**Synphony: A video telephony encoding system**

Edson Bárcenas, Rafael Carrillo, Mauricio Díaz, Ricardo Solano, Carolina Soto, Luis Valderrama, Javier Villegas and Pedro Vizcaya.

Research group on bioengineering, signal and image processing
Pontificia Universidad Javeriana, Bogotá, Colombia
pvizcaya@javeriana.edu.co

Abstract

In this paper we present a visual telephony system, *Synphony*, which uses a new coding method based on the re-synthesis of visual speech at the receiver. This method allows the transmission of visual telephony scenes over a very low capacity channel (under 64kbps). The coding method proposes a new paradigm for sequence coding in which the objective is to re-synthesize smooth sequences coherent with the audio speech signal instead of minimizing the average distortion.

1. Introduction

In the past decades there was a growth in multimedia communications and in the development of many video coding standards such as H26X and MPEG [1]. Most of these coding standards have data rates over 64kbps. However, some of the most common channels have lower capacity than this.

It was shown that in videoconference communication the audio speech is perceptually more important than the visual information [2]. The relation between the visual features and the audio characteristics of the speech has been exploited to encode the articulatory movements, transmit them and then use animation models to reproduce them [3],[4].

A big development has been made in the visual speech synthesis field in the last years [5]. Different methods have been proposed such as those based on human models, the photorealistic approaches and parametric approximations [6]-[9]. In this paper we present a video telephony coding system where the information capacity used by the video is less than the one used by the audio, being coherent with the human perceptual system. This work proposes a fusion between video coding methods and photorealistic visual speech synthesis approaches to encode video sequences. This approach can reach a very low bit rate while presenting a very high quality video (720x480, 30fps, color).

In the next three sections we present an overview of the system and detailed descriptions of the training and online phases (sections 2, 3 and 4). Results and conclusion are given in section 4 and 5 respectively.

2. Overview

*Synphony* is a visual telephony system that allows the full-duplex transmission of visual speech scenes (video and audio) in real time between 2 terminals over a LAN. The key feature of the system is the use of re-synthesis of the visual speech to achieve very low transmission rate. The visual speech scenes must be acquired under controlled conditions of motion, lighting and background. The system operation has two main stages: training and on-line execution. At the first stage, a linguistically representative video must be acquired to obtain a codebook. The codebook is a set of images selected to represent the visual speech sequences. Figure 1 shows the main parts of the system:

![Figure 1 Block Diagram of the system](image)

3. Training

In the training phase the user should record a video with controlled conditions for motion, lighting and background, uttering sentences that contain all the visemes of the language and some transitions between them.

In the training phase the video is acquired using DirectX® 9.0 in DV format or webcam format (720x480 30fps or 320x240 15fps respectively), and a codebook is generated. The codebook images are compressed using JPEG2000 and the compressed codebook is transmitted off line over the LAN to the other terminal using TCP sockets. In the receiver the codebook is expanded. The complete process is shown in Figure 2. The training process takes typically 3 minutes to run.

![Figure 2 Training phase](image)
3.1. Segmentation

In previous works was shown that processing only the mouth region produces good results in visual speech analysis and synthesis [8], [9]. Therefore in Synphony only the mouth region is processed and an automatic real time segmentation algorithm for this region is called upon.

Facial feature segmentation has been widely explored by the image processing community [10]; however real time segmentation is not entirely developed. To obtain the mouth region Synphony uses two weak but fast classifiers in cascade: one that uses anthropometric face relations and another that seeks for the maximum motion region. The cascade connection boosts the segmentation results.

