Parametric temporal alignment for the detection of facial action temporal segments

Bihan Jiang[1]
bi.jiang09@imperial.ac.uk
Brais Martinez[1]
b.martinez@imperial.ac.uk
Maja Pantic[12]
m.pantic@imperial.ac.uk

[1] Computing Department
Imperial College
London, UK

[2] Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
Netherlands

Abstract

In this paper we propose the very first weakly supervised approach for detecting facial action unit temporal segments. This is achieved by means of behaviour similarity matching, where no training of dedicated classifiers is needed and the input facial behaviour episode is compared to a template. The inferred temporal segment boundaries of the test sequence are those transferred from the template sequence. To this end, a parametric temporal alignment algorithm is proposed to align a single exemplar sequence to the test sequence. The proposed strategy can accommodate flexible time warp functions, does not need to exhaustively align all frames in both sequences, and the optimal warp parameters can be found by an efficient Gauss-Newton gradient descent search. We show that our approach produces the best results to date for the problem at hand, and provides a promising opportunity to studying facial actions from a new perspective.

1 Introduction

Facial expression recognition has been an active topic in computer vision due to its wide applications in human-computer interaction, security, and health care. The Facial Action Coding System (FACS) [6] is one of the most comprehensive and objective ways to describe facial expressions. It associates facial expressions with actions of muscles that produce them by defining a set of atomic movements called Action Units (AUs). The dynamics of facial expressions are crucial for the interpretation of human facial behaviour [7]. The most common way of modelling facial dynamics is by detecting the boundaries of temporal segments (namely neutral, onset, apex and offset) of each AU activation. This article presents a method for automatic detection of AU temporal segments in a novel way, by means of behaviour similarity matching, where no training of dedicated classifiers is needed and the input facial behaviour episode is compared to a template for similarity measurement.

The temporal segments are defined as the *increasing*, *stabilising*, and *relaxation* of the facial muscle related to an AU. They are therefore by definition dynamic events that cannot be effectively analysed at a frame level. More formally, if we consider the intensity of an

action unit as a function depending on time, then analysing the temporal segments corresponds to studying the sign of its derivatives. As a consequence, we intend to reason about the temporal segments of an AU by analysing the facial action as a whole.

The majority of methods related to automatic analysis of AUs break the problem down so that the patterns used for learning relate only to a single frame. AU detection is typically attained by training a frame-level binary classifier for the targeted AU, combined with information on temporal correlation of the data, e.g. by using a graphical model like Conditional Random Field (CRF) [15] or Hidden Markov Model (HMM) [14]. Some works also propose to use features encoding temporal information. For example, [12] proposed to use motion features, designed to capture the flexible face motion pattern typical of the target AU, while [10] and [13] proposed to use dynamic appearance descriptors to this problem. The latter captures the appearance of a spatio-temporal volume defined as a temporal neighbourhood of a frame, therefore having enough information to reason about the AU temporal segments.

Here we propose the very first weakly supervised approach to encoding facial behaviour dynamics. More specifically, we formulate the "behaviour similarity problem" in which the aim is to address the question "are these two behaviours similar?" instead of "what is the displayed behaviour?". Pioneering efforts to address such research questions in areas such as face verification [8] and human bodily action recognition [2] have been recently reported. However, the concept of facial behaviour similarity has not been previously defined in the literature. The main value of such an approach is that it could represent the solution to the long-standing problem in machine analysis of facial behaviour – the lack of annotated data to learn from. To wit, in contrast to the standard approach, in which machine learning algorithms are trained using (usually unavailable or excessively expensive) large amounts of facial recordings manually annotated in terms of the target behaviour (e.g. offset of smile, onset of blink, apex of eyebrow raise, etc.), in the behaviour-similarity-matching paradigm, minimal annotation of training data is needed (it is only required to pinpoint "typical" example(s) of the target behaviour) and "templates" of the target behaviour are compared to the currently observed behaviour for similarity measurement.

