Learning Temporal Signatures for Lip Reading

Eng-Jon Ong and Richard Bowden
The Centre for Vision, Speech and Signal Processing,
University of Surrey,
Guildford GU27XH, Surrey, UK
e.ong,r.bowden@surrey.ac.uk

Abstract

This paper attempts to tackle the problem of lipreading by building visual sequence classifiers that are based on salient temporal signatures. The temporal signatures used in this paper allow us to capture spatio-temporal information that can span multiple feature dimensions with gaps in the temporal axis. Selecting suitable temporal signatures by exhaustive search is not possible given the immensely large search space. As an example, the temporal sequence used in this paper would require exhaustively evaluating 2^{2000} temporal signatures which is simply not possible. To address this, a novel gradient-descent based method is proposed to search for a suitable candidate temporal signature. Crucially, this is achieved very efficiently with $O(nD)$ complexity, where D is the static feature vector dimensionality and n the maximum length of the temporal signatures considered. We then integrate this temporal search method into the AdaBoost algorithm. The results are spatio-temporal strong classifiers that can be applied to multi-class recognition in the lipreading domain. We provide experimental results evaluating the performance of our method against existing work in both subject dependent and subject independent cases demonstrating state of the art performance. Importantly, this was also achieved with a small set of temporal signatures.

1. Introduction

This paper presents a machine learning approach to Lip Reading and proposes a novel learning technique called temporal gradient descent boosting that allows us to very efficiently search and combine a set of temporal patterns to form strong spatio-temporal classifiers. Lip reading is a difficult task, especially in the subject independent case as both the appearance and motion of the lips varies significantly between subjects. Any attempt at automatic lip reading needs to address a number of demanding challenges. The first involves the inherently temporal nature of the problem. It is not possible to simply find some set of static visual features that can differentiate between two sets of spoken speech. Instead, it is crucial to model and use spatio-temporal information. However, other challenges arise from motion and appearance varations: The degree of movement of the mouth due to speech also tends to be less than that of emotions and other typical forms of actions recognised and variations are present across different individuals in terms of different mouth shapes, possible presence of beards and different styles of lip movements whilst speaking the same words.

1.1. Related Work

There already exists a body of work that deals directly with the task of lip reading. Due to the temporal nature of the problem at hand, all existing approaches attempt to model and exploit the dynamics of the lip movements in order to automatically lip read. One popular method is to model the temporal information using Hidden Markov Models (HMM). Matthews et al [4] proposed a method that simultaneously models the mouth shape information and underlying texture information using Active Shape Models. Here, the mouth texture is warped into a normalised shape before being combined with the mouth shape. Dimensionality reduction using Principal Component Analysis is then performed giving a feature vector for an image. The dynamics of the feature vectors are then modelled using HMMs. One disadvantage with this approach is the direct use of texture and shape, which can vary across different individuals. This results in poor subject-independent results. In order to overcome this, Lan et al [3] proposed to incorporate short term temporal information into the feature vector as well as performing Linear Discriminant Analysis in addition to PCA. Another possibility is to use geometric information, for example mouth width, area within the inner lip, etc.. [1]. However, this requires accurate tracking of both the inner and outer lip shape, a non-trivial task.

An alternative would be to utilise simpler visual features that can be extracted within a bounding-box area contain-

2011 IEEE International Conference on Computer Vision Workshops
978-1-4673-0063-6/11/$26.00

ing the mouth. This is the approach that is taken in this paper, where binary comparison features are used. Recently, Local Binary Patterns (LBPs) have been popular as visual features for lipreading. LBPs have the advantage that they encode the relative intensity differences between a pixel and its surrounding pixels, making it less reliant on the absolute intensity value. This in turn provides robustness to lighting changes and can potentially help generalise across subjects. Zhao et al[11] proposed a method that uses LBPs that span both time and space, effectively modelling local spatio-temporal information. Histograms of LBP responses around various sub-blocks of the mouth region were extracted and concatenated into a high dimensional feature vector. Support Vector Machines (SVM) were then used to build classifiers using a set of training feature vectors. Another interesting approach is to model the sequence of mouth images as a graph embedding problem, as proposed by Zhou et al[12]. Our proposed method differs from SVM-based methods in that we attempt explicit temporal feature selection by means of the boosting algorithm. This allows us to build classifiers that are smaller and thus more efficient, whilst providing similar or better accuracy.

1.2. Overview

The rest of the paper is organised as follows: In Section 2, we first discuss and highlight important differences between our proposed methods and existing spatio-temporal boosting work. Next, Section 2.1 describes the general temporal pattern used known as a temporal signature. We then describe how weak temporal classifiers are formed by means of detecting temporal signatures present in an input sequence of example frames. An efficient gradient descent greedy search method is proposed for searching for a suitable weak temporal classifier within the Boosting framework in Section 2.3 before describing the novel temporal pattern Boosting algorithm in Section 2.4. Experimental results demonstrating state-of-the-art performance in the domain of lipreading is described in Section 3 before concluding in Section 4.

