Available online at www.sciencedirect.com
ScienceDirect
Signal Processing 86 (2006) 3644–3656
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

Real-time language independent lip synchronization method using a genetic algorithm

Goranka Zorić*, Igor S. Pandžić

Department of Telecommunications, Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, HR-10000 Zagreb, Croatia

Received 1 July 2005; received in revised form 5 December 2005; accepted 1 February 2006
Available online 24 May 2006

Abstract

Lip synchronization is a method for the determination of the mouth and tongue motion during a speech. It is widely used in multimedia productions, and real time implementation is opening application possibilities in multimodal interfaces. We present an implementation of real time, language independent lip synchronization based on the classification of the speech signal, represented by MFCC vectors, into visemes using neural networks (NNs). Our implementation improves real time lip synchronization by using a genetic algorithm for obtaining a near optimal NN topology. The automatic NN configuration with genetic algorithms eliminates the need for tedious manual NN design by trial and error and considerably improves the viseme classification results. Moreover, by the direct usage of visemes as the basic unit of the classification, computation overhead is reduced, since only visemes are used for the animation of the face. The results are obtained in comprehensive validation of the system using three different evaluation methods, two objective and one subjective. The obtained results indicate very good lip synchronization quality in real time conditions and for different languages, making the method suitable for a wide range of applications.

Keywords: Lip synchronization; Lip sync; Facial animation; MPEG-4 FBA; Human–computer interaction; Virtual characters; Speech processing; Neural networks; Genetic algorithms

1. Introduction

A human speech is bimodal in its nature [1]. A speech that is perceived by a person depends not only on the acoustic information, but also on the visual information such as lip movements or facial expressions. In noisy environments, a visual component of a speech can compensate for a possible loss in speech signal. This combination of the auditory and visual speech recognition is more accurate than only auditory or only visual. Use of multiple sources generally enhances a speech perception and understanding. Consequently, there has been a large amount of research on incorporating bimodality of a speech into the human–computer interaction interfaces. Lip synchronization is one of the research topics in this area.

The goal is to animate the face of a speaking avatar (i.e. a synthetic 3D human face) in such a way that it realistically pronounces the given text, which is based only on the speech input. Especially

*Corresponding author. Tel.: +385 1 6129 801; fax: +385 1 6129 832.
E-mail addresses: Goranka.Zoric@fer.hr (G. Zorić), Igor.Pandzic@fer.hr (I.S. Pandžić).

0165-1684/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2006.02.038

important component of facial animation is the movement of lips and the tongue during speech. For a realistic result, lip movements must be perfectly synchronized with the audio. However, in the real time use, some time delay must be accepted, since a speech has to be spoken before it can be classified.

In this section the problem of the lip synchronization is introduced. Next section gives the background information and related work. Section 3 explains the proposed lip synchronization algorithm. Implementation of our system is briefly described in Section 4, while achieved results and description of the system behaviour in different conditions is presented in Section 5. The paper closes with the conclusion and the discussion of the future work.

2. Background

Lip synchronization is the determination of the motion of the mouth and tongue during a speech [2]. Intonation characteristics, a pitch, an amplitude and voiced/whispered quality, are dependent on the sound source, while the vocal tract determines the phoneme. A phoneme is the basic unit of the acoustic speech. A visual representation of the phoneme is called viseme. There are many sounds that are visually ambiguous when pronounced. Therefore, there is a many-to-one mapping between phonemes and visemes. To make lip sync possible, position of the mouth and tongue must be related to characteristics of the speech signal. Positions of the mouth and tongue are functions of the phoneme and are independent of intonation characteristics of a speech.

The basic idea of lip synchronization is shown in Fig. 1. The process of the automatic lip sync consists of two main parts. The first one, audio to visual mapping, or more specific speech to lip shape mapping, is the key issue in the bimodal speech processing. In this first phase a speech is analysed and classified into viseme categories. In the second part, calculated visemes are then used for the animation of virtual character's face. The animation is not the topic of the interest in this work as it is already implemented in the Visage Technologies [3] software on which our application is based, so it is only briefly described in this paper.

The problem of converting a speech signal to the lip shape information can be solved on several different levels, depending on the speech analysis that is being used [4]. These levels are:

- Front end (signal level)
- Acoustic model (phoneme level)
- Language model (word level)

Each of the three levels can be applied within the speech-driven face animation system. However, the choice will depend on a specific application, considering characteristics of the individual solution. In addition, a balance between time needed for the signal processing and the quality to be achieved must be found.

A signal level concentrates on a physical relationship between the shape of the vocal tract and the sound that is produced. The speech signal is segmented into frames. A mapping is then performed from acoustic to visual feature, frame by frame. This method uses a large set of audio-visual parameters to train the mapping. There are many algorithms that can be modified to perform such mapping—Vector Quantization (VQ), the Neural Networks (NN), the Gaussian Mixture Model (GMM), etc.

At the second level, speech is observed as a linguistic entity. The speech is first segmented into a sequence of phonemes. Mapping is then found for each phoneme in the speech signal using a lookup table, which contains one visual feature set for each phoneme. The standard set of visemes is specified in

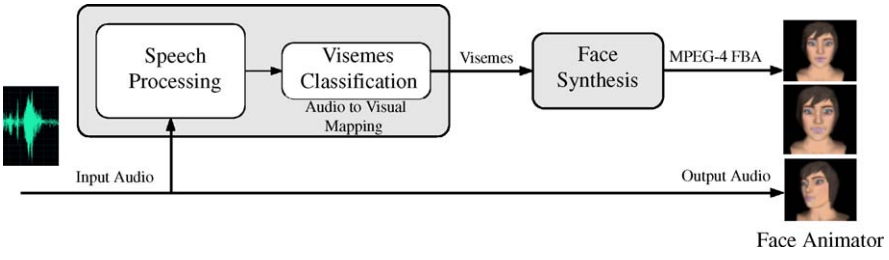

Fig. 1. The basic idea of lip sync.