To this end, a parametric temporal alignment algorithm is proposed to align between a single exemplar of the target action and the test sequence. The Dynamic Time Warping [11] and related methods are the *de facto* standard for computing an explicit time alignment. However, they show some inherent restrictions that hinder its application to our problem, such as the requirement to align the full sequences, and the lack of flexibility of the time warp function. We propose a novel parametric temporal alignment approach by extending the framework in [5]. In [5], the authors define a linear parametric time warp and find the optimal parameters though a gradient descent strategy. The major limitation of this methodology is its reliance on frame interpolation, and the coarseness of the numerical approximations necessary to compute the derivatives. We extend here this method to accommodate for a more general family of time warp functions. Further, we combine this with the framework presented in [17], which can be used to avoid frame interpolation and to obtain analytical approximations of the derivatives of the frame appearance with respect to time. While the authors of [17] only considered temporal scaling and translation, we propose here richer temporal transformations that can account for the intra-action structure. We also substitute the sliding window exhaustive search approach by an efficient Gauss-Newton gradient descent method. Finally, two novel alignment strategies tailored to the problem at hand are provided.

The remainder of this paper is organised as follows. The problem at hand is formally stated in Sec. 2. Then for completeness, the work of [17] is reviewed in Sec 3. Sec. 4 defines specific loss and warp functions for the problem of AU temporal segment detection

and finally the Gauss-Newton parameter search is presented in Sec. 5 .

2 Problem formulation

Our aim is to find the boundaries of the AU temporal segments by aligning the test sequence with a sequence containing the activation of the target AU with known temporal segment boundaries. More specifically, we assume the existence of one such sequence, to which we refer to the template sequence, noted $\mathbf{X}^{\text{tmpl}} = \{\mathbf{x}_1^{\text{tmpl}}, \ldots, \mathbf{x}_n^{\text{tmpl}}\}$[1]. For example, $\mathbf{X}^{\text{tmpl}}$ might contain a full episode of an AU so that the onset, apex and offset of the AU are fully included in the sequence. Assume as well that the temporal boundaries of each of the temporal segments within the template sequence are known. Our approach relies on finding an explicit temporal alignment between the template sequence and the test sequence, and then transferring the temporal boundaries from the template sequence to the test sequence. This is a totally novel way of tackling this problem by behaviour similarity matching, being therefore one of the main contributions of this work.

We also present a novel parametric temporal alignment algorithm to align two sequences. To this end, we consider a family of possible temporal warp functions, W, parametrised by means of θ. Then temporal alignment results in warping one sequence onto the other by means of the function $W(-;\theta)$. For example, if we consider W to be a linear warp, and we parametrise it using translation and speed (scale) parameters, then a time stamp i is warped into $W(i;\theta) = \theta_1 i + \theta_2$, aligning frame i from one sequence to frame $W(i;\theta)$ from the other. If we aim to warp the template into the test sequence, we can define the alignment error as the accumulation of the frame-to-frame alignment errors as

$$\sum_i \mathcal{L}\left(\mathbf{x}_i^{\text{tmpl}}, \mathbf{x}^*\left(\mathcal{W}(i;\theta)\right)\right) \tag{1}$$

where $\mathbf{x}^*\left(\mathcal{W}(i;\theta)\right)$ notes the appearance of the test sequence at time $\mathcal{W}(i;\theta)$. The temporal alignment results from minimising this loss w.r.t. the parameters θ.

It is clear however that $\mathbf{x}^*\left(\mathcal{W}(i;\theta)\right)$ is typically unobserved, and that frame interpolation is thus required for its computation. In order to overcome this problem, we propose to make use of the framework presented in [17]. In particular, this framework proposes a graph embedding technique that, given a sequence $\mathbf{X}$, defines a continuous and differentiable function $\mathcal{X}(t), t \in \mathbb{R}$ that can be easily evaluated and that provides an approximation of the original frames at corresponding time stamps, i.e., $\mathcal{X}(i) \approx \mathbf{x}_i$, $i = 1, \ldots, n$.

