Text2Video: A SMS to MMS Conversion

Dipl.-Ing. Jürgen Rurainsky and Dr.-Ing. Peter Eisert
Fraunhofer Institute for Telecommunication Heinrich-Hertz Institute, Berlin, Germany

Abstract

With the fast growing high data rate networks, such as UMTS, new communication types are provided by mobile communication providers. Video communication as well as broadcast over DVB-H are only two examples of these new services. Video messages created from enhanced text messages also belongs to these new services. In this context, we have developed a complete system for the automatic creation of talking head video sequences from text messages like SMS. Our system converts the text into MPEG-4 Facial Animation Parameters and synthetic voice. A user selected 3D character will perform lip movements synchronized to the speech. The 3D models created from a single image vary from realistic people to cartoon characters. A voice selection for different languages and gender as well as a pitch shift component enables a personalization of the animation. The animation is shown using the 3GPP player of mobile devices. Therefore, our system can be used in mobile communication for the conversion of regular SMS messages to MMS animations.

1 Introduction

SMS messages are a widely used type of communication. These messages consist of simple text which might be additionally personalized by emoticons stressing the content and expressing the feelings of the author. Since there is a high correlation between the text, speech, and lip movements of a person, an artificial video can be synthesized purely from the text.

We have developed a scheme, which allows the user to communicate by means of video messages created from the transmitted text. A Text-To-Speech engine (TTS) converts the message into a speech signal for different languages and markup information, like phonemes, phoneme durations, as well as stress levels for each phoneme. These side information are used to estimate MPEG-4 Facial Animation Parameters that are applied onto the 3D head model. Rendering of the head model leads to a realistic facial animation synchronized with the speech signal [1, 2, 3].

A similar system extracts the MPEG-4 FAPs at the receiver [4]. Realistic voices for TTS engines require a large set of speech samples, which have to be stored locally. The usage of a TTS engine for devices like PDAs and cellular phones requires either more memory than regular provided or the acceptance of less quality of the synthetic voice. Human-Machine Interfaces with a TTS engine for face animation are developed as well [5, 6].

With the following description the conversion of a SMS message to a MMS animation will be explained. In case the rendering is implemented on the mobile device, our scheme can also be used for a new type of Human-Machine interfaces. Therefore, we included a section covering the real-time rendering on mobile devices.

2 SMS to MMS Conversion

The system presented in this paper allows the user to animate written text with a selected character using the framework of the Short Message Service (SMS). If a SMS is received it analyzes the written text for the major language. Together with the selected character, a voice is chosen to synthesize the speech signal and to render an animation with the 3D model. The Multimedia Messaging Service (MMS) is used from this point on to transmit the video message to the receiver. The internal video player on the receiver cellular phone displays the movie with the animated text message. An example of an animation displayed on a 3GP player is given in **Figure 1**.

Figure 1: SMS to MMS conversion with existing infrastructure. Field of view modifications give more freedom during conversion.

Figure 2: SMS to MMS conversion system. The mobile communication provider receives regular SMS messages with meta data and creates short audio-visual animations which are encoded considering provider specify file size limitations, wrapped into 3GP/3GPP container, and sent as MMS to the recipient.

In the following sections, we describe how our system requests input data as well as selections in order to individualize the video clips for this application. Further, we describe the facial animation parameter estimation and the rendering on the mobile phone.

3 System Description

The system for facial animation, that converts written text into an audio-visual animation allows the user to select a character (3D model), who will speak the text. Emoticons added to the text enable to control the 3D model besides the lip movements and to personalize the message for the receiver. Different voices with different languages and genders can be selected. Our pitch shift adjusts the synthetic voice to the character and increase the variability of the system. From the user input, MPEG-4 FAPs are created which efficiently describe the animation. Dependent on the application, the video can be rendered on the output device or a server creates a video which is streamed to the client. A system overview is given in **Figure 2**. It follows a brief description of the system before we explain each element of the system more in detail.

Written text is the major input provided by the user. In addition to text, a set of emoticons (e.g. smile and sad) can be inserted between words and allow to personalize a message. A set of selections characterize the second major input. The user has to select a character from a list of already available 3D models. A gender and language selection allows to increase the field of usage and completes the set of required input data of our system. Predefined associations of gender to 3D face/head model and automated language detection are implemented as well.

