VISUAL SPEECH RECOGNITION USING SUPPORT VECTOR MACHINES

Mihaela Gordan *Constantine Kotropoulos* *Ioannis Pitas*

Technical University of Cluj-Napoca
15 C. Daicoviciu Str.
Cluj-Napoca RO-3400, ROMANIA

Department of Informatics
Aristotle University of Thessaloniki
Box 451, Thessaloniki 540 06, GREECE

mihag@bel.utcluj.ro {costas, pitas}@zeus.csd.auth.gr

Abstract: In this paper we propose a visual speech recognition network based on Support Vector Machines. Each word of the dictionary is described as a temporal sequence of visemes. Each viseme is described by a support vector machine, and the temporal character of speech is modeled by integrating the support vector machines as nodes into a Viterbi decoding lattice. Experiments conducted on a small visual speech recognition task show a word recognition rate on the level of the best rates previously reported, even without training the state transition probabilities in the Viterbi lattice and using very simple features. This proves the suitability of support vector machines for visual speech recognition.

1. INTRODUCTION

The recognition of speech from the visual information only is referred as visual speech recognition or lipreading. Different shapes of the mouth (i.e. different mouth openings, different position of the teeth and tongue) realized during speech cause the production of different phones. A mouth shape and mouth dynamics corresponding to the production of a phone or a group of phones indistinguishable in the visual domain defines a *viseme* [6]. One can establish a correspondence between visemes and phonemes, even if this correspondence is not one-to-one, but one-to-many, due to the involvement of non-visible parts of the vocal tract in the speech production. Still, for word dictionary of small size, we can perform good quality speech recognition using only a viseme-level description of the words.

Many methods have been proposed for solving the visual speech recognition problem in the literature. The different types of solutions adopted vary widely with respect to: the feature types; the classifier used; the class definition. For example, Bregler uses time-delayed neural networks (TDNN) for visual classification, and the outer lip contour coordinates as visual features [4]. Luettin uses active shape models for representing different mouth shapes, gray level distribution profiles (GLDPs) around the outer and/or inner lip contours as feature vectors, and finally builds whole-word hidden Markov models (HMM) for visual speech recognition [5]. Movellan employs also HMMs for building visual word models, but uses directly the gray levels of mouth images as features after some simple preprocessing to exploit the vertical symmetry of the mouth [3].

Despite the variety of existing strategies for visual speech recognition, there is still ongoing research in this area, attempting to: 1) find the most suitable features and classification techniques to discriminate as good as possible between different mouth shapes, but to keep in the same class the mouth shapes corresponding to the same phone produced by different individuals (i.e., to be individual-independent) thus leading to higher visual speech recognition rates; 2) require as few processing of the mouth image as possible, to allow the implementation in real time of the mouth shape classifier considering that the end use of mouth shape classifier is in audio-visual speech recognition systems, which are supposed to work in real-time; 3) facilitate the easy integration of audio and video speech recognition.

In this paper, we aim to contribute to the first two aspects mentioned above by examining the suitability of support vector machines (SVMs) for visual speech recognition tasks motivated by the fact that SVMs have been proved powerful classifiers in various pattern recognition applications such as face detection, face recognition, etc., to mention a few. Very good results in audio speech recognition using SVMs were recently reported in [1]. No attempts in applying SVMs for visual speech recognition have been reported so far, although a somehow closely related application is described in [2],where SVMs were applied for detecting the degree of opening/smile of mouth images in videosequences. This work uses SVMs for linear regression, not for classification task. Thus, according to the best of the author's knowledge, the use of SVMs as visual speech classifiers is a novel idea.