3 Graph embedding representation

For completeness, we review in here the graph embedding framework proposed in [17]. The authors make use of the Laplacian eigenmaps technique [4] to find the projection of a set of data points onto a lower dimensional space, so that the projected points respect the *distances* within the original space with respect to neighbouring points. Neighbouring relations are established by means of a graph, so nodes correspond to data points, and edges connect two nodes if their corresponding points are considered to be neighbouring. Given a graph,

[1] We use bold lower-case symbols to note vectors (understood as column vectors), bold upper-case symbols to denote matrices, and calligraphic upper-case symbols to denote functions

the mapping of the original points results from the computation of the eigenvectors of the Laplacian of the graph. The Laplacian is defined as matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}_{i,i} = \sum_j \mathbf{W}_{i,j}$, and $\mathbf{W}_{i,j}$ denotes the weight of the edge between nodes i ad j (for unweighted graphs $\mathbf{W}_{i,j} = 1$ if the edge exists, 0 otherwise). The projected points, $\mathbf{y}_i$, minimise $\sum_{i,j}^{n}(\mathbf{y}_i - \mathbf{y}_j)^2\mathbf{W}_{i,j}$. It is interesting to note that this technique directly yields the projected points, while the explicit mapping between the original data and the projected one is not computed.

In here a sequence is represented as an unweighted undirected chain graph. Because of the specific case of graph topology considered, it is possible to show that the eigenvectors of the Laplacian can be computed as the following set of trigonometric functions

$$\mathbf{e}_k^n(u) = sin(\pi ku/n + \pi(n-k)/2n) \tag{2}$$

where $u = 1, 2, \ldots, n$, n being the number of frames in the sequence, and $k = 1, 2, \ldots, n-1$.

Then the projection of the i-th datapoint, $\mathbf{x}_i$, can be computed as the i-th elements of the eigenvectors. By performing a simple variable substitution so $t = u/n$ in the above equation, we can write:

$$\mathbf{y}_i = \left(\mathbf{e}_1^n(t = i/n), \ldots, \mathbf{e}_{n-1}^n(t = i/n)\right) \tag{3}$$

That is to say, if the eigenvectors are stored as columns in a matrix, then the rows of the matrix are the points projected into the lower-dimensional space. However, due to the parametric form of the eigenvectors, this definition can be extended to values of t that do not correspond to the original frames in the sequence

$$\mathcal{Y}(t) = \left(\mathbf{e}_1^n(t), \ldots, \mathbf{e}_{n-1}^n(t)\right) \quad t \in [0,1] \tag{4}$$

By this definition, $\mathcal{Y}(t)$ is a parametric one-dimensional curve within $\mathbb{R}^{n-1}$, and it is a parametric **continuous** representation of the action.

The rest of the framework is based the explicit computation of the linear mapping, which is done following the derivations in [9] and, on inverting this mapping as to be able to back-project any point of the curve back into frame space. The reader is referred to [17] for further details. This results in a linear function

$$\mathcal{X}(t) = A\mathcal{Y}(t) + \bar{\mathbf{x}} \tag{5}$$

where $\mathcal{X}(i/n) \approx x_i$, and where $\bar{\mathbf{x}}$ denotes the mean frame in the sequence. That is to say, we are capable of synthesising the in-between-frames appearance as cheaply as evaluating the trigonometric function $\mathcal{Y}(t)$ and multiplying the outcome by a matrix. As warned in [17], it is assumed that the data points $\mathbf{x}_i$ are linearly independent. This assumption usually holds when dealing with high-dimensional data such as images.

4 Proposed warp functions

In here we describe the proposed parametric alignment algorithm. To this end, we need to specify the family of valid warping functions, W, and the loss function employed. We define two different alignment strategies:

Model 1: This first model considers the template to be a representative instance of the full activation process of the target AU. Each segment is composed of n_{on}, n_{ap} and n_{off} frames. It is also assumed that the transition points between stages are known.