MPEG-4 and contains 15 static visemes that can be easily distinguished [5].

The language model is more concerned about context cues in the speech signals. A speech recognizer must be first used for segmenting the speech into words. Then a Hidden Markov Model (HMM) can be created to represent the acoustic state transition in the word. In the next step, one of the methods used in the first level, can be applied for each state in this model to perform mapping from audio to visual parameters, frame by frame. Because the mapping is modelled inside individual words, better results can be achieved using this solution.

The latter two approaches are providing more precise speech analysis. Acoustic speech signal is explored together with the context, so that co-articulations (co-articulation is a process by which one sound effects production of the neighbouring sounds) are incorporated. However, higher input signal level requires a more complex system. At the same time, because the motion of the lips, tongue and mouth can be found from the speech signal without previous recognition of phonemes or spoken words, these methods produce a certain amount of computation overhead. Another problem with phoneme level approach is the definition of different phonemes in different languages, so that there is no standard phoneme set [6]. Additionally, speaker's gender, dialect or co-articulation could be an obstacle for obtaining a correct segmentation of the phonemes for individual's speech.

On the other hand, an approach based on the low level acoustic signals is simple, language independent and suitable for the real-time implementation, what is not the case in acoustic model where a speech engine have to be incorporated in the system in order to obtain a phoneme sequence for a given speech.

2.1. Related work

Various solutions have been proposed in the field of the automatic lip synchronization. A basic difference between them is the level on which the speech signal is analysed. There out arise some other characteristics of the system, such as applicability on real time systems, language independence, co-articulation, computation complexity/accuracy etc.

For example, systems that use Hidden Markov Model take into consideration the audio contextual information, which is very important for modelling mouth co-articulation during the speech. That is not the case with the vector quantization and the Gaussian mixture model. NNs can be trained for audio to visual mapping so that they take into account the audio contextual information (e.g. time-delay neural networks—TDNN [7]). TDNN is more computationally efficient than HMM, but requires a large number of hidden units, which results in high computational complexity during training phase. Many approaches use a combination of the different techniques.

Hong et al. [8] use NNs together with Motion Units (MUs). MUs are visual representation of the facial deformation and are obtained from the video. When audio-visual database is built, three layer perceptrons are trained to estimate MU parameters (MUPs) from audio. A similar technique is presented in [9] by the same authors. It uses Gaussian mixture model and multilayer NN to perform auditory-visual speech recognition in the real time. Massaro [10] trains three layer feed-forward ANN with a huge number of hidden units and as input takes parameters from the previous and the next frame, in addition to the current time frame.

Huang et al. [11] combine HMMs with sequence searching in order to animate lip movements from the speech in the real time. Acoustic feature vectors are calculated from the input voice and then the minimum distance between the vectors and the vocal data of the sequences in the base is found. If the distance is larger than a threshold, the face is synthesized by HMM-based method. Otherwise, the face in the corresponding sequence is exported. Huang et al. [4] implemented a real-time audio to visual mapping using Hidden Markov Model together with Gaussian mixture model. Brand [12] introduces method for speech-based full facial animation (lip sync, upper-face expressions) from a video. The video is analysed once, for training, and then facial HMM is used to construct the vocal HMM. Tamura et al. [13] propose a technique based on an algorithm for parameter generation from HMM with dynamic features for synthesizing synchronized lip movements from auditory input speech signal. Generated parameter sequence contains information of both static and dynamic features of several phonemes before and after the current one, so the synthetic lip motion becomes smooth and realistic, but it is not applicable on the real time systems.

Lewis [14] describes a lip sync approach based on a linear prediction. In this approach speech is effectively deconvolved into the sound source and

vocal tract filtering components. Kshirsagar et al. [15] train three-layer NN to classify coefficients derived with the linear predictive (LP) analysis into the vowels. Also, the average energy in the speech signal is used to modulate vowel–vowel and vowel–consonant lip-shape transition and zero crossing rate is used to detect fricatives. The same authors in [16] use Principal Components Analysis (PCA) of facial capture data extracted using a tracking system to form a vector space representation.

3. The proposed Lip Sync algorithm

The proposed system for automatic lip synchronization is suitable for real-time and offline applications. It is speaker independent and multilingual. Our system is in between signal and phoneme level, as we use visemes as the main classification target. Visual representation of phonemes, visemes, defined in MPEG-4 FA, is used for face synthesis. Database used for viseme classification is not audio-visual but auditory only.

Speech is first segmented into frames. For each frame most probable viseme is determined. Classification of speech into viseme classes is performed by NNs. Then MPEG-4 compliant facial animation is produced.

Next sections present the components of the system.

3.1. Viseme database

As a training data, a set of phonemes is collected. These phonemes are manually mapped onto MPEG-4 visemes, and in doing so the database is organized in 15 classes, each corresponding to one MPEG-4 viseme.

For each phoneme, nine test subjects (six male and three female) with different age and accent recorded three different words containing the specific phoneme as the first letter [17]. Next, phoneme is extracted from each of the three words corresponding to the specific phoneme. This gives 27 versions of each phoneme in the database. The words are recorded in a noise free environment and with top quality equipment with sampling frequency of 16 kHz with 16 bits accuracy and saved into wav files.

