ERROR CONCEALMENT FOR FACIAL ANIMATION BASED ON PREDICTION OF MUSCLE DATA

Insu Park, Shahram Shirani and David W. Capson

Department of Electrical and Computer Engineering, McMaster University

ABSTRACT

An error concealment algorithm is proposed based on flow of facial expression to improve communication of animated facial data over a limited bandwidth and error prone channel. Facial expression flow is tracked using dominant muscles which are those with maximum change between two successive frames. The receiver uses linear interpolation and the information on facial expression flow to interpolate the erroneous facial animation data. Experimental results are provided to show that the proposed error concealment method improves the quality of an animated face communications.

1. INTRODUCTION

Communication of human faces is essential in many multimedia applications such as videoconferencing, distance learning and virtual reality. All of these applications demand rendering of a realistic human face for efficient and comfortable face-to-face communication. Distortion in rendered facial animation will occur at the receiver terminal if compressed data is lost or changed during transmission. Under this circumstance, error concealment is recommended to reduce the visual artifacts which caused by transmission errors. A forward error concealment method and an error concealment by post processing for synthetic facial data are proposed in this paper. In forward error concealment, in addition to muscle data, the transmitter sends side information of the facial expression flow. To determine the facial expression flow, the transmitter classifies the current facial expression based on the dominant muscle data which are those muscle data with maximum change between two successive frames. In error concealment by post processing, the decoder uses facial data in the frame before an erroneous frame to estimate the missing facial data.

2. BACKGROUND

2.1. Facial animation and facial expression

The three-dimensional face model consists of a number of non-overlapped polygons. As a matter of convenience in representing moving facial expression the facial data set (F) and facial object ($\mathcal{S}(F)$) in three dimension are expressed as

$$\begin{aligned} F &= \{\mathsf{v}_i, \psi_j | i = 1, 2, \cdots, N_V, j = 1, 2, \cdots N_P\} \\ \mathcal{S}(F) &= \textstyle\bigcup_{j=1}^{N_P} \psi_j \end{aligned} \tag{1}$$

where ψ_j is jth polygon of object F, and N_V and N_P represent the number of vertices and the number of polygons, respectively. Face muscle data is a useful tool for facial animation. Waters [1] developed a muscle-based facial model for realistic facial animation. The expression of the face changes by changing the position of vertices of the polygon forming the face. The vertices of each polygon is determined by a linear combination of muscles:

$$\mathsf{v}_i = \sum_{k=1}^{N_M} \alpha_{i,k} m_k \tag{2}$$

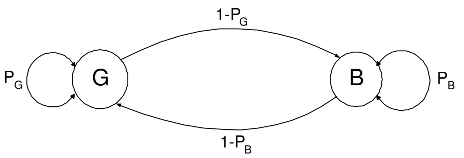
where v_i is the position of ith vertex, N_M is the number of muscle data, m_k is kth muscle data and $\alpha_{i,k}$ is the coefficient of the kth muscle data related to the ith vertex. By representing facial expression using several muscle data instead of using all of the geometrical and topological data, facial animation is rendered with a reduced data set. So it is possible to control facial expression with muscle data after transmitting all of the geometrical and topological data in the initial stage.

2.2. Error prone channels and transmission errors

In data transmission over error prone channel, errors are inevitable. The error rate of these channel is time varying. The underlying processes that lead to data losses over error prone channel is complex [2]. However Gilbert-Elliott model give us good approximation for loss behavior of error prone channels. Figure 1 shows Gilbert-Elliott channel model. In Figure 1 G and B denote the good and bad state of channel, respectively. Also the channel error probabilities in good (G) and bad (B) states are represented with $\mathsf{P_G}$ and $\mathsf{P_B}$, respectively. The Gilbert-Elliott channel model processes with an error-free interval distribution. The error-free interval is defined with the length of bit l_s (i.e, the number of bits between two consecutive errors) and the length

0-7803-9332-5/05/$20.00 ©2005 IEEE

of packet size l_p. More detail on Gilbert-Elliott channel can be found in [3].



(a)

Fig. 1. Gilbert-Elliott channel model.