One of the reasons for not using SVMs in audiovisual speech recognition so far is the fact that they are inherently static classifiers, whilst speech is a dynamic process, where the temporal information is essential for recognition. A solution to mitigate this deficiency is presented in [1], where a combination of HMM with SVM is proposed. In this paper we adopt a similar strategy for modeling the visual speech dynamics with the difference that we shall use only the Viterbi algorithm employed by an HMM to create dynamically visual word models. Another novel aspect in the visual speech recognition approach proposed here refers to the strategy adopted for building the word models: while most of the applications presented in the literature [1, 5, 3] build whole word mod-

els as basic visual models, our basic visual models are viseme models, and the visual word model is obtained by the combination of these basic models into a temporal dynamic sequence. This approach offers the advantage of an easier generalization to large vocabulary word recognition tasks without significantly increasing the storage requirements by maintaining the dictionary of basic visual models needed for word modeling into a reasonable limit.

The word recognition rate obtained is on the level of the best previous reported rates in literature, although we will not attempt to learn the state transition probabilities. In the case of using very simple features (i.e. pixels), our word recognition rate is superior to the ones reported in the literature. The viseme-oriented approach also facilitates the integration of audio and visual speech recognition.

2. OVERVIEW OF SUPPORT VECTOR MACHINES

SVMs is a principled technique to train classifiers that stems from statistical learning theory [7, 8]. Their root is the optimal hyperplane algorithm. They minimize a bound on the empirical error and the complexity of the classifier at the same time. Accordingly, they are capable of learning in sparse high-dimensional spaces with relatively few training examples. Let $\{\mathbf{x}_i, y_i\}, i = 1, 2, \ldots, N$, denote N training examples where $\mathbf{x}_i$ comprises an M-dimensional pattern and y_i is its class label. Without any loss of generality we shall confine ourselves to the two-class pattern recognition problem. That is, $y_i \in \{-1, +1\}$. We agree that $y_i = +1$ is assigned to positive examples, whereas $y_i = -1$ is assigned to counterexamples.

The data to be classified by the SVM might be linearly separable in their original domain or not. If they are separable, then a simple linear SVM can be used for their classification. However, the power of SVMs is demonstrated better in the nonseparable case, when the data cannot be separated by a hyperplane in their original domain. In the latter case, we can project the data into a higher dimensional Hilbert space and attempt to linearly separate them in the higher dimensional space using kernel functions. Let Φ denote a nonlinear map $\Phi : \mathcal{R}^M \rightarrow \mathcal{H}$ where $\mathcal{H}$ is a higher-dimensional Hilbert space. SVMs construct the optimal separating hyperplane in $\mathcal{H}$. Therefore, their decision boundary is of the form:

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{N} \alpha_i \, y_i \, K(\mathbf{x}, \mathbf{x}_i) + b\right) \qquad (1)$$

where $K(\mathbf{z}_1, \mathbf{z}_2)$ is a kernel function that defines the dot product between $\Phi(\mathbf{z}_1)$ and $\Phi(\mathbf{z}_2)$ in $\mathcal{H}$, and α_i are the nonnegative Lagrange multipliers associated with the quadratic optimization problem that aims to maximize the distance between the two classes measured in $\mathcal{H}$ subject to the constraints

$$\begin{aligned} \mathbf{w}^T\Phi(\mathbf{x}_i) + b \geq 1 \qquad & \text{for } y_i = +1 \\ \mathbf{w}^T\Phi(\mathbf{x}_i) + b \leq 1 \qquad & \text{for } y_i = -1. \end{aligned} \qquad (2)$$

Frequently used kernel functions are:

1. the polynomial kernel:
   $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = (m\mathbf{x}_i^T\mathbf{x}_j + n)^d$;
2. the Radial Basis Function (RBF) kernel:
   $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = exp\{-\gamma|\mathbf{x}_i - \mathbf{x}_j|^2\}$.

In the following, we will omit the sign function from the decision boundary (1) that simply makes the optimal separating hyperplane an indicator function.

