Published in final edited form as:
*Image Vis Comput*. 2014 October 1; 32(10): 659–670. doi:10.1016/j.imavis.2014.02.008.

# Classification and Weakly Supervised Pain Localization using Multiple Segment Representation

**Karan Sikka**,
University of California San Diego, 9450 Gilman Drive # 0440, La Jolla, CA 92093, USA

**Abhinav Dhall**, and
Australian National University, CSIT, Building 108, North Road, Canberra, Australia 2601

**Marian Stewart Bartlett**
University of California San Diego, 9450 Gilman Drive # 0440, La Jolla, CA 92093, USA

Karan Sikka: ksikka@ucsd.edu; Abhinav Dhall: abhinav.dhall@anu.edu.au; Marian Stewart Bartlett: mbartlett@ucsd.edu

## Abstract

Automatic pain recognition from videos is a vital clinical application and, owing to its spontaneous nature, poses interesting challenges to automatic facial expression recognition (AFER) research. Previous pain vs no-pain systems have highlighted two major challenges: (1) ground truth is provided for the sequence, but the presence or absence of the target expression for a given frame is unknown, and (2) the time point and the duration of the pain expression event(s) in each video are unknown. To address these issues we propose a novel framework (referred to as MS-MIL) where each sequence is represented as a bag containing multiple segments, and multiple instance learning (MIL) is employed to handle this weakly labeled data in the form of sequence level ground-truth. These segments are generated via multiple clustering of a sequence or running a multi-scale temporal scanning window, and are represented using a state-of-the-art Bag of Words (BoW) representation. This work extends the idea of detecting facial expressions through ‘concept frames’ to ‘concept segments’ and argues through extensive experiments that algorithms such as MIL are needed to reap the benefits of such representation.

The key advantages of our approach are: (1) joint detection and localization of painful frames using only sequence-level ground-truth, (2) incorporation of temporal dynamics by representing the data not as individual frames but as segments, and (3) extraction of multiple segments, which is well suited to signals with uncertain temporal location and duration in the video. Extensive experiments on UNBC-McMaster Shoulder Pain dataset highlight the effectiveness of the approach by achieving competitive results on both tasks of pain classification and localization in videos. We also empirically evaluate the contributions of different components of MS-MIL. The paper also includes the visualization of discriminative facial patches, important for pain detection, as discovered by our algorithm and relates them to Action Units that have been associated with

**Publisher's Disclaimer:** This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

pain expression. We conclude the paper by demonstrating that MS-MIL yields a significant improvement on another spontaneous facial expression dataset, the FEEDTUM dataset.

### Keywords

Emotion classification; Action classification; Pain; Temporal Segmentation; Bag of Words; Weakly Supervised Learning; Boosting; Bagging

## 1. Introduction

Pain is one of the most challenging problems in medicine and biology and has substantial eco-social costs associated with it [10]. It has been estimated that there might be more than 30 million people in USA with chronic or recurrent pain [35]. Also nearly half of Americans seeking treatment from a physician report pain as their primary symptom. The United States Bureau of the Census estimated the total cost for chronic pain to exceed $150 billion annually in year 1995–96 [35, 10]. Thus there has been a significant research effort in improving pain management over the years.

Identifying pain among patients is considered critical in clinical settings since it is used for regulating medications, long-term monitoring, and gauging the effectiveness of a treatment. Pain assessment in most cases involves patient self-report, obtained either through clinical interview or visual analog scale (VAS) [10]. For the latter case the nurse asks the patient to mark his pain on a linear scale with ratings from 0 to 10, denoting no-pain to unbearable-pain. The fact that VAS is easy to use and returns a numerical rating of pain has made VAS the most prevalent pain assessment tool. However VAS suffers from a number of drawbacks such as subjective differences, and patient idiosyncrasies. Therefore it cannot be used for unconscious or verbally-impaired patients [7] and may suffer from high individual bias. These drawbacks have led to a considerable research effort to identify and quantify objective pain indicators using human facial expression [34]. However most of these methods entail manual labeling of facial action units or evaluations by highly trained observers, which in most cases is time consuming and unfit for real-time applications.

Over the years there has been a significant progress in analyzing facial expressions related to emotions using machine learning (ML) and computer vision [20]. Most of this work has focused on posed facial expressions that are obtained under controlled laboratory settings and differ from spontaneous facial expression in a number of ways [8, 5]. We refer our readers to a survey on automatic facial expression recognition (AFER) by Bartlett et al. [5] that has identified the difficulties faced by AFER on spontaneous expressions. A major challenge of spontaneous expressions is temporal segmentation of the target expressions. Videos may exist in which the target emotion or state was elicited, but the onset, duration, and frequency of facial expressions within the video are unknown.

A significant contribution to in research on spontaneous expressions was the introduction of UNBC-McMaster Shoulder Pain dataset [22] that involves subjects experiencing shoulder pain in a clinical setting. This dataset was provided with two levels of annotations for measuring pain- (1) per-frame pain ratings based on a formula applied to Action Unit (AU) annotations, and (2) per-video pain ratings as measured by experts (see Section .5.1). This work utilizes the per-video pain ratings for training a binary pain classification system. Pain localization is then evaluated using the per-frame pain ratings based on AU labels, which are more costly to obtain. Thus our setting is such that each video is labeled for presence or absence of pain, but there is no information about the location or duration of facial expressions within each video. This setting is referred to as weakly labeled data and poses a challenge for training sliding window classifiers and further limits the performance of the standard approach of obtaining fixed length features through averaging and training a classifier. Previous approaches [23, 2] follow a common paradigm of assigning each frame the label of the corresponding video and using them to train a support vector machine (SVM). Pain is detected in a video if the average output score (distance from separating hyperplane) of member frames is above a pre-computed threshold. Such approaches suffer from two major limitations: (1) not all frames in a video have the same label, (2) averaging output scores across all the frames may dampen the signal of interest. This paper proposes to address these challenges by employing multiple instance learning (MIL) learning framework [36].

MIL is an approach for handling ’weakly labeled’ training data. In such cases the training data only specifies the presence (or absence) of a signal of interest in the data without indicating where it might be present. For instance in the case of pain vs no-pain detection, a sequence label only specifies if a subject is/not in pain without any details regarding the time point or duration of pain. Other techniques for tackling weakly labeled data includes part-based models [12] and latent models such as pLSA and LDA [38]. Most of these approaches try to identify the signal of interest by inferring the values of some latent variables while minimizing a loss function. MIL was introduced to address the problem of weakly supervised object detection [36] [14]. Compared to other approaches, MIL offers a tractable way to train a discriminative classifier that avoids complex inference procedures. MIL has been successfully employed for face recognition from video [36] and more recently has been proposed for handling labeling noise in video classification [19].

This work focuses at detecting spontaneous pain expression in video when given only sequence level ground-truths. The phrase *detection* is used throughout the paper to denote the joint tasks of pain classification and localization in time. Explicitly, classification refers to predicting absence/presence of pain in a video, while localization refers to predicting pain/no-pain at the frame level. The novelty of this work lies in combining MIL with a dynamic extension of concept frames, into a novel framework called **Multiple-Segment Multiple Instance Learning (MS-MIL)**. Our major contributions are as follows:

1. Inherent drawbacks in previous approaches for pain detection in videos are identified and a pipeline has been proposed to address these concerns. The most salient feature of our approach is that it can jointly classify and localize pain by using only sequence level labels (Section. 2.
2. For addressing the demands of the pain detection task, we propose to represent each video as a bag containing multiple segments which are modeled using MIL. The multiple segment based representation and MIL are able to address

spontaneous expressions, such as pain, that can have uncertain locations, durations and occurrences (Section. 4).

3. The performance of MS-MIL is compared on the detection task with other competitive algorithms. We also perform systematic evaluation to highlight the contribution of multiple segment representation and MIL, in MS-MIL, separately. These results indicate the advantage of using the MS-MIL approach along with some interesting insights. (Section. 6)

The problem of detecting pain through facial expressions in general includes many challenges and this work is trying to focus on a particular aspect of the problem. Other challenges in objective pain measurement include differences between acute and chronic pain, as well as differences in personality including pain catastrophizing, which may affect the intensity of pain expression. We are undertaking a separate study to begin to address some of these factors [15].

## 2. Related Work and Motivation

The first computer vision work on automatic pain detection in videos on the UNBC Mc-Master Pain dataset was by Ashraf et al. [2]. Their approach started by first extracting AAM based features from each frames and using these to cluster the frames in order to create a training data with size that is manageable by a SVM. Following this, each of these clustered frames were assigned the label of their corresponding sequence and used to train a linear SVM. Finally during prediction each test-frame was assigned a score based on its distance from separating hyperplane. Then a test-video was predicted to be in pain if the average score of its member frames exceeded a threshold. Lucey et al. [23] extended this work by borrowing ideas from the related field of visual speech recognition and proposed to compress the signal in the spatial rather than temporal domain using the Discrete Cosine Transform (DCT). Lucey et al. [23] used the system in [2] as their baseline system and showed significant improvement in performance using their idea.

Previous works didn’t address the ambiguity introduced by weakly labeled data, and each member frame was assigned the label of the sequence. Such approaches lead to a lower performance compared to the case when ground-truth for each frame is known [2][1]. We address this particular concern by proposing to use MIL (in-place of SVM) which has been designed specifically to handle weakly labeled data.

Secondly, [23] highlighted that incorporating the dynamics of the pain signal is difficult since there is no information about the number of times pain expressions can occur or their location and duration in a sequence. Following this, [23] suggested to add temporal information by appending adjacent frames onto the frame of interest, as input to the SVM [26]. [23] tested this idea of appending adjacent frames in their paper, however they found that their performance degraded. One possible explanation is that SVM classifiers are not well suited to weakly labeled training data and may suffer from mislabels when the data is in this form.

Motivated by the last idea we propose to incorporate temporal dynamics by representing each sequence not as individual frames (as done earlier) but as sets of frames, referred to as ‘multiple segments’. The benefits of such a representation are reaped by using MIL, which can efficiently handle data in such form. Since MIL handles data as bags, we can visualize every sequence as a bag containing multiple segments. Multiple segments (MS) has two fold advantages: (1) it allows pain expression to have random duration and occurrence, and (2) it incorporates temporal information by pooling across multiple frames in a segment. Thirdly, the earlier work performed prediction for each sequence using the average decision score of its frames. Such an approach may not be optimal in all situations since the averaging operation tends to dampen the signal of interest. The MIL framework employed in this work avoids this limitation by using the max operation to predict the label of a bag based on the posterior probability of its instances (see Section. 3).