2. Boosting Temporal Patterns

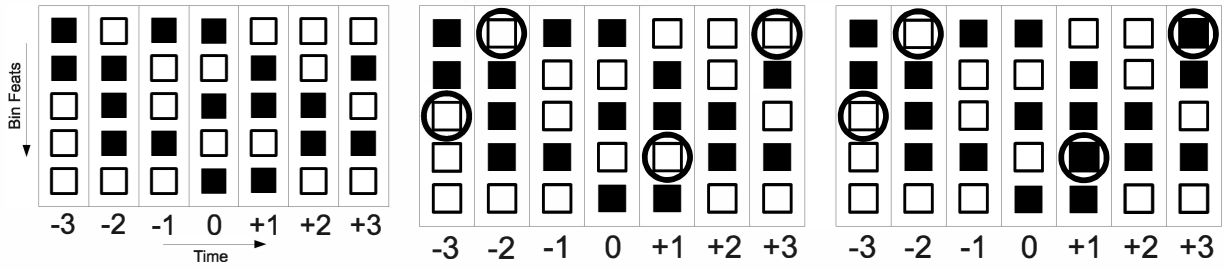
In this section, we propose a novel method that will construct a strong temporal pattern classifier by learning temporal weak classifiers within the AdaBoost[2] framework. The work of TemporalBoost by Smith et al[9] is related to the work proposed here. Here, temporal information is modelled by extending a single static feature back in time. The temporal weak classifier in TemporalBoost also considers a static feature's values in a temporal window that extends into the past. However, there is an important difference between the proposed method and TemporalBoost. All the temporal weak classifiers in TemporalBoosts are contiguous time segments fixed to a particular feature starting at the present time frame going into the past. The temporal signatures proposed here (Section 2.1) utilises information from the past, present and future. Crucially, temporal signatures that form temporal weak classifiers can be disjoint and have gaps in the temporal axis as well as spanning different feature dimensions. This allows our method to capture temporal signatures that can span across different features and ignore areas in time where features are ambiguous. We will describe how temporal signatures can be used to form temporal weak classifiers in Section 2.1.

It is important to note that the search space for possible temporal weak classifiers is enormous and thus computationally impossible to perform by exhaustive search. Recently, a number of approaches were proposed for tackling the problem of searching for suitable temporal patterns given its enormous search space. These approaches have in common the use of data mining methods in combination with boosting algorithms to learn strong classifiers that can be applied to recognising temporal patterns. Nowozin et al[5] proposed weak classifiers based on sequential patterns or subsequences. In order to search for the optimal subsequence, a modified data mining method based on PrefixSpan[8] was used. This introduces a number of search space pruning conditions that allows one to search for a suitable subsequence pattern with respect to the example weight distribution. However, this approach effectively requires data mining to be performed at every boosting iteration, which can be computationally demanding, depending on the complexity of the problem. The weak classifiers based on the temporal signatures proposed in this paper will invalidates the pruning conditions used by PrefixSpan. Our search approach described in Section 2.3 is different in that the complexity of our algorithm is $O(T^{Max}D)$ and depend on the maximum length of the temporal pattern (T^{Max}) and the static feature dimensionality (D).

More recently, Yuan et al[10] proposed to initially perform data mining on features, with the resulting mined rules producing compositional features. Our approach differs from this in that we do not "pre-select" a fixed set of temporal classifiers tso later perform boosting on. Instead, the TGD-Boosting algorithm described in Section 2.4 selects weak temporal classifiers with respect to the example weights, allowing us to be more efficient in locating suitable temporal classifiers for tackling difficult examples.

2.1. Weak Temporal Classifiers

In order to build a suitable weak temporal classifier, we start by assuming that a static image is represented as a D-dimensional binary feature vector: $\mathbf{x}_i = (x_{i,j})_{j=1}^{D}$ with $x_{i,j} \in \{0, 1\}$. In this paper, the binary feature vector takes the form of binary comparison features that implicitly capture textural variations within the mouth (Section 3.1). Suppose the frame currently considered is denoted by $\mathbf{x}_0$, we



(a) (b) (c)

Figure 1. (a) shows an example of a temporal sequence of radius 3 with each column representing a 5-dimensional binary vector (black is 0 and white is 1). The frame offsets for this temporal pattern is shown below, with the "current" frame having zero offset. (b) shows an example temporal signature: ((-3,3),(-2,1),(1,4),(3,1)). (c) The same temporal signature, but on another temporal pattern.

then define an input *temporal sequence* of radius R centred around $\mathbf{x}_0$ as $\mathbf{X} = (\mathbf{x}_{-R}, ..., \mathbf{x}_0, ...\mathbf{x}_R))$, where $\mathbf{x}_r$ with $r < 0$ are feature vectors for frames preceeding $\mathbf{x}_0$ and $\mathbf{x}_r$ with $r > 0$ denote feature vectors of images after the frame currently considered. A simple example of a temporal sequence can be seen in Figure 1a. The "future" frames could be obtained by buffering frames and delaying the processing of a particular frame by $2R$ frames.