The database consists of 48 different phonemes (taken from Swedish and Croatian language) and the silence. Each viseme class is represented by certain number of samples in the database, depending on the number of phonemes it contains. Number of samples ranges from zero (viseme class 3 does not contain phonemes) to 216 (viseme class 6 contains eight phonemes), what makes average more than 80 samples per viseme group. In the viseme class 3 there are no phonemes at the moment, since neither one Swedish or Croatian phoneme is best described with this class.

For fine tuning of animation, phonemes specific for certain language might be added in the database.

Accuracy of the speech classification depends greatly on the quality and the size of the recorded database.

3.2. Audio to visual mapping

In order to perform audio to visual mapping, the speech is first segmented into the frames. Then, preprocessing and classification into the viseme classes is performed on every frame of the input speech.

3.2.1. Speech analysis

MFCC representation of the speech is chosen as first step in preprocessing the speech.

The Mel-Frequency Cepstrum Coefficients (MFCC) is audio feature extraction technique which extracts parameters from speech similar to ones that are used by humans for hearing speech, while, at the same time, deemphasizes all other information. As MFCCs take into consideration the characteristics of the human auditory system, they are commonly used in the automatic speech recognition systems [17].

Additionally, Fisher linear discriminant transformation (FLDT) is done on MFCC vectors to separate classes [18]. If there is no separation between classes before FLDT, transformation will not enhance separability, whereas if there is only slight distinction between classes, the FLDT will separate them satisfactory.

The overall procedure of the coefficients calculation is shown in Fig. 2.

In order to use MFCCs on the speech signal, frame length and the dimension of the MFCC vectors must be determined. The frame length must be chosen, so that the frame contains enough information. The choice is frame length of 256 samples and 12 dimensional MFCC vector. The coefficients in MFCC vectors are ranked according to significance and the information content diminishes towards the end of the vector.

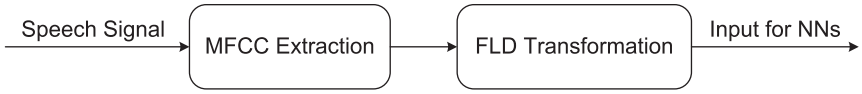

Fig. 2. An audio preprocessing used.

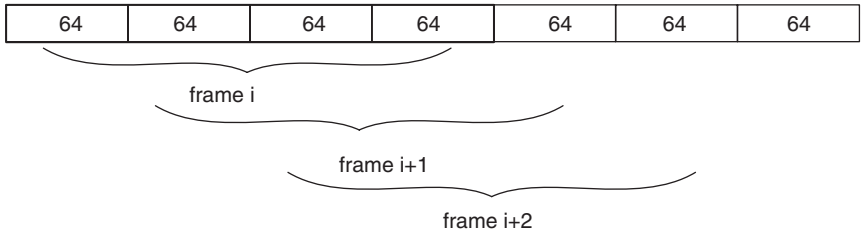

Fig. 3. 75% overlapping of the frames. The frame length is 256 samples [17].

In the following FLDT, no reduction of dimensions is made. Overlapping of the frames is used to smooth transition from frame to frame (Fig. 3), where 75% of the samples in the current frame are reused. The viseme database, 15 subsets of 12-dimensional MFCC vectors, is now used as a training set in order to train NNs.

3.2.2. Training NNs

In order to be able to separate inputs from each other, the network needs to be trained to learn from its environment. The training process is repeated a number of times. In every iteration, the vectors in the training set are processed through the network and the corresponding outputs are compared with the desired outputs, followed by adjustments of the weights.

In this approach, multilayer feedforward networks are used to map the speech to lip movements. The training algorithm for adjusting the weights is the Levenberg-Marquardt backpropagation algorithm. Two different kinds of activation functions are used. For the neurons in the first two layers, the *tansig* function is used and for the output layer, the *logsig* function is used as it produces outputs within the range of [0,1], which is desired here.

In our lip sync system, the visemes are used for the speech animation of the synthetic face model. Final goal of the speech analysis is to get visemes. Phonemes that are visual ambiguous, do not need to be separated, since it does not influence animation. We took advantage of this idea and decided to classify the speech directly into visemes.

The 12-dimensional MFCC vectors are used as inputs to 15 different networks. For each viseme class, a NN with 12 inputs, a number of hidden nodes and 1 output is trained. The network for each subset is expected to give 1 as output when the corresponding viseme is present at the input and 0 otherwise.

The number of hidden layers and the number of nodes per each layer should have been determined for each network. This is laborious and time consuming work since the training session must be run until the result is satisfactory. In order to avoid time consuming trial and error method, we introduced simple genetic algorithm to help find suitable topology for our NNs.

3.2.3. GA and NNs in our approach

Neural networks (NNs) are widely used for mapping between the acoustic speech and the appropriate visual speech movements. Many parameters, such as weights, topology, learning algorithm, training data, transfer function and others can be controlled in NN [19]. A major unanswered question in NN research is how best to set a series of configuration parameters so as to maximize the network's performance.

As training NN is an optimization process where the error function of a network is minimized, genetic algorithms can be used to search optimal combination of parameters.

Genetic algorithms (GA) are a method for solving optimization or search problems inspired by biological processes of inheritance, mutation, natural selection and genetic crossover. A conventional GA consists of coding of the optimization problem and set of the operators applied on the set of possible solutions [20]. The algorithm's three main operators

are selection, crossover and mutation. Only individuals that are good enough get the possibility to survive.

GAs might be used to help design NNs by determining [21]:

- *The weights*. Algorithms for setting the weights by learning from presented input/output examples with given fixed topology often get stuck in local minima. GAs avoid this by considering many points in the search space simultaneously.
- *Topology* (number of hidden layers, number of nodes in each layer and connectivity). Determining an optimal topology is even more difficult—most often, an appropriate structure is created by intuition and time consuming trial and error.
- A suitable learning rule.