Another potential approach to the problem of pain detection comes from the classical approach to action recognition from computer vision literature [18, 40]. This approach is based on BoW architecture and composed of three steps: feature extraction, encoding features using a dictionary of visual words and pooling with $l_1$ normalization. Since each video is represented as a fixed length vector, we shall refer to these techniques as global-feature based approaches. [40] has provided a systematic evaluation of different components of this pipeline on two human action datasets. These techniques are known to work well for problems with uniform actions that span the entire video such as CK+ facial expression dataset [21] or KTH human action dataset [17]. However their performance falls down when actions have high intra-class variations and are localized in the video, which is true for the pain detection problem as well. We also found this hypothesis to be true during our experiments and attribute it to the argument that pooling features across the entire video tends to reduce discriminative ability of the features.

In a recent paper [32] Tax et al. explored the question of whether it is always necessary to fully model the entire sequence, or whether the presence of specific frames, called ‘concept frames’, might be sufficient for reliable detection of facial expressions. In their study two different approaches for AFER were investigated: (1) modeling full sequences using approaches such as Hidden Markov Models and Conditional Random Fields, and (2) modeling only certain frames, for AU detection in sequences. The author in [32] also suggested that for modeling only particular key frames, algorithms such as MIL are required and investigated one such approach. Through extensive experiments the authors showed that for reliable classification, modeling certain key frames is sufficient compared to modeling the entire sequence. A limitation of ‘concept frames’, however, is that they do not incorporate temporal information, which could potentially be exploited by learning algorithms such as MIL (and to some extent SVM [31]).

The present paper takes a leap forward by proposing a dynamic variant of ’concept frames’. Here we extend the idea of ‘concept frames’ to ‘concept segments’ consisting of multiple frames. These ‘concept segments’ can be thought of as localized sub-expressions that contain the expression of interest in a sequence. We propose that Reliable detection of facial expression can be achieved by detection of key localized segments using tailored algorithms such as MIL. [31] explored a segment based approach, called k-Seg SVM, and employed a structured-SVM to detect temporal events (AU segments in their case). Our work differs from this work in several respects, most notably that [31] is a completely supervised algorithm requiring location information in the training data, whereas the approach presented here operates on weakly labeled data. Authors in [9] represented a video by concatenating features from 6 key-frames (segments) that were identified by clustering based on the output of an emotion classification task. We overcome the possible limitations of this work by allowing the videos to be represented by a variable number of segments of varying lengths and performing classification by explicitly spotting the segment containing target expression.

## 3. MIL

The general machine learning paradigm involves finding a classification function that minimizes a loss function $\mathscr{L}(D, h(x))$ over training data provided as $N$ samples and their corresponding labels, $D=\{x_i, y_i\}_{i=1}^{N}$, where $x_i \in X$ and $y_i \in Y$. Rather than handing training data in the form of individual samples, the MIL paradigm is designed to handle problems involving training data in the form of bags, $B=\{X_i, y_i\}_{i=1}^{N}$, where $X_i=\{x_{ij}\}_{j=1}^{N_i}$, $y_i \in Y$ and $N_i$ are the number of instances in $X_i$. Since this work deals with only binary classification problems, the output space $Y \in \{-1, 1\}$. Such problems occur frequently in computer vision since it is easier to obtain a group label for the data compared to individual labels and such labels can also suffer from annotator bias and noise [19]. Recently several works have adopted MIL to address these concerns in domains such as handling label noise in video classification [19], face recognition in videos with subtitles [41], and object localization [14], etc.

As shown in Figure. 1 the MIL framework defines two kinds of bags, positive and negative, in a similar fashion to positive and negative instances in traditional machine learning. A bag is a positive bag if it contains at least one positive instance, while a negative bag contains no positive instance.

We have employed Multiple Instance Learning based on boosting (MilBoost) algorithm proposed by Viola et al. [36] for this work. In the next two sections we shall give an overview of Friedman's gradient boosting framework [13], which is the backbone of MilBoost. This will be followed by the description of MilBoost.

### 3.1. Gradient Boosting

We shall define the gradient boosting in the realm of traditional learning framework and then discuss its extension to the MIL framework.

Boosting involves constructing a strong classifier $H_T(x)$ by iteratively combining many weak classifiers $h_t(x)$, where the subscript $t$ ($t = 1\ldots.T$) represents the index of the classifier added at the $t^{th}$ iteration. All weak classifiers are constrained to belong to a certain family of functions $\mathscr{H}$, such as stumps or trees.

$$H_T(x)=\sum_{t=1}^{T}\alpha_t h_t(x) \quad (1)$$

$$H_T(x)=H_{T-1}(x)+\alpha_T h_T(x) \quad (2)$$

Equation. 2 can be seen as a numerical optimization strategy that iteratively minimizes a loss function $\mathscr{L}(D, H_{T-1}(x))$ over training data $D$ by moving in certain optimal direction given by $h_T$. Under this strategy, the loss function at step $T$ can either be seen as a function of the current classifier $H_{T-1}$ or the parameters that define the family of functions $\mathscr{H}$.

Friedman suggested following the latter approach since it offers an intuitive way to solve the above optimization problem. $H_{T-1}(x)$ can be considered as a $n$ dimensional vector whose $i^{th}$ component is $H_{T-1}(x_i)$. Following this idea, the gradient descent strategy is employed to minimize the loss function by moving some steps in the direction of the negative-gradient of the loss function wrt $H_{T-1}(x)$. This negative gradient is denoted by $w_i$ in Equation. 3. In the remaining sections of this paper we shall refer to $w$ as weights and the rationale behind this will be evident in Section. 3.2.

$$w_i=-\left.\frac{\partial \mathscr{L}}{\partial H_{T-1}(x)}\right|_{x=x_i} \quad (3)$$

Thus the gradient boosting framework prescribes to minimize the loss function by moving in the direction $w$ computed at each iteration. Since $H_T$ is a linear combination of $H_{T-1}$ and $w$, it would be smooth only when $w \in \mathscr{H}$ However it will be too idealistic to assume this in all cases. Friedman proposed to tackle this problem by projecting $w$ over the function space $\mathscr{H}$ by finding the best approximation $h_t \in \mathscr{H}$ to $w$.

$$h_t=\arg\max_h \sum_{i=1}^{N} w_i h(x_i) \quad (4)$$

We shall refer to Equation. 4 as the ’projection step’ and note that $h_t$ has the maximum correlation with $w$. Once $h_t$ is computed, step size $\alpha_t$ is found via a line search to minimize $L(D, H_T(x))$. In the next section we shall discuss how gradient boosting is extended to the MIL framework.

### 3.2. MilBoost

MilBoost combines the gradient boosting framework with the concept of MIL, where training data occurs as bags. As defined in Section. 3, the $i^{th}$ bag is denoted by $X_i$ and the $j^{th}$ instance inside it is represented as $x_{ij}$. The posterior probabilities over bags and instances are defined as:

$$p_i=Pr(y_i=1|X_i) \quad (5)$$

$$p_{ij}=Pr(y_{ij}=1|x_{ij}) \quad (6)$$

We shall be using the original formulation defined in [36] for the loss function given by the negative log-likelihood:

$$\mathscr{L}=-\sum_{i}^{N}t_i log p_i+(1-t_i)log(1-p_i) \quad (7)$$

where $t_i = 1$ if $y_i = 1$ and $t_i = 0$ if $y_i = -1$.

This formulation for the loss function seems intuitive since the only information available about a MIL dataset is label information for each bag ($y_i$). We lack any information about the probabilities (or labels) of individual instances ($p_{ij}$). These instance probabilities can also be seen as latent variables, whose values are inferred during the boosting process [4].

MIL assumes that a positive bag contains at least one positive instance. Hence the probability of a bag being positive ($p_i$) is defined in terms of individual instances as:

$$p_i=\max_j(p_{ij}) \quad (8)$$

Since the max function is not differentiable, a number of differentiable approximations to the max function have been proposed for MilBoost [36, 41, 4]. In this work we shall refer to these approximations as softmax functions $g(p_{ij})$. The most common choice of soft-max function in earlier works is noisy-or (NOR). A major disadvantage with NOR is that it deviates from the max function as the size of the bag increases, which we shall refer to as 'bagsize-bias'. To illustrate this shortcoming we consider a toy example which consists of two bags $B_1$ and $B_2$ of sizes of 3 and 5. The instance probabilities for these bags are given by $B_1$ = [.15 .15 2] and $B_2$ = [.15 .15 .15 2]. As is evident, the max for both cases is 2, however the NOR formulation yields the maximum as .45 and .53 respectively. This observation clearly highlights the bagsize-bias associated with NOR. Such a problem is critical for cases where bag sizes might differ across training examples and ours is such a case since the number of frames per sequence vary from 60–600. Thus in this work we have addressed this problem by employing another soft-max function called Generalized mean (GM), which is known to be a better approximating function than NOR [4].

The instance probabilities ($p_{ij}$) for instance $x_{ij}$ are obtained by the application of a sigmoid function over the raw classifier score $h_{ij}$:

$$p_{ij}=\sigma(h(x_{ij})) \quad (9)$$

As described in Section. 3.1, the negative gradient of the loss-function (for instance $x_{ij}$) is obtained as:

$$w_{ij} = -\frac{\partial \mathscr{L}}{\partial h_{ij}} \quad (10)$$

We can easily calculate $w_{ij}$ by exploiting the chain rule of differentiation and calculating each components as:

$$w_{ij} = -\frac{\partial \mathscr{L}}{\partial h_{ij}} = -\frac{\partial \mathscr{L}}{\partial p_i}\frac{\partial p_i}{\partial p_{ij}}\frac{\partial p_{ij}}{\partial h_{ij}} \quad (11a)$$

$$\frac{\partial \mathscr{L}}{\partial p_i} = \begin{cases} \frac{1}{p_i} & t_i = 1 \\ \frac{1-t_i}{1-p_i} & t_i = 0 \end{cases} \quad (11b)$$

$$\frac{\partial p_i}{\partial p_{ij}} = \frac{\partial g(p_{ij})}{\partial p_{ij}} \quad (11c)$$

$$\frac{\partial p_{ij}}{\partial h_{ij}} = \frac{\partial \sigma(h_{ij})}{\partial h_{ij}} = \sigma(h_{ij})(1-\sigma(h_{ij})) \quad (11d)$$

Next we explain the rationale behind referring to the negative instance-wise gradients ($w_{ij}$) as weights, using the NOR softmax function as an example.. From Table. 1, $w_{ij}$ for the NOR soft-max function is defined as $w_{ij} = \frac{1-p_i}{p_i} p_{ij}$ for a positive bag and $w_{ij} = -p_{ij}$ for a negative bag. Thus these weights describe, (1) the label of the bag containing instance $x_{ij}$ and, (2) the importance of the instance in learning procedure, by being high for an instance that lies in a positive bag but has a low classifier score and vice-versa. The idea of weighting instances during learning is common in boosting procedure [13].