2.3. Distortion measure tool

Metro is a distortion measurement tool that can quantify the difference between two 3-D objects $\mathcal{S}_1$ and $\mathcal{S}_2$. *Metro* which is based on the Hausdorff distance was introduced by Cignoni et al. [4]. Given a point set ($\Gamma(\mathcal{S})$) on the object $\mathcal{S}$ the distance from one point v to $\mathcal{S}$ is defined as:

$$\mathfrak{D}(\mathcal{S}, \mathsf{v}) = \min_{\mathsf{w} \in \Gamma(\mathcal{S})} \|\mathsf{v} - \mathsf{w}\|, \tag{3}$$

where $\|.\|$ denotes Euclidean distance. Then one-sided distance between two objects $\mathcal{S}_1, \mathcal{S}_2$ is defined as:

$$\mathfrak{D}(\mathcal{S}_1, \mathcal{S}_2) = \max_{\mathsf{v} \in \Gamma(\mathcal{S}_1)} \mathfrak{D}(\mathcal{S}_2, \mathsf{v}). \tag{4}$$

Then, the Hausdorff distance between two 3-D objects $\mathcal{S}_1$ and $\mathcal{S}_2$ is:

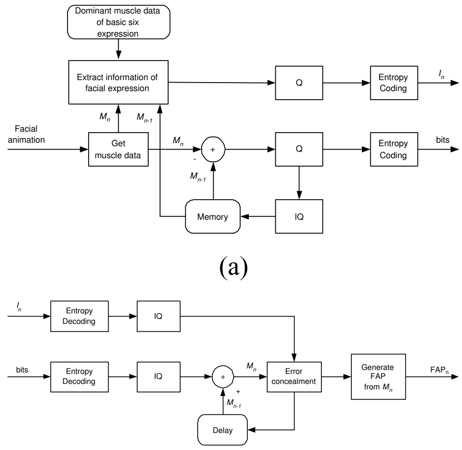
$$\mathfrak{E}(\mathcal{S}_1, \mathcal{S}_2) = \max\left(\mathfrak{D}(\mathcal{S}_1, \mathcal{S}_2), \mathfrak{D}(\mathcal{S}_2, \mathcal{S}_1)\right). \tag{5}$$

We use the Hausdorff distance $\mathfrak{E}(,)$ to define $PSNR$ in Section 4.

3. ERROR CONCEALMENT METHOD

If compressed muscle data is affected during transmission, the error concealment block is activated to reduce the visual effects of error. Our error concealment method employs both forward error concealment and post processing.

In forward error concealment, in addition to muscle data, the transmitter sends side information describing the facial expression. Figure 2 (a) shows the proposed encoding process for forward error concealment. In this figure M_n is the muscle data set of the nth frame (e.g., $M_n = \{m_k(n), k = 1, 2, \cdots, N_M\}$) and I_n is side information about facial expression flow. The side information (I_n) consists of two parts: (1) start expression (SE) and end expressions (EE) which can be any of seven expressions (natural, happiness, sadness, surprise, fear, anger and disgust) and (2) muscle data slope (β_n). To extract the side information the encoder tracks the facial expression flow using dominant muscle data as will be explained in Section 3.1.



(b)

Fig. 2. General error concealment processing: (a) encoding process, (b) decoding process.

3.1. Tracking of 3-D facial expression

Since we use seven expressions, there is $\begin{pmatrix} 7 \\ 2 \end{pmatrix} = 42$ possible different expression flows. The expression information table (EIT) consists of dominant muscle data for 42 different facial expressions changes. The EIT has index number of dominant muscle data. Determining dominant muscle data is as follows. The difference of muscle data between two expressions is calculated as:

$$d_k^{(a,b)} = m_k^a - m_k^b \tag{6}$$
$$\{a, b\} \in \{N, H, A, D, Sad, Sur, F\}$$

where m_k^a is kth muscle data of facial expression a, a (and b) is one of seven expression; N: natural, H:happiness, A: anger, D: disgust, Sad: sadness, Sur: surprise and F: fear. Using the difference muscle data, dominant muscle data of two facial expressions are formed as:

$$\mathcal{D}_{N_L}^{(a,b)} = \mathfrak{S}_L(d_1^{(a,b)}, d_2^{(a,b)}, ..., d_{N_M}^{(a,b)}) \tag{7}$$

where $\mathcal{D}_{N_L}^{(a,b)}$ is the dominant muscle data for a facial expression starting with expression a and ending with expression b. The $\mathfrak{S}_L(\cdot)$ is the operator that sorts the data in decreasing order and returns the indices to N_L largest (dominant increased) and N_L smallest (dominant decreased) values.

To initially classify the facial expression, the transmitter defines the dominant increased and decreased muscle data between two successive frames. That is:

$$\mathcal{D}_{N_L}(n) = \mathfrak{S}_L(d_1(n), d_2(n), ..., d_{N_M}(n)) \tag{8}$$
$$d_k(n) = m_k(n) - m_k(n-1)$$

where $\mathcal{D}_{N_L}(n)$ is the set of dominant muscle data for nth frame, $m_k(n)$ is the kth muscle data of nth frame, N_M is the number of muscle data. By comparing dominant muscle data $\mathcal{D}_{N_L}(n)$ with the dominant muscle data of EIT ($\mathcal{D}_{N_L}^{(a,b)}$) the transmitter determines the start and end expressions (SE and EE). Increasing the value of N_L increases the accuracy of our classification algorithm.