We now propose a representation for a *temporal signature* that may be present within a binary feature vector sequence. Specifically, a temporal signature ($\mathbf{t}$) of length T, can be defined as the set of pairs: $\mathbf{t} = \{(t_i, s_i)\}_i^T$, where t_i defines the time offset from the current frame and s_i denotes the dimension index of the feature vector at the respective offset frame (see Figure 1b,c for an example).

A temporal weak classifier ($h_\mathbf{t}(\mathbf{x})$) is formed using the detection of a temporal signature ($\mathbf{t}$) within an input temporal pattern ($\mathbf{x}$). To this end, two types of temporal weak classifiers are defined: the "AND" and "OR" temporal classifiers. They differ by the requirement of how much of the temporal signature is required to be present within the input sequence. The "AND" temporal classifier denoted with the superscript A (i.e. $h_\mathbf{t}^A$) requires an entire temporal signature to be present within the input pattern. Thus, given a temporal signature $\mathbf{t}$ and an input sequence $\mathbf{x}$, the "AND" weak temporal classifier can be defined as:

$$h^A(\mathbf{x})_\mathbf{t} = \begin{cases} 1 & \text{if } x_{t_i,s_i} = 1, \forall (t_i, s_i) \in \mathbf{t} \\ -1 & \text{otherwise} \end{cases} \quad (1)$$

Similarly, an "OR" weak temporal classifier, which is denoted by the superscript O (i.e. $h_\mathbf{t}^O$), can be defined as follows:

$$h^O(\mathbf{x})_\mathbf{t} = \begin{cases} 1 & \text{if } x_{t_i,s_i} = 1, \exists (t_i, s_i) \in \mathbf{t} \\ -1 & \text{otherwise} \end{cases} \quad (2)$$

As an example of both the "AND" and "OR" temporal classifier difference, an "AND" classifier associated with the temporal signature in Figure 1 applied to the temporal pattern in Figure 1c will give -1, as not all the binary features under the temporal signature is 1. However, an "OR" classifier will give 1, since at least one element under the respective temporal signature is 1.

2.2. Temporal Pattern Boosting

The Boosting algorithm takes as input a set of M training example pairs: $(\mathbf{x}_i, y_i)_{i=1}^M$, where $\mathbf{x}_i$ is a temporal sequence, as defined in Section 2.1 and $y_i \in \{-1, +1\}$ is the label of the respective training example. We assume that all the temporal sequence examples $\mathbf{x}_i$ have the same length, defined as R. Additionally, the training set is also associated with a distribution of "difficulty" weights, where each example in the training set is assigned the weight: w_i with $\sum w_{i\,i=1}^M = 1$. The weights are updated accordingly by the Boosting algorithm based on the performance of the weak classifier chosen at a particular iteration of the algorithm. Thus, at each iteration of the Boosting algorithm, the task is to select a weak classifier, or in our case, a weak temporal classifier, $h_\mathbf{t}^\beta(\mathbf{x}), \beta \in \{A, O\}$, associated with the temporal signature $\mathbf{t}$ that minimises the following error:

$$\epsilon_{h_\mathbf{t}^\beta} = \sum_{i=1}^{M} w_i [h_\mathbf{t}^\beta(\mathbf{x}_i \neq y_i)] \quad (3)$$

where $[a \neq b] = 1$ if $a \neq b$ and 0 otherwise. In traditional Boosting algorithms, ϵ in Eq. 3 is minimised by an exhaustive search of all possible weak classifiers. However, this is not possible in case of the temporal weak classifiers that have a total number of 2^{TD} possible configurations, where D is the dimension of the binary feature vector for each frame and T is the maximum possible length for temporal signatures considered. As an example, for our experiments, we have considered the case where $D = 400$ and $T = 5$, giving 2^{2000} possible temporal weak classifiers.

