pain outcome measures," *Pain*, vol. 88, no. 3, pp. 287–294, 2000.

[5] M. P. Jensen, C. Chen, and A. M. Brugger, "Interpretation of visual analog scale ratings and change scores: a reanalysis of two clinical trials of postoperative pain," *The Journal of Pain*, vol. 4, no. 7, pp. 407–414, 2003.

[6] M. P. Jensen, S. A. Martin, and R. Cheung, "The meaning of pain relief in a clinical trial," *The Journal of Pain*, vol. 6, no. 6, pp. 400–406, 2005.

[7] K. D. C. et al., *The facial expression of pain*. Guilford Press, 2011.

[8] M. Kunz, S. Scharmann, U. Hemmeter, K. Schepelmann, and S. Lautenbacher, "The facial expression of pain in patients with dementia," *Pain*, vol. 133, no. 1, 2007.

[9] K. M. Prkachin, "The consistency of facial expressions of pain: a comparison across modalities," *Pain*, vol. 51, no. 3, 1992.

[10] K. M. Prkachin and P. E. Solomon, "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain," *Pain*, vol. 139, no. 2, pp. 267–274, 2008.

[11] P. Ekman, W. Friesen, and J. Hager, *Facial Action Coding System: The Manual on CD ROM*, 2002.

[12] Z. Hammal and J. F. Cohn, "Automatic, objective, and efficient measurement of pain using automated face analysis," *Ken Prkachin, Zina Trost and Kai Karos (EDs.), Handbook of Social and interpersonal processes in pain: We don't suffer alone*, p. 121–146, 2018.

[13] P. Werner, D. Lopez-Martinez, S. Walter, A. Al-Hamadi, S. Gruss, and R. Picard, "Automatic recognition methods supporting pain assessment: A survey," *IEEE Trans. on Affective Computing*, pp. 1–1, to appear 2019.

[14] D. L. Martinez, O. Rudovic, and R. W. Picard, "Personalized automatic estimation of self-reported pain intensity from facial expressions," in *IEEE Conf. on Computer Vision and Pattern Recognition Workshops CVPR*, 2017, pp. 2318–2327.

[15] D. Liu, F. Peng, O. O. Rudovic, and R. W. Picard, "Deepfacelift: Interpretable personalized models for automatic estimation of self-reported pain," in *AffComp@IJCAI*, ser. Proceedings of Machine Learning Research, vol. 66. PMLR, 2017, pp. 1–16.

[16] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. A. Matthews, "Painful data: The unbc-mcmaster shoulder pain expression archive database," in *IEEE Int. Conf. on Automatic Face and Gesture Recognition (FG)*, 2011, pp. 57–64.

[17] B. Szczapa, M. Daoudi, S. Berretti, A. Del Bimbo, P. Pala, and E. Massart, "Fitting, comparison, and alignment of trajectories on positive semidefinite matrices with application to action recognition," in *IEEE Int. Conf. on Computer Vision (ICCV) Workshops*, Oct 2019.

[18] H. Drucker, C. J. C. Burges, L. Kaufman, A. J. Smola, and V. Vapnik, "Support vector regression machines," in *Advances in Neural Information Processing Systems (NIPS)*, 1996, pp. 155–161.

[19] A. Kacem, M. Daoudi, B. Ben Amor, S. Berretti, and J. C. Alvarez Paiva, "A novel geometric framework on gram matrix trajectories for human behavior understanding," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 1–14, 2020. [Online]. Available: https://doi.org/10.1109/TPAMI.2018.2872564

[20] S. Bonnabel and R. Sepulchre, "Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 1055–1070, 2009.

[21] M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre, "Low-rank optimization on the cone of positive semidefinite matrices," *SIAM Journal on Optimization*, vol. 20, no. 5, pp. 2327–2351, 2010.

[22] E. Massart and P.-A. Absil, "Quotient geometry with simple geodesics for the manifold of fixed-rank positive-semidefinite matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 41, no. 1, pp. 171–198, 2020.

[23] E. Massart, J. M. Hendrickx, and P.-A. Absil, "Curvature of the manifold of fixed-rank positive-semidefinite matrices endowed with the Bures-Wasserstein metric," in *4th Conference on Geometric Sciences of Information (GSI 2019)*, 2019, pp. 739–748.

[24] B. Vandereycken, P.-A. Absil, and S. Vandewalle, "Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank," in *IEEE/SP Workshop on Statistical Signal Processing (SSP)*, 2009, pp. 389–392.

[25] B. Vandereycken, P.-A. Absil, and S. Vandewalle, "A Riemannian geometry with complete geodesics for the set of positive semidefinite matrices of fixed rank," *IMA Journal of Numerical Analysis*, vol. 33, no. 2, pp. 481–514, 2013.

[26] M. Faraki, M. T. Harandi, and F. Porikli, "Image set classification by symmetric positive semi-definite matrices," in *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2016, pp. 1–8.

[27] G. Meyer, S. Bonnabel, and R. Sepulchre, "Regression on fixed-rank positive semidefinite matrices: a Riemannian approach," *Journal of Machine Learning Research*, vol. 12, no. Feb, pp. 593–625, 2011.

[28] P.-Y. Gousenbourger, E. Massart, A. Musolas, P.-A. Absil, L. Jacques, J. M. Hendrickx, and Y. Marzouk, "Piecewise-Bézier C^1 smoothing on manifolds with application to wind field estimation," 2017, pp. 305–310.

[29] E. Massart, P.-Y. Gousenbourger, N. T. Son, T. Stykel, and P.-A. Absil, "Interpolation on the manifold of fixed-rank positive-semidefinite matrices for parametric model order reduction: preliminary results," in *European Symposium on Artifical Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, 2019, pp. 281–286.

[30] P.-Y. Gousenbourger, E. Massart, and P.-A. Absil, "Data fitting on manifolds with composite Bézier-like curves and blended cubic splines," *Journal of Mathematical Imaging and Vision*, vol. 61, no. 5, pp. 645–671, 2018.

[31] M. Cuturi, J. Vert, Ø. Birkenes, and T. Matsui, "A kernel for time series based on global alignments," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing ICASSP*, 2007, pp. 413–416.

[32] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, "Manopt, a matlab toolbox for optimization on manifolds," *Journal of Machine Learning Research*, vol. 15, pp. 1455–1459, 2014. [Online]. Available: http://jmlr.org/papers/v15/boumal14a.html