The second part of side information is the muscle data slope. After determining the start and end expressions the transmitter calculates muscle data slope using:

$$\beta_n = \frac{m_d(n) - m_d^{SE}}{m_d^{EE} - m_d^{SE}} \tag{9}$$

where m_d is the largest dominant muscle data. Then, SE, EE and β_n form the side information I_n employed in forward error concealment.

3.2. Error detection and concealment

If error occurs during data transmission, distortion of facial animation may start and propagate until the receiver gets a new full facial data set. To minimize this distortion, the receiver must constantly check the received muscle data set for the presence of error. Here we propose a mechanism for error detection. Two kinds of transmission error are considered: (1) data loss and (2) bit alteration. To detect bit errors caused by bit alteration automatically, we define the maximum (T_{max}) and minimum (T_{min}) of the acceptable range of change of muscle data. Whenever the difference between current muscle data and previous muscle data is larger than T_{max} or smaller than T_{min}, or there is no muscle data, the receiver treats the muscle data as erroneous.

Whenever the receiver detects an error in the data of a frame, it checks whether side information exists or not. If side information is available, the receiver extracts the start expression (SE), the end expression (EE) and the muscle data slope (β_n). In this case the erroneous muscle data is then interpolated as:

$$m_k(n) = \beta_n \times (m_k^{EE} - m_k^{SE}) + m_k^{SE} \tag{10}$$

where $m_k(n), m_k^{SE}, m_k^{EE}$ are the kth muscle data, the kth muscle data of the start expression, and the kth muscle data of the end expression, respectively and β_n is the muscle data slope of the nth frame.

If there is no side information, the erroneous muscle data is estimated using the following method. Since in the facial data compression considered here the transmitter sends a difference of muscle data, the kth muscle data of the nth frame is generated as:

$$m_k(n) = m_k(n-1) + d_k(n) \tag{11}$$

where $d_k(n)$ represent the kth difference muscle data of the nth frame. When the receiver encounters erroneous muscle data of the nth frame, the error is corrected by setting:

$$d_k(n) = d_k(n-1). \tag{12}$$

Since $d_k(n)$ is not necessarily the same as $d_k(n-1)$, this approach is not always effective. Nevertheless, it mitigates the propagation of distortion.

4. RESULTS AND DISCUSSIONS

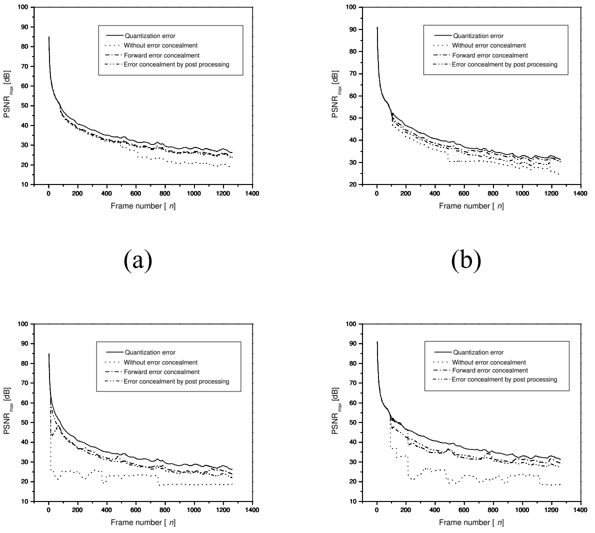
In our experiment, to verify the efficiency of the proposed error concealment algorithm, we consider three scenarios: (1) transmission of compressed facial muscle data over an error-free channel, thus, there exists just quantization error, (2) transmission of compressed facial muscle data over a channel in which bit alteration error occurs and decoding is performed without error concealment, and (3) transmission of compressed facial muscle data over the same channel as (2) but using error concealment when error occurs during transmission. We compare the resultant distortion of each scenario in terms of $\mathfrak{E}$. We made a test sequence having 1260 frames of facial animation from six basic facial expressions and natural expression. The range of the geometrical data of facial model are: (4.36, -4.36), (-8.74, 7.522) and (0.149, 9.855) in v_x, v_y and v_z, respectively.

In the encoder (Figure 2 (a)), we consider 256 and 1024 quantization levels (L_Q) after differentiation each muscle data. The average bit rate of source coder is 81.5 and 85.5 bit per frame with side information and without side information, respectively. In our simulation, bit error is introduced into the compressed muscle data using Gilbert-Elliott channel error model. For our channel model, the packet size l_p is 32, the error-free interval l_s is 30. To change the error rate, we use different probability of good state $\mathsf{P_G}$ and bad state $\mathsf{P_B}$; Error condition 1 ($EC1$): $\mathsf{P_G} = 0.995$ and $\mathsf{P_B} = 0.0001$, and Error condition 2 ($EC2$): $\mathsf{P_G} = 0.995$ and $\mathsf{P_B} = 0.0005$.