However, they are generally not used in all three problems at the same time, since they are computationally very expensive.

The design of NN is optimized for a specific application, so we had to find suitable network for our lip sync system. As determining a good or optimal topology is even the most difficult task in design of NN, we tried to solve this problem with GAs.

For each viseme class, a NN with 12 inputs, a number of hidden nodes and 1 output is trained.

In our example, given the learning rule, we used GA for training a backpropagation feedforward network to determine near optimal network topology, including the number of hidden layers and the number of units within each layer.

We use simple genetic algorithm [22], where number of genes specify the number of hidden layers (n). Gene maximum and minimum values are defined in the range from zero to m, determining the number of nodes per layer. If a value of the single gene is set to zero, the number of hidden layers is decreased, so practically it ranges from zero to n. Other parameters that have to be specified are population size, maximum number of generation and mutation rate.

Experiments have shown that it is not necessary to have more than two hidden layers ($n = 2$). From the same reason, maximum number of nodes per layer is set to 30 ($m = 30$). Such configuration of GA seems suitable since larger network increases computation time, but does not give better results.

By using genetic algorithms, the process of designing NN is automated. Once the problem to be solved is coded and GA parameters are determined, the whole process is automated. Although it is still a time consuming work, much time is saved by making the process automatic.

3.2.4. Lip shape generator

For every frame of the speech that is classified in one of the viseme classes, the corresponding viseme need to be determined and sent to the animated face model.

Because of the network imperfection, output lies within the interval [0,1] and the network that produces the largest output is picked as the correct phoneme [17]. Errors may occur, and an incorrect phoneme can be identified as the correct one. Choosing a viseme in each frame, could cause a sudden discontinuous facial expression. To avoid this, the outputs from NN for four consecutive frames (1024 samples) are analysed. The viseme class with the largest output sum is chosen as a correct viseme. This results in some time delay from input to output.

3.3. MPEG-4 face animation

A face animation (FA) is supported in MPEG-4 standard. In this work, MPEG-4 standard is used for generating facial animation. Once the required information is extracted from the speech and the proper visemes are identified, any parameterised face model can be animated. More details on the MPEG-4 standard (i.e. MPEG-4 compatible 3D faces, MPEG-4 FA player) can be found in [5].

4. Implementation

Database construction and creation of 15 NNs have to be done only once. In the training process, network's biases and weights are extracted and saved for later use. Together with Fisher matrix (obtained by calculating FLDT), biases and weights matrix are loaded in the application.

Application captures speech from the microphone and segments it into frames of 256 samples. When a frame has been captured, data is stored and calculations are performed during capturing of the next frame. These calculations consist of MFCC extraction and simulation of 15 networks. The outputs are added to outputs from the previous frame. Every fourth frame, the viseme class that has the largest sum of output values from NNs is presented on the screen. It is important that

calculation time does not exceed time needed for recording of a frame.

In the offline mode, 75% overlapping between frames is introduced. Since computational time is unlimited, more complex calculations are performed. On the other hand, in the real time mode time delay between the sound and the animated lip movements plays significant role. The total time delay consists of the time needed to perform calculations and the time length of the frames taken into account for viseme determination. As we have 16 kHz sampled frames of 256 samples, the time needed for playing one frame is 16 ms. Consequently, a calculation time must be less than 16 ms. In the real time mode, when overlapping is not used, additional time delay is 64 ms, since we analyse 4 frames before deciding about correct viseme. That makes a total time delay less than 80 ms, what is short enough not to lose a real time impression.

5. System validation

A validation of the proposed lip synchronization system consists of two parts (Fig. 4).

The first one is performed in the process of generation NNs in order to obtain validation results of the classification with the NNs. In this part, two methods were used. The first one is based on the functions available in Matlab, while the second one compares results of viseme recognition obtained with our Lip Sync algorithm with the ground truth, as it will be described later.

Once the suitable NNs are found, the effects of various factors, such as background noise, language or personality, are tested on our lip sync system. In this second part we have used a subjective test, as it is based on the opinions of different persons on videos made from the animations generated in different conditions.

5.1. NN simulation

To perform some analysis of the network response, the generated NNs are simulated with the Matlab function *sim*. The function *sim* takes the network input, and the network object, and returns the network output [23]. The network object is the NN we want to analyse and the network input is the data we put through the network to get output. Since our database consists of recordings made by nine persons where seven of them are used in the training process, the rest of recordings are used as validation data.

Validations results are expressed by the percent the NNs recognize specific viseme class. Ideally, a NN trained to recognize viseme class 0, would recognize it with 100% probability and others with 0%. However, as it was not case for the most of the classes, NN parameters were fine tuned and training was repeated until satisfactory results were obtained. The number of training cycles needed to get good results varies from viseme class to viseme class. For some viseme classes (i.e. viseme class 0), several training cycles were enough, while for some (i.e. viseme class 5) even several hundred of them was not enough to obtain good results, no matter of the chosen network parameters. Results shown in the following tables are based on the evaluation of the four frames at the time and with overlapping used.

The final configuration of all NNs can be seen in Table 1 (all NNs are two layer networks).

Simulation of NNs gave results as Table 2 shows.