The first classifier finds an image region, bigger than the mouth region and smaller than the total image, where the mouth will be searched. The first step in this classifier is to locate the eyes and head position from the luminance plane. The Absolute Accumulative Difference Profiles (AADP) in rows and columns are used to determine the eye line and the head limits respectively. The AADP’s for columns and rows in an \( N \times M \) intensity image are defined as follows:

\[
\text{AADP}_c(k) = \sum_{i=1}^{N} |I(i,k) - I(i-1,k)| \quad \text{with} \quad k = 1, 2, \ldots M
\]

\[
\text{AADP}_r(l) = \sum_{j=1}^{M} |I(l,j) - I(l,j-1)| \quad \text{with} \quad l = 1, 2, \ldots N
\]

Once the eye line is located, the position of the eyes and the distance, \( d \), between them are calculated using a row PDAA and smooth with a low pass filter. Finally the classifier places a square of 200x200 (image search region) pixels in a central point. The horizontal coordinate of the point is the mid point between the eyes, and the vertical coordinate of the point is the vertical coordinate of the eye line plus \( d \), where the origin of the image is on the left upper corner. Figure 3 shows examples of the row and column AADP (a) and the mouth search region (b).

![Image of PDAAs and Search region of the mouth.](image)

3.2. Parameterization

Once the images are segmented they are parameterized; the objective of this process is to reduce the signal space dimension and so, to reduce the computational complexity of the training problem. The images are transformed to a lower dimensional space using a decimation by 4 in the two dimensions. This is accomplished by applying a low pass filter to the image and then sub-sampling the original image to obtain a 16x32 new image. The DCT (Discrete Cosine Transform) of the sampled image is calculated and the first 8x16 coefficients are taken in order to obtain a 128 dimensional vector.

3.3. Vector Quantizer Construction

Synphony uses vector quantization to encode images as others systems do [11]. The main difference between our system and others is that classical methods choose the centers based on minimizing the average distortion while Synphony makes the choice based on the reconstruction of credible sequences. Credible sequences are those with smooth transitions, specifically in the mouth region, and coherent with the speech.

In order to achieve smooth sequences, the optimization condition is to minimize the maximum error within the class samples and their representative image. This is done using a modified version of the Lloyd-Max algorithm, changing the definition of the representative sample. The modified algorithm is explained below.

Let \( X = \{ \mathbf{x}_i \}_{i=1}^{N} \) be the training set, where \( \mathbf{x}_i \in \mathbb{R}^M \) (representation of the \( n \)-th image in the \( M \) dimensional parametric space) and \( Y = \{ \mathbf{y}_k \}_{k=1}^{K} \) be the codebook set with \( K \) classes. Then the conditions of the quantizer are the following:

1. The center, \( \mathbf{y}_k \), is defined as:

\[
\mathbf{y}_k = \arg \min_{\mathbf{x} \in \mathcal{Y}_k} \left\{ \max_{\mathbf{x} \in \mathcal{Y}_k} \| d_{\mathcal{Y}_k}(\mathbf{x},\mathbf{x}_i) \| \right\}
\]

with

\[
d_{\mathcal{Y}_k}(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{M} |x(i) - y(i)|.
\]

where \( \mathcal{Y}_k = \{ \mathbf{x}_j \}_{j=1}^{N_k} \) is the set of \( N_k \) samples in the training set assigned to the \( k \)-th class.

2. The \( i \)-th class defined as:

\[
\mathcal{Y}_i = \left\{ \mathbf{x}_n \in X \left| d_{\mathcal{Y}_k}(\mathbf{x}_n,\mathbf{y}_k) = \min_{k=1,\ldots,K} \left\{ d_{\mathcal{Y}_k}(\mathbf{x}_n,\mathbf{y}_k) \right\} \right. \right\}
\]

As a consequence of the new choice, the images in the codebook tend to cover all the image space and non typical but possible images are included in it. The codebook size used in the system is 256.
4. Online execution

Once the vector quantizer is trained, the system is ready to start the online execution where the user should remain in the same controlled conditions. The block diagram of the online execution can be seen in Figure 4.