To enable the use of SVMs in visual speech recognition, when we model the speech as a temporal sequence of symbols corresponding to the different phones produced, we shall employ the SVMs as nodes in a Viterbi lattice. The nodes of such a Viterbi lattice are supposed to generate the posterior probabilities of the corresponding symbols to be emitted [10], and the standard SVMs do not provide such probabilities as output. A good solution to map the SVM outputs into probabilities is proposed by Platt [11]. Having a trained SVM, we can convert its output to probability by training the parameters of a sigmoidal mapping function. In general, the class-conditional densities on either sides of the SVM's hyperplane are exponential. So, Bayes' rule on two exponentials suggests the following parametric form of a sigmoidal function:

$$P(y = +1|f(\mathbf{x})) = \frac{1}{1 + \exp(a_1 f(\mathbf{x}) + a_2)} \qquad (3)$$

where a_1 and a_2 are the parameters of the sigmoidal mapping to be derived for the trained SVM under consideration with $a_1 < 0$. $P(y = +1|f(\mathbf{x}))$ gives directly the posterior probability to be used in the Viterbi decoder. The parameters a_1 and a_2 are derived from the training set $\{f(\mathbf{x}_i), y_i\}$, $i = 1, 2, \ldots, N$, using maximum likelihood estimation. The detailed description of the training algorithm can be found in [11].

3. THE PROPOSED APPROACH TO VISUAL SPEECH RECOGNITION

The problem of discriminating between different mouth shapes during speech production can be viewed as a pattern recognition problem. In this case, the set of patterns is a set of feature vectors $\{\mathbf{x}_i\}, i = 1, 2, \ldots, P$, each of them describing some mouth shape. The feature vector $\mathbf{x}_i$ is a representation of the mouth image (either low-level, such as the gray levels from a rectangular image region containing the mouth, geometric parameters such as the mouth width, height, perimeter, or the coefficients of a linear transformation of the mouth image). All the feature vectors from the set have the same number of components, M.

Let us denote the pattern classes by $\mathcal{C}_j, j = 1, 2, \ldots, Q$ where Q is the total number of classes. Each pattern class $\mathcal{C}_j$ is a group of patterns that represent mouth shapes corresponding to the same viseme. The class label of the class $\mathcal{C}_j$ is denoted by l_j.

A network of Q parallel SVMs is designed where each SVM is trained to classify test patterns in class $\mathcal{C}_j$ or its complement $\mathcal{C}_j^C$ (i.e., not in class $\mathcal{C}_j$). To derive an unambiguous classification we assign $\mathbf{x}_k$ to the class $\mathcal{C}_l$

according to a maximum a posteriori classification rule, such that:

$$P\left(y_l = 1 \,|f_l(\mathbf{x}_k)\right) = \arg\max_{j=1}^{Q} P\left(y_j = 1 \,|f_j(\mathbf{x}_k)\right) \quad (4)$$

where the right side probabilities are given by the SVMs probabilistic outputs.

This pattern recognition problem can be applied to visual speech recognition in the following way: each unknown pattern represents the image of the speaker's face at a certain time instant; each class label represents one viseme. Accordingly, we shall identify what viseme is produced at any time instant in the spoken sequence.

By its nature, speech is a temporal process. Each spoken word can be modelled in the visual domain as a sequence of visemes corresponding to some basic sounds, called here visemic model. The most natural way of representing the word models in the temporal domain, starting only from the symbolic visemic model and from the total number of T frames in the word pronunciation, is to assume that the duration of each viseme in the word pronunciation can be whatever, but **necessarily not zero**. Thus, we can create a temporal network of models corresponding to the different possible durations of the visemes in the model, containing as many states as many frames we have in the videosequence, that is, T. The most straightforward way to represent such a network of models is the Viterbi algorithm [10]. The resulting Viterbi lattice is shown in Figure 1 for the same example of the word "one", where the visemes present in the word pronunciation have been denoted with the same letters as the underlying phones. The paths formed by the solid lines show the possible model realizations. Each node signifies the realization of the corresponding viseme at that particular time instant. Each word from the dictionary of D words, w_d, $d = 1, 2, \ldots, D$, will have its own Viterbi lattice model. Let us interpret each node in the lattice of Figure