The data creation step converts the written text into animation data for a 3D head model. First, a Text-To-Speech (TTS) system is used to convert the text into audio and phoneme data of the specified language. In a second step, MPEG-4 animation parameters are estimated from the provided phoneme data from the TTS. For low bandwidth connections, these parameters are streamed and real-time rendering is performed on the client. However, the videos can be created at the server also, which requires additional audio-video interleaving and compression. Our system currently supports MPEG-4 and 3GP formats. The transmission channel and mobile communication provider regulations are used for the adaptive video and audio encoding. After creating the video message, these data are transmitted to the receiver of the animated text message. UMTS and GPRS transmissions to mobile phones are realized via a mobile communication network provider. The already integrated 3GP player for video MMS messages is used to display the enhanced SMS message.

3.1 Input Interface

The input interface requests the necessary information from the user/sender. The input data can be categorized into two types. First, arbitrary text input as well as the free usage of emoticons. Second, selections and side information like character, gender, and language.

The input text is specified by the format of the SMS message. The emoticons are represented by a character string. In **Table 1** a table is given, which shows the used mapping from either textual description or images of emoticons to the appropriate character strings. An example text could look like: "*Hello Mr. Smith, :-)) I am very happy about your message.*".

Table 1: Mapping from emoticons specified by ASCII strings to facial expressions and visemes of the 3D model.

ASCII string	textual description
:-)	smile
:-))	laugh
:-(	sad
:-((	very sad
:-|	neutral

Selections are the second type of input, which allow the user/sender to control and individualize the output. A set of selections is given in **Table 2**. The selections *character* and *voice* are mandatory for all cases of usage or target display interfaces.

Table 2: Available selections to be made by the user/sender.

option	example	textual description
character	woman1 woman2 man1 cartoon robot1	female persons movie actor former president non realistic figure robot
voice (male/female)	US UK French German	US English pronunciation British pronunciation
Sender Recipient	name, email, phone number	

Using the above given requirements, an information message can be created. Such text message encodes all needed data in order to fit the device and channel properties. An example is given in **Table 3**. A voice or language selection is not necessary, because the system evaluates text for the major language used for the written text of the message and sets the selection in combination with the gender information from the selected character. In **Figure 1** such enhanced SMS message is shown as well as the created MMS video message. The field *contact information* can be used in cases of eCard applications, where the video message will be send to the email account of the receiver.

Table 3: Information message used as input interface.

message	textual description
XXXXXXX	identification
smith@hotmail.com	contact information
+49 30 1234567	
cartoon	character, see **Table 2**
Hello Peter, ...	text message

3.2 Creation of Animated Message

Using the input data provided by the user/sender the data creation step converts the given information into the desired animation. The type of animation data depends on the display or render unit at the receiver. For 3GP movie players, the video data must be created, interleaved with the speech data, and finally encoded according to the target player properties. With the future capability of mobile device to render a 3D model in real-time, only animation parameters plus side information as well as speech data have to be transmitted.

3.2.1 Text-To-Speech Conversion

The initial part of the data creation step is the conversion of the given text into synthetic speech data as well as side information in the form of phonemes and phoneme duration. A phoneme is the smallest part of a spoken word and dependents on the language. Speech and lip movements are related in time. The duration of each phoneme can be measured either in time or samples of the speech data. Both types allow a precise synchronization between animation and audio data. The speech synthesis is realized by a Text-To-Speech engine. We support the AT&T Natural Voices engine, as well as BrightSpeech from Babel Technologies, and RealSpeak from ScanSoft [7, 8, 9].
All these TTS engines come with prepared voices. Some of the TTS engines allow the adjustment of the voices for special needs. We post-process the speech data with a pitch shift mechanism. This mechanism allows us to change the main frequency of the speech, so that the voice sounds deeper or brighter adding one more degree of freedom to the system. The result is a adjusted voice, which fits more the personality of the selected character.

3.2.2 MPEG-4 Parameter Extraction

In order to create an animation, the phonemes, which are extracted from the text (see 3.2.1), are converted into MPEG-4 Facial Animation Parameters (FAP). Each FAP describes a particular facial action related to a region in the face as, e.g., chin , mouth corner, or eyes. In the 3D model, deformations are specified for all these actions, which define deviations from the neutral state. In MPEG-4, there are 66 different low level FAPs, which can be superposed in order to create realistic facial animations.[10]
MPEG-4 describes the set of facial animation parameters and the range of usage. How the FAPs perform the desired facial action is not defined in the standard, but can be specified by transformation tables. Vertex movements in the 3D triangle mesh model associated to particular deformations can be taken from these tables for rendering. In order to estimate the FAPs for each frame of the video sequence from the phoneme data, the phonemes are first converted into 15 different visemes. For each viseme, measurements from real people are taken as input to estimate the corresponding set of FAPs that result in the same lip shape as given by the captured images. An analysis-by-synthesis technique is used that exploits knowledge from the 3D model as well as calibration data in order to optimize the parameter set.[11]
The lip shape associated to a particular viseme is stored as a vector of 14 different FAPs. Transitions between different phonemes are interpolated by particular blending functions that allow the creation of mesh deformations for each video frame. For additional realism, random eye blinking and head motion is added to the parameter set.