![Block diagram of online execution](image)

*Figure 4 Online Execution*

4.1. Transmitter

The first process in the transmitter is the acquisition which is done just like in the training phase. Then each image is segmented using the region determined in the training phase and it is represented using the same parameterization. Once the image is parameterized, it is classified using the vector quantizer. The index of the image is packed with the compressed audio (8Kbps). The audio is compressed in segments of 30ms using DSP Group TrueSpeech (TM) Audio Codec from the Windows® ACM. Finally the packets are sent through the LAN using RTP (Real Time Protocol Library).

If the video acquisition rate is lower than 30fps, the system uses an interpolation method to simulate 30fps, so that the video rate is always 30bytes/s. A basic difference between the proposed method and others [8] is that the intermediate images are searched in the codebook instead of generating new images. The main idea is to find images that maintain credible sequence between a pair of images. Formally the proposed interpolation algorithm is the following:

Let \( I = \{i_n\}_{n=1,...,M} \) be the codebook, \( i_n, i_m \in I \) the pair of targeted images, \( S^k \) the \( k \)-th possible sequence of \( N \) images \((N<M)\) between \( i_n \) and \( i_m \) and \( D(S^k) \), the weight of the \( k \)-th sequence

\[
D(S^k) = \left[\sum_{l=1}^{N} d(S^k_l, s^k_{l-1})\right]^{2} + \left[\sum_{l=1}^{N} d(S^k_l, t_{l-1})\right]^{2} \tag{6}
\]

where \( s^k_l \) is the \( l \)-th image in \( S^k \). We select the smoothest sequence as the sequence with the minimum weight.

\[
s_{opt} = \arg \min_{s^k_l, l=1,...,N} \left\{ D(S^k) \right\} \tag{7}
\]

The definition of the sequence distance assures that the smoothest sequence will be obtained. The problem of finding the minimum weight sequence using direct calculation has \(M^{N-1} \) candidates and it is time consuming. To perform this search we use the Viterbi algorithm which is a computationally efficient method.

The final step corresponds to the packet assembling where each segment of 30ms of compressed audio is attached to the corresponding video index. The packet size is 88 bytes where 34 correspond to the audio and video before the overhead for the RTP transmission.

\[
\text{Avg. Bit Rate} = \text{Packet Size} \div \text{Packet Arrival Time} = 88 \text{ bytes} \div 32 \text{ ms} = 22 \text{ kbps} \tag{8}
\]

4.2. Receiver

In the receiver the packets are disassembled. The audio information of the packet is reproduced using DirectSound®, the index of the packet is used to look up an image in the codebook and then the image is presented using Windows® APIs. The synchronization made in the transmitter guarantees the coherence between audio and video in the presentation.

5. Results

Two kinds of evaluation were made in order to quantify the results of the system: subjective and technical evaluations.

5.1. Subjective Evaluation

A group of 56 people, half male and half female, were asked about the characteristics from different reconstructed videos. The test was designed following the ITU-T P.910 and P.911 recommendations to evaluate the audio-visual quality for multimedia applications. According to the recommendation each question was repeated twice in inverted order. The test videos show four different people. In the following paragraph some of the most relevant results will be explained.

Table 1 shows the opinions about the naturalness of the coded video: DV and Webcam acquisition at 30fps and 15fps in the first and second presentation of the question (1°T. - 2°T.). The qualitative results shows that the interpolated video (webcam) has an acceptance score higher than the direct coded video (DV) and both of them have good results.

<table>
<thead>
<tr>
<th></th>
<th>Excellent</th>
<th>Good</th>
<th>Acceptable</th>
<th>Bad</th>
<th>Very Bad</th>
</tr>
</thead>
<tbody>
<tr>
<td>DV 1°T</td>
<td>0.00%</td>
<td>25.00%</td>
<td>62.50%</td>
<td>10.71%</td>
<td>1.79%</td>
</tr>
<tr>
<td>Webcam 1°T</td>
<td>3.57%</td>
<td>50.00%</td>
<td>42.86%</td>
<td>3.57%</td>
<td>0.00%</td>
</tr>
<tr>
<td>DV 2°T</td>
<td>3.57%</td>
<td>50.00%</td>
<td>46.43%</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>Webcam 2°T</td>
<td>5.36%</td>
<td>55.36%</td>
<td>39.29%</td>
<td>0.00%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

Table 2 shows the opinion about the jerkiness of the video. These results are important because the coder design paradigm is the creation of smooth sequences. Although the jumps are perceived, they are not annoying. It also can be inferred that people who use the system tend to become used to the jerkiness of the video.

