Texturing of Surface of 3D Human Head Model

Ján MIHALÍK, Viktor MICHALČIN

Laboratory of Digital Image Processing and Videocommunications,
Dept. of Electronics and Multimedia Telecommunications, FEI TU Košice,
Park Komenského 13, 041 20 Košice, Slovak Republic

Jan.Mihalik@tuke.sk, michalcin@vk.upjs.sk

Abstract. *The paper deals with an algorithm of texturing of the surface of 3D human head model. The proposed algorithm generates a texture consequently of several camera frames of the input video sequence. The texture values from the camera frames are mapped on the surface of the 3D human head model using perspective projection, scan line and 3D motion estimation. To decrease the number of camera frames a filling of empty places by a simple interpolation method has been done in the texture plane.*

Keywords

Texturing, surface, 3D human head model, perspective projection, 3D motion estimation.

1. Introduction

MPEG-4 [1] has shortly come up with a clear position in favor of detailing special classes of synthetic objects, which were considered of particular importance in audio-visual scenes. Among these, the human face has deserved major attention and specific parameters have been standardized for its calibration and motion. Besides calibration and motion of a generic 3D human head model in order to allow a photo-realistic appearance of the model a texture have to be mapped to its surface. The texture is similar to a still image and can be compressed by any technique suitable for compression of still images [2]. While a texture, extracted from the single frontal view camera frame of the input video sequence, is very simple on the other side a texture obtained by 3D scanner is full one.

The paper presents an algorithm of texturing of the surface of a specific 3D wire frame model using a combination of textures from several camera frames. Such a way we keep possibility of generation of more complex texture from the input video sequence without the need of a 3D scanner.

2. An Algorithm of Texturing

The texture gives 3D human head model the final appearance and represents color values of surface points. The textured 3D human head model should be the same as a real human head in the input video sequence, therefore its texture must be obtained from the video sequence. Texturing of the surface of 3D human head model is based on mapping of the texture values of the points in a texture plane to the corresponding ones on the surface. Assuming constant luminance conditions we can obtain the texture for whole surface of 3D human head model from a few frames.

If motion parameters Θ_h, Θ_v, Θ_r, t_h, t_v, t_r of the 3D human head model [3] are known then the moved model point $\boldsymbol{M}'=(h', v', r')^T$ is perspective projected to the corresponding one (i',j') in a frame. Afterwards the texture value of the nearest point to (i',j') on the frame lattice is assigned to the initial model point $\boldsymbol{M}=(h, v, r)^T$. Note that we do not know the coordinates of all surface points, because only the vertices define the 3D human head model in the model coordinate system (MCS). For obtaining a texture for the whole surface of the 3D human head model, first its polygonal structure is projected polygon by polygon to the frame. Then the inside of each polygon of the 3D human head model in MCS is projected to the one of the corresponding polygon in the frame (Fig. 1).

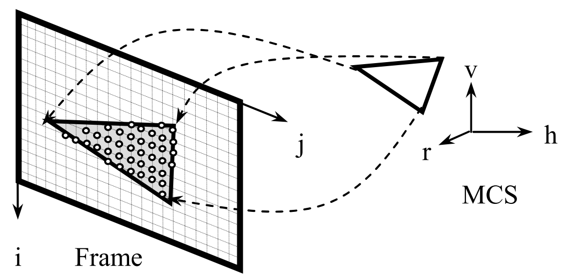

Fig. 1. Projection of a polygon of 3D human head model from MCS to the frame.

By the scan line algorithm [4] we can obtain consequently coordinates (i',j') of all lattice points inside of each projected polygon in the frame. The relationship between the coordinates (i',j') and the initial ones (h,v,r) of the corresponding model point for the known 3D motion parameters Θ_h, Θ_v, Θ_r, t_h, t_v, t_r is given by a linear system of two equations [3]

$$\begin{bmatrix} f_y\Theta_v+\Theta_r(i'-i_0) & f_y-\Theta_h(i'-i_0) & -f_y\Theta_h-(i'-i_0) \\ -f_x+\Theta_v(j'-j_0) & f_x\Theta_r-\Theta_h(j'-j_0) & -f_x\Theta_v-(j'-j_0) \end{bmatrix}\begin{bmatrix} h \\ v \\ r \end{bmatrix}=$$
$$=\begin{bmatrix} -d(i'-i_0)-f_y t_v+t_r(i'-i_0) \\ -d(j'-j_0)+f_x t_h+t_r(j'-j_0) \end{bmatrix} \tag{1}$$

where f_x and f_y are the scaled focal lengths of a camera, and d is the distance between the centers of MCS and camera coordinate system (CCS). Rewriting eq.(1) to the matrix form we get