2.3. Temporal Pattern Greedy Gradient Descent

Given that an exhaustive search of possible temporal weak classifiers is not possible, we now propose a novel

Algorithm 1 TemporalPatternSearch

Require: Training Data:$\mathbf{x}_i, y_i)_{i=1}^M$, Example Weights: $(w_i)_{i=1}^M, T^{Max}$, Temporal Classifier Type: $\beta \in \{A, O\}$

Obtain starting pattern $\mathbf{t}^S$ and its error $\epsilon_{h^{\beta}_{\mathbf{t}^S}}$ (Eq. 4)
$T^{cur} = 0$
$\mathbf{t}^{Cur} = \mathbf{t}^S$
$\epsilon_{best} = \epsilon_{h^{\beta}_{\mathbf{t}^S}}$
while $T^{cur} < T^{Max}$ **do**
  Expand backwards (Eq. 5), obtaining new pattern $\mathbf{t}'$ and its error $\epsilon_{h^{\beta}_{\mathbf{t}'}}$
  if $\epsilon_{h^{\beta}_{\mathbf{t}'}} < \epsilon_{best}$ **then**
    $\mathbf{t}^{Cur} = \mathbf{t}'$
    $\epsilon_{best} = \epsilon_{h^{\beta}_{\mathbf{t}'}}$
  end if
  Expand in front (Eq. 6), obtaining new pattern $\mathbf{t}''$
  if $\epsilon_{h^{\beta}_{\mathbf{t}''}} < \epsilon_{best}$ **then**
    $\mathbf{t}^{Cur} = \mathbf{t}''$
    $\epsilon_{best} = \epsilon_{h^{\beta}_{\mathbf{t}''}}$
  end if
end while
return $\mathbf{t}^{Cur}$

search method that considers potential temporal weak classifiers up to a maximum length of T^{Max}. The search procedure is initialised by locating a suitable temporal weak classifier associated with an initial temporal signature. This is achieved by finding the optimal temporal signature of length 1 given the training examples and their weights. Formally, suppose the initial temporal signature is defined as $\mathbf{t}^S = \{(t^S, y^S)\}$ that satisfies:

$$\mathbf{t}^S = \underset{(t^S \in [-R,R], y^S \in [0,D])}{\operatorname{argmin}} \epsilon_{h^{\beta}_{\mathbf{t}^S}} \tag{4}$$

The search procedure then proceeds by iterative greedy expansion of an existing temporal signature ($\mathbf{t}^{Cur} = (t_i, y_i)_{i=1}^{T^{Cur}}$) of length T^{Cur}, such that the error of its associated weak classifier is maximally reduced with respect to the example weight distribution, $(w_i)_{i=1}^M$. Two possible expansions of an existing temporal signature are considered in this paper. The first involves adding a new feature backwards in time to $\mathbf{t}^{Cur}$ and the second expansion involves adding a new feature forwards in time to $\mathbf{t}$. To achieve this, we first define $t^B = \min t_i, \forall i \in \{0, ..., T\}$ and $t^F = \max t_i, \forall i \in \{0, ..., T\}$. In the first case of adding a new feature (t'_i, y'_i) preceeding $\mathbf{t}$, we define the expanded temporal signature to be: $\mathbf{t}' = \mathbf{t} \cup \{(t'_i, y'_i\}$, with the temporal signature length of $T' = T + 1$. The task is then to find the optimal configuration for (t'_i, y'_i). To this end, we search for a binary feature y'_i that lies within a window preceeding $\mathbf{t}^{Cur}$ such that the error of the weak classifier based on $\mathbf{t}'$ is minimised:

$$\mathbf{t}' = \underset{(t'_i \in [\max(t^B - S, R), t^B], y'_i \in [0,D])}{\operatorname{argmin}} \epsilon_{h_{\mathbf{t}'}} \tag{5}$$

where ϵ_h was defined in Eq. 3. The second possible expansion is to similarly locate the optimal feature (t''_i, y''_i) that lies in a window succeeding $\mathbf{t}^{Cur}$. Suppose the "appended" feature is defined as $\mathbf{t}'' = \mathbf{t}^{Cur} \cup \{(t''_i, y''_i\}$, then we select (t''_i, y''_i) such that the following is satisfied:

$$\mathbf{t}'' = \underset{(t''_i \in [t^F, \min(t^F + S, R)], y''_i \in [0,D])}{\operatorname{argmin}} \epsilon_{h_{\mathbf{t}''}} \tag{6}$$

The temporal pattern greedy gradient descent algorithm is given in Algorithm 1. It can be seen above that both the expansion procedures above only considers a fixed amount of possibilities that is proportional to the dimensionality of the feature vector. Given that they are then used in an iterative manner, this gives Algorithm 1 a complexity of $O(T^{Max} D)$.

2.4. TGD-Boosting Algorithm

The temporal pattern gradient descent search method can now be easily integrated into the AdaBoost algorithm, although integration into other Boosting methods (e.g. GentleBoost, LPBoost) could be achieved similarly. Here, the step for selecting a suitable weak classifier at a given iteration is replaced by the temporal pattern gradient descent method proposed in Section 2.3. Thus, the Temporal Gradient-Descent Boosting (TGD-Boosting) algorithm detailed in Algorithm 2.