3.2.3 Rendering of the Video

The rendering of the animation requires a frame-based deformation of the selected 3D model. The FAPs computed for each frame are applied in order to deform the mesh. Since a 3D description of the scene is available, additional features can be applied to enhance the animation. Camera pans or zoom are only a few of the possible changes with this feature. If the animation is created on a PC based server, rendering is performed on the graphics hardware and the resulting images are saves as frames of a movie.

A selection of 3D models created from a single picture is given in **Figure 3**.

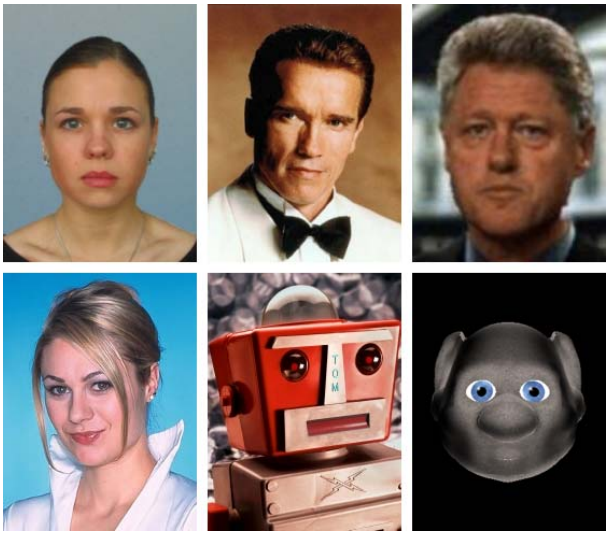

Figure 3: A set of 3D models created from single pictures.

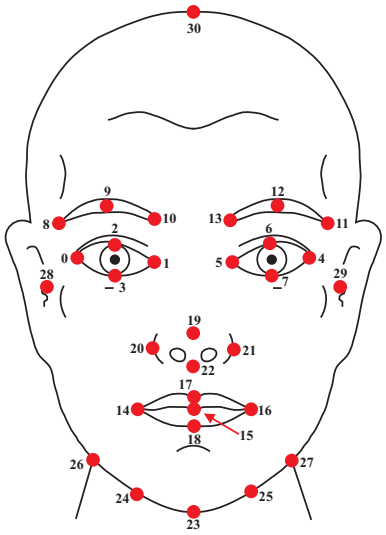

Figure 4: Feature points for 3D model adaptation.

In order to robustly deal with a wide variability of different input images, the actual geometry fitting is a semi-automatic process. The user has to select 31 feature points as shown in **Figure 4**. From these 2D positions, the 3D geometry is computed. Since the generic head model provides only a smooth contour, hair and ears are cut off for model adaptation. Third order Bezier curves are fitted through the 8 contour points using additional continuity constraints at the transitions as illustrated in **Figure 5**. Areas outside the curves are later represented with billboards while the inner area is used as approximation for the model silhouette.

Figure 5: Bezier curve fitting through contour points.

3.2.4 3D Model Creation

The 3D models used for rendering the video can be created from a single image. If no explicit 3D information is available the generic head model shown on the left side of **Figure 8** is adapted to the individual's picture. For that purpose, parameters for the shape modification are added to the model similar to the FAPs for facial expressions. In particular, 17 different morphs affecting shape, position, and dimensions of the head and facial features have been implemented. The shape adaptation to a new person is then equivalent to finding an optimal set of these 17 parameters. Once the 3D geometry is aligned, texture is mapped onto the model from the input image and extrapolated at the borders. Hair and other fine structures at the silhouette are not described by 3D polygons but by billboarding techniques.