As described in Section. 3.1, the next step involves finding a new weak learner ($h(x_{ij})$) that has the highest correlation with the weights $w_{ij}$ using the projection step (Equation. 4). This work employs binary decision stumps as weak learners, which perform classification by assigning a threshold to a single feature and are a common choice in boosting frameworks [36]. Thus $\mathscr{H}$ belongs to the class of decision stumps. A simple mathematical formulation has been provided in Borris et al. [4] on how Equation. 4 (the projection step) can be transformed into:

$$h_t = \arg\min_h \sum_{ij} [h(x_{ij}) \neq sgn(w_{ij})] w'_{ij} \quad (12)$$

where [.] is the Iverson bracket, $w'_{ij} = \frac{|w_{ij}|}{\sum_{ij}|w_{ij}|}$ and $sgn(l)$ is the signum function.

Equation. 12 is a general formulation for any learning algorithm that has training data with binary labels $sgn(w_{ij})$ and weights $w'_{ij}$. Thus we can easily find a function $h_t(x_{ij})$ at $t^{th}$

iteration that has the highest correlation with $w_{ij}$ by using training procedure for a decision stumps. All the steps of the MilBoost algorithm are mentioned in a sequential order in Algorithm. 1.

# 4. Multiple Instance Learning based on Multiple Segments (MS-MIL)

## 4.1. Overview

Each sequence $S_i$ is represented as a bag containing many segments or sub-sequences $\{s_{ij}\}_{j=1}^{N_i}$, where $N_i$ is the number of segments in sequence $S_i$. Temporal consistency is maintained inside a segment $s_{ij}$ by restricting it to contain only contiguous frames (see Section. 4.3), $s_{ij}=\{f_i^k, f_i^{k+1}, \ldots.., f_i^{N_{ij}-k-1}\}$, where $k$ represents the time index (in the video) of the first frame inside segment $s_{ij}$, $f_i^k$ represents the $k^{th}$ frame in the sequence $S_i$ and $N_{ij}$ is the number of frames in subsequence $s_{ij}$. Thus a sub-segment $s_{ij}$ is characterized by length of the segment (number of frames) $N_{ij}$ and the time index $k$ of first frame in the video. Two approaches are outlined in Section 4.3 for constructing multiple segments- (1) overlapping temporal scanning windows, and (2) multiple clustering. Depending upon the approach the number of frames inside a segment can either be fixed (in scanning windows) or sequence-dependent (in multiple clustering). Also the frames inside two different segments are allowed to overlap.

**Algorithm 1**

MilBoost Algorithm

```
Data: Bags and labels $\{X_i, y_i\}_{i=1}^{N}$
Initialization: Initialize $w_{ij}$, $|H_0(x_{ij}) = 0 \ \forall x_{ij}$
for $t=1$ to $T$ do
    Train a weak classifier
        $h_t = \arg\min_{h} \sum_{ij} (h(x_{ij}) \neq sgn(w_{ij})) \frac{|w_{ij}|}{\sum_{ij} |w_{ij}|}$   (13)
    Perform Line Search
        $\alpha_t = \arg\min_{\alpha} \mathcal{L}(H_{t-1} + \alpha h_t)$   (14)
    Update Rule
        $H_T = H_{t-1} + \alpha h_t$   (15)
    Compute Weights
        $w_{ij} = -\left.\frac{\partial \mathcal{L}}{\partial H_T(x)}\right|_{x=x_{ij}}$   (16)
end
```

The only information available about a sequence during training is whether it has a pain expression i.e. $y_i = 1$ or not i.e. $y_i = -1$. We shall give a brief overview of the entire algorithm here.

**Representation—**The feature extraction process for a frame shall be denoted by a mapping $\varphi_{Fr} : R^{m \times n} \to R^d$ that maps frames in image space $R^{m \times n}$ to a $d$–dimensional vector space $R^d$. The feature representation for a subsequence (or segment) is represented as a mapping $\varphi_S : \mathscr{S} \to R^d$ that transforms subsequences in space $S$ to a $d$–dimensional vector space.

**Training—**Training data in the form of bags is trained using the MilBoost framework described in Section. 3.2. This process yields a classifier $H_T : R^d \to R$. The number of iterations/weak-learners for MilBoost have been empirically set to 100 in our experiments.

**Prediction—**Suppose we have a test sequence $S_i = \{s_{i1}, \dots, s_{iN_i}\}$. Each subsequence $s_{ij}$ is assigned a posterior probability $p_{ij}$ using the trained classifier $H_T$ and a sigmoid function σ as:

$$p_{ij} = \sigma(H_T(\phi_S(s_{ij}))) \quad (17)$$

Here $\varphi_S$ is the feature mapping for a sub-sequence.

The posterior probability of test sequence $S_i$ is predicted by using a soft-max function, as described in Section. 3.2, over instance probabilities:

$$p_i = g(p_{ij}) \quad (18)$$

**Avoiding Local-Minima—**MilBoost algorithms can often overfit and converge to local minima. This issue is more critical for problems such as pain detection since theoretically the algorithm can converge even after learning a single instance of pain expression in a sequence, since the loss function is defined over bags. In such cases the learned function won't be able to generalize well over unseen data. Hence we draw parallel ideas from bagging predictors proposed by Brieman [6], in which multiple versions of a predictor are combined to get an aggregated prediction. They showed improvement for predictors that are unstable/get caught up in multiple local minima. Since the problem formulation is very similar to ours, we also ran MilBoost over multiple initializations and bootstrapped data (random 90% subset). The final predictions for each segment were obtained by averaging the predictions $p_{ij}$ made from multiple MilBoost classifiers. Using this approach we found an improvement in predictions, and moreover this procedure allowed us to report results that would be reproducible. Based on our experiments we opted to run MilBoost 30 times. In practice we found that any number about this size or larger worked equally well

**Pain Localization—**The prediction process estimates the posterior probability of each segment $s_{ij}$ in $S_i$. For assigning posterior probability to any frame in the sequence, we first identify the segments containing that frame. Following this, the frames are assigned a score based on their proximity to the center of that segment. We employ a hamming window, pivoted at the center of the segment, to assign a smoothly varying score to different frames in a segment. Since a frame could belong to multiple segments, it is assigned the maximum score from all these segments. In mathematical notations, the probability of frame $f_i^k$ in pain is predicted using the following formula:

$$p_{f_i^k}=p(y=1|f_i^k)=\max_j(\tilde{w}(s_{ij}) \times p_{ij}|f_i^k \in s_{ij}) \quad (19)$$

where $w\tilde{(}s_{ij})$ is the hamming window function centered at the middle frame of segment $s_{ij}$. $p_{f_i^k}$ is a discrete probability measure since it is bounded by 0 and 1 since $w\tilde{(}s_{ij}) \in (0, 1]$ and $p_{ij} \in [0, 1]$. Secondly $\sum_y p(y|f_i^k)=1$. Thus our algorithm not only yields the probability for a sequence but also the probability for each frame that can be used to localize painful expression frames in a video using just sequence-level labels.

### 4.2. Bag of Words based Representation (BoW)

Recently computer vision has witnessed significant research in BoW models and their extensions, and as a result they have been applied across multiple domains. Sikka et al. [30] presents a survey of different BoW Architectures for AFER. They identified many advantages of BoW based approaches over previous approaches to AFER based on Gabor wavelets, or local binary patterns, passed directly through a classifier and have proposed a state-of the-art feature pipeline through experimental analysis.

We employed the system proposed in [30] for the feature extraction and image representation. This representation consists of a spatial pyramid of level 4 on top of highly discriminative multi-scale dense SIFT (MSDF) features, which are encoded using LLC encoding followed by max-pooling. We also employed a separate dataset ($CK$ + [21]) for building a codebook (size 200 in this case) for encoding features. By using a separate dataset for creating the codebook, the feature extraction process is completely independent of the dataset. Our experiments yielded that MSDF features at two scales are sufficient for this problem and hence extracted MSDF features with window sizes of 4 and 8 and strides of 2 pixels. As mentioned in Section. 4, the feature extraction operation using BoW is denoted as a mapping $\sigma_{Fr}$. We refer readers to [30] for more information about feature extraction and image representation in the BoW model including empirical comparisons of alternative feature extraction methods for AFER.

### 4.3. Multiple Segment (MS) Representation

This work defines a segment as a subset of an original sequence that contains only contiguous frames. Thus a sequence is represented as a bag of segments which are allowed to overlap. As highlighted in Section 2, the motivation behind the MS representation is that it allows random onset of pain expression, incorporates dynamic information, and can be efficiently handled by the MIL framework. It is assumed that for a sequence labeled as pain, at least one of the segments will contain a painful expression, and such a positive segment is referred to as a ‘concept segment’.

**Construction—**We propose two ways to generate multiple segments. A naive procedure is to run overlapping temporal scanning windows at multiple scales across the sequence and represent each subset of frames as a segment. This idea is motivated by the traditional approach in computer vision of running multi-scale scanning windows prior to a detection task. This idea has been exploited in previous work on weakly-supervised object localization [12] [36]. A parallel approach for generating multiple segments was explored in [14], where an image was segmented into many clusters using the idea of multiple stable segmentation. Each segmentation was obtained by varying the parameters of normalized cuts (referred to as Ncuts) [14]. We explored an analogous approach by clustering the frames in a sequence using Ncuts. Since we wanted to restrict a segment to contain only contiguous frames, the weight/similarity matrix used in Ncuts was defined to incorporate the similarity between the time index of two frames along with their feature similarity. Each element of this weight matrix $W_i(r, s)$ defines the similarity between frames $f_i^r$ and $f_i^s$ of sequence $S_i$:

$$W(r,s)=\exp\left(-\left|\frac{\phi_{F_r}(f_i^r)-\phi_{F_r}(f_i^s)}{\sigma_f}\right|^2-\left|\frac{t_r-t_s}{\sigma_t}\right|^2\right) \quad (20)$$

where $t_r$ refers to time index of frame $f_i^r$.