5.2. Lip Sync Test Application

Lip Sync Test Application is developed in order to evaluate generated NNs in more natural environment. In this method, the whole sentence is used

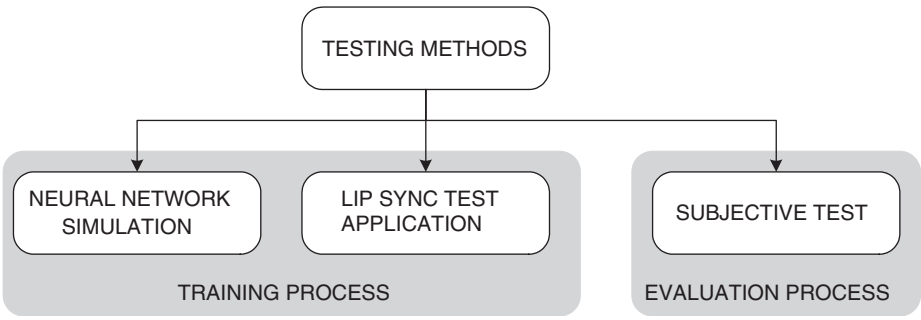

Fig. 4. Methods used for validation of the lip sync system.

Table 1
Characteristics and result of final neural networks

Viseme class	Number of hidden nodes	Number of training iterations	Result
0	18	32	90
1	28	32	63
2	5	26	57
3	—	—	—
4	7	39	69
5	14	29	37
6	24	25	86
7	3	27	86
8	12	33	55
9	9	24	68
10	16	25	94
11	22	27	81
12	17	26	64
13	10	27	77
14	10	33	46

Table 2
Validation results obtained by neural networks configured with genetic algorithms and with extended database used

Exp. Cl.	Recognized class														
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	90	1	3	—	1	1	2	0	0	2	0	0	0	0	0
1	2	63	6	—	8	3	15	0	2	2	0	0	0	0	0
2	6	5	57	—	4	10	6	1	1	4	0	0	4	1	1
3	—	—	—	—	—	—	—	—		—	—	—	—	—	—
4	5	6	8	—	69	5	0	4	0	0	0	0	0	3	0
5	1	9	7	—	21	37	6	1	4	11	0	1	2	0	1
6	0	1	1	—	1	5	86	0	0	4	1	0	1	0	0
7	0	0	4	—	7	2	1	86	0	0	0	0	0	0	0
8	0	4	6	—	4	2	0	0	55	2	2	0	9	0	17
9	0	1	3	—	1	6	0	0	5	68	1	6	3	0	5
10	0	0	0	—	0	0	0	0	0	0	94	2	0	5	0
11	0	0	0	—	1	3	0	0	0	3	0	81	6	0	6
12	0	0	6	—	11	2	0	0	2	0	0	6	64	0	9
13	0	0	3	—	2	0	0	0	2	0	8	1	1	77	7
14	0	2	2	—	1	1	1	0	3	12	1	13	20	0	46

as testing data and not just isolated phonemes as in NN simulation method. The idea is to compare the visemes calculated by the Lip Sync algorithm with the ground truth data. The comparison is made by the simple algorithm that had to be developed for that purpose. Fig. 5 shows a basic idea of the Lip Sync Test Application.

As a ground truth we use a single Croatian sentence recorded by the microphone in the silent room by the person which did not participate in the database creation. The sentence is than segmented into phonemes by manually marking the time-stamps for the beginning and the end of every phoneme. Each phoneme is represented with its associate viseme for later comparison. The sentence as shown in Fig. 5 is used (translated on English—*I am Reana, the first virtual person to talk Croatian*):

For the same sentence we use our Lip Sync algorithm to calculate visemes. The visemes are calculated for four consecutive frames meaning that one viseme contains 1024 samples.

Next some alignment must be performed since ground truth viseme and calculated one differ in the number of samples it encloses as well as the timestamps for the beginning and the end. Therefore, we have developed algorithm that normalizes ground truth visemes on calculated ones.

The output of the Lip Sync Test Application is the rate of the correct visemes and the total number of them in the ground truth sentence calculated by the Lip Sync module. Total number of visemes in our ground truth sentence is 71. In the case when the online mode was used 26 visemes out of 64 were recognized correctly, while in the offline mode 29 of them (seven of them were marked as questionable).

Beside overall correctness of recognized phonemes, correctness per viseme class is measured. Fig. 6 shows results obtained in the offline and online mode in the form of percentage of correctly recognized visemes, while Fig. 7 shows results as number of correctly recognized visemes compared with the total number of visemes that appeared in certain viseme class.

Ground truth sentence did not contain visemes from all viseme groups. Those groups are of viseme class three, six and eight (on the graph displayed with zero—actually not displayed). The third column in Fig. 7 (labelled *total*), denote the total number of visemes, for each viseme group that appeared in the ground truth sentence. Every viseme group is represented with different number of samples. The choice was accidental.

5.3. Subjective testing using video generated from animation

Numerical results provide a very good base, for the validation of the system. A visual impression still remains important indicator of our lip synchronization system quality, since the observers are ones to whom our virtual characters talk. Therefore, a subjective test is conducted. The goal was to test our system in different conditions, with emphasis on different languages.