3. Experiments

This section will describe the experiments that were performed to evaluate the performance of the proposed TGD-Boosting algorithm when applied to the problem of lip reading. For our experiments, the OuluVS database [11] was chosen since it has been used by a number of recent lip reading methods [11, 12] for the purpose of experimental evaluation. This makes it a suitable platform for comparing the performance of our proposed method against existing state-of-the-art methods. This database contains video sequences of 20 subjects in total (see Figure 3). Each subject read out 5 repetitions of 10 different phrases: *'excuse me', 'goodbye', 'hello', 'nice to see you', 'have a nice day', 'I'm sorry', 'thank you', 'have a good time'* and *'your welcome'*. Each phrase was segmented, resulting in a total of 50 sequences for each subject and 1000 example sequences in total for the entire database. The resolution of the images in all sequences is 720x576 captured at 25fps. The aim of the experiments described in this section is to recognise which of the 10 phrases was spoken. This will be done in the context of subject dependence and subject independence as detailed in Section 3.2.

Algorithm 2 TGD-Boosting

Require: Training Data:$\mathbf{x}_i, y_i)_{i=1}^{M}$

Initialise example weights: $w_i = \frac{1}{M}, \forall i \in [1, M]$

for $i = (1, ..., I)$ **do**

Get best "AND" temporal classifier $h^A_{\mathbf{t}_i}(\mathbf{x})$ with error ϵ^A, and its temporal pattern ($\mathbf{t}^A_i$) using TemporalPatternSearch with $\beta = A$ (Algo. 1)

Get best "OR" temporal classifier $h^O_{\mathbf{t}_i}(\mathbf{x})$ with error ϵ^O, and its temporal pattern ($\mathbf{t}^O_i$) using TemporalPatternSearch with $\beta = O$ (Algo. 1)

if $\epsilon^A < \epsilon^O$ **then**

$\epsilon_i = \epsilon^A$, $\beta_i = A$, $\mathbf{t}_i = \mathbf{t}^A_i$

else

$\epsilon_i = \epsilon^O$, $\beta_i = O$, $\mathbf{t}_i = \mathbf{t}^O_i$

end if

if $\epsilon_i \geq 0.5$ then stop

Choose suitable weak classifier weight:

$$\alpha_i = \frac{1}{2} ln \frac{1 - \epsilon_i}{\epsilon_i}$$

Update example weights:

$$w_j = \frac{w_j exp(-\alpha_i y_j h^{\beta_i}_{\mathbf{t}_i}(\mathbf{x}_i))}{W}$$

where W ensures w_j is a distribution.

end for

Output strong temporal classifier:

$$H(\mathbf{x}) = sign(\sum_{i=1}^{I} \alpha_i h^{\beta_i}_{\mathbf{t}_i}(\mathbf{x})) \quad (7)$$

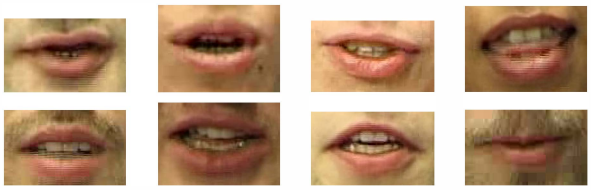


Figure 2. Examples of the mouths for various subjects in the OuluVS database (Note, this is not the entire image, only the mouth segment of the image).

3.1. Visual Features

The visual features selected for the experiments take the form of simple comparative binary features. These features are similar to LBPs in that binary descisions based on relative intensity differences are used. However, unlike LBPs, our visual features also allow us to capture non-local intensity relations. Whereas an LBP, by nature, is limited to using local surrounding pixels, the visual features here can compare two parts of the mouth that are non-local.

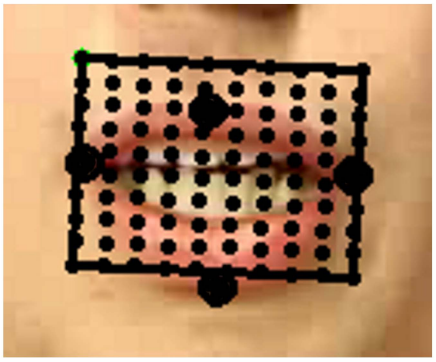


Figure 3. Visualisation of the grid of points in the mouth bounding box based on four tracked points on the mouth (shown as black points).

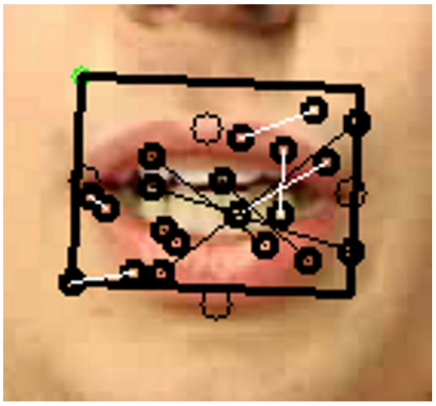


Figure 4. Visualisation of the 10 binary comparison features used (200 were used in the experiments) described in Section 3.1.