Once the segments are constructed using either of the two approaches, it is important to represent them as fixed-length vectors while also preserving temporal information. [23] have highlighted that an elegant way of doing this is to append features from adjacent frames. We employed this idea along with max feature pooling, proposed for AFER in [30], for feature extraction. This process is represented as a mapping $\varphi_S$: $\mathscr{S} \rightarrow R^d$ that maps a segment $s_{ij}=\{f_i^k, f_i^{k+1}, \ldots.., f_i^{N_{ij}-k-1}\}$ belonging to set $\mathscr{S}$ to a $d$-dimensional vector space and can be shown as:

$$\phi_S(s_{ij})=max_k(\phi_{F_r}(f_i^k)|f_i^k \in s_{ij}) \quad (21)$$

Tthe idea of using a max operation for temporal pooling has also been explored in spatio-temporal deep learning approaches [33]. Also a number of recent works [29, 39, 30] have highlighted the performance advantages of the max pooling operation compared to average pooling.

## 5. Experimental Design

### 5.1. Dataset

Our experiments employed data from the UNBC-McMaster Pain Shoulder Archive that was distributed to the research community in [22], and included 200 sequences from 25 subjects. Each subject was undergoing some kind of shoulder pain and was asked to perform a series of active and passive movements of their affected and unaffected limbs. Active tests were self-initiated shoulder movements and in passive tests the physiotherapist was responsible for the movement. For complete details of the experimental settings we refer the readers to [22]. These sequences were then coded on a number of levels by experts. The coding of interest to this work was the Observer Pain Intensity (OPI) rating that was assigned to each sequence on a level of 0 (no-pain)−5 (strong pain) by an independent observer trained in identification of pain expressions. Following the protocol proposed in [23] [2], labels were binarized into 'pain' and 'no pain' by defining training instances with OPI≥ 3 as the positive class (Pain) and OPI= 0 as the negative class (No-Pain). Only those subjects were included in our experiments who had a minimum of one trial with an OPI rating of 0 (no pain) and one trial with an OPI rating of either 3, 4 or 5 (pain). Intermediate pain intensities of 1 and 2 were omitted, per the protocol in [23] [2]. This yielded 147 sequences from 23 subjects for our experiments. Since this work addressed two joint tasks i.e classification and localization of pain, two different performance metrics were employed to evaluate each tasks separately.

## 5.2. Performance Metrics

**Classification—**The classification task focuses on pain predictions at video-level. Experiments were conducted in a leave-one-subject-out cross-validation strategy. Thus there was no overlap between subjects in the training and testing data. For reporting the results, we followed the strategy employed in [23] [2], where they reported total classification rate or accuracy, which refers to the percentage of correctly classified sequences, computed at Equal Error Rate (EER) in the Receiver Operation Curve (ROC).

**Localization—**The localization task focuses on pain predictions at frame-level. This task was evaluated by employing the Prkachin and Solomon pain intensity index (PSPI) that combines intensities of 4 Action units (AUs) from Facial Action Coding System (FACS) [27]. In particular PSPI combines the intensities of four "core" AUs for pain which are brow lowering(AU4), orbital tightening (AU6 and AU7), levator contraction (AU9 and AU10) and eye closure (AU43) [22]. The UNBC McMaster dataset provided FACS expert codes and PSPI metrics for each frame. We would like the readers to note that our algorithm used only OPI labels (sequence-level groundtruth) for training, while the PSPI labels were solely used for evaluation. The localization performance was evaluated across two subtasks, as explained below, with experiments conducted in leave-one-subject-out fashion.

The first task was designed to predict presence/absence of pain in each frame and compare these predictions against binarized PSPI score (where PSPI> 0 means pain). A similar idea of evaluating localization performance, when training with only sequence-level groundtruth, was also explored in [3]. The first metric for this frame based pain classification experiment was classification accuracy computed at EER in the ROC curve. Several previous works focusing on detection [31] have noted that metrics based on ROC curve are designed for balanced binary classification rather than detection tasks, and hence are unable to take into account the effect of the proportion of positive to negative samples. Thus in this work we also incorporated maximum $F1$ score (given by $\frac{2 \times Recall \times Precision}{Recall+Precision}$) for evaluating pain detection task. The $F1$ score is known to give a trade-off between high recall rates and accuracy for predictions [31].

The second task measured how well the the per-frame classification scores can predict PSPI pain intensities. This was accomplished by measuring the correlation between predictions and PSPI pain intensities for each frame. We opted for Spearman's rank correlation [16] instead of Pearson correlation since the PSPI score occurs as ranked values in the range of

1–16. For these experiments we reported Spearman’s rank correlation coefficient, which is calculated between two observations $X_i$ and $Y_i$ as:

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \acute{\mathrm{O}}\acute{\mathrm{r}})}{\sqrt[2]{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \acute{\mathrm{O}}\acute{\mathrm{r}})^2}} \quad (22)$$

s.t $-1 \leq \rho \leq 1$. $\rho = 0$, $\rho = 1$ and $\rho = -1$ correspond to no-correlation, perfect correlation and perfect negative correlation respectively.

## 6. Results and Discussion

### 6.1. Performance Evaluation of Pain Classification

MS-MIL was compared with related algorithms for the problem of pain classification. We divided these related algorithms into 3 groups and have provided implementation details for each of these in the following Subsections. The result for MS-MIL is reported for the best configuration of the multiple segment representation, which was empirically estimated to be a combination of segments of length 31, 41 and 51 frames, generated using overlapping scanning windows (see Section. 6.3).

**6.1.1. Previous State of the Art—**MS-MIL was first compared with previous state of the art algorithms by Ashraf et al. [2] and Lucey et al. [23] as shown in Table. 2. We have reported results for Ashraf et al. as were reported by authors in [23] using their own implementation.

**6.1.2. Global-feature based Approaches—**MS-MIL was also compared for pain classification performance with two global-feature based approaches constructed using BoW [18, 40], as discussed in Section. 2Global-feature based methods represent a video by a fixed length vector. Hence by themselves these methods can only be used for pain classification task and not for pain localization. We used the same frame features, constructed using BoW, as used in MS-MIL. These frame features were then pooled using average [18, 40] and max pooling [40] to obtain a fixed dimensional representation for the entire video. Following feature extraction, classification was performed using a linear SVM [40]. Depending on the pooling strategy, these approaches are referred to as BoW+Avg+SVM or BoW+Max+SVM in Table. 2. These approaches serve as a good baseline since they are amongst the classical approaches for action classification in computer vision [25].

**6.1.3. Evaluating the Contribution of MIL—**We have argued the aptness of MIL to handle sequences represented as multiple segments compared to traditional ML algorithms. This argument was validated by using the same MS representation but replacing MIL with a linear SVM. All the segments in the training data were assigned the label of the sequence and used to train this SVM. This strategy, if not same, is in spirit similar to that employed in previous works ([23] ([2]). Finally during prediction a combining rule was used to assign each sequence a decision score based on the score of its member segments [32]. We had explored two common combining rules, namely maxima (similar to MIL and used in [32]) and average ([23] [2]) and the corresponding SVMs are referred to as MS-SVM$_{max}$ and MS-

SVM$_{avg}$. Table. 2 reports the accuracy for both SVMs with the same MS representation as used in MS-MIL.

**6.1.4. Overview of Pain Classification Task—**Although it could be argued that a direct comparison with previous algorithms for pain detection by [23] and [2] is not possible owing to a different number of samples, some inferences could still be made since the sample set differs by only a small amount of data. Firstly the results of [2] (as published in [23]) and [23] showed an accuracy of 68.31% and 80.99% respectively, compared to 83.7% performance of MS-MIL. Thus it could be argued that MS-MIL shows significant performance improvement over [2] and is comparable to (or better) than [23]. This improvement can be attributed to the algorithmic improvements that MS-MIL has over these approaches (see Section. 2). The two global-feature based approaches, BoW+Avg+SVM and BoW+Max+SVM, yielded a performance of 65.22% and 78.26% respectively. Our argument that global-feature based approaches discard discriminative information as a result of pooling is supported by the observation that they have a lower performance compared to MS-MIL (65.2% and 78.26% vs 83.7%). Also these results provide additional support that max pooling is preferable to average pooling.

Lastly the argument that the MS representation is efficiently handled by MIL is validated by the comparison of MS-MIL with SVM applied to the MS representation as shown in Table. 2. Here MS-MIL outperformed both MS-SVM$_{avg}$ and MS-SVM$_{max}$ by a margin of at least 6% points. The results also indicate that MS-SVM$_{max}$ performs better than MS-SVM$_{avg}$ for all cases since the averaging operation is known to dampen the signal of interest (Section. 2).

### 6.2. Performance Evaluation of Pain Localization

We evaluated the localization performance of MS-MIL for two different sub-tasks of (1) predicting presence/absence of pain, and (2) predicting pain intensity using per-frame classification scores, as discussed in Section. 5.2. The SVM based MS-SVM$_{max}$ algorithm was selected for comparison with MS-MIL. Both algorithms used the same MS representation, which was a combination of segments of length 31, 41 and 51 frames generated using overlapping scanning windows (Section. 6.3).

We were also interested in performance comparison of MS-MIL with a system that was trained particularly for a frame-by-frame pain prediction task. This was accomplished by training a linear SVM over the same frame features as used in MS-MIL, using two versions of frame-level groundtruth. The first version, referred to as Frame-SVM$^1$, was trained using binarized PSPI labels (PSPI> 0 is pain). While for the second version, referred to as Frame-SVM$^2$, the frames were assigned the label of the video that contained them. Thus Frame-SVM$^1$ represents a fully supervised algorithm with complete label information, and Frame-SVM$^2$ represents a weakly-supervised algorithm (such as MS-MIL). Both methods had the same experimental settings as MS-MIL. We handled the massive amount of data (around 35$K$ frames) for this task by training the linear SVM in its primal form using LIBLINEAR SVM library [11]. The results from these experiments are shown Table. 3.

Although the primary interest in this Section is pain-localization performance, we have also reported video-level classification accuracy for each of these methods so as to supplement current analysis. For MS-MIL and MS-SVM$_{max}$, the classification accuracy is the same as that reported in Table. 2. For the two frame based algorithms (Frame-SVM$^1$ and Frame-SVM$^2$), the video scores were estimated by taking a *max* over the scores of member frames, as was done for MS-SVM$_{max}$ in Section. 6.1.

It is evident from Table. 3 that MS-MIL outperforms all other algorithms across both pain localization tasks. The performance of Frame-SVM$^1$ was lower than MS-MIL as reported by pain localization metrics. This was contrary to our expectations since Frame-SVM$^1$ was trained on actual (binarized PSPI) frames labels compared to weak-labels used for MS-MIL. The possible reason for higher performance of MS-MIL could be the use of the MS representation in MS-MIL, that is able to achieve some degree of temporal smoothing. This also shows that MIL framework used in MS-MIL is able handle label ambiguity elegantly. However one cannot neglect the benefit of having complete frame labels, and this is evident in the classification accuracy of Frame-SVM$^1$ (84.78%), which is slightly above MS-MIL (83.7%) and surpasses its weakly-supervised counterpart (Frame-SVM$^2$) (73.91%) by a large margin.