The geometry fitting is then performed by finding an optimal set of shape parameters such that the Euclidian distance of the model features to the selected 2D feature points is minimized. The relation from the 3D geometry features to the 2D image plane is specified by perspective projection. A linear set of equation is set up using a-priori 3D scene information and the unknowns are efficiently determined in a least-squares sense. Inherent non-linearities are handled by an iterative framework that converges after two or three iterations.
With this optimization, a 3D head model is obtained that optimally fits to the specified 2D feature positions. However, it is not assured that the geometry lies completely

within the silhouette of the person. This might lead to annoying artifacts if background is mapped onto the model and moves with the person's head. Therefore, the silhouette information from the Bezier curve fitting is also incorporated into the optimization. Inequality constraints are added to the linear equations that make sure that all vertices strictly lie within the pre-defined area. On the other hand, additional equations are added describing the distance of the model from the contour. Thus, the contour of the generic model is attracted to the Bezier curves without crossing them at a single point. The linear system with additional non-linear constraints is efficiently solved using a LSI estimator. **Figure 6** shows an example of this model adaptation. The upper row is obtained by removing all pixels outside the Bezier curves shown in **Figure 5**. The geometry fitting is illustrated in the lower row by means of synthetic frames from the textured 3D model. Hair and body objects are now added with billboarding leading to natural output as shown in **Figure 3**.

Figure 6: Upper row: original frames with hair cut off; lower row: adapted 3D head model.

3.3 Video and Audio Encoding

In the case, the rendering is performed on server side, transmission of the video is necessary. In order to create a movie, the rendered frames are encoded, interleaved with the speech data, and finally packed to the target file container. Appropriate video and audio encoder as well as file container have to be used for this step. Two different types are implemented: 3GPP container with H.263 and MPEG-4 encoders for mobile devices like mobile phones and AVI containers with MPEG-4 encoders for standard personal computer. Audio data are encoded within the 3GP container using the AMR narrow band speech encoder and MP3 for the AVI container. For Multimedia Messaging Service (MMS) messages, special constraints on the maximum file size apply. Here, automatic rate control for video and audio are used during encoding in order to fulfill the requirements.

4 Human-Machine Interface

The previous description shows the conversion of text to a video MMS message. With the increasing hardware performance of mobile devices real-time rendering on a mobile device becomes possible and with this capability also a new type of Human-Machine interfaces.
Already known Human-Machine interfaces usually lack of personality, e.g., talking wizards, jumping paperclip. This can be circumvented by adding avatars, that help with problems and react on the user input with a voice and additional written information. The PC got a face and is not longer only a machine. Even a car can get a face for travel information, entertainment and road side assistant. Favorite persons are able to present the latest news transmitted to the car a text message and kids can be entertained by the latest Walt Disney character. Examples of such human-machine interfaces are given with the two middle sub-images of **Figure 7**.

Figure 7: Different Human-Machine interfaces. (left) real-time 3D render engine implemented on a mobile communicator , (middle) in car display units for information and entertainment.

In order to enable such performance the adaptation of the model as well as the 3D player have to be considered. We are going to show, with the following description, what and how such modifications can be realized in order to fit the hardware properties.

4.1 Rendering on Mobile Devices

In our system, we also support rendering on mobile devices. This requires 3D graphics operations like transformations and rasterizations of a textured mesh to be im-

plemented in software. The left sub image of **Figure 7** shows a realization on a common Personal Digital Assistant (PDA). This device does not have a graphic chip on board, but is able to render the animation at approximately 30 fps (frames per second). The speech data are played synchronously to the 3D animation. In order to achieve such frame rates for display devices with no graphic acceleration, modifications of the 3D model and associated texture map are required like polygon reduction, color mapping of the texture map, and 16 bit integer arithmetic. A polygon reduction of 50% as shown in **Figure 8** increases the render rate of about 30%.

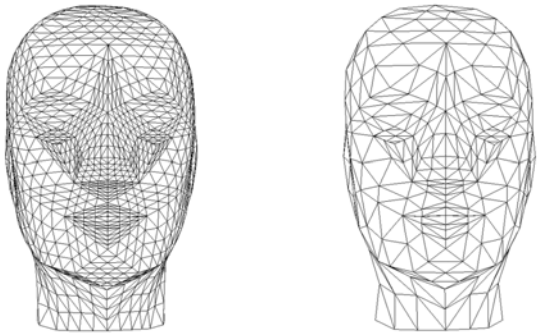

Figure 8: Polygon reduction for mobile communicator in order to increase display performance. (left) original polygon model with approximately 4000 polygons, (right) the reduced version with approximately 2200 polygons.

Another burst in frame rate is the conversion from floating point calculations to 16 bit integer arithmetic, since these low power mobile devices usually lack of an floating point unit and have to emulate the calculations in software. Therefore, the mesh representation and deformation calculations are done in 16 bit integer arithmetic resulting in a drastic speedup.