In order to extract suitable visual features (see Fig. 3), four points on the mouth of the subjects were first automatically tracked using the linear predictor tracking method [6]. Following this, a bounding box aligned with the orientation of the mouth is extracted. The mean width and height of the mouth bounding box of a subject is then obtained and this determines the size of the final bounding box used throughout all the sequences for a particular subject. Following this, a grid of 20x20 points is formed on the bounding box. Next, two random sets of 200 grid point indices are obtained and fixed for all subjects and experiments. A binary feature vector is then formed by performing 200 binary comparisons of the underlying intensities for the two grid point index sets (see Figure 4). An inverse feature vector is then obtained by inverting the original binary feature vector. The final feature vector is then obtained by concatenating the original binary feature vector with its inverse feature vector, resulting in a 400-dimensional binary feature vector for each

image frame. All the example video sequences were then converted into temporal patterns of radius 12, with the middle of the example sequence being the zero-offset frame. Thus, all the resulting temporal patterns will have in total 25 frames. If an example sequence has less than 25 frames, the frames in the temporal pattern falling outside the sequence is assigned a feature vector containing -1 for all dimensions.

3.2. Experimental Setup

In this paper, we performed two classes of experiments. The first evaluates the performance of the proposed method in a subject-dependent setting, whilst the second set of experiments evaluates the performance of our method in a subject-independent setting. For the subject dependent experiments, a leave-one-out video cross-validation was performed. Specifically, for each subject, a single video was used as a test sequence. The remaining video sequences for this subject was used as training examples. In this work, the video sequences of the other subjects were not included as training examples. In the case of subject-independent testing, a 20-fold cross validation was performed. In each cross validation fold, all example sequences for a particular subject were reserved for testing. The example sequences for the remaining subjects were used as training examples.

For each cross validation fold, a set of one-vs-one classifiers (90 in total) using the proposed method described in Section 2 was trained. A voting scheme was then used to obtain a class label for each test sequence. In both subject dependent and subject independent tests, these classifiers were trained for temporal patterns with a heuristically chosen maximum length of 5 (i.e. $T^{Max} = 5$). All the classifiers were boosted with a maximum of 100 iterations. In order to analyse the contribution of using temporal information, classifiers with only temporal signatures of length 1 ($T^{Max} = 1$) were also trained, again with a maximum of 100 iterations. These classifiers are equivalent to performing the traditional non-temporal boosting.

3.3. Results

All the one-vs-one classifiers were trained on a single Quad-core PC. The total training time for all 90 classifiers was approximately 15 minutes in total. However, this time can be trivially reduced by training in parallel different one-vs-one classifiers if more computers are available. It was found that for every one-vs-one classifiers trained, in both subject dependent and subject independent tests, the training errors when using temporal classifiers decreased approximately twice as quickly as when no temporal information was used. Examples of the error curves for various classifiers trained in the subject independent case can be seen in Figure 5. It was also found that approximately 99% of the weak classifiers selected in the temporal set were associated with temporal signatures with the maximum length of 5. The remaining 1% classifiers were associated with temporal signatures of length 4. There were no shorter temporal signatures chosen.

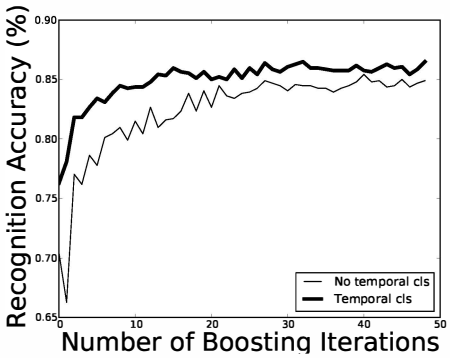


Figure 6. Shown here are the average accuracy for subject dependent tests over different boosting iterations. The thick line shows the accuracy when using temporal classifiers and the thin line represents the accuracy without (i.e. length 1 temporal classifiers).

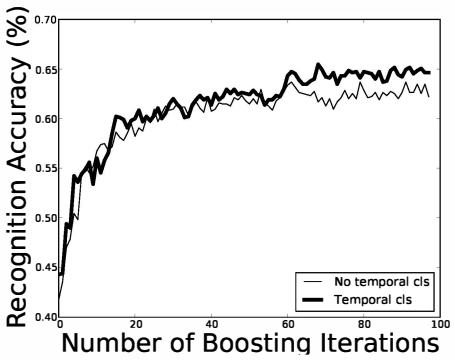
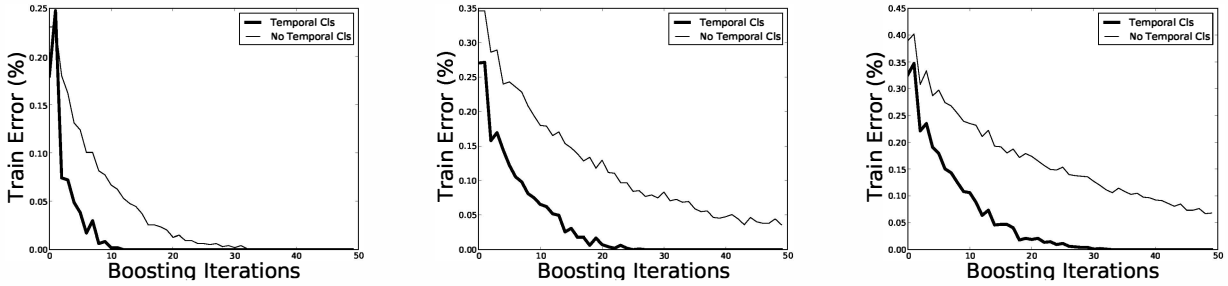