The advantage of using the MS representation is also evident in the higher performance of MS-SVM$_{max}$ compared to Frame-SVM$^2$, where the two algorithms were trained on the same sequence-level labels but employed the MS and the frame representation respectively. It was also interesting to note that MS-MIL was able to achieve a correlation of .432 with the PSPI intensity when it was trained using only weak-labels in the form of video-level labels. Moreover this correlation was higher as compared to the correlation achieved by the supervised frame-by-frame algorithm Frame-SVM$^1$ (.432 vs .385). Thus these results conclude that MS-MIL has a performance advantage over its weakly supervised counterparts as well as over supervised frame-by-frame algorithms.

We have also shown visualization for 2 cases in Fig. 5 to highlight the ability of our algorithm to localize pain. These visualizations compare the per frame posterior probability as predicted by MS-MIL against the PSPI index (Section. 5.2). In order to facilitate a direct comparison between probabilities and the PSPI index on the same vertical scale, the PSPI index was normalized in the range of [0, 1] by dividing by maximum PSPI score of 16 ([22]). These visualizations qualitatively support our claims that MS-MIL is capable of joint classification and localization of pain. It is evident from Fig. 5a that our algorithm is able to identify multiple occurrences of pain. Secondly the posterior probabilities predicted by MS-MIL seem to correlate well with the PSPI index. Fig. 5b shows a case of a pain sequence whose PSPI ground-truth score was zero across all frames but the observer rated the facial expression as showing pain (OPI= 3). Our algorithm, which was trained on observer ratings, was able to localize pain in this case. On further analysis we found that there was a FACS coding error for this particular sample. This is an intriguing example highlighting the advantage of using automatic computational methods compared to humans.

### 6.3. How does the Multiple Segment Representation effect MS-MIL

The novelty of this work lies in combining multiple segment representation with MIL. In Section 6.1 we have already highlighted the advantage of MIL by replacing MIL with SVM (MS-SVM). Here our aim is to empirically evaluate the benefits of the MS representation in MS-MIL. We have tried to show this by analyzing the performance of MS-MIL across different configurations of the MS representation. Here we compare different lengths of the multiple segments in Section. 4.3, we restricted ourselves to the use of multi-scale temporal scanning windows (Scan-wind) for generating MS for this experiment. (Two approaches for generating MS are evaluated independently in the next Section.)

Two parameters are required for Scan-wind: (1) window size, and (2) overlap between two windows. The overlap was fixed to 50% of the window size in all cases and the window size parameter was swept to generate results. The parameters were selected so as to cover a broad range of window sizes starting from short windows of length 10 frames to large windows of length 100 frames. This was done keeping in mind the large variation in video lengths and temporal extent of pain signal in the dataset. We had also tried several combinations of windows sizes to generate multi-scale MS and included results for the case having best performance for both classification and localization tasks. The results are shown in Figure. 2, with Figure. 2a showing plots for classification and localization accuracy metric, and Figure. 2b showing plots for correlation and $F1$ score metric. These metrics are the same as those discussed in Section. 5.2.

Looking at the results in Figure. 2, it is evident that the performance across all metrics goes up as the window size is increased from 11 to 61 frames and thereafter the performance starts to fall down. These results are quite intuitive to interpret as features pooled over small windows will not encode sufficient temporal information, showing lower performance. While for very large window sizes, pooling tends to pack too much information in the features making them less discriminative (as discussed in Section.6.1). The algorithm performed consistently high across window sizes of lengths 41 to 61 frames. One possible reason for this observation could be that most subjects in the dataset present facial action related to pain within intervals of length 41–61 frames. Finally the result corresponding to combination of MS of length 31, 41 and 51 frames yielded the highest results across all the metrics. Although it had the same classification accuracy as segments from 41–61 frames, it showed significant improvement in the metrics evaluating localization task. Thus one could argue the advantage of using multi-scale MS for the pain detection task since it tries to capture all possible pain expressions in a scale independent manner.

Overall these results empirically support the advantage of the multiple segment representation in MS-MIL for the problem of pain detection. Moreover it is evident that the advantage of MS is best reaped at segments of medium length or a combination of these. It was also interesting to empirically verify our hypothesis regarding the importance of pooling over segments of the right length as discussed in Section. 2.

### 6.4. Approaches for Generating Multiple Segments (Normalized Cuts vs Scanning Windows)

Two methods for generating the multiple segment representation were discussed in Section. 4. The first method was Ncuts that generated segments through clustering. Since the number of frames differs across videos, we determined the number of clusters for Ncuts by fixing the minimum number of elements (frames) in a cluster. The values of other parameters were kept constant for all experiments ($\sigma_t = 100$ and $\sigma_f = 10k$). The second approach is the multi-scale temporal scanning windows. We employed the same parameters for multi-scale temporal scanning window as taken in Section. 6.3. For both cases the parameter of interest is the length of the segment to be used.

To systematically study the effect of these approaches on performance, four scenarios were considered by varying the length of segments in our MS representation. These configurations were selected to cover a wide variety of temporal scales. They are referred to as:

1. *short*- segments of short length (11 frames)
2. *med*- segments of medium length (41 frames)
3. *long*- segments of long length (81 frames)
4. *combine*- combination of segments of length 31,41 and 51.

We have also included results from MS-SVM$_{max}$ and MS-SVM$_{avg}$, along with results from MS-MIL in Table. 4, for making further inference.

From results in Table. 4 it is evident that MS-MIL has a low performance for short and long settings and high for medium and combine settings, for both Ncuts and Scan-wind. This observation is in line with the results presented in the previous Section. We didn’t observe any clear trends for the SVM based approaches. It is also interesting to note that the performance of both MS-SVM$_{max}$ and MS-MIL is similar (78.26%) for Ncuts with the short setting. Thus it is possible that there isn’t much difference between MS-MIL and MS-SVM$_{max}$ for short segments since features pooled over short-segments are less informative. Finally MS-MIL shows a consistent performance of 83.7% for combine segments for both Ncuts and scan-wind, highlighting a consistent benefit of multi-scale MS. Although Ncuts and Scan-wind show similar classification performance, Ncuts lags behind Scan-wind on the localization task. This is because Ncuts employs windows/segments that are sparsely located in time, compared to dense sampling in Scan-wind, and localization in the former case will only be approximate (see Section.4).

## 7. Visualizing the Classifier and Benefits of Bagging

Since different expressions are associated with different facial muscles, we wanted to visualize the facial regions contributing most towards pain detection. To accomplish this we selected the weights and indices of the weak-learners learned during the gradient boosting procedure (Section. 3.2). Since our features are based on the spatial-pyramid BoW framework [30], each of these indices represents a word that lies in a localized image patch at one of the 4 scales (see Section.4.2). Next we formed an intensity image by back-mapping each index to its facial patch, and then aggregating weights over all facial patches. We further converted the intensity image into a RGB image with the color hue encoding the magnitude of the weights. The intensity image corresponding to the MS-MIL classifier is shown in Figure. 4a with the color encoding shown in Figure. 4b. We have also shown an intensity image in Figure. 4c overlaid with the image of a subject to aid in visualization of discriminative facial regions. Please note that we have shown both the overlaid and non-overlaid intensity images since the overlaid intensity image could include some extra intensity owing to the texture on subject's face image.

We have tried to interpret the visualization in Figure. 4c by relating regions, identified important for pain detection, to Action Units previously known to be associated with pain [27].

1. The red-most region near the lower-corner of right eye seems to be picking up levator contraction and naso-labial furrow changes associated with AU 9 and AU 10 respectively. This region also seems to capture orbital contraction movements related to AU 6.
2. The eye corner (left-eye) seems to be picking up eye squinting (AU 7 and also AU 43).
3. The chin area seems up to be picking up a chin raise related to AU 17 or mouth opening related to AU 25.

Thus it is evident that the visualization showing discriminative facial regions (learned by the algorithm) seems to correlate well with the prior knowledge about Action Units related to pain.

These visualizations have also been used for highlighting the advantage of using bagging step in MS-MIL. The bagging step works by training multiple MilBoost predictors with different initialization and bootstrapped data as discussed in Section. 4. The final classification score for a segment is obtained by averaging scores from each predictor. To emphasize the contribution of the bagging step, we visualize the weights learned by 3 individual predictors in Fig. 3a, Fig. 3b and Fig. 3c, and the final average predictor obtained after averaging (bagging) in Figure. 3d respectively. These visualizations have been generated using the same procedure as discussed in previous paragraph. It is evident from these visualizations that the weights learned by individual classifiers have high variance and the bagging step helps by averaging and lowering the variance in weights. It is also interesting to note that these results support the argument, posed in several works that analyzed bagging theoretically [24], that bagging can be seen as a kind regularization operation. The weight patterns also reveal the discovery that we made during our experiments that MS-MIL, being a latent variable model, is unstable with respect to initializations and prone to local-minima. And this instability is the vital component that causes bagging to work well in our case as noted in [6].

## 8. Experiments on FEEDTUM dataset

From extensive experiments it is evident that MS-MIL gives appreciable results on the UNBC-McMaster pain dataset. However it could be argued by a machine learning practitioner that the reason for good results could be over-fitting by MS-MIL for this particular setting of features and dataset. Thus we evaluated MS-MIL on a different dataset of spontaneous expressions. We compared the performance of MS-MIL with its global-features based counterpart on a different problem with different set of features. The rationale behind opting for a different problem and different set of features is to exhibit that MS-MIL can also be generalized to a different yet connected problem.

This experiment was conducted on a subset of FEEDTUM facial expression dataset [37] that consists of videos of 19 subjects (320 videos) showing six basis emotions, namely- anger, disgust, fear, happiness, sadness and surprise. The dataset exhibits natural (or spontaneous) expressions, which were elicited by showing the subjects several carefully selected video stimulus. This is different from datasets like CK+ [21], where the subjects were asked to move specific facial muscles. The rationale behind selecting this dataset is that the subjects exhibit spontaneous expressions and the videos are unsegmented, yielding no information about the onset, duration and frequency of the facial expressions. Thus AFER on this dataset poses similar challenges as were discussed in the motivation for current work (see Section . 1).