As shown in Fig 1, A piecewise linear model is used to map the template into a sub-sequence of the test sequence. More specifically, θ is a 4-dimensional parameter vector indicating the points within the test sequence to which the phase limits are aligned to. In this case it is necessary to project the test sequence, obtaining the function $\mathcal{X}^*(t)$. The loss function is defined as

$$\hat{\theta} = \arg\min_{\theta} \sum_{i}^{n} \|\mathbf{x}_i^{\text{tmpl}} - \mathcal{X}^*\left(W(i;\theta)\right)\|_2^2 \tag{6}$$

where $W(i;\theta)$ is defined as:

$$W(i;\theta) = \begin{cases} \frac{\theta_2-\theta_1}{n_{\text{on}}} i + \theta_1 & : \theta_1 \leq i < \theta_2 \\ \frac{\theta_3-\theta_2}{n_{\text{ap}}} i + \theta_2 & : \theta_2 \leq i < \theta_3 \\ \frac{\theta_4-\theta_3}{n_{\text{off}}} i + \theta_3 & : \theta_3 \leq i \leq \theta_4 \end{cases} \tag{7}$$

Model 2: Our first model aligns the full exemplar to the test sequence despite both AU potentially reaching different amplitudes. In other words, the maximum AU intensity levels in both sequences is in general different. Now we propose a model capable of accounting for this variation. To this end, we consider a template that contains an AU activation from the neutral to the apex. That is to say, the apex and offset frames are not included. Furthermore, the template exemplar should attain a high-intensity apex. Similarly as before, the warp is defined to be piecewise linear. However, we include a variable that implicitly accounts for the maximum intensity of the test sequence. Thus, θ is 5-dimensional. Furthermore, instead of aligning the template to a sub-sequence of the test sequence, we align a sub-sequence of the template to the full test sequence. More specifically, we project the template sequence to obtain the function $\mathcal{X}^{tmpl}(t)$ and use the loss function defined as:

$$\hat{\theta} = \arg\min_{\theta} \sum_{i}^{m} \|\mathcal{X}^{\text{tmpl}}\left(W(i;\theta)\right) - \mathbf{x}_i^*\|_2^2 \tag{8}$$

where the warp $W(i;\theta)$ function given by the following piecewise function:

$$W(i;\theta) = \begin{cases} 0 & : i < \theta_1 \text{ or } \theta_4 \leq i \\ \frac{\theta_5}{\theta_2-\theta_1}(i-\theta_1) & : \theta_1 \leq i < \theta_2 \\ \theta_5 & : \theta_2 \leq i < \theta_3 \\ -\frac{\theta_5}{\theta_4-\theta_3}(i-\theta_3)+\theta_5 & : \theta_3 \leq i < \theta_4 \end{cases} \tag{9}$$

As we can see in Fig. 1, we align a portion of the template sequence until a point delimited by θ_5. Ideally, at this point the template sequence should be of the same intensity as the apex of the test sequence. Then, the apex segment of the test sequence is aligned to the same point of the template sequence. Finally, the offset corresponds to aligning the same sub-sequence of the template we used for the onset, but traversed backwards. Under this model, any frame of the template beyond the one defined by θ_5 is not used in the alignment, while the offset is restricted to be a mirrored version with different speed of the onset segment.

