Abstract

This paper presents a novel method for automatic spotting (temporal segmentation) of facial expressions in long videos comprising of continuous and changing expressions. The method utilizes the strain impacted on the facial skin due to the non-rigid motion caused during expressions. The strain magnitude is calculated using the central difference method over the robust and dense optical flow field of each subjects face. Testing has been done on 2 datasets (which includes 100 macro-expressions) and promising results have been obtained. The method is robust to several common drawbacks found in automatic facial expression segmentation including moderate in-plane and out-of-plane motion. Additionally, the method has also been modified to work with videos containing micro-expressions. Micro-expressions are detected utilizing their smaller spatial and temporal extent. A subject’s face is divided in to sub-regions (mouth, cheeks, forehead, and eyes) and facial strain is calculated for each of these regions. Strain patterns in individual regions are used to identify subtle changes which facilitate the detection of micro-expressions.

1. Introduction

Expression segmentation is vital for many applications that analyze temporal facial changes, including micro and macro expression recognition, genuine and feigned emotion detection, animation, artificial expression generation, among others. Largely unsolved problems that need to be addressed include large movements of the head, which can include large translation, out of plane rotations, and uneven lighting conditions. In this paper, we propose a two step approach to segment temporal expressions from face videos. First, facial strain maps are calculated based on the facial deformation observed in video sequences using the methods described in [1][2]. Next, the strain magnitude is calculated and used to segment expressions from the video. This approach has the advantage of eliminating motion vectors due to in-plane head movement, thus making the method more stable in real-world scenarios.

One way to analyze the observed facial tissue deformation is to calculate optical strain patterns under the application of a force, a condition that naturally happens during facial expressions [1]. This has several advantages: (i) the strain pattern can easily be used in conjunction with existing face recognition methods; (ii) the strain pattern is related to the biomechanical properties of facial tissues, and (iii), as is demonstrated empirically, the strain pattern is stable under lighting variations.

There are two main approaches for calculating optical strain: (1) Recover the absolute values of elastic moduli by solving an inverse problem; (2) Compute strain from the measured displacement (motion) observed in the video sequence, and use the spatial variation as an indicator of the underlying tissue properties. Since an inverse problem is often ill-posed and highly nonlinear, the computational complexity of the first approach is relatively high. Various regularization techniques must then be used to stabilize an inverse solution [3]. The second approach is essentially a forward problem and can be implemented with conventional imaging methods.

In this paper, we chose the second approach and used a robust optical flow method [4] for calculating displacement. The central difference method is then used to estimate optical strain. This method has several advantages: (i) it is efficient and can be used to process large amounts of video in a reasonable time framework. With further optimization, it can be considered for real time; (ii) video data can be acquired using standard optical camorders; (iii) because of its non-invasive imaging nature, this approach is suitable for security checkpoints, access control and surveillance applications.

2. Related Work

In [4], De la Torre et al. addressed the generalized problem of segmenting facial behavior. As pointed out in their work, very little research has been conducted on temporal segmentation in face videos. However, a substantial amount of work has been done on hand-segmented expression recognition, including both temporal and static approaches [6][7][8].
In [1][2], facial strain patterns were shown to be useful as supplementary biometric evidence for forensic identification. It was shown that adverse lighting conditions, which typically reduce accuracy in appearance based recognition methods, were not nearly as severe on strain map computations. Moreover, it was shown that the strain maps were consistent across conditions where the subject was both wearing and not wearing camouflage.

Using FACS and optical flow, Essa and Pentland [9] were able to develop two models (muscle and motion energy) for representing facial motion, and were able to achieve high classification accuracy for expression recognition. However, their results are based on pre-segmented videos and do not address the problem of initially spotting the expressions.

Stephen et al [10] conducted one of the earliest experimental studies to validate the existence of micro-expressions. They concluded that it is important to analyze the upper and lower halves of the face separately. This conclusion is supported by the experiments performed in [11], since the authors segmented the face into upper and lower regions in order to recognize micro-expressions.