### 8.1. Experimental setting and Results

The experiments were conducted in leave-one-subject-out fashion. The classification was performed in 1-vs-all format and thus involved solving a different binary classification task for each of the 6 expressions. Different from BoW features, we opted for features based on the displacement of facial landmarks points [28]. 49 landmark points were obtained for each frame by using a state-of-the-art facial feature detector based on supervised gradient descent [42]. Displacement features for each frame were obtained by subtracting *x* and *y* coordinates of the landmark points in that frame from the landmark coordinates in the first (neutral frame) in that video. It is been shown in the expression recognition literature that this subtraction from a person-specific neutral face is vital to normalize landmark features and remove subject-dependent bias [28]. The final feature dimension of 98 is obtained by concatenating displacements of both *x* and *y* coordinates.

In order to highlight the efficacy of MS-MIL, we compared the performance of the following two implementations:

1. geom.+MS-MIL: This is essentially MS-MIL with landmark displacement features. We extracted multiple segments of length 9, 15, and 21, using the overlapping scanning window approach as discussed in Section .4.3. The features inside a multiple segment were obtained by averaging the landmark features of all of the frames inside that segment. This is in spirit similar to the averaging operation used for pooling BoW features (see Section .4.3). We also tried using operators like max instead of averaging to obtain the fixed length features and didn’t see significant change in results.

2. geom.+MilBoost: This version is similar to the global-feature based approaches as discussed in Section. 6.1.2. The fixed length features that represent each video are obtained by averaging the landmark features over all the frames. Once the features are obtained, MilBoost is used as the binary classifier.

It is important to note that MilBoost functions as a generic classifier while working with training data organized as positive and negative instances (as for geom.+MilBoost). Also fixing the classifier allows us to perform a fair comparison for highlighting the performance different with (geom.+MS-MIL) and without (geom+MilBoost) multiple segments.

All the experimental settings for MilBoost have been kept same as those of MS-MIL (see Section), except the number of weak learners which is set to 60 (feature dimension is 98) since we found the performance to saturate approximately at 60 weak learners. The threshold for assigning a positive label to a video based on the probabilistic output (see Equation .18) was set to a standard value of 0.5. The performance metric for this experiment is mean classification accuracy over the 6 expression classes.

The results are shown in Table 5. MS-MIL gives a mean classification accuracy of 84.55(±0.98) compared to 81.78(±1.31) of MilBoost. Thus it is evident from the results that MS-MIL utilizing multiple segments outperforms its fixed length feature counterpart even for a expression classification problem on a different dataset and with different feature set. Such a result was expected since FEEDTUM is a spontaneous expression dataset and holds the assumption that not all frames in a video exhibit the expression of interest. Thus it is evident from this experiment that MS-MIL is capable of generalizing to other classification problems with similar assumptions.

## 9. Conclusion

This paper proposed a novel approach to the problem of detecting spontaneous expressions of pain in videos, based on multiple instance learning (MIL). We presented a novel framework called multiple-segment multiple instance learning (MS-MIL) which incorporated with MIL a dynamic extension of concept frames, referred to as multiple segments (MS). This work targeted the joint problem of, (1) classifying the expression in a video as pain/no-pain (classification), and (2) predicting pain in each frame (localization). The problem is particularly challenging since the algorithm is trained using only sequence-level groundtruth, which provides no information regarding the presence/absence of pain for a given frame.

The paper first highlighted some limitations of previous approaches and how they motivated the design of the proposed algorithm. Next, an overview of multiple instance learning was presented, followed by the description of the proposed approach, MS-MIL. Rigorous experiments were conducted to compare the performance of MS-MIL against related algorithms on both classification and localization tasks on the UNBC Mc-Master Shoulder Pain dataset. The benefits of our algorithm were evident by having significant performance advantages compared to its counterparts across both tasks. Following this we also empirically validated the contributions of both multiple segments representation and multiple instance learning in MS-MIL independently. The results from these experiments supported our argument that MS-MIL is able to tackle the twin challenges of (1) label ambiguity, and (2) incorporating temporal information, in current problem efficiently. To highlight that our algorithm is actually learning meaningful facial structures for pain detection, we showed the visualization for the discriminative facial patches that were learned by our algorithm. We further showed that these discriminative facial patches were related to Action Units known to be associated with Pain.

From our experiments it is evident that pain detection in videos is a challenging problem owing to the variability associated with how pain can be expressed by different subjects at different times and scenarios. The present algorithm is able to do an appreciable job of not only detecting pain, but also identifying the temporal location of pain expressions within the video clip. The most salient contribution of this work is that pain localization is achieved without any human intervention and employing only sequence level labels.

## Acknowledgments

Support for this work was provided by NSF grant IIS-0905622 and NIH grant NIH R01NR013500. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We would also like to thank Tingfan Wu, Gwen Littlewort, Mohsen Malmir, Deborah Forster and Ritwik Giri for helpful discussions.

## References

1. Ashraf A, Lucey S, Cohn J, Chen T, Ambadar Z, Prkachin K, Solomon P. The painful face–pain expression recognition using active appearance models. Image and Vision Computing. 2009; 27(12):1788–1796. [PubMed: 22837587]

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

3. Ashraf AB, Lucey S, Cohn JF, Chen T, Ambadar Z, Prkachin KM, Solomon PE. The painful face pain expression recognition using active appearance models. Image and Vision Computing. 2009; 27(12):1788–1796. [PubMed: 22837587]

4. Babenko B, Dollár P, Tu Z, Belongie S. Simultaneous learning and alignment: Multi-instance and multi-pose learning. Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition. 2008

5. Bartlett M, Littlewort G, Frank M, Lainscsek C, Fasel I, Movellan J. Automatic recognition of facial actions in spontaneous expressions. Journal of Multimedia. 2006; 1(6):22–35.

6. Breiman L. Bagging predictors. Machine learning. 1996; 24(2):123–140.

7. Cornelius, R. The science of emotion: Research and tradition in the psychology of emotions. Prentice-Hall, Inc.; 1996.

8. Craig K, Hyde S, Patrick C. Genuine, suppressed and faked facial behavior during exacerbation of chronic low back pain. Pain. 1991; 46(2):161–171. [PubMed: 1836259]

9. Dhall, A.; Asthana, A.; Goecke, R.; Gedeon, T. Emotion recognition using phog and lpq features; Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on. IEEE; 2011. p. 878-883.

10. E Lynch, M.; Craig, KD.; WH Peng, P. Clinical Pain Management: A Practical Guide. Wiley-Blackwell; 2010.

11. Fan R-E, Chang K-W, Hsieh C-J, Wang X-R, Lin C-J. Liblinear: A library for large linear classification. The Journal of Machine Learning Research. 2008; 9:1871–1874.

12. Felzenszwalb P, Girshick R, McAllester D, Ramanan D. Object detection with discriminatively trained part-based models. Pattern Analysis and Machine Intelligence, IEEE Transactions on. 2010; 32(9):1627–1645.

13. Friedman J. Greedy function approximation: a gradient boosting machine. Annals of Statistics. 2001:1189–1232.
14. Galleguillos, C.; Babenko, B.; Rabinovich, A.; Belongie, S. Proceedings of the 10th European Conference on Computer Vision: Part I. Springer-Verlag; 2008. Weakly supervised object localization with stable segmentations; p. 193-207.
15. Huang J, Craig K, Sikka K, Ahmed A, Terrones L, Littlewort G, Goodwin M, Bartlett S. Automated facial expression analysis can detect clinical pain in youth in the postoperative setting. Pediatric Academic Societies Annual Meeting. 2014
16. Kendall MG. A new measure of rank correlation. Biometrika. 1938; 30(1/2):81–93.
17. Laptev, I.; Lindeberg, T. Space-time interest points; Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. IEEE; 2003. p. 432-439.
18. Laptev, I.; Marszalek, M.; Schmid, C.; Rozenfeld, B. Learning realistic human actions from movies; Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE; 2008. p. 1-8.
19. Leung, T.; Song, Y.; Zhang, J. Handling label noise in video classification via multiple instance learning; Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE; 2011. p. 2056-2063.
20. Li, SZ.; Jain, AK.; Tian, Y-L.; Kanade, T.; Cohn, JF. Handbook of Face Recognition. New York: Springer; 2005. Facial expression analysis; p. 247-275.
21. Lucey, P.; Cohn, J.; Kanade, T.; Saragih, J.; Ambadar, Z.; Matthews, I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression; Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on. IEEE; 2010. p. 94-101.
22. Lucey, P.; Cohn, J.; Prkachin, K.; Solomon, P.; Matthews, I. Painful data: The unbc-mcmaster shoulder pain expression archive database; Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on. IEEE; 2011. p. 57-64.
23. Lucey, P.; Howlett, J.; Cohn, J.; Lucey, S.; Sridharan, S.; Ambadar, Z. Improving pain recognition through better utilisation of temporal information; In the Proceedings of the International Conference on Auditory-Visual Speech Processing; 2008.
24. Poggio, T.; Rifkin, R.; Mukherjee, S.; Rakhlin, A. Bagging regularizes. Tech. rep., DTIC Document. 2002.
25. Poppe R. A survey on vision-based human action recognition. Image and Vision Computing. 2010; 28(6):976–990.
26. Potamianos G, Neti C, Iyengar G, Senior A, Verma A. A cascade visual front end for speaker independent automatic speechreading. International Journal of Speech Technology. 2001; 4(3): 193–208.
27. Prkachin KM, Solomon PE. The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. PAIN. 2008; 139(2):267–274. [PubMed: 18502049]
28. Saeed, A.; Al-Hamadi, A.; Niese, R. The effectiveness of using geometrical features for facial expression recognition; Cybernetics (CYBCONF), 2013 IEEE International Conference on. IEEE; 2013. p. 122-127.
29. Serre, T.; Wolf, L.; Poggio, T. Object recognition with features inspired by visual cortex; Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. Vol. 2. IEEE; 2005. p. 994-1000.
30. Sikka, K.; Wu, T.; Susskind, J.; Bartlett, M. Computer Vision–ECCV 2012. Workshops and Demonstrations. Springer; 2012. Exploring bag of words architectures in the facial expression domain; p. 250-259.
31. Simon, T.; Nguyen, MH.; De La Torre, F.; Cohn, JF. Action unit detection with segment-based svms; Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE; 2010. p. 2737-2744.
32. Tax, D.; Hendriks, E.; Valstar, M.; Pantic, M. The detection of concept frames using clustering multi-instance learning; Pattern Recognition (ICPR), 2010 20th International Conference on; 2010 Aug. p. 2917-2920.

33. Taylor, GW.; Fergus, R.; LeCun, Y.; Bregler, C. Computer Vision–ECCV 2010. Springer; 2010. Convolutional learning of spatio-temporal features; p. 140-153.