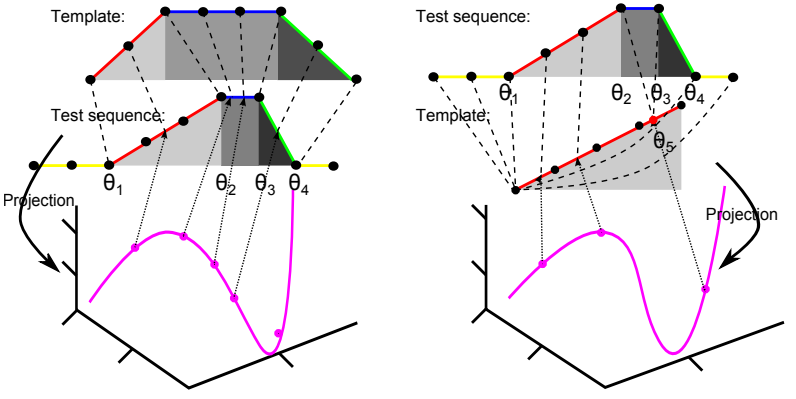

Figure 1: Depiction of the temporal alignment strategy for both of the models presented here (left: model1, right: model2).

5 Gauss-Newton parameter search

The temporal alignment results from the minimisation of the loss function with respect to the warp parameters. In order to minimise the expression, we follow a gradient descent procedure parallel to that in [3]. The differentiability of the loss function w.r.t. the warp parameters is guaranteed by the smoothness of $\mathcal{Y}$ and $\mathcal{W}$. We use the shorthand $\mathcal{L}(\theta)$ to refer to the loss function specified by Eq. 6 (the derivations in the case of the other model are equivalent).

The Gauss-Newton gradient descent approach only requires the computation of the Jacobian of the loss function, $\nabla\mathcal{L}(\theta) = \left(\frac{\partial\mathcal{L}(\theta)}{\partial\theta_1},\ldots,\frac{\partial\mathcal{L}(\theta)}{\partial\theta_n}\right)$. It can be computed as follows.

$$\frac{\partial\mathcal{L}(\theta)}{\partial\theta_j} = 2\sum_{i=1}^{n}\left(\mathbf{x}_i^{\text{tmpl}} - \mathcal{X}^*(\mathcal{W}(i;\theta))\right)\frac{\partial}{\partial\theta_j}\mathcal{X}^*(\mathcal{W}(i;\theta)) \tag{10}$$

and, recalling Eq. 5, we only need to compute

$$\frac{\partial\mathcal{X}^*(\mathcal{W}(i;\theta))}{\partial\theta_j} = \mathbf{A}^*\frac{\partial\mathcal{Y}(\mathcal{W}(i;\theta))}{\partial\theta_j} \tag{11}$$

and

$$\frac{\partial\mathcal{Y}(\mathcal{W}(i;\theta))}{\partial\theta_j} = \left.\frac{\partial\mathcal{Y}(t)}{\partial t}\right|_{\mathcal{W}(i;\theta)}\frac{\partial\mathcal{W}(t;\theta)}{\partial\theta_j} \tag{12}$$

where $\left.\frac{\partial\mathcal{Y}(t)}{\partial t}\right|_{\mathcal{W}(i;\theta)}$ refers to the evaluation of the function $\frac{\partial\mathcal{Y}(t)}{\partial t}$ on $t = \mathcal{W}(i;\theta)$. Finally, recall from Eq. 4 that $\mathcal{Y}(t)$ is expressed in terms of the functions $\mathbf{e}_k^n(t)$. Therefore, the only thing left to be computed is:

$$\frac{\partial}{\partial t}\mathbf{e}_k^n(t) = k\pi\cos(k\pi t + \pi(n-k)/2n) \tag{13}$$

This last equation shows that, remarkably, the derivatives of $\mathcal{X}^*(t)$ are computed analytically, therefore not requiring any numerical approximation.

Now we are in the position of defining the iterative gradient descent procedure. Given the current estimate of the parameters, noted $\theta^{(it)}$, and by noting $\mathbf{J}_{\theta^{(it)}} = \nabla\mathcal{L}(\theta^{(it)})$, then the new estimate of the warp parameters can be expressed as:

$$\theta^{(it+1)} = \theta^{(it)} - \left(\mathbf{J}^t_{\theta^{(it)}}\mathbf{J}_{\theta^{(it)}}\right)^{-1}\mathbf{J}^t_{\theta^{(it)}}\mathbf{r}\left(\theta^{(it)}\right) \tag{14}$$

where $\mathbf{r}\left(\theta^{(it)}\right) = (r_1,\ldots,r_n)$, and $r_i = \mathbf{x}_i^{\text{tmpl}} - \mathcal{X}^*(\mathcal{W}(i;\theta^{(it)}))$.