$$\underline{\mathbf{C}}\mathbf{M}=\overline{\mathbf{D}} \ . \tag{2}$$

The number of unknowns in eq. (1) is bigger than the number of equations therefore it is impossible to compute the coordinates (h,v,r) exactly by it. The (h,v,r) of a surface point of a polygon in MCS can be substituted by its barycentric coordinates (b_1,b_2,b_3) [5] as follows

$$\begin{bmatrix} h \\ v \\ r \end{bmatrix}=\begin{bmatrix} h_1 & h_2 & h_3 \\ v_1 & v_2 & v_3 \\ r_1 & r_2 & r_3 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \tag{3}$$

or in the matrix form

$$\mathbf{M}=\underline{\mathbf{S}}\mathbf{B} \ , \tag{4}$$

where (h_1,v_1,r_1); (h_2,v_2,r_2); (h_3,v_3,r_3) are the coordinates of the polygon vertices in MCS. In general, for barycentric coordinates it is valid

$$b_1+b_2+b_3=1 \ , \tag{5}$$

$$0\le b_i\le 1, \ i=1,2,3 \ . \tag{6}$$

From eq. (2) and (4) we get the system of two linear equations

$$(\underline{\mathbf{C}}\underline{\mathbf{S}})\mathbf{B}=\overline{\mathbf{D}} \ . \tag{7}$$

If $\underline{\mathbf{C}}\ \underline{\mathbf{S}}=\underline{\mathbf{E}}$ then eq. (7) in conjunction with eq. (5) gets the system of three linear equations with three unknown barycentric coordinates (b_1,b_2,b_3)

$$\begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ 1 & 1 & 1 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}=\begin{bmatrix} d_1 \\ d_2 \\ 1 \end{bmatrix}, \tag{8}$$

or in the matrix form

$$\underline{\mathbf{E}}\mathbf{B}=\overline{\mathbf{G}} \ . \tag{9}$$

For the coordinates (i',j') of an inner point of the polygon in the frame we can compute the barycentric coordinates (b_1,b_2,b_3) of the corresponding inner point of the polygon on the surface of the 3D human head model from eq. (9). Finally, by eq. (3) from (b_1,b_2,b_3) we get the coordinates (h,v,r) of the surface point in MCS to which the texture value from (i',j') in the frame will be assigned.

Assuming known 3D motion parameters Θ_h, Θ_v, Θ_r, t_h, t_v, t_r in a frame of the input video sequence, the proposed algorithm of texturing of the surface of a polygon (triangle) of the 3D human head model includes 6 steps:

1 3D moving of vertices of the polygon.

2 Perspective projection of the rotated and translated vertices of the polygon on the frame.

3 Scan line algorithm to obtain integer coordinates (i',j') of the inner points of the projected polygon in the frame. The coordinates of polygon vertices in the frame are in the float format.

4 Computing of barycentric coordinates (b_1,b_2,b_3) by eq. (9) of the corresponding surface point for each point (i',j') from the third step.

5 Getting coordinates (h,v,r) of the inner point of the polygon on the surface of the 3D human head model by eq. (3) from barycentric coordinates computed in the fourth step.

6 Finally we assign the texture value of the point (i',j') in the frame to the corresponding surface point with coordinates (h,v,r).

The algorithm of texturing of the surface of the 3D human head model is done polygon by polygon. We cannot generate a texture of the whole surface only from one frame of the input video sequence because in each frame only a particular part of the human head is seen. Therefore a texture for the whole surface of 3D human head model is generated by a combination of textures from a few frames of the input video sequence [6]. The combined texture is stored in the texture plane in a decoder, where it is used for texturing of the surface of the 3D human head model. Then the proposed algorithm of texturing is composed of a generation of the texture plane and mapping of its texture content on the surface of the 3D human head model.

In a decoder the texture plane is empty by the beginning of generation. The extracted texture value for the surface point $\boldsymbol{M}=(h, v, r)$ of the 3D human head model will be assigned to the projected point in the texture plane with cylindrical coordinates

$$s=round(v) \ , \tag{10}$$

$$t=round(m\arctan(h/r)) \ , \tag{11}$$

where m is a horizontal scale of the texture plane and *round* means round operator. If there is no texture value at the projected position (s,t), the extracted texture value is stored at this position. Otherwise, the following equation is used to incorporate the new texture value into the texture plane

$$Y_M=(1-\alpha)Y_M+\alpha Y_S \ , \tag{12}$$

where Y_M is a texture value in the texture plane, Y_S is an extracted one from the frame, and α is the blending factor. It is set α=1.0 if there is no previous texture value in a texture plane, and α=0.5 otherwise.

