

Lipreading: A Graph Embedding Approach

Ziheng Zhou, Guoying Zhao and Matti Pietikäinen

*Machine Vision Group, Department of Electrical and Information Engineering
University of Oulu, Finland
{ziheng.zhou, gyzhao and mkp}@ee.oulu.fi*

Abstract

In this paper, we propose a novel graph embedding method for the problem of lipreading. To characterize the temporal connections among video frames of the same utterance, a new distance metric is defined on a pair of frames and graphs are constructed to represent the video dynamics based on the distances between frames. Audio information is used to assist in calculating such distances. For each utterance, a subspace of the visual feature space is learned from a well-defined intrinsic and penalty graph within a graph-embedding framework. Video dynamics are found to be well preserved along some dimensions of the subspace. Discriminatory cues are then decoded from curves of the projected visual features to classify different utterances.

1. Introduction

It is known that speech perception is a multimodal process which involves information not only from what we hear (audio) but from what we see (visual) [1]. Although languages, e.g., English, cannot be completely distinguished only by visual cues (such as lip movements and facial expressions), techniques for interpreting speech using visual information only (namely, lipreading) still have a wide range of applications in the real world. For instance, such a system can be used to understand someone's speaking in a highly noisy environment (e.g., among a large crowd or in a moving vehicle). It can also be used to improve the quality of the lives of people with hearing impairments.

In the case that there are a limited number (e.g., dozens) of words or phrases to be recognized, the problem of lipreading can be considered as a problem of classifying video sequences of utterances. It differs from the problem of classifying subjects in videos whose frames all belong to one subject (class) (e.g., [2]). For lipreading, each frame contains a certain appearance of a speaker's mouth when he/she speaks. The same appearance could occur when uttering different words or phrases, indicating that the video dynamics or the

temporal change of mouth appearance play an important role for the problem of lipreading.

There have been a few systems developed for lipreading [3-6]. From the point of view of characterizing video dynamics, most of the methods [3-5] considered speech as a stochastic process and modeled the video dynamics using the hidden Markov models. More recently, Zhao et al. [6] analyzed video sequences in the XYT space. Here XY stands for the image plane and T for temporal positions of individual frames. They found out that the dynamic information was contained in the textures formed by image frames in the XT and YT plane. The LBP texture descriptor was used to extract such information. In all of these methods, audio information of training data was considered as redundancy and therefore, discarded.

In this paper, we propose to use graph embedding to capture video dynamics. Instead of ignoring audio information, we use it to better align video sequences of the same utterance. A new distance metric on a pair of frames is then defined based on the alignment results. Graphs are constructed based the distance metric to characterize the temporal connections among video frames. For each utterance, the high-dimensional visual features are mapped into a low-dimensional subspace of which video dynamics are well preserved by the curves of the mapped features along each dimension. Discriminatory cues are decoded from the curves for classification. Note that the visual features mentioned here are not meant to be restricted to any particular kind of features extracted from images.

2. Proposed Method

The overview of the proposed method can be found in Figure 1. The training of the system includes two major issues: the graph representation of video sequences and the graph embedding. They will be described in Sections 2.1 and 2.2, respectively. After that, Section 2.3 will give details about how to decode the discriminatory information within the subspaces learned for various utterances.