6 Experiments

Dataset: In this paper, we use the MMI database. It contains 264 videos of 10 subjects fully FACS-coded in terms of AU activation and temporal segments by two FACS experts. Following the previous related studies, only sequences that have the target AUs activated are considered for testing. Other datasets with annotated temporal segments include the Cohn-Kanade (CK) database and the SAL dataset. However, the offset segment is not included in the CK database and very few sequences of the SAL dataset is annotated in terms of temporal segments. Therefore they are not used in this paper.

Preprocessing: This paper focuses on mouth-related AUs as they cover most facial actions (15 out of 32). In order to remove variability due to rigid motions of the head such as translation, scaling and in-plane head rotations, we pre-process the image frames in the following way. The first step is to localise the facial landmarks in every frame of the sequence. To this aim the work presented in [1] is used. A Procrustes transformation (i.e. a combination of translation, rotation and isotropic scaling) is then computed by aligning the coordinates of the mouth landmarks to a set of anchor points. After that the mouth region are cropped and resized to 30×50 pixels.

Experiment setup: For each AU, the template with the maximum intensity is selected from the activated sequences. These templates are then aligned to the test sequences following both our models. The alignment parameter θ is initialised by independently aligning the onset and offset segments. The only constrain is that the offset always occurs after the onset. To this end, instead of trying to find a global minimum when the alignment is performed, all the local minimums obtained by aligning the onset and offset template are kept, and the pair yielding the lowest combined alignment error is used to infer the initial boundary of the actions. Additionally, the parameter θ_5 controlling the intensity magnitude of model 2 is initialised to 0.5.

Experimental results: Table 1 summarises the per AU performance for the detection of the AU temporal segments, computed on the MMI database. The table includes the performance for both of the models presented in Section 4, showing the superior performance for the detection of each of the temporal segments using the second model. This is unsurprising, however, given the extra capability of the second model to handle AU intensity differences.

Table 1: F1-measure for the frame-level classification of the AU temporal segments on the MMI database. n is the number of activated sequences. $\mathbf{F1}_{\text{act}}$ is the F1-measure after converting the labelling into a binary labelling of AU activation. Results shown are for model 1 / model 2

AU	n	Neutral	Onset	Apex	Offset	$\mathbf{F1}_{\text{act}}$
10	14	92.19 / 92.40	57.55 / 55.33	82.09 / 84.10	60.64 / 59.63	83.46 / 87.65
12	18	86.26 / 92.24	67.51 / 65.14	84.45 / 77.55	75.73 / 74.49	83.04 / 88.33
13	9	95.38 / 84.40	57.14 / 49.17	95.24 / 79.51	76.19 / 65.10	95.71 / 90.38
14	16	81.77 / 86.71	56.61 / 59.60	71.58 / 81.24	41.60 / 68.42	91.67 / 91.73
15	12	80.69 / 87.24	70.07 / 62.62	89.70 / 81.92	59.69 / 53.12	72.91 / 74.16
16	13	89.83 / 87.67	60.43 / 58.47	75.60 / 76.57	62.40 / 52.37	83.97 / 83.88
18	21	94.74 / 95.95	28.57 / 53.88	90.91 / 86.49	57.14 / 61.67	75.65 / 79.62
20	11	69.97 / 73.38	36.41 / 38.04	64.32 / 73.04	37.96 / 33.46	71.09 / 70.41
22	10	66.25 / 70.26	48.91 / 51.12	64.81 / 82.43	44.60 / 57.72	84.83 / 86.28
23	12	90.68 / 84.45	53.30 / 46.97	81.54 / 75.42	55.46 / 46.71	55.52 / 63.30
24	18	76.68 / 81.58	57.61 / 51.80	72.81 / 83.50	50.24 / 62.89	51.26 / 59.59
25	44	93.62 / 90.94	51.21 / 52.38	79.22 / 80.23	57.78 / 69.40	94.27 / 88.34
26	25	71.16 / 83.88	42.76 / 49.96	70.47 / 80.73	50.23 / 58.24	72.00 / 72.03
27	13	92.42 / 92.99	63.86 / 85.07	86.77 / 76.81	68.38 / 43.55	63.67 / 84.97
28	30	73.63 / 84.07	60.33 / 65.27	73.42 / 76.66	70.01 / 77.54	88.48 / 88.61
AVG	-	83.42 / 85.88	54.15 / 56.32	78.86 / 79.75	57.87 /58.95	77.83 / 80.62