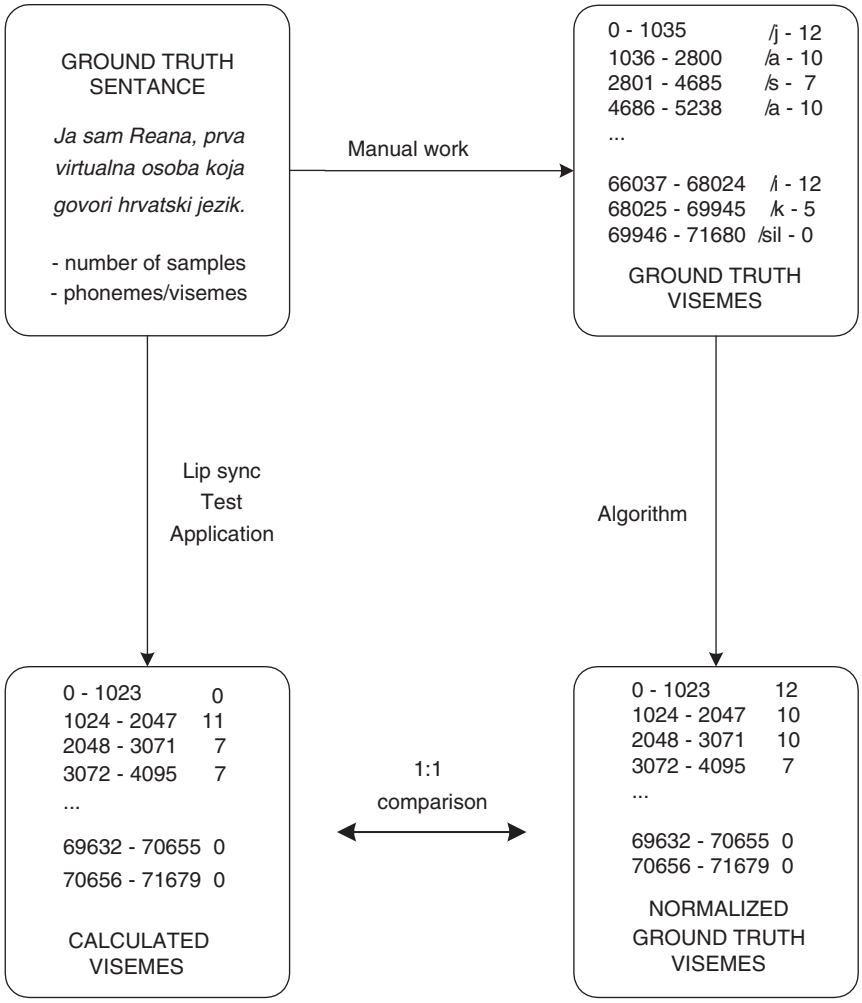

Fig. 5. A basic idea of the Lip Sync Test Application.

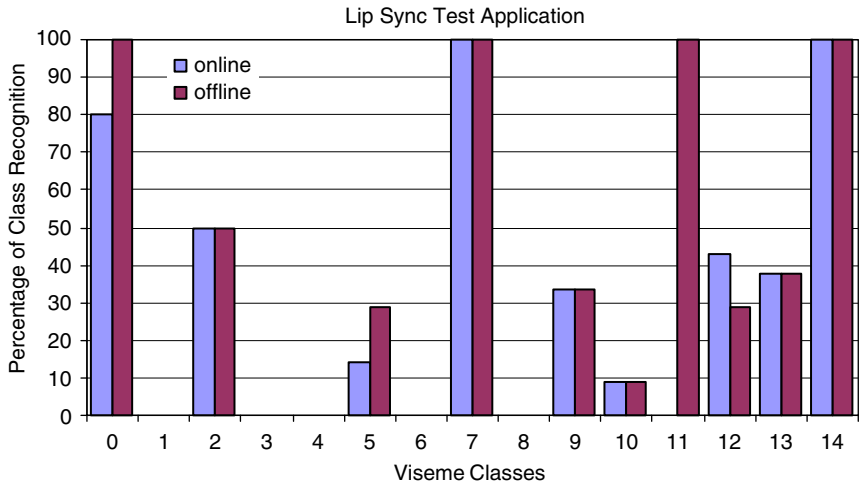

Fig. 6. Percentage of correctly recognized visemes in the online and offline mode obtained by Lip Sync Test Application.

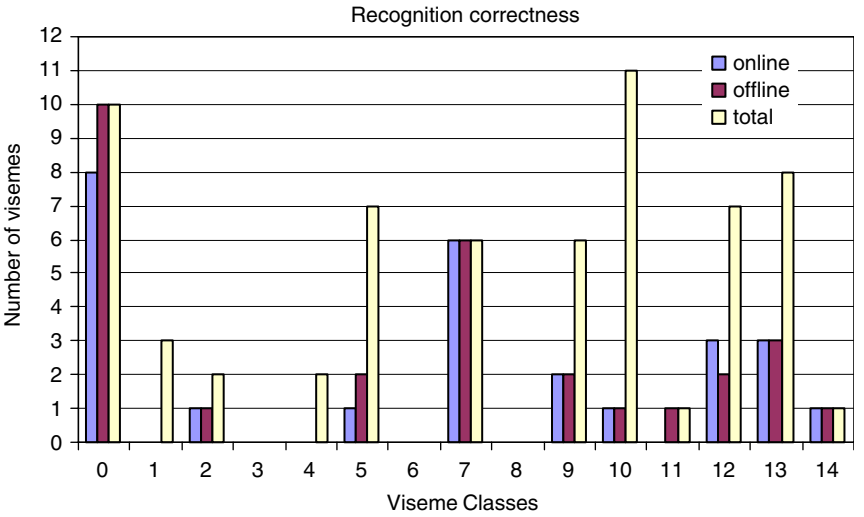

Fig. 7. Number of correctly recognized visemes in the online and offline mode compared with total number of visemes that appear in the ground truth sentence.

We have synthesized a facial animation of the face model using our Lip Sync algorithm. The input is audio file and the output animation file. Both files are than used to create videos with 3ds max [24] software. Several videos are generated using different audio files, presented to 25 testing subjects. Subjects were generally not connected with the topic.

Audio files have been created in the way to differ in characteristics, that can influence speech perception and recognition, such as background noise, language etc.

For every video, subjects are asked to answer previously prepared questions (Table 3). In questions 1 and 2, the score was graded on the scale from 5 to 1. The higher score correspond to more positive answers. Question 3 is about general impression of all videos shown. We created eight videos from eight different audio and animation files, whose characteristics are shown in the Table 4.

Fig. 8 summarizes the average score for the first two questions.