Figure 7. Shown here are the average accuracy for subject independent tests over different boosting iterations. The thick line shows the accuracy when using temporal classifiers and the thin line represents the accuracy without (i.e. length 1 temporal classifiers).

3.3.1 Subject Dependent Results

The leave-one-out cross validation results for the subject dependent tests are summarised using a confusion matrix shown in Figure 8. From this, it can be seen that all but one classes performed well, with accuracy rates above 80%. The "See you" class had the lowest accuracy, with the largest confusion with "Thank you". This is not surprising and this ambiguity was also observed in [11], with the cause for confusion being the similar nature in which both phrases are spoken. However, it is interesting that the converse was not observed. The degree of confusion between the "Thank you" class and "See you" class is not as large. Additionally, we have also compared the performance of strong classifiers



ExM vs Wl Gb vs Wl Syu vs Tk

Figure 5. Examples of training errors with and without using weak temporal classifiers.

ExM	Excuse Me
Gb	Goodbye
Hlo	Hello
Hw	How are you
Nc	Nice to see you
Syu	See you
Sry	I'm sorry
Tk	Thank you
Tm	Have a good time
Wl	Your welcome

	ExM	Gb	Hlo	Hw	Nc	Syu	Sry	Tk	Tm	Wl
ExM	83.3	0.0	3.2	0.0	5.3	2.1	1.1	2.1	2.1	1.1
Gb	0.0	95.8	0.0	0.0	1.1	1.1	0.0	0.0	1.1	1.1
Hlo	1.1	0.0	86.3	0.0	1.1	3.12	2.1	6.3	0.0	0.0
Hw	0.0	0.0	1.1	90.5	2.1	1.1	2.1	0.0	0.0	3.1
Nc	4.2	2.1	0.0	0.0	83.2	2.1	3.2	2.1	1.1	2.1
Syu	3.1	1.1	3.2	1.1	1.1	69.4	1.1	9.45	1.1	0.0
Sry	3.1	1.1	0.0	0.0	1.1	0.0	92.6	0.0	1.1	1.1
Tk	0.0	1.1	4.2	1.1	3.2	6.3	0.0	82.1	0.0	2.1
Tm	0.0	4.2	0.0	0.0	1.1	0.0	4.2	0.0	89.5	1.1
Wl	0.0	2.1	0.0	3.2	1.1	1.1	0.0	0.0	1.1	91.6

Figure 8. Confusion matrix for subject dependent tests

that uses longer lengths temporal information (i.e. temporal signature length 5) in comparison to those without (i.e. temporal signature length 1). This was achieved by comparing the accuracies of both the above classifiers as the number of weak classifiers vary. The results seen in Figure 6 show that using longer temporal signatures results in a higher accuracy as well a faster convergence to peak performance.

3.3.2 Subject Independent Results

The leave-one-subject-out cross validation results for the subject independent case are similarly summarised using the confusion matrix shown in Figure 9. The overall accuracy rates for subject independent tests are lower than those in the subject dependent test due to the increased difficulty of the problem. This arises from differences in visual appearance of the mouth and more importantly, the different ways in which different subjects utter the same set of speech. However, the accuracy for both "Welcome" and "Have a nice time" are very good, considering the difficulty of the problem. As observed in the subject dependent tests, the two classes with the lowest accuracy are "See you" and "Thank you", again due to the similar way in which they are both spoken. We find that the difference in the strong classifiers performance with and without longer term temporal information is less, as can be seen in Figure 7. Whilst it can be seen that ultimately the strong classifiers with temporal information converge to a higher peak performance, this only happens after 60 boosted iterations.

3.3.3 Results Summary

Method	SD	SI
TGD-Boosting	86.5%	65.6%
ST-LBP SVM[11]	70.2%	62.4%
Graph Embedding[12]	90.6%	N/A

A summary of the proposed methods results in comparison to existing approaches for subject dependent(SD) and subject independent(SI) tests can be seen in the above table. The subject dependent score is significantly higher than the original method using temporal LBP-based SVMs. It is important to note that this accuracy was achieved using a small number of very simple binary comparison features. It is also worth noting that although our subject dependent accuracy is marginally behind than the graph embedding approach, it is unclear whether that approach is better when it comes to subject independent tests, since no results along these lines were available.