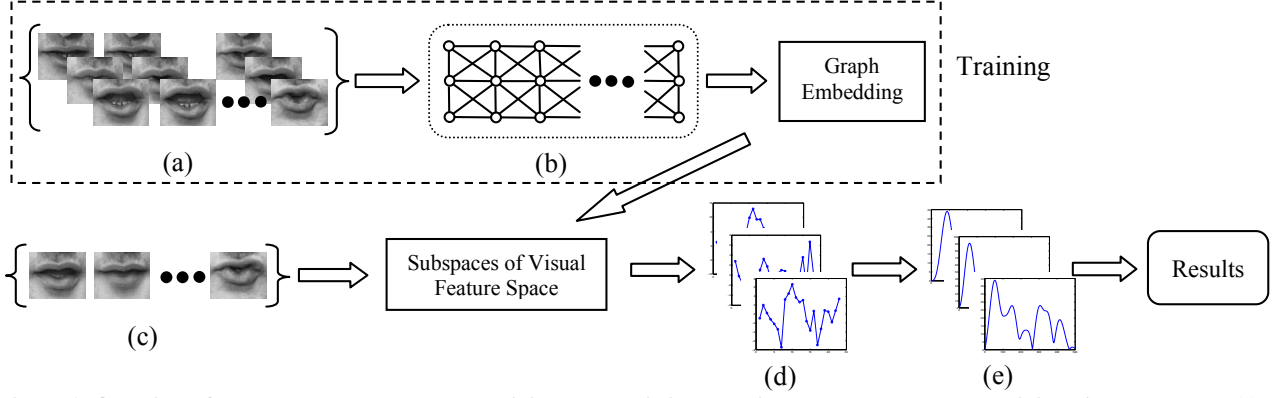


Figure 1. Overview of the proposed method. The training process is included in the dashed box. The training video sequences (a) of different utterances are firstly represented by graphs (b) which are then used to learn some subspaces of the visual feature space through the graph embedding framework described in [7]. Given a test sequence (c), the extracted visual features are mapped into the subspaces. The spectra (e) of the mapped feature curves (d) along some particular dimensions are calculated and the magnitudes within certain frequency bands are used as cues to determine the associated utterance.

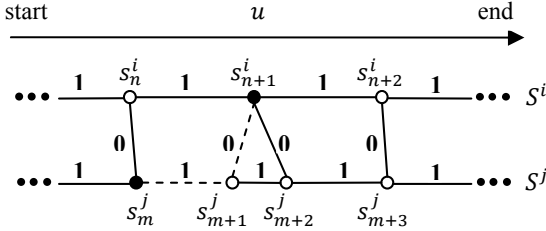


Figure 2. An example of how to calculate distances between inter-sequence frames. Here, S^i and S^j are two video sequences of the same utterance u , and s_n^i and s_m^j are the n^{th} and m^{th} frames of S^i and S^j respectively. The distance between two connected vertices is shown alongside the edge.

2.1. Graph Representation

Graph representation of high-dimensional data has been widely used for dimensionality reduction [7]. When it is used for solving a classification problem, within a graph, edges are often added to connect vertices (data points) of the same class and the associated weights are given according to the distances between vertices. In this way, data points of the same class are expected to be close to each other in the learned low-dimensional feature space. Very often, the distance metric is defined in the original feature space (e.g., the Euclidean distance metric).

Such a distance metric, however, is not suitable for preserving the dynamics among video frames of the same class (or utterance) since we cannot assign any frame uniquely to one utterance. Here, we introduce a new distance metric on a pair of frames. There are two types of distances to be considered: the intra-sequence distance – the distance between two frames within the same video sequence and the inter-sequence distance – the distance between two frames belongs to two different video sequences of the same utterance.

Let S denote a video sequence and s_1, \dots, s_L be its L frames. The intra-sequence distance between the m^{th} frame, s_m and the n^{th} frame, s_n ($m < n$) is simply

defined as $n - m$. The closer two frames are in the sequence, the shorter their distance is. For the inter-sequence distance, let us consider the case when there are only two sequences of the same utterance. For each frame in one sequence, we want to find among all the frames in the other sequence its correspondence such that they correspond to the same part of the utterance spoken in the two videos. To do that, their audio data are firstly aligned. Audio samples are divided into a number of short-term states and the Mel-frequency cepstral coefficients are calculated as features. The dynamic time warping is implemented to align the two sequences of states and the results are then used to find correspondences for video frames. Note that audios are only used when constructing graphs during training. For testing, visual information will be the only source from which we make classification decisions.