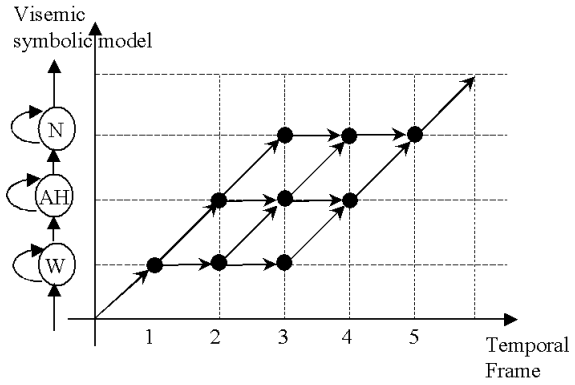

Fig. 1. The temporal Viterbi lattice for the pronunciation of the word "one" in a videosequence of 5 frames

1 as the probability that the corresponding symbol o_k is emitted at the time instant k. We denote this probability by $b_{o_k k}$. Each solid line between the nodes corresponding to the symbol o_k at the time instant k and o_{k+1} at the time instant $k+1$ represents the transition probability from the state that is responsible for the generation of o_k to the state that generates the symbol o_{k+1}. We denote the latter probability by $a_{o_k o_{k+1}}$, where o_k and o_{k+1} may be different or not.

Having a videosequence of T frames for a word pronounced and such a Viterbi model for each word w_d, $d = 1, 2, \ldots, D$ from the dictionary, we can compute the probability for the word w_d to be produced following a path ℓ in the Viterbi lattice as:

$$p_{d,\ell} = \prod_{k=1}^{T} b_{o_k k} \cdot \prod_{k=1}^{T-1} a_{o_k o_{k+1}|d,\ell} \,. \quad (5)$$

and the probability for the word w_d to be produced as the maximum over all possible $p_{d,\ell}$ s. Among the words that can be produced following all the possible paths in all the D Viterbi lattices, the most probable word, that is, whose probability p_d, $d = 1, 2, \ldots, D$, is maximum is finally recognized.

In the visual speech recognition approach discussed in this paper, the symbol emission probabilities $b_{o_k k}$ are given by the corresponding SVMs, SVM_{o_k}. To a first approximation, we assume equal transition probabilities $a_{o_k o_{k+1}}$ between whatever two symbol emission states.

4. EXPERIMENTAL RESULTS

To evaluate the recognition performances of the proposed SVM-based visual speech recognizer, we choose to solve the task of recognizing the first four digits in English from the small audiovisual database Tulips1 [3], frequently used in similar visual speech recognition experiments. First we define the viseme classes for each word, based on their phonetic descriptions [12] trough the manual annotation of the training set. The visual speech recognizer requires the training of 12 different SVMs, one for each distinct mouth shape considered. We used for our experiments SVMs with polynomial kernel of degree 3. We used two types of features: 1) The first type comprises the gray levels of a rectangular region of interest around the mouth, downsampled to the size 16×16 and scanned raw by raw. Each mouth image is represented by a feature vector of length 256. 2) The second type represents each mouth image frame at the time T_f by a vector of double size i.e. $2 \times 256 = 512$, that comprises the gray levels of the rectangular region of interest as previously, and the temporal derivatives of the gray levels normalized to the range $[0, L_{Max} - 1]$ (where L_{Max} is the maximum gray level value in mouth image). The temporal derivatives are simply the pixel by pixel gray level differences between the frames T_f and $T_f - 1$ and are called **delta features**.

The complete visual speech recognizer was implemented in C++. We used the publicly available SVM-Light toolkit modules for the training of the SVMs [9] and implemented in C++ the module for learning the sigmoidal mapping of the SVMs output to probabilities and the module for generating the Viterbi decoder lattice based on SVMs with probabilistic outputs.

We performed speaker-independent visual speech recognition tests, using the leave-one-out testing strategy for the 12 subjects in the Tulips1 database. More precisely, the testing strategy was as follows: we trained the system 12 times separately, each time using 11 subjects in

Table 1. The overall WRR of the SVM dynamic network compared to other techniques.