<table>
<thead>
<tr>
<th></th>
<th>Imperceivable</th>
<th>Perceivable but not annoying</th>
<th>Slightly annoying</th>
<th>Annoying</th>
<th>Very annoying</th>
</tr>
</thead>
<tbody>
<tr>
<td>1°T</td>
<td>0.00%</td>
<td>32.14%</td>
<td>60.71%</td>
<td>7.14%</td>
<td>0.00%</td>
</tr>
<tr>
<td>2°T</td>
<td>0.00%</td>
<td>46.43%</td>
<td>37.50%</td>
<td>16.07%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>
Another important result is shown in Table 3. In this question the people were asked to select the most natural video between a direct coded video and an interpolated video, both of them trained with the same sequence. In this case people prefer the interpolated video. This shows that the interpolation algorithm generates smooth and natural sequences.

Table 3: Preference between direct and interpolated coding.

<table>
<thead>
<tr>
<th></th>
<th>Video 1*</th>
<th>Video 2 §</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st time</td>
<td>33.93%</td>
<td>66.07%</td>
</tr>
<tr>
<td>2nd time</td>
<td>23.79%</td>
<td>73.21%</td>
</tr>
</tbody>
</table>

* Video 1. Trained 30fps video acquired at 30fps
§ Video 2. Trained 30fps video acquired at 15fps

5.2. Technical Evaluation

The PSNR of the reconstructed videos were calculated for direct coding, interpolated 15fps acquired video and interpolated 7.5fps acquired video. The results are shown in Table 4.

Table 4: PSNR of re-synthesized videos.

<table>
<thead>
<tr>
<th></th>
<th>30fps ac.</th>
<th>15fps ac.</th>
<th>7.5 ac.</th>
</tr>
</thead>
<tbody>
<tr>
<td>DV</td>
<td>Webcam</td>
<td>Webcam</td>
<td></td>
</tr>
<tr>
<td></td>
<td>37.731dB</td>
<td>33.903dB</td>
<td>34.156dB</td>
</tr>
</tbody>
</table>

To have a quantitative measure of the smoothness of the reconstructed sequence, an experiment with control points was run. In Figure 5 can be seen the four control points placed in the user face (a) and the change of position in the vertical coordinate of the lower mouth point in the natural video (b) and the re-synthesized one (c). It could be seen that the reconstructed sequence movement is very similar to the natural one.

Figure 5 Experiment to measure the smoothness.

6. Conclusions

A system capable of transmit an intelligible and acceptable quality video over a very low capacity channel was developed. The coding method proposed allows the reconstruction of smooth and natural sequences coherent with the corresponding audio. The interpolation algorithm is computational efficient and yields smooth and natural sequences.

The parameterization used achieves good results when the motion condition is controlled because the segmentation algorithm works properly and all the codebook images are with the same head position. Under uncontrolled motion conditions the reconstructed sequence is not natural.

It is very important to maintain lightning, motion, background and eye blink conditions under control in the training phase because any change in the images apart from the mouth region cause lack of naturalness in the re-synthesized video. A change in the training sequence could cause non uniform images in the codebook and therefore unnatural sequences (e.g. uncontrolled eye blinking or head movement). The technical results show that the longer the training video the less natural the re-synthesis looked because it is more difficult to keep controlled conditions.

The visemical richness of the phrases uttered in the training phase is important to assure the quality of the final video. According to the results the best reconstructed video will be obtained with a short phrase which contains all the visemes in the language uttered without over-articulation.

7. Acknowledgement

This work has been partially funded by COLCIENCIAS under contract RC. No. 35-2003.

8. References