After that, we set the distance between a frame and its correspondence as zero. Once the distance between two frames is defined, an edge will be added to connect them in the graph. Figure 2 illustrates the graph constructed for S^i and S^j by far. (Note that we do not show any edge that connects two frames having a distance larger than 1 in the figure.) It can be seen that it is possible that multiple frames have the same correspondence. For any pair of frames having not been connected (e.g., s_{n+1}^i and s_m^j in Figure 2), we calculate their inter-sequence distance as the minimum sum of the distances along the paths that connect them. One of the shortest paths found for s_{n+1}^i and s_m^j is marked by the dashed lines as an example. The way we calculate distances can be directly applied to the cases when there are more than two video sequences of the same utterance. Let d denote the distance between two frames, the weight, w associated with their edge in the graph is computed by the heat kernel [8]:

$$w = e^{-\frac{d^2}{\rho}} \quad (1)$$

where ρ is a positive real-value constant determined empirically.

2.2. Graph Embedding

Recently, Yan et al. [7] proposed a graph embedding framework for dimensionality reduction. Within this framework, two graphs are constructed: an intrinsic graph G that contains the desired statistical and geometric properties of its vertices and a penalty graph G^P that contains the properties to be suppressed. In this work, we adopt the linear extension of the direct graph embedding. Let \mathbf{X} be the matrix that contains the visual feature vectors extracted from all of the video frames in the training dataset. (Here, each column of \mathbf{X} is a feature vector.) Given G and G^P , in the original feature space, the directions α^* along which the high-dimensional visual features are to be projected can be calculated as:

$$\alpha^* = \underset{\alpha^T \mathbf{X} \mathbf{L}^P \mathbf{X}^T \alpha = c}{\arg \min} \alpha^T \mathbf{X} \mathbf{L} \mathbf{X}^T \alpha \quad (2)$$

where \mathbf{L} and \mathbf{L}^P are the Laplacian matrices of G and G^P and c is a constant.

For each utterance u_i , we learn a linear map from the original feature space to a D -dimensional subspace. Here D is much smaller than the original feature dimension. To build the intrinsic graph G_i , we just simply connect and calculate weights for all the video frames belonging to u_i following the way described in Section 2.1. In the penalty graph G_i^P , we do the same operation for all the utterances except u_i . In such a way, the temporal connections among frames of the current utterance are emphasized while the video dynamics of other utterances are suppressed. Those frames correspond to non-speaking periods in videos are connected in the penalty graph if the continuous length of such frames are longer than, for example, five frames.

2.3. Decoding

After graph embedding, we can now map video frames from the original visual feature space into the learned D -dimensional subspaces. It is expected that the temporal connections among video frames of the same utterance would be preserved in these subspaces. In this subsection, we give details of how to decode the discriminatory information from the projected features.

Once again, we cannot use distances between video frames to classify utterances in the subspaces since frames do not uniquely belong to any of the utterances as explained in Section 1. We have found out that the visual features extracted from training sequences are projected in a desired way along some dimensions of the subspace learned for the utterance they belong to. Figure 3 shows an example of a training sequence mapped into the subspace learned for its associated utterance. The projected curves alongside the first six dimensions are given in Figure 3(a)-(f). The curves (except the one in the third dimension) can be well characterized by sine waves with gradually increased frequencies. Here, we call such dimensions the *dominant dimensions (DDIMs)*. We then