Question 3 contains the general impression of all seen videos. Some interesting remarks that come out from answers on the Q3 can be seen in Table 5.

Table 3
Questions in the Questionnaire for subjective test

Q1—To what extent do the generated lip movements follow the speech?
Q2—Did you notice any disturbing or discontinuous lip movements during the playback of the video?
Q3—In short, describe your general impression of the videos you have just seen (with particular focus on achieved naturalness, quality and accuracy of animation, etc.).

Table 4
Characteristics of created videos

	Language	Mode	Conditions
Croatian on	Croatian	Offline	Silent room
Croatian off	Croatian	Online	Silent room
English	English	Offline	Silent room
German	German	Online	Silent room
Swedish	Swedish	Offline	Silent room
Mobile	Croatian	Offline	Mobile phone
Noisy	Croatian	Online	Noisy room
Lab	Croatian	Offline	Professional studio

5.4. Discussion

In this chapter, we have presented detailed validation of our Lip Sync system. What follows is concise comment on them.

Results obtained with NN simulation method for the final NNs are far away from the ideal one. The percentage range of recognition varies from 37 (viseme class 5) to 94 (viseme class 10), with average of 70%. It is interesting to compare the results of NN simulation and Lip Sync Test method. For

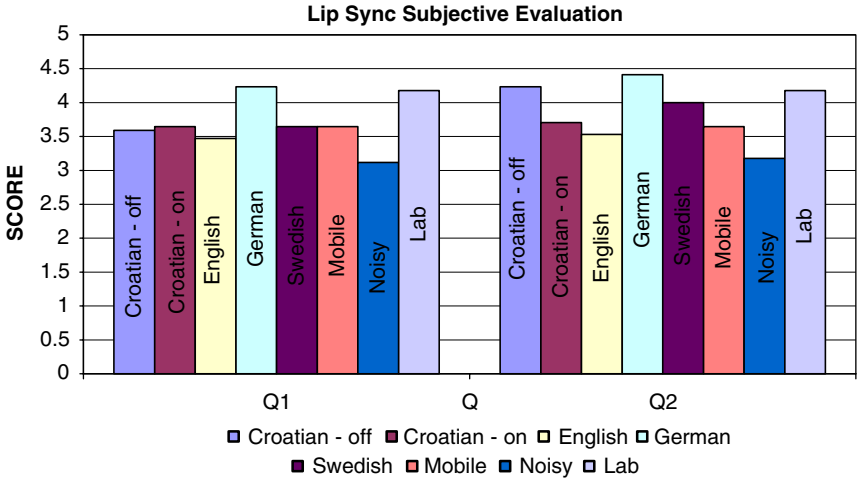

Fig. 8. Scores achieved for the first and the second question in the subjective testing.

Table 5
General impression of seen videos of several tested subjects

1	The lips sometime move although they are not supposed to move, probably because of presence of noise in the audio files (like convulsion).
2	If there is only short break between words (or before new sentence), lips do not stop moving.
3	It is noticeable that some phonemes are not correctly animated either because they are not correctly recognized or because the form of the animated lips does not seem to perfectly follow the form of a real speaker.
4	Animation accurately follows the speech in the sense of timing.
5	Generally, if subjects didn't know the language spoken in the video, animation seemed more accurate and natural.
6	On the face only lips were animated. Adding some other movements (eyebrows, head or eye blinking) would improve general impression a lot, no matter of the lip synchronization accuracy.

some viseme classes the Lip Sync Test results are as expected considering the ones of NN simulation method, while for others they are completely opposing. For example, viseme class 0 (silence) is very good recognized no matter of the method—90% with NN simulation and 80% or 100% (10 total appearances) with Lip Sync Test Application in online and offline mode, respectively. On the other hand, viseme class 10 achieved very good results in NN simulation (94%) while in our testing application results came out to be poor—only 9.1% (1 out of 11 visemes) in both modes. This can be explained as consequence of not precise segmentation, i.e. the border between phonemes was not accurately determined, either because of the human mistake or because there is not a really clear border. Another problem of our Lip Sync Test Application is short ground truth sentence. The number of visemes in it is too small for any general conclusion. Moreover the viseme distribution is not uniform distributed over viseme groups (some viseme groups do not appear at all in the ground truth sentence).

The serious shortcoming of our system is low recognition of viseme class 1, containing phonemes /m, /p and /b (63% with NN simulation and zero—0/3 with Lip Sync Test). This class is important since the lips are closed while pronouncing those phonemes. Having wrong recognition and because opened lips, the synchronization is definitely disrupted. Even one of tested subjects recorded in the third answer for the subjective test group that some phonemes are not correctly recognized, specifying phoneme /m as example.

The results of subjective testing proved that visual impression is satisfactory, however. According to achieved results, subjects evaluated the generated animations generally positively. Average grade for the first question, related on the connectivity of created animation with the original speech, is 3.69 and for the second question, more concerned with sudden disturbing movements, 3.86 (with maximum grade 5). Three, of eight videos for the subjective test, are generated in the online mode and the rest in the offline. Animations generated in the real time were graded with scores, almost as high as for offline animation. Average grade for the online mode is 3.72, and for the offline mode 3.81. In the

real time animations, it is visible in some videos that speech precedes the animation.

As we expected, the video generated in the laboratory conditions with professional speaker and equipment achieved the best results. As well, results for the video in noisy environment were consistent with our expectations. Rather poor animation was created because different sounds that were present in the room were not recognized as noise. The system was trained with Swedish and Croatian phonemes. However, some tested subjects liked the video generated in German the best. There are several possible explanations why English made video did not achieve good results. Reason for this could be, that the characteristic of English language, that words are not pronounced clear and separate from each other, but with tendency of putting them all together in one big word (in contrary with German, which is more articulate). We were uncertain for results of creating animation with the audio file recorded on the mobile phone. Results, however, came out rather good. Lower sound quality caused by 8 bit recording available on the mobile phone, did not affect much the accuracy of animation.