3. Expressions

Facial expressions can be both voluntary and involuntary, and usually convey the emotional state of an individual. In particular, we are interested in expressions with facial deformation corresponding to the activation of facial muscles. The six universal facial expressions are happy, sad, fear, surprise, anger, and disgust [12]. Expressions can vary in intensity and time, and naturally many combinations and mixtures of these expressions can occur. In this paper, we distinguish between expressions which are easily noticed and occur over a substantial period of time (macro-expression) and those that occur over a brief period of time and can commonly go unnoticed (micro-expression).

3.1. Macro-Expressions

In general, the dynamics of a facial expression consists of three main phases: start, peak, and stop [6]. These phases occur for any expression, either at the macro or micro level. In our data collected, the macro level expressions are typically between three fourths of a second to two second in duration (10 - 60 frames). Hence, these expressions are generally easy to manually identify in videos. The identifying characteristics of a macro level expression can be found over the entire face depending on the expression. For instance, surprise, fear, disgusts, and anger generally creates more facial motion than sadness and smiling. In contrast, micro-level expressions rarely cause much motion except in the forehead and eye regions [10].

3.2. Micro-Expressions

A micro-expression is a short lived, mostly involuntary facial expression, shown when a person is trying to hide ones true emotions [13][14]. Micro-expressions usually last anywhere between one fifth to one twenty-fifth of a second before it is recognized and suppressed [10]. The identifying characteristic of a micro-expression is its duration and spatial locality [10][11]. Micro-expressions last for very short a period and usually occur in one part of the face which makes it hard to detect them with the naked eye. Microexpressions can internally be classified into three major types [15]: Simulated Expressions: when a microexpression is not accompanied by a genuine expression. Masked Expressions: When a genuine expression is replaced by a falsified expression and Neutralized Expressions: When a genuine (emotional) expression is suppressed and the face remains neutral.

4. Method

Two solution strategies can be employed to estimate strain magnitude. The first is to integrate the strain definition into the formulation of optical flow equations so that strain can be derived directly from the image intensity. This approach skips intermediate steps and is potentially more efficient. However, computing high order derivatives from original images without appropriate processing could cause numerical difficulties because such a solution is very sensitive to image noise. The second approach is to compute motion and strain separately. This approach is more flexible because it allows us to examine the quality of motion data before they are processed by the strain filters. In this paper, we use the second approach.

4.1. Compute Motion Using Robust Optical Flow

Optical Flow is a well known motion estimation technique that is based on the brightness conservation principle [4]. Two conditions must be satisfied in order to obtain a reliable solution: (i) the intensity of a point on a moving object remains constant across a pair of frames, and (ii) pixels in a small image window move with a similar velocity. The optical flow equation is often expressed as:

\[(\nabla I)^T \mathbf{p} + I_t = 0, \quad (1)\]

where \(I(x,y,t)\) is the image intensity as a spatial and temporal function, \(x\) and \(y\) are the image coordinates and \(t\) is time. \(\nabla I\) and \(I_t\) are the spatial and temporal gradients of the intensity function. \(\mathbf{p} = [p = \partial x / \partial x, q = \partial y / \partial x]^T\) denotes horizontal and vertical motion.

To ensure the high quality of motion data, we experimented with a few different implementation
methods as discussed in [16]. We found that the method formulated in a robust estimation framework yielded consistent and reliable results [4] Therefore, we used this method to generate all motion data for the subsequent strain computations.

4.2. Strain Computation Using Finite Difference Method

Considering a deformable object in two dimensional space, its motion can be described by a displacement vector \( \mathbf{u} = [u, v]^T \). Assuming a small motion, a finite strain tensor can be defined as:

\[
\varepsilon = \frac{1}{2} [\nabla \mathbf{u} + (\nabla \mathbf{u})^T],
\]

or in an expanded form:

\[
\varepsilon = \begin{bmatrix}
\varepsilon_{xx} &= \frac{\partial u}{\partial x} \\
\varepsilon_{yy} &= \frac{\partial v}{\partial y} \\
\varepsilon_{xy} &= \frac{1}{2} \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right)
\end{bmatrix},
\]

Where \( \varepsilon_{xx}, \varepsilon_{yy} \) are normal strain components and \( \varepsilon_{xy} \) are shear strain components.