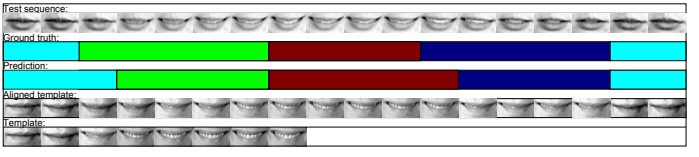

Figure 2: Illustration of the results of the temporal alignment using model 2, and a comparison of the temporal segment detection respect to the ground truth

The F1-measure is used to compute performance. Furthermore, we show results when turning the prediction on the temporal segments into a prediction of the AU activation. That is to say, each neutral frame is labelled as 0, while non-neutral frames are labelled as 1. Among the AUs considered, AU23 (lip tighten) and AU24 (lip pressor) have the lowest detection rate in average. This is logical because the appearance changes in these two AUs are very subtle. The poor performance for AU20 (lip stretcher) and AU22 (lip funneler) is likely to be caused by the fact that both AUs cause variations in the mouth shape (elongation and shortening respectively) that are diminished by the mouth size normalisation step.

An illustration of the detection process is shown in Fig. 2. We show on it how a template is aligned to the test sequence, yielding a prediction on the temporal boundaries, which are a precise approximation of the ground truth. It also demonstrates that the potential of aligning the test sequence to the template although the test sequence does not reach the maximum intensity (when using model 2).

We also show a performance comparison of our method with respect to other state-of-the-art approaches. Table 2 offers a direct comparison of our method to earlier works reporting results on the same dataset. For a fair comparison, the results are computed over the 15 mouth-related AUs. As we can see, the models proposed in this article outperform all the existing works. It is important to note that, as opposed to other methods, we only use one exemplar of the action to produce these results and no other training is needed.

Table 2: Comparison of AU temporal segment detection methods on the MMI database. **F1**$_{act}$ is the F1-measure after converting into AU activation.

Systems	Neutral	Onset	Apex	Offset	F1$_{act}$
Model1	83.42	54.15	78.86	57.87	77.83
Model2	85.88	56.32	79.75	58.95	80.62
Jiang et al. 2013[10]	78.50	53.38	72.12	48.73	67.53
Valstar et al. 2012[14]	76.60	56.75	69.38	48.87	-
Koelstra et al. 2010[12]	-	-	-	-	62.5

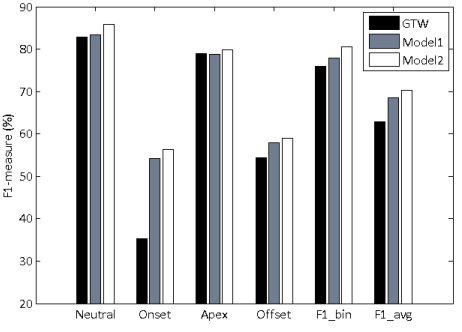

Figure 3: Performance on temporal segment detection when using the proposed models and GTW [16] for temporal alignment

Finally, we experiment whether the proposed alignment method performs better than previously existing temporal alignment methods. We show a direct comparison with the Generalised Time Warp (GTW) [16]. This is the best-performing time warping methodology and the most closely related to the alignment method proposed here. This is due to the family of parametric functions used such that the produced alignment is smooth without a requirement to exhaustively align both sequences (although it tends to do so), and the estimation of the alignment parameters through gradient descent. However, such method is still unable to encode intra-sequence structure into the alignment process. These results are shown in Fig. 3, where the superior performance of the proposed alignment method, especially for the onset phase, can be observed.