34. Turk, D.; Melzack, R. Handbook of pain assessment. The Guilford Press; 2010.

35. Turk DC, Dworkin RH, et al. What should be the core outcomes in chronic pain clinical trials? Arthritis Research and Therapy. 2004; 6:151–173. [PubMed: 15225358]

36. Viola P, Platt J, Zhang C. Multiple instance boosting for object detection. Advances in neural information processing systems. 2006; 18:1417.

37. Wallhoff F. Feedtum facial expression and emotion dataset. 2004 URL http://cotesys.mmk.e-technik.tu-muenchen.de/isg/content/feed-database.

38. Wang, G.; Zhang, Y.; Fei-Fei, L. Using dependent regions for object categorization in a generative framework; Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Vol. 2. IEEE; 2006. p. 1597-1604.

39. Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; Gong, Y. Locality-constrained linear coding for image classification; Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE; 2010. p. 3360-3367.

40. Wang, X.; Wang, L.; Qiao, Y. A comparative study of encoding, pooling and normalization methods for action recognition. In: Lee, K.; Matsushita, Y.; Rehg, J.; Hu, Z., editors. Computer Vision ACCV 2012. Vol. 7726 of Lecture Notes in Computer Science. Springer; Berlin Heidelberg: 2013. p. 572-585.

41. Wohlhart P, Köstinger M, Roth P, Bischof H. Multiple instance boosting for face recognition in videos. Pattern Recognition. 2011:132–141.

42. Xiong X, De la Torre F. Supervised descent method and its applications to face alignment. CVPR. 2013

## Highlights

1. Target problem of pain classification and localization using weakly-labeled pain videos.
2. Algorithm combines multiple instance learning with multiple segment representation.
3. Rigorous experiments show that our algorithm achieves state-of-the-art performance.
4. Empirically evaluate the contributions of different components of our algorithm.
5. Visualize discriminative facial patches, as learned by our algorithm.

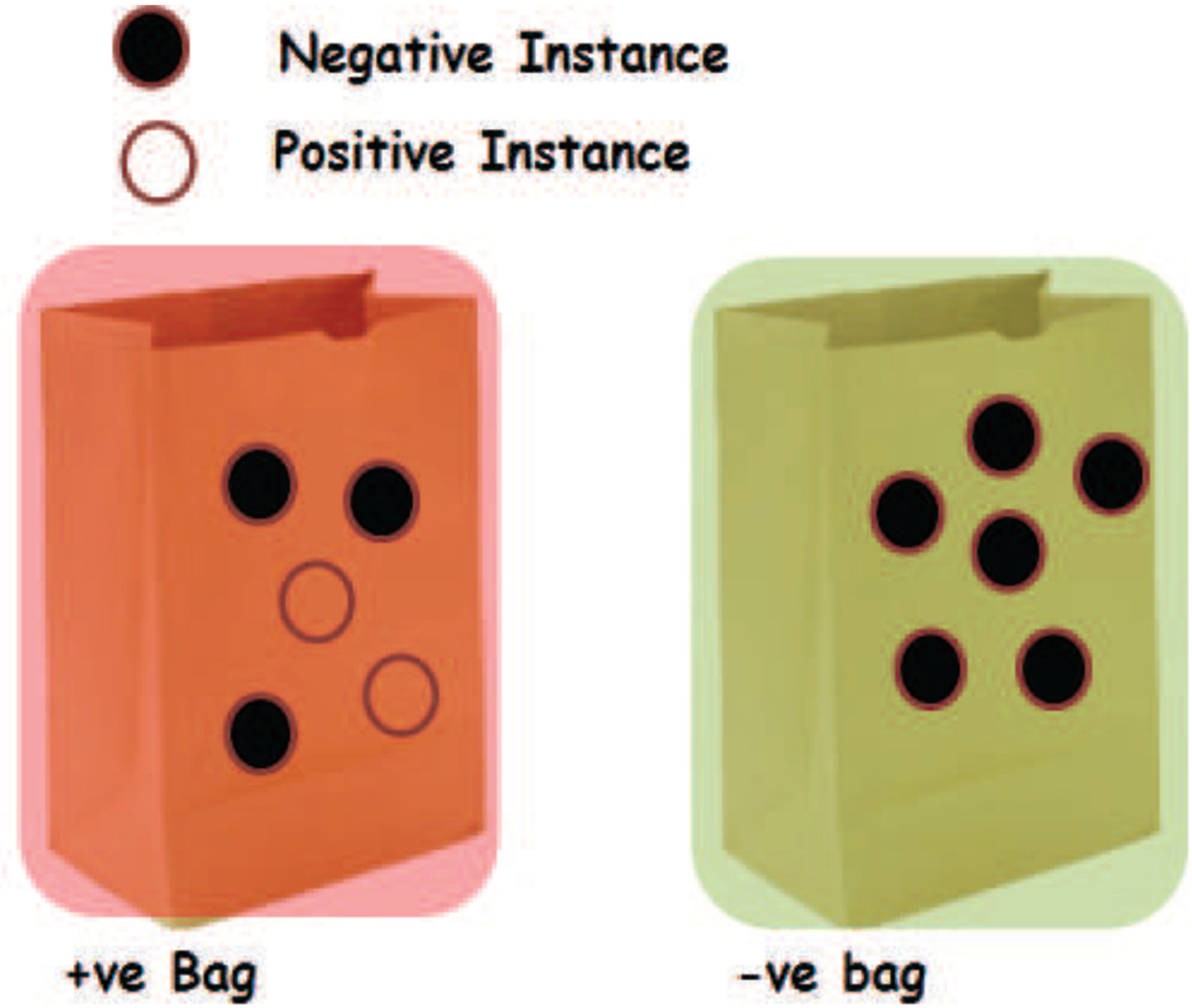
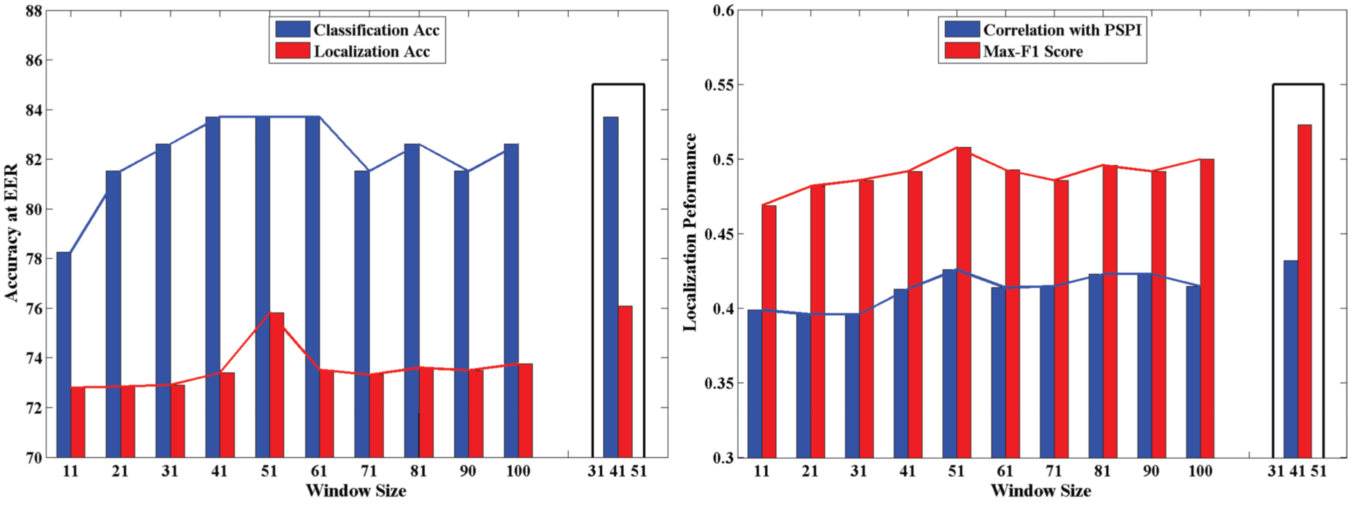


**Figure 1.**
Figure showing positive and negative bags used in MIL. A positive bag contains at least one positive instance and negative contains only negative instance.



(a) Accuracies for both tasks

(b) Performance metric for localization task

**Figure 2.**

Plots for classification and localization performance across different configurations of multiple segment representation (Section. 6.3).