Since strain is defined with respect to the displacement vector \((u, v)\) in continuous space, we make the following approximation in order to estimate strain from the discrete optical flow data \((p, q)\):

\[
p = \frac{dx}{dt} = \frac{\Delta x}{\Delta t}, \quad u = p \Delta t,
\]

where \( \Delta t \) is the elapsed time between two image frames.

If we compute the optical flow and strain using a fixed frame interval throughout a particular video sequence, we can treat \( \Delta t \) as a constant and estimate the partial derivatives as follows:

\[
\frac{\partial u}{\partial x} = \frac{\partial p}{\partial x} \Delta t, \quad \frac{\partial u}{\partial y} = \frac{\partial p}{\partial y} \Delta t,
\]

\[
\frac{\partial v}{\partial x} = \frac{\partial q}{\partial x} \Delta t, \quad \frac{\partial v}{\partial y} = \frac{\partial q}{\partial y} \Delta t,
\]

The above computation scheme can then be implemented by using any spatial derivate over a finite number of points such as the forward difference method, central difference method, or the Richardson extrapolation method. We chose the central difference method due to its accuracy and efficiency. Then,

\[
\frac{\partial u}{\partial x} = \frac{u(x + \Delta x) - u(x - \Delta x)}{2\Delta x} = \frac{p(x + \Delta x) - p(x - \Delta x)}{2\Delta x} \quad (7)
\]

\[
\frac{\partial v}{\partial y} = \frac{v(y + \Delta y) - v(y - \Delta y)}{2\Delta y} = \frac{q(y + \Delta y) - q(y - \Delta y)}{2\Delta y} \quad (8)
\]

where \((\Delta x, \Delta y)\) are preset distances of 2-3 pixels.

Under the uniform stress, large strain values correspond to low elastic moduli and vice versa. Therefore, elastograms based on the absolute strain value or relative strain ratio can be used to reveal underlying elastic property changes. For this purpose, we compute a strain magnitude as follows:

\[
\varepsilon_m = \sqrt{\varepsilon_{xx}^2 + \varepsilon_{yy}^2 + 2 \varepsilon_{xy} \varepsilon_{yx}}. \quad (9)
\]

5. Algorithm

The algorithm consists of five stages: first, optical flow is computed between consecutive pairs of images over the entire video sequence using the Matlab version of Michael Blacks optical flow code available at his website [17]. Then vector linking is used to calculate the flow between the starting frame of each sequence to all other frames in the same sequence. Next, the optical strain magnitude is calculated over each flow field (e.g. Frame\(i\)-Frame\(j\), \(i \in [2, n]\)), where \(n\) is the number of frames in the sequence. Lastly, the strain magnitude calculated over the sequence is then segmented using a local threshold strategy.

The video sequence is divided into separate intervals \([a, b]\) every \(m\) frames in the video sequence, starting at the first frame. Then a threshold \(T\) is calculated based on a percentage \(p\) of the maximum minus the minimum values observed in each interval. Hence, given:

\[
S_{\text{max}} = \text{Max} \{S_j\}_{j=a}^b \quad (10)
\]

\[
S_{\text{min}} = \text{Min} \{S_j\}_{j=a}^b \quad (11)
\]

where \([b-a] = m\), then the threshold \(T\) is defined as

\[
T = S_{\text{min}} + p \times (S_{\text{max}} - S_{\text{min}}), p \in (0, 1) \quad (12)
\]

Hence, time points at which the strain magnitude rises above this threshold is considered for being a macro-expression. In order to reduce false positives from small
peaks and valleys, a global threshold is also used to
determine a minimum strain magnitude acceptable for an
expression, and is calculated by taking a percentage \( \varepsilon \) of
the max strain magnitude observed over the entire
sequence.