Method	SVM-based dynamic network without delta features	SVM-based dynamic network with delta features	AAM and HMM system (shape + intensity model, inner + outer lip contour) without delta features [5]	AAM and HMM system (shape + intensity model, inner + outer lip contour) with delta features [5]	Stochastic networks without delta features [3]	Stochastic networks with delta features [3]
WRR [%]	76	90.6	87.5	90.6	60	89.93

the training set and leaving the 12^{th} subject out for testing. In this way, we obtained a total of 96 video test sequences. We examine the overall percentaged word recognition rate WRR, comparing this result with the ones reported in literature under similar conditions (i.e., using the same features, the same database and the same testing procedure) [5, 3] in Table 1. We can see that our results are on the same level as the best ones reported in the literature ($WRR = 90.6\%$). However the features used by us are simpler than those used in literature to obtain the same WRR. For the shape $+$ intensity models [5] the gray levels should be sampled in the exact subregion of the mouth image containing the lips, around the inner and outer lip contours, and should exclude the skin areas. Accordingly, the method reported in [5] requires the tracking of the lip contour in each frame, which increases the processing time of visual speech recognition. Moreover we notice that our very good WRR was obtained without training the transition probabilities in the Viterbi decoding lattice from whole-word models. An improvement of the WRR is expected when training of the transition probabilities is implemented and the trained transition probabilities are incorporated in the Viterbi decoding lattices.

5. CONCLUSIONS

We examined the suitability of SVMs with probabilistic outputs in visual speech recognition by employing them into a dynamic temporal network implemented by a Viterbi decoding lattice as nodes and testing the proposed method on a small visual speech recognition task. We obtained good word recognition rates as compared to the state of the art results from the literature. This demonstrates that SVMs are promising classifiers for visual speech recognition tasks. Another advantage of the viseme-oriented modeling method proposed is the possibility of easier generalization to larger vocabularies. In our future research, we will try to improve the performance of the visual speech recognizer by using other kernel functions and learning the state transition probabilities of the Viterbi decoding lattice.

REFERENCES

[1] A. Ganapathiraju, J. Hamaker, and J. Picone. "Hybrid SVM/HMM architectures for speech recognition," in *Proc. of Speech Transcription Workshop*, College Park, Maryland, USA, May 2000.

[2] V. P. Kumar and T. Poggio. "Learning-based approach to real time tracking and analysis of faces," in *Proc. of AFGR*, 2000.

[3] J. R. Movellan. "Visual speech recognition with stochastic networks," in *Advances in Neural Information Processing Systems*, (G. Tesauro, D. Toruetzky, and T. Leen, Eds.), Vol 7, MIT Press, Cambridge, MA, 1995

[4] C. Bregler and S. Omohundro. "Nonlinear manifold learning for visual speech recognition," in *Proc. IEEE ICCV*, 1995, pp. 494-499.

[5] J. Luettin and N. A. Thacker. "Speechreading using probabilistic models," *Computer Vision and Image Understanding*, 65(2):163-178, February 1997

[6] C. Benoît, T. Lallouache, T. Mohamadi, and C. Abry. "A set of French visemes for visual speech synthesis," in *Talking machines: Theories, Models, and Designs*, (G. Bailly and C. Benoît, Eds.), 485-504, North Holland, Elvsevier, Amsterdam, 1992

[7] V.N. Vapnik. *Statistical Learning Theory*, J. Wiley, N.Y., 1998

[8] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines*, Cambridge University Press, Cambridge, U.K., 2000

[9] T. Joachims. "Making large-scale SVM learning practical," in *Advances in Kernel Methods - Support Vector Learning*, (B. Scoelkopf, C. Burges, and A. Smola, Eds.), MIT-Press, 1999

[10] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK book*, Entropic, Ltd., Cambridge, UK, HTK version 2.2 edition, 1999

[11] J. Platt. "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, (A. Smola, P. Bartlett, B. Scholkopf, and D. Schuurmans, Eds.), MIT Press, Cambridge, MA, 2000

[12] The Carnegie Mellon University Pronouncing Dictionary v. 0.6. http://www.speech.cs.cmu.edu/cgi-bin/cmudict