Although the subjects were asked only to evaluate animation of the lips, many of them commented the general animation of the face, pointing that it was hard to concentrate only on lips if eyes or head stood still. So many comments in Q3 were generally about unnatural face, what points out our next goals.

6. Conclusion

In this article we have described our system for automatic lip synchronization of synthetic 3D avatars based only on a speech input.

In our approach for the lip synchronization system by speech signal analysis, the speech is classified into viseme classes by neural networks. The genetic algorithm is used for obtaining a near optimal neural network topology. By introducing segmentation of a speech directly into viseme classes instead of phoneme classes, computation overhead is reduced, since only visemes are used for the animation of the face, i.e. lips. The automatic design of neural networks with genetic algorithms saves much time in the training process.

According to the feedback that we have received from testing subjects, we can conclude that our lip sync system can be used in various applications since animations generated in different conditions are fairly convincing.

Since natural speech always involves some facial gestures, a face that only moves its lips looks extremely unnatural. Our next step will be to extend the automatic lip synchronization system with facial gestures and emotions. Therefore, following efforts will be focused on the extraction of face expressions in addition to lip movements from the speech signal. For the animation of the speech driven facial gesturing more information is needed. Driven with that fact, the speech prosody will also be taken into consideration.

Acknowledgement

The initial version of this lip sync system has been implemented by A.Axelsson and E. Björhall as part of their master thesis of Linköping University [17] and in collaboration with Visage Technologies AB, Linköping, Sweden. This work is also partly supported by Visage Technologies.

References

[1] T. Chen, R. Rao, Audio-visual integration in multimodal communication, Proceedings of IEEE, Special Issue on Multimedia Signal Processing (1998) 837–852.
[2] D.F. McAllister, R.D. Rodman, D.L. Bitzer, A.S. Freeman, Lip synchronization for animation, Proceedings of SIGGRAPH 97, Los Angeles, CA, 1997.
[3] Visage Technologies, www.visagetechnologies.com
[4] F.J. Huang, T. Chen, Real-time lip-synch face animation driven by human voice, Proceedings of IEEE Multimedia Signal Processing Workshop, Los Angeles, CA, 1998.
[5] S. Pandžić, R. Forchheimer (Eds.), MPEG-4 Facial Animation—The Standard, Implementation and Applications, Wiley, New York, 2002.
[6] Y. Li, F. Yu, Y. Xu, E. Chang, H. Shum, Speech-driven cartoon animation with emotions, Proceedings of the ninth ACM international conference on Multimedia, Ottawa, Canada, 2001.
[7] S. Curinga, F. Lavagetto, F. Vignoli, Lip movements sythesis using Time Delay, Proceedings of EUSIPCO-96, Trieste, 1996.
[8] P. Hong, Z. Wen, T.S. Huang, Real-time speech-driven face animation, in: S. Pandžić, R. Forchheimer (Eds.), MPEG-4 Facial Animation—The Standard, Implementation and Applications, Wiley, New York, 2002 (Chapter 7).
[9] P. Hong, Z. Wen, T.S. Huang, Real-time speech driven avatar with constant short time delay, Proceedings of International Conference on Augmented, Virtual Environments and 3D Imaging, Greece, 2001.
[10] D.W. Massaro, J. Beskow, M.M. Cohen, C.L. Fry, T. Rodriguez, Picture my voice: Audio to visual speech

synthesis using artificial neural networks, Proceedings of AVSP'99, Santa Cruz, CA, 1999.
[11] Y. Huang, X. Ding, B. Guo, H. Shum, Real-time face synthesis driven by voice, Proceedings of Computer-Aided Design and Computer Graphics, Kunming, PRC, 2001.
[12] M. Brand, Voice Puppetry, Proceedings of SIGGRAPH'99, 1999.
[13] M. Tamura, T. Masuko, T. Kobayashi, K. Tokuda, Visual speech synthesis based on parameter generation from HMM: Speech-driven and text-and-speech driven approaches, Proceedings of Auditory-Visual Speech Processing AVSP'98, 1998.
[14] Lewis, Automated Lip-Sync: background and techniques, Proc J. Visual. Comput. Anim. 2 (1991).
[15] S. Kshirsagar, N. Magnenat-Thalmann, Lip synchronization using linear predictive analysis, Proceedings of IEEE International Conference on Multimedia and Expo, New York, 2000.
[16] S. Kshirsagar, N. Magnenat-Thalmann, Viseme space for realistic speech animation, Proceedings of Auditory-Visual Speech Processing AVSP'01, 2001.
[17] Axelsson, E. Björhall, Real time speech driven face animation, Master Thesis at The Image Coding Group, Department of Electrical Engineering at Linköping University, Linköping, 2003.
[18] Fisher, Fisher linear discriminant and dataset transformation, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/FISHER/FLD/fld.html
[19] J.J. Dávila, Genetic optimization of neural networks for the task of natural language processing, dissertation, New York, 1999.
[20] R. Rojas, Neural networks, A Systematic Introduction, Springer, Berlin Heidelberg, 1996.
[21] J. Jones, Genetic algorithms and their applications to the design of neural networks, Neural Comput. Appl. 1 (1) (1993) 32–45.
[22] Black Box Genetic algorithm, http://fdtd.rice.edu/GA/
[23] Mathworks documentation for NNs, http://www.mathworks.com/access/helpdesk/help/toolbox/nnet/
[24] 3ds max, http://www4.discreet.com/3dsmax/