The method for recognizing micro-expressions is similar,
except for a few key differences. The major difference is
that spatial regions are considered, hence separate strain
values are calculated for the forehead, cheeks, and mouth
regions of the face. Moreover, temporal regions that have
been spotted to be a macro-expression are removed. This
in turn decreases the variability of the strain magnitude
and increases the sensitivity to temporally quick facial
movements. Another difference is that the micro-
expression should last a certain number of frames. A
temporal threshold \( \delta \) is used to define the max number of
frames a micro expression can last. Finally, micro-
expressions should only occur in one or two regions of the
face. Hence, if a high strain magnitude is detected in more
than two regions of the face, it is not considered to be a
micro-expression.

6. Experiment and Evaluation

6.1. Data Sets

Two datasets were used for experimentation. The first
dataset (BU) [18] consists of 40, frontal-view macro-
expressions with moderate head translations. The second
dataset (USF) consists of 60 macro-expressions, which are
performed in video sequences around a minute in length,
and includes moderate to high amounts of head
translations.

6.2. Strain Magnitude Characteristics

As described in Section 3, an expression consists of
start, peak, and stop phases. The strain magnitude over the
expression contains these attributes in Figure 2. This
figure also illustrates the effectiveness of using optical
strain by comparing the magnitude of both optical flow
and optical strain calculated over a video containing back
and forth head motion. In Figure 2.a, the observed
expression cannot be easily determined by the optical flow
magnitude. However, using the strain magnitude (Figure
2.b), three peaks can be seen for each expression, and
hence were accurately spotted using the algorithm given in
Section 5.

6.1. Spotting Results

Detected expression intervals are compared with
manually defined ground truth intervals, and considered
spotted if it falls within a buffer of two thirds of a second
(approx. 20 frames) before and after the endpoints.
Otherwise, it is considered missed. For macro-expression
spotting, we found that \( p = .2 \), and \( \varepsilon = .5 \) led to promising
results, within local range of 50 frames (\( m = 50 \)). Figure 4
illustrates two cases: one where the algorithm was
successful and one producing errors.
Figure 4. Example cases where spotting was successful (a) and where it produced errors (b). Dashed line indicates where threshold was calculated. In (a), the threshold creates a successful split. However, in (b), large and fast head translation and facial movement along with merging expressions resulted in noisy strain magnitude calculations.

Table 1. Macro-Expression Spotting

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Total Exp.</th>
<th>Spotted</th>
<th>Missed</th>
<th>False Spot</th>
</tr>
</thead>
<tbody>
<tr>
<td>BU</td>
<td>60</td>
<td>60</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>USF</td>
<td>40</td>
<td>29</td>
<td>11</td>
<td>8</td>
</tr>
</tbody>
</table>

All expressions were spotted in the BU dataset, which contained 60 expressions. Faces contained only moderate head translations. The accuracy on the USF dataset was much lower due to much larger head movements over a longer period of time. Possible improvements for this method are discussed in the next section.

Figure 5. Strain Magnitude of sequence which contained micro-expressions. Dashed lines indicate global threshold used.

Table 2. Micro-Expression spotting.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Total Exp.</th>
<th>Spotted</th>
<th>Missed</th>
<th>False Spot</th>
</tr>
</thead>
<tbody>
<tr>
<td>USF</td>
<td>7</td>
<td>7</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Micro expression spotting was performed on a video from the USF video collection. Figure 5 shows the thresholds for an example sequence which contained micro-expressions. The spotter achieved 100% accuracy in detecting all hand marked ground truth micro-expressions, however one false spot was also reported (Table 2). For these experiments, \( p \) was also .2, and \( \delta = 20 \) (max number of frames allowed for a micro-expression).

7. Discussion and Conclusions

In this paper, a novel method for automatic spotting facial expressions in video sequences was presented. Testing was performed on a two datasets containing 100 expressions, and the initial results are positive. Furthermore, the algorithm was modified to work with micro-expressions and initial results look promising. The method is robust to moderate head translation, however, large and fast head translation and facial movement still pose a problem. This is due, at least in part, to the small motion assumption inherent with calculating optical flow. One possible solution that we will explore is using an eye
tracking algorithm to accurately segment the facial region from the rest of the image. This should have the dual benefit of decreasing computation time and increasing robustness to larger head movements.

Acknowledgment
This work was supported in part by the USF Computational Tools for Discovery Thrust and by the Graduate Multidisciplinary Scholars Program.

References