The distortion can be quantified by $PSNR$ which is defined as:

$$PSNR \quad = \quad -20 \log \frac{\mathfrak{E}(\mathcal{S}, \bar{\mathcal{S}})}{\mathfrak{L}(\bar{\mathcal{S}})} \tag{13}$$

where $\mathfrak{E}(,)$ is maximum error defined by Equation 5. $\mathfrak{L}(\bar{\mathcal{S}})$ denote diagonal length of bounding box enclosing facial animation $\bar{\mathcal{S}}$, which is 20.8291. Figure 3 shows the $PSNR$ of the facial animation when the quantization levels used in muscle data compression is 256 and 1024 with different channel condition. From Figures 3 (a) and (b) we can see that forward error concealment and error concealment by post processing both offer little improvement. The quality of rendered face after error concealment by post processing can not overcome the quality of rendered face when there is only quantization error. Due to less quantization errors



(a) (b)

(c) (d)

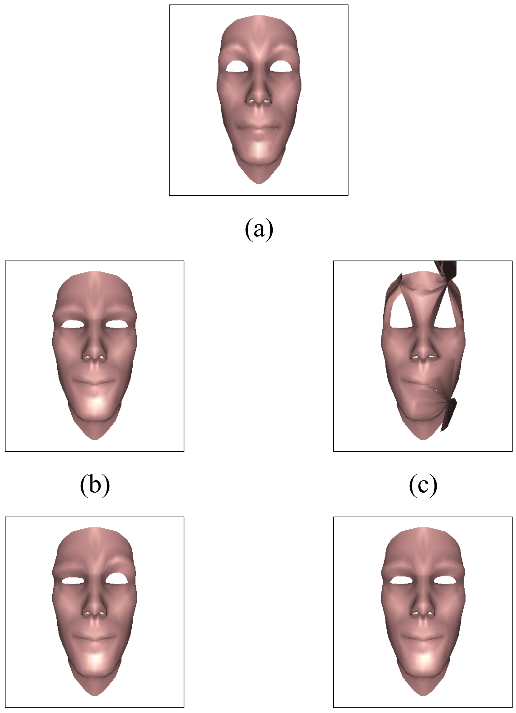
Fig. 3. Simulation result for $N_P = 876$ with different quantization levels and channel condition: (a) $L_Q = 256$, $EC1$, (b) $L_Q = 1024$, $EC1$, (c) $L_Q = 256$, $EC2$ and (d) $L_Q = 1024$, $EC2$.

the result of Figure 3 (b) is better than the result of Figure 3 (a). When transmitter sends side information (I_n) error concealment by post processing improves $PSNR$ around 10 dB. Figures 3 (c) and (d) show another results when there is large channel error during data transmission. In this case, both error concealment methods perform satisfactory and $PSNR$ improves around 15 and 20 dB by error concealment by post processing and forward error concealment, respectively.

In fact, when we study the final rendered 3-D facial animation, the effectiveness of the proposed error concealment algorithm is more clear. Figure 4 shows an example of the performance of our proposed method using 1024 quantization levels with error condition 2. Note the significant distortion of the eye and mouth in Figure 4 (c) is compensated in Figure 4 (e). Comparing Figure 4 (d) and 4 (e) with 4 (c) it is clear that error concealment has improved the quality of the rendered frame.

5. CONCLUSION

In this paper, we have proposed an error concealment algorithm for facial animation. Linear interpolation is employed to interpolate the lost data using the side information. The proposed error concealment method is simple and simulation results show that the proposed error concealment method is effective in preventing propagation of error while improving the quality of rendered facial sequences.



(a)

(b) (c)

(d) (e)

Fig. 4. Distortion of facial animation and the performance of forward error concealment and error concealment by post processing when $L_Q = 1024$ and $EC2$, the last frame with: (a) the original facial animation, (b) with quantization error, (c) with transmission error and without error concealment, (d) with transmission error after error concealment by post processing and (e) with transmission error after forward error concealment.

6. REFERENCES

[1] K. Waters, "A muscle model for animating three-dimensional facial expressions," *Comput. Graph.*, vol. 21, no. 4, pp. 17–24, 1987.

[2] G. Al Regib, Y. Altunbasak, and J. Rossignac, "Error-resilient transmission of 3d models," to appear in ACM Transactions on Graphics, Apr. 2005.

[3] M. Mordechai and B.-D. Israel, "Capacity and coding for the gilbert-elliott channels," *IEEE Trans. Inform. Theory*, vol. 35, no. 6, pp. 1277–1290, Nov. 1989.

[4] P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: measuring error on simplified surfaces," *Computer Graphics Forum*, vol. 17, no. 2, pp. 167–174, June 1998.