In order to fit texture colors with associated shading information into a 16 bit word, some color quantizations from the original 8x8x8 bit RGB texture are necessary. Since the display of the used PDA only supports a color resolution of 5x6x5 bits, only 65535 different colors can be shown. For performance reasons, the texels in the texture map only contain 3x3x3 bit of colors - the 7 remaining bits are used for internal processing. Since a color quantization with 8 steps per channel would lead to significant artifacts especially in the smoothly colored skin areas, a color map was introduced. A set of 512 colors is computed from the original texture map in order to get an optimal color quantization as shown in **Figure 9**.

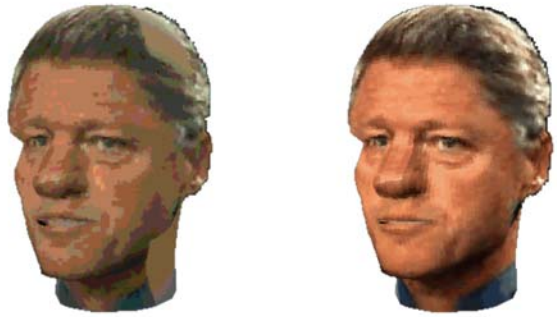

Figure 9: Color mapping for mobile devices. (left) texture map quantized to the target resolution of 3x3x3 bits, (right) optimized color mapping from 3x3x3 texture resolution to 5x6x5 display colors.

5 Conclusion

The need of communication is essential for humans. With the fast growing connectivity (cellular phones, broadband connections to homes, etc.) new types of communications can be realized. The presented text-driven video animation is one such example. The system is able to animate an avatar or real person from written text. The 3D models can be created from a single image or photograph. With emoticons, language, gender, pitch shift, and character selections, we are able to individualize the clips which can be used for a wide range of applications. Not only the conversion from regular SMS messages to MMS animations have been implemented. Human-Machine interfaces on mobile devices could also profit from this technology.

6 Acknowledgments

The work presented was developed within VISNET, a European Network of Excellence (http://www.visnet-noe.org), funded under the European Commission IST FP6 program.
The authors would like to thank Jens Güther for providing fruitful ideas, software as well as hardware knowledge for this work.

References

[1] Thierry Dutoit, "High-Quality Text-to-Speech Synthesis : an Overview," *Journal of Electrical & Electronics Engineering, Australia: Special Issue on Speech Recognition and Synthesis*, vol. 17, no. 1, pp. 25–37, 1997.

[2] S. Kshirsagar, M. Escher, G. Sannier, and N. Magnenat-Thalmann, "Multimodal Animation System Based on the MPEG-4 Standard," in

Proceedings of the International Conference on Multimedia Modeling (MMM 99), Ottawa, Canada, October 1991, pp. 215–232.

[3] J. Ostermann, M. Beutnagel, A. Fischer, and Y. Wang, "Integration of Talking Heads and Text-to-Speech Synthesizers for Visual TTS," in *Proceedings of the International Conference on Speech and Language Processing (ICSLP 98)*, Sydney, Australia, December 1998.

[4] S. Kshirsagar, C. Joslin, W. Lee, and N. Magnenat-Thalmann, "Personalized Face and Speech Communication over the Internet," *IEEE Signal Processing Magazine*, vol. 18, no. 3, pp. 17–25, May 2001.

[5] J. Ostermann, "E-Cogent: An Electronic Convincing aGENT," in *MPEG-4 Facial Animation - The Standard Implementation and Applications*, Igor S. Pandzic and Robert Forchheimer, Eds. Wiley, 2002.

[6] Simon Beard, John Stallo, and Don Reid, "Usable TTS for Internet Speech on Demand," in *Proceedings of the Talking Head Technology Workshop (OZCHI)*, Perth, Australia, November 2001.

[7] AT&T, *Natural Voices^{TM}; Text-To-Speech Engines (Release 1.4)*, 2002.

[8] Babel Technologies home page, *http://www.babeltech.com*.

[9] ScanSoft home page, *http://www.scansoft.com/realspeak/demo/*.

[10] ISO/IEC International Standard 14496-2, *Generic Coding of Audio-Visual Objects – Part2: Visual*, 1998.

[11] P. Eisert and B. Girod, "Analyzing facial expressions for virtual conferencing," *IEEE Computer Graphics & Applications*, vol. 18, no. 5, pp. 70–78, September 1998.