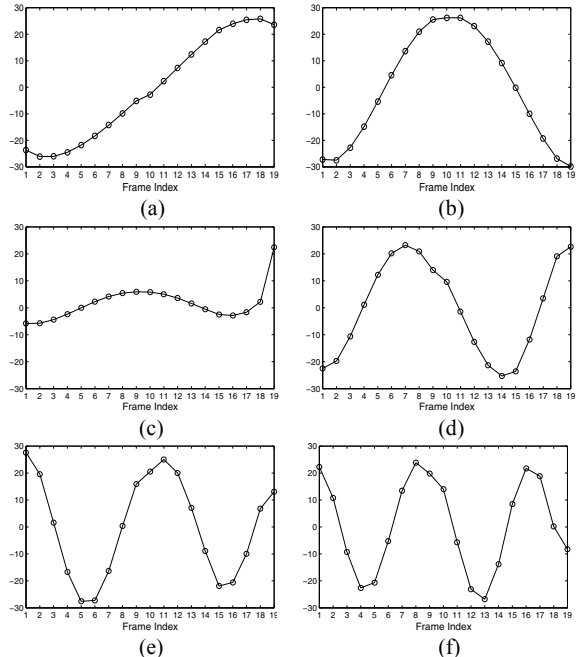


Figure 3. Curves of the projections of the visual features extracted from a training sequence alongside the first six dimensions (from (a) to (f)) of the subspace learned for the utterance to which the sequence belongs.

refine the map by only keeping the DDIMs. Moreover, it has been found out that video frames of other utterances in the training dataset are unanimously mapped to some values very close to zero in all the DDIMs

It is assumed that the curves from an unseen test sequence along the DDIMs should have similar characteristics as the curves from training sequences. For utterance u_i , along the j^{th} DDIM of its corresponding subspace, the Fourier transform is applied to the curve mapped from each of its training sequences. We then record the frequencies corresponding to the maximum magnitude in each of the obtained spectra, from which the mean f_i^j and standard deviation σ_i^j are computed for u_i . During testing, an unseen video sequence is mapped into the (refined) subspace learned for each u_i . Along the j^{th} DDIM, we expect a dominant peak around f_i^j in the spectrum if the sequence belongs to u_i . To detect such dominance, we record the maximum magnitude in the frequency band $(f_i^j - 2\sigma_i^j, f_i^j + 2\sigma_i^j)$ as the evidence of the sequence belonging to u_i . The measured magnitudes are then summed up as output. The test video sequence is assigned to the utterance with the maximum sum of magnitudes.

3. Experiments and Results

In our experiments, the OuluVS database [6] was used for performance evaluation. The database includes 817 sequences from 20 speakers. Each of them was asked to speak 10 different utterances including words, phrases and short sentences. See [6] for details of the database.

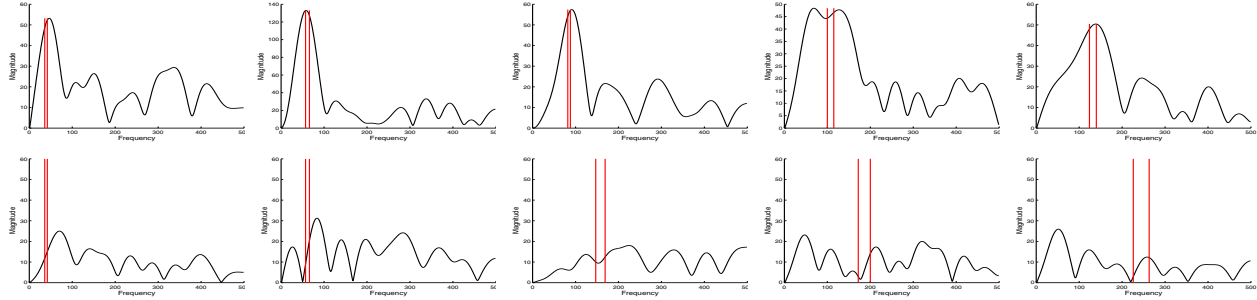


Figure 4. Spectra of a test video sequence along dimensions of the subspace learned for the utterance it belongs to (the first row) and of the subspace learned for another utterance (the second row). The vertical lines outline the boundaries of the frequency bands in which the maximum magnitudes are used as discriminatory cues.