ExM	Excuse Me
Gb	Goodbye
Hlo	Hello
Hw	How are you
Nc	Nice to see you
Syu	See you
Sry	I'm sorry
Tk	Thank you
Tm	Have a good time
Wl	Your welcome

	ExM	Gb	Hlo	Hw	Nc	Syu	Sry	Tk	Tm	Wl
ExM	54.0	2.0	1.0	0.0	7.0	7.0	6.0	8.0	13.0	2.0
Gb	3.0	65.0	0.0	1.0	3.0	2.0	2.0	2.0	7.0	15.0
Hlo	1.0	0.0	59.0	8.0	1.0	5.0	4.0	19.0	0.0	3.0
Hw	0.0	1.0	7.0	66.0	7.0	3.0	2.0	7.0	0.0	7.0
Nc	6.0	0.0	0.0	3.0	64.0	4.0	4.0	1.0	9.0	9.0
Syu	12.0	1.0	9.0	1.0	4.0	49.0	5.0	16.0	1.0	2.0
Sry	10.0	2.0	2.0	1.0	2.0	3.0	61.0	2.0	16.0	1.0
Tk	8.0	3.0	21.0	4.0	1.0	16.0	0.0	41.0	3.0	3.0
Tm	3.0	2.0	0.0	1.0	3.0	0.0	7.0	2.0	80.0	2.0
Wl	0.0	7.0	1.0	5.0	4.0	2.0	2.0	4.0	4.0	71.0

Figure 9. Confusion matrix for subject independent tests

4. Future Work and Conclusion

This paper has proposed a novel machine learning algorithm (TGD-Boosting) to tackle the problem of lipreading by building visual sequence classifiers that are based on salient temporal signatures. The temporal signatures used in this paper allow us to capture spatio-temporal information that spans multiple feature dimensions with possible gaps in the temporal axis. Given that selecting suitable temporal signatures by exhaustive search is not possible due to the immense search space, a novel highly efficient gradient-descent based method is proposed to search for a suitable candidate temporal signature with $O(T^{Max}D)$ complexity, where the maximum length of the temporal pattern is T^{Max} and the static feature dimensionality is D. This temporal search method is then integrated into the boosting framework resulting in the TGD-Boosting algorithm. The resulting spatio-temporal strong classifiers were applied to lipreading by performing multi-class recognition on the OuluVS database. Experimental results show that our method achieves state of the art performane in both subject dependent and subject independent cases, using only a small set of temporal signatures. Future work will concentrate on evluating the method on different features, for example spatio-temporal LBPs that capture short-term information. Additionally, we will aim to evaluate the performance of our method on a wider range of lip-reading datasets such as AVLetters[4] and CUAVE[7] datasets.

References

[1] N. Brooke. Using the visual component in automatic speech recognition. In *Proceedings of International Conference on Spoken Language*, 1996. 1

[2] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28, 1998. 2

[3] Y. Lan, B. Theobald, E. Ong, and R. Harvey. Improving visual features for lip-reading. In *Proceedings of International Conference on Auditory-Visual Speech Processing*, 2010. 1

[4] I. Matthews, T. Cootes, J. Bangham, and S. Cox. Extraction of visual features for lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):198–213, 2002. 1, 8

[5] S. Nowozin, G. Baklr, and K. Tsuda. Discriminative subsequence mining for action classification. In *Proceedings of IEEE Conference on Computer Vision*, 2007. 2

[6] E. Ong, Y. Lan, B. Theobald, R. Harvey, and R. Bowden. Robust facial feature tracking using selected multi-resolution linear predictors. In *Proceedings of the 12th International Conference on Computer Vision*, 2009. 5

[7] E. Patterson, S. Gurbuz, Z. Tufekci, and J. Gowdy. Cuave: A new audio-visual database for multimodal human-computer interface research. In *Proceedings of ICASSP*, volume 2, pages 2017–2020, 2002. 8

[8] J. Pei, J. Han, B. Mortazavi-asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In *Proceedings of International Conference on Data Engineering*, 2001. 2

[9] P. Smith, N. Lobo, and M. Shah. Temporal boost for event recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2005. 2

[10] J. Yuan, J. Luo, and Y. Wu. Mining compositional features for boosting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008. 2

[11] G. Zhao, M. Barnard, and M. Pietikainen. Lipreading with local spatiotemporal descriptors. *IEEE Transactions on Multimedia*, 11(7), 2009. 2, 4, 6, 7

[12] Z. Zhou, G. Zhao, and M. Pietikinen. Lipreading: A graph embedding approach. In *Proc. 20th International Conference on Pattern Recognition*, 2010. 2, 4, 7