7 Conclusions

We have presented a new method for the detection of the facial action temporal segments by facial behaviour similarity matching, where no training of dedicated classifier is needed. This method could potentially alleviate long-standing problem in machine analysis of facial behaviour – the lack of annotated data to learn from. To this end, we perform an explicit temporal alignment with a novel parametric technique. This approach can accommodate a larger variety of warp functions and an efficient Gauss-Newton gradient descent search. We show that for the problem of AU temporal segment detection, our approach produces the best results to date.

8 Acknowledgement

This work has been funded by the EPSRC project EP/J017787/1 (4DFAB). The work by Maja Pantic is funded in part by the European Community 7th Framework Programme [FP7/2007-2013] under grant agreement no. 611153 (TERESA). The work of Brais Martinez was also funded in part by the EPSRC grant EP/H016988/1: Pain rehabilitation: E/Motion-based automated coaching.

References

[1] J. Alabort, G. Tzimiropoulos, S. Zafeiriu, and M. Pantic. Generic active appearance models revisited. In *Proc. IEEE Asian Conf. Computer Vision*, pages 650–663, 2012.

[2] Z. Ambadar, J. W. Schooler, and J. F. Cohn. Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions. *Psychological Science*, 16(5):403–410, 2005.

[3] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *Int'l Journal of Computer Vision*, 56(3):221 – 225, 2004.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in Neural Information Processing Systems*, pages 585–591, 2001.

[5] Y. Caspi and M. Irani. Spatio-temporal alignment of sequences. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(11):1409 – 1423, 2002.

[6] P. Ekman, W.V. Friesen, and J. C. Hager. *Facial action coding system*. A Human Face, 2002.

[7] O. Kliper-Gross et al. The action similarity labeling challenge. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 34(3):615–621, 2012.

[8] G. Huang et atl. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, 2008.

[9] X. He, D. Cai, S. Yan, and H.J. Zhang. Neighborhood preserving embedding. In *Proc. IEEE Int. Conf. Computer Vision*, volume 2, pages 1208–1213, 2005.

[10] B. Jiang, M. F. Valstar, B. Martinez, and M. Pantic. Dynamic appearance descriptor approach to facial actions temporal modelling. *IEEE Trans. Systems, Man and Cybernetics, Part B*, 44(2):161–174, 2014.

[11] Eamonn J. Keogh and Michael J. Pazzani. Derivative dynamic time warping. In *SIAM International Conference on Data Mining*, 2001.

[12] S. Koelstra, M. Pantic, and I. Patras. A dynamic texture based approach to recognition of facial actions and their temporal models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(11):1940–1954, 2010.

[13] G. Sandbach, S. Zafeiriou, and M. Pantic. Binary pattern analysis for 3D facial action unit detection. In *British Machine Vision Conference*, 2012.

[14] M. F. Valstar and M. Pantic. Fully automatic recognition of the temporal phases of facial actions. *IEEE Trans. Systems, Man and Cybernetics, Part B*, 42(1):28–43, 2012.

[15] L. Van der Maaten and E. Hendriks. Action unit classification using active appearance models and conditional random field. *Cognitive processing*, 13:507–518, 2012.

[16] F. Zhou and F. de la Torre. Generalized timewarping for multi-modal alignment of human motion. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[17] Z. Zhou, G. Zhao, and M. Pietikaïnen. Towards a practical lipreading system. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 137–144, 2011.