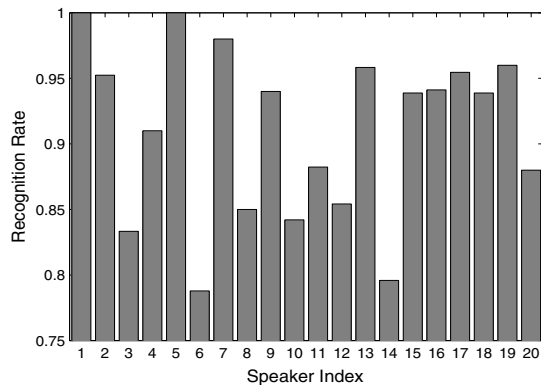


Figure 5. Lipread results for individual speakers.

The video sequences were collected using 25 fps and each frame was of resolution 720×576 . For normalization, eye positions were marked manually every five frames and a mouth region of size 70×84 was cropped from each of the video frames.

Some of the parameters used in the test were set as follows. To extract visual features, a normalized mouth image was divided into 5×5 blocks. The $LBP_{8,2}^{u2}$ and $LBP_{8,3}^{u2}$ [9] descriptors were used to calculate histograms of the dominant pattern from each block. We then concatenated all the histograms to form the visual feature vector for the image. To compute edge weights, constant ρ in Equation 1 was set as $\rho = 2$. In graph embedding, we first chose $D = 20$ to learn a subspace for an utterance. The subspace was then refined by the first five DDIMs found.

For each of the 20 speakers, the leave-one-video-out cross validation was carried out, i.e., only one video sequence of the speaker was used for testing and the rest for training. Figure 4 shows the spectra of a test sequence calculated in two subspaces. The spectra in the first row are computed in the DDIMs of the subspace learned for the utterance it belongs to and the second row shows the spectra calculated in the subspace of another utterance. In the first row, we can clearly see dominant peaks around the learned frequency bands (outlined by the vertical lines) and the obtained magnitudes are significantly larger than those from the second row. Figure 5 shows the result for each individual speaker. In total, we achieved 90.66%

recognition rate over the whole database. To compare our method with the state-of-the-art, we implemented Zhao’s method [6] which was reported to have superior performance over other methods. We then tested their approach on the same normalized data with the same test protocol and achieved 82.2% recognition rate.

3. Conclusions

We have represented a novel graph embedding method for the problem of lipreading. A new distance metric has been defined to preserve the temporal connections among video frames of the same utterance. Low-dimensional subspaces are learned for each utterance within the graph-embedding framework described in [7]. Discriminatory information is then decoded from curves of the projected visual features along the dominant dimensions of the learned subspaces.

References

- [1] H. McGurk, and J. MacDonald, “Hearing lips and seeing voices,” *Nature*, 264, 746-748, 1976.
- [2] R. Wang *et al.*, “Manifold-manifold distance with application to face recognition based on image set,” in *CVPR*, 2008, pp. 1-8.
- [3] G. I. Chiou, and J. N. Hwang, “Lipreading from color video,” *TIP*, 6(8), 1192-1195, 1997.
- [4] K. Saenko, K. Livescu, M. Siracusa *et al.*, “Visual speech recognition with loosely synchronized feature streams,” in *ICCV*, 2005, pp. 1424-1431.
- [5] I. Matthews *et al.*, “Extraction of visual features for lipreading,” *TPAMI*, 24(2), 198-213, 2002.
- [6] G. Zhao *et al.*, “Lipreading with local spatiotemporal descriptors,” *TMM*, 11(7), 1254-1265, 2009.
- [7] S. Yan *et al.*, “Graph embedding and extensions: a general framework for dimensionality reduction,” *TPAMI*, 29(1), 40-51, 2007.
- [8] M. Belkin, and P. Niyogi, “Laplacian Eigenmaps and spectral techniques for embedding and clustering,” in *Adv. Neural Inform. Proc. Syst.* 14, 2001, pp. 585-591.
- [9] T. Ojala *et al.*, “Multiresolution gray scale and rotation invariant texture analysis with local binary patterns,” *TPAMI*, vol. 24, no. 7, pp. 971-987, 2002.