3. Simulation Results

The proposed algorithm of texturing of the surface of 3D human head model Candide [7], [8] was verified for the input video sequence Claire of 166 frames of the size 288x352 pels. Before texturing of the surface of 3D model Candide it was calibrated according to the reference frame of the video sequence Claire. Afterwards calibration of a camera has been done for which we get the scaled focal lengths f_x=8752 and f_y=1072 pels assumed the distance d=10000 pels between CCS and MCS [3].

In Fig. 2a there is a texture generated by the proposed algorithm only from the first frame of the video sequence Claire together with the polygonal mesh of 3D human head model Candide. The horizontal scale of the texture plane was set up on the value m=80. At the beginning of generating of a texture there are lots of empty places in the texture plane which are consequently filled in by next frames of the video sequence Claire, as it is seen in Fig. 2b. Another way how to reduce empty places in the texture plane is by decreasing of its horizontal scale which is seen in Fig. 2c as m=30. For small horizontal scales m generated textures are more continuous without empty places but on the other side texture values are multiple updated as follows out from eq. (12). It is caused by projection of the same number of texture values on the texture plane with different horizontal scales. Better solution is keeping the horizontal scale big enough and filling empty places of a texture generated after a number frames of the video sequence Claire, by using interpolation method. In Fig. 2d we have the interpolated texture obtained from the one in Fig. 2b by the average interpolation over the window of the size 3x3 pels.

The texture from Fig.2d generated of 40 selected frames of the video sequence Claire after the average interpolation with the horizontal scale m=80 has been used for texturing of the surface of 3D model Candide. In Fig. 3 there is a frame of the video sequence Claire together with

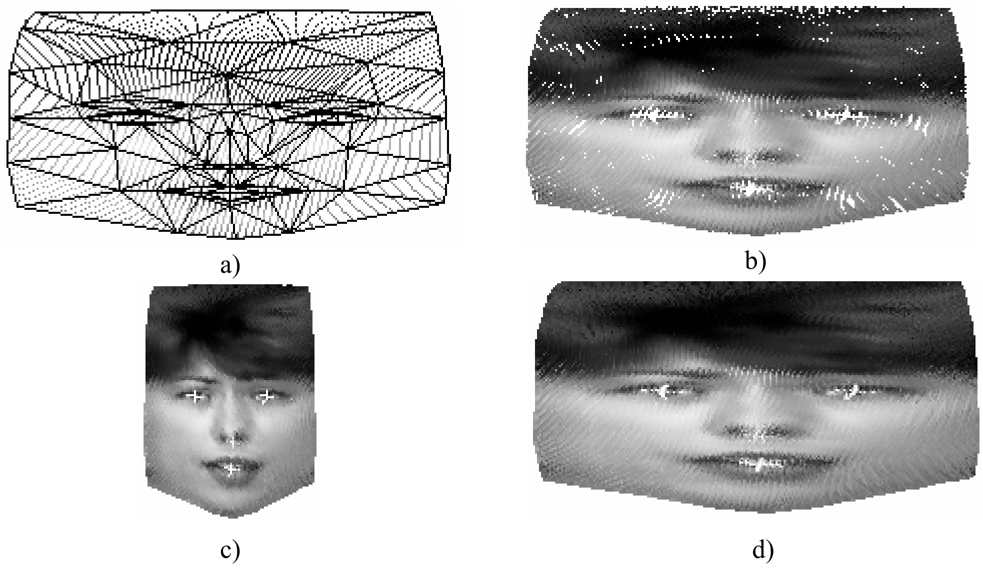

Fig. 2. Generated textures from a) the first frame, m=80, b) 40 frames, m=80, c) 40 frames, m=30, d) 40 frames, m=80 after average interpolation.

the textured 3D model Candide perspective projected on the synthesized frame. The textured 3D model Candide in the synthesized frame was created by 3D motion of the model Candide according to the motion of the real human head in the frame of video sequence Claire, then by its perspective projection on the synthesized frame and mapping of the texture values from the plane of generated texture in Fig. 2d.

The error measure that we used for its evaluating was the peak signal noise ratio (PSNR), which is defined as

$$PSNR = 10\log\left(255^2 / \frac{1}{N}\sum_{(i,j)}\left[I_{orig(i,j)} - I_{synth(i,j)}\right]^2\right). \qquad (13)$$

In this equation $I_{orig(i,j)}$ is the luminance component of pel in the original camera frame with values from 0 to 255, and $I_{synth(i,j)}$ is the corresponding value in the synthesized frame. N denotes the number of pels in facial area. The PSNR between each original and synthesized frames computed in facial area is in Fig. 4 and its average value is 20,9 dB.

Fig. 3. A frame of videosequence Claire and the textured 3D model Candide perspective projected on the synthesized frame.