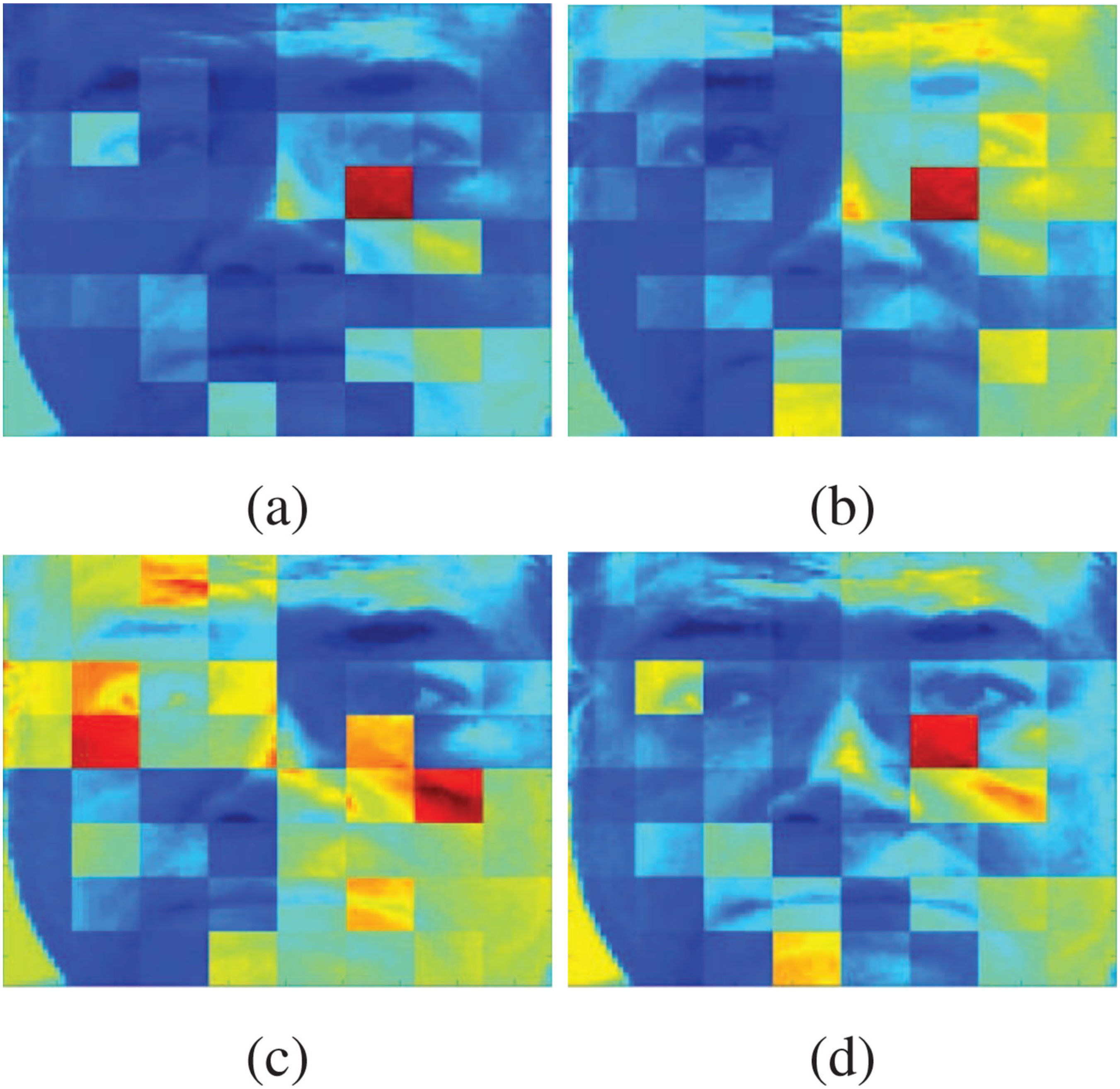


**Figure 3.**

Visualization of the weights learned by MS-MIL classifier. Fig. 3a, Fig. 3b and Fig. 3c show the weights learned by 3 individual classifier, while Fig. 3d shows the weights learned by final classifier obtained after bagging(Section. 4). Color coding is shown in Fig. 4b

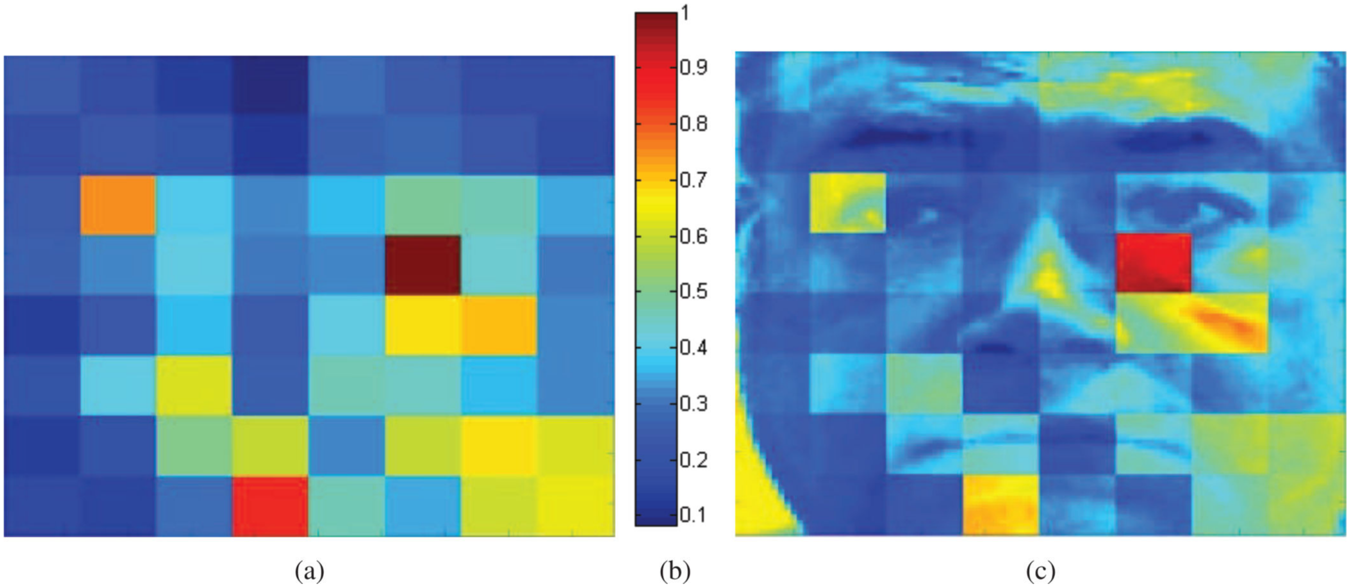
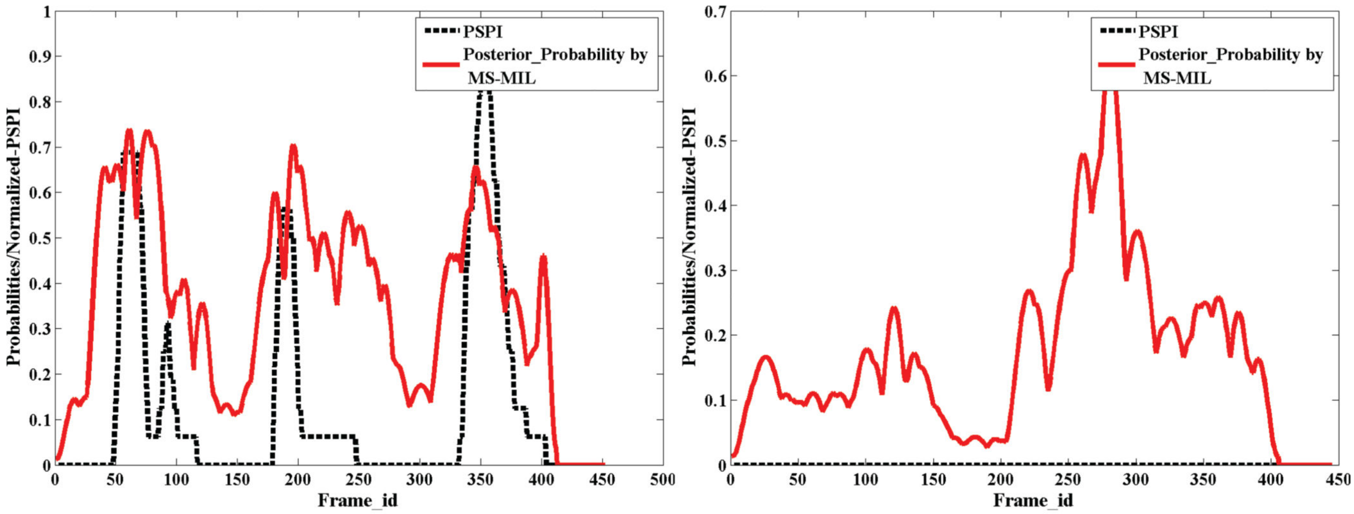


**Figure 4.**

Discriminative facial patches for pain detection as learned by our algorithm (Section7). Fig. 4a shows an intensity image with hue of the color encoding importance of each facial region as discovered by MS-MIL. The colorbar is shown in Fig. 4b, with blue and red denoting lowest and highest weights respectively. Fig. 4c shows the same intensity image overlaid over a subject’s image for better visualization.



(a) Case1: Pain expression with multiple occurences

(b) Case2: Case where PSPI index showed no pain but MS-MIL could correctly localize pain.

**Figure 5. Pain Localization**

Example showing the performance of our algorithm for pain localization vs ground-truth frame labels (PSPI).

### Table 1

Formulation of different soft-max functions along with $w_{ij}$ in each case.

| **Soft-max** | $g(p_{ij})$ | $w_{ij}$ |
|---|---|---|
| NOR | $1 - \prod_j (1 - p_{ij})$ | $\frac{t_i - p_i}{p_i} p_{ij}$ |
| GM | $(\frac{1}{n} \sum_k p_{ij}^r)^{\frac{1}{r}}$ | $\frac{1 - p_i(2 - t_i)}{1 - p_i} \frac{p_{ij}^r - p_{ij}^{r+1}}{\sum_j p_{ij}^r}$ |

## Table 2

Comparison of MS-MIL with different algorithms for pain classification in videos.

| Method | Accuracy (%)<br>(at EER) | #subjects-<br>#samples |
|---|---|---|
| Lucey et al. [23] | 80.99 | 20 – 142 |
| Ashraf et al. [2] (shown in [23]) | 68.31 | 20 – 142 |
| MS-SVM$_{max}$ | 77.17 | 23 – 147 |
| MS-SVM$_{avg}$ | 71.73 | 23 – 147 |
| BoW+Avg+SVM [40] | 66.30 | 23 – 147 |
| BoW+Max+SVM [40] | 81.52 | 23 – 147 |
| **MS-MIL** | **83.7** | 23 – 147 |

## Table 3

Comparison of MS-MIL with different methods for the pain localization.

| Method | Localization Accuracy (%) | Correlation | Max-F1 | Video-level Classification accuracy (%) |
|---|---|---|---|---|
| MS-SVM$_{max}$ | 72.64 | .390 | .471 | 77.17 |
| **MS-MIL** | 76.08 | .432 | .523 | 83.70 |
| frame-SVM$^1$ | 70.47 | .385 | .477 | 84.78 |
| frame-SVM$^2$ | 66.76 | .282 | .403 | 73.91 |

## Table 4

Evaluation across different methods for generating multiple segments. Classification accuracy is reported in %.

| Setting | MS-MIL | | MS-SVM$_{max}$ | | MS-SVM$_{avg}$ | |
|---|---|---|---|---|---|---|
| | **Ncuts** | **scan-wind** | **Ncuts** | **scan-wind** | **Ncuts** | **scan-wind** |
| *short* | 78.26 | 78.16 | 78.26 | 73.91 | 73.91 | 73.91 |
| *medium* | 82.61 | 83.7 | 78.26 | 77.17 | 70.65 | 75.00 |
| *long* | 80.43 | 82.61 | 79.34 | 73.91 | 75.00 | 75.00 |
| *combine* | 83.70 | 83.7 | 77.17 | 76.08 | 71.73 | 73.91 |

## Table 5

Experiments on FEEDTUM dataset highlighting the generality of MS-MIL. Classification accuracy is reported in % for six different facial expressions. p-value for the paired ttest between the two classification accuracies is 0.0162, showing that the difference is significant.

| Algo | Anger | Disgust | Fear | Hapiness | Sadness | Surprise | All |
|---|---|---|---|---|---|---|---|
| geom.+MilBoost | 79.86 | 80.95 | 87.88 | 83.98 | 77.20 | 80.73 | 81.78±1.31 |
| geom.+MS-MIL | 84.28 | 85.41 | 85.01 | 86.36 | 83.04 | 83.18 | 84.55±0.98 |