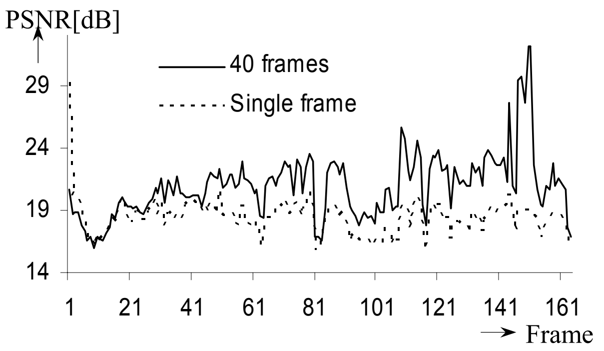

Fig. 4. PSNR between original and synthesized frames by using the textured 3D model Candide.

For comparison, the PSNR for the reconstructed video sequence Clair that has been generated using only the single frontal view frame for texturing of the model Candide is also shown in Fig.4 at the same calibration and 3D motion parameters [3]. It is seen from Fig.4 that virtual quality measured by PSNR increases in average about 2,5 dB if we use more complex texture composed of 40 camera frames by the proposed algorithm.

Texturing of the surface of the 3D human head model assumes the knowledge of 3D global motion parameters. In general, the quality of the generated texture depends on the accuracy of 3D motion estimation, the horizontal scale, the number of selected frames used for its generation and the interpolation method.

4. Conclusion

In this paper we have presented the algorithm of texturing of the surface of the 3D human head model for MPEG-4. The proposed algorithm consequently extracts textural information from several camera frames of the input video sequence. The texture values from the camera frames are mapped on the surface of the 3D human head model using perspective projection, scan line and 3D motion estimation. To decrease the number of camera frames a filling of empty places by the average interpolation method has been done in the texture plane. Since only the incrementally updated texture information is transmitted during the whole session, the bandwidth requirement is very small. This approach, in comparison with the previous schemes, has the advantages of larger permissible viewing angle of virtual scenes and higher virtual image quality. The experimental results of texturing of the surface of the 3D human head model Candide show acceptable quality of the synthesized video sequence Claire. Further increasing of quality can be achieved by animation and more complex wire frame model of the human head.

Acknowledgement

The work was supported by the Scientific Grant Agency of the Ministry of Education and the Academy of Science of the Slovak Republic under No. 1/0384/03.

References

[1] *Signal Processing: Image Communication.* 2000, vol.15, no.4-5. (Tutorial Issue on MPEG-4 Standard).

[2] MIHALÍK, J. *Image Coding in Videocommunications*. Košice: Mercury - Smékal, 2001. ISBN 80-89061-47-8(in Slovak).

[3] MIHALÍK, J., MICHALČIN, V. 3D Motion Estimation and Texturing of Human Head Model. *Radioengineering*, 2004, vol.13, no.1, p.26- 31, ISSN 1210-2512.

[4] FOLEY, J. D, VAN DAM, A., FEINER, S. K, HUGHES, J. F. *Computer Graphics, Principles and Practices*. Addison-Wesley, 2nd edition, 1990.

[5] GALLIER, J. *Curves and Surfaces in Geometric Modeling: Theory and Algorithms*. Morgan Kaufmann Publishers, 2000.

[6] TSAI, C. J., EISERT, P., GIROD, B., KATSAGGELOS, A. K. Model-Based Synthetic View Generation from a Monocular Video Sequence. In *Int. Conf. on Image Proc*., Santa Barbara, vol.1, 1997, p.444-447.

[7] RYDFALK, M. *CANDIDE: A Parameterised Face.* Dep. Elec. Eng. Rep. LiTH-ISY-I-0866, Linköping Univ., 1987.

[8] AHLBERG, J. *Candide-3 – An Updated Parameterised Face*. Report No. LiTH-ISY-R-2326, Linköping Univ., 2001.

About Authors…

Ján MIHALÍK graduated from the Technical University in Bratislava in 1976. Since 1979 he has joined the Faculty of Electrical Engineering and Informatics, Technical University of Košice, where he received his PhD degree in Radio electronics in 1985. Currently, he is Full Professor of Electronics and Telecommunications and the head of the Laboratory of Digital Image Processing and Videocommunication at the Department of Electronics and Multimedia Telecommunications. His research interests include information theory, image and video coding, digital image and video processing and multimedia videocommunication.

Viktor MICHALČIN was born in 1976 in Ukraine. He received the Ing. Degree from the Technical University of Košice in 2000. Currently he is PhD student at the Department of Electronics and Multimedia Telecomunications, Technical University, Košice. His research interests are vector quantization and model based image coding.