Automatic Image-Based 3D Head Modeling with a Parameterized Model Based on a Hierarchical Tree of Facial Features

Peter Kán*
Supervised by: Andrej Ferko†

Faculty of Mathematics, Physics and Informatics
Comenius University
Bratislava

Abstract

The automatic modeling of a 3D human head has challenged researchers in computer graphics for many years, because 3D head models are useful in several application areas to assist in different tasks. In this paper we present the novel system for automatic 3D head model creation from two images based on a parameterized head model with a hierarchical tree of facial features. The proposed system is divided into three parts. In the first one we use computer vision techniques like Skin-Tone based image segmentation or Haar Cascade Classifiers to detect head parameters for parameterized model update from frontal and profile head photograph. Then a 3D head model is reconstructed by updating a parameterized model with parameters detected from images and a texture is mapped. In the second part the reconstructed head model is rendered with real-time rendering techniques simulating skin illumination. In the third part the model is exported to the Collada format for further use in other applications. The system for automatic image-based 3D head modeling can be used e.g. in computer games development, movies, telecommunications, medicine, or security systems for human identification.

Keywords: 3D head modeling, face detection, facial features detection, face reconstruction, parameterized model, real-time rendering, Collada

1 Introduction

A 3D head model is an essential part of many computer graphic applications and its easy creation is an important task for developers. Manual 3D head model creation is very time consuming, especially when good precision is needed. The Automatic 3D head creation algorithm with few or no user interaction is a challenging task.

A parameterized head model [6] provides us possibility of creation many variable 3D heads and we can obtain a model representing concrete person if correct parameters and texture are used. In this work we created a new parameterized head model based on a hierarchical tree of facial features which enable application to easily estimate correct parameters for a specific human head. This model defines vertex weights for each facial feature, which gives us the information about transformations intensity in each vertex.

To gather concrete head parameters we use computer vision and image processing techniques like Haar cascade classifiers [23] and skin tone based image segmentation [14, 12, 17, 18]. Then we analyze detection results and estimate final head model parameters.

According to detected parameters from input images we calculate global and local transformations and apply them to create new model. Then we interpolate between old and changed model with parameters defined as vertex weights for each facial feature.

Our parameterized model consists of 32672 triangles to reach good reconstruction precision and to gives us high resolution head model. This model can be rendered with texture-space diffusion [11] technique even approximating subsurface scattering in interactive frame rates.

The paper is organized as follows. In section 2 previous works are discussed. In section 3 main parts of our modeling system are described, our novel parameterized head model based on hierarchical tree of facial features is discussed and rendering techniques simulating skin illumination are shown. In section 4 our reconstruction system results are shown and in section 5 we discuss conclusion and future work.

2 Related Work

The automatic head detection and reconstruction is an important task solved in research area for a long time. Many techniques were created to reach the best results. Head detection and reconstruction are in some approaches strongly related. The automatic head model creation techniques use head detection results and vice versa.

Reconstruction results within different approaches are variable in quality and computation time according to the purpose of use. Techniques used in telecommunications

*peterkan@peterkan.com
†ferko@sccg.sk

create a low-polygonal face or head model to achieve lower data transfer in videoconferences. For this purpose Mikael Rydfalk [22] created a parameterized face model named Candide in 1987. His model was a low-resolution adjustable model, which can be updated by several parameters. This model was later improved by Jörgen Ahlberg [6] to simplify animation and it was named Candide-3. An advantage of the Candide-3 model is fast reconstruction time, but disadvantage of this technique is low precision and few details.

Another approach generating high quality 3D head models, called a Morphable Model, was proposed by Volker Blanz and Thomas Vetter [8] to reach photorealistic model appearance and high precision in reconstruction. In this approach head model is represented as a multidimensional 3D morphing function based on the linear combination of large number of the 3D face scan. The main disadvantage of a Morphable Model is high computational time. This model was later updated by Yong-Li Hu et al. [15], who used the mesh resampling method to overcome the key problem of model construction, the pixel-to-pixel alignments of prototypic faces. When applying the method to several images of a person, the reconstructions reach almost the quality of laser scans.

Image-based head model reconstruction techniques are much more available and applicable. In some of them a head is reconstructed from one image like the AvMaker [16] program or reconstructed from several images. The special case of image-based head reconstruction is gathering 3D coordinates of points from two 2D images using stereo vision [20]. Advantages of image-based head model reconstruction techniques are generally fast computational time and low reconstruction cost. The main disadvantage is low reconstruction accuracy.

An Active Appearance Models [10] is an example of head model reconstruction technique based on predefined model and it is used mainly for head tracking in video. Liu et al. [21] use video sequences to generate 3D head models. A disadvantage of this approach is low acceptable head rotation angle.

In our solution image-based 3D head model reconstruction technique with predefined parameterized head model based on a hierarchical tree of facial features is used. This approach is capable of giving us the reconstructed 3D head model in a few seconds.

3 Virtual reconstruction of a human head

We created the Modeling system, which consists of the three main phases (Figure 1). In the first phase input images are loaded, head is detected in both images and textured 3D model is created. Then the reconstructed model is rendered in second phase with real-time rendering techniques, which simulate skin lighting. In the third phase the

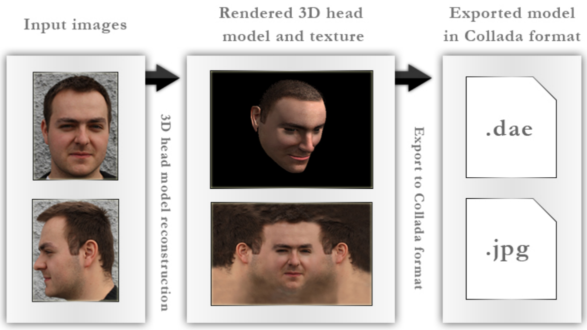

Figure 1: Three main phases of the modeling system

model can be exported to Collada format for further use in another application. The texture is also exported with a model given.

The modeling system detects facial features positions and head attributes automatically from input images and also allows user to adjust detected positions to enhance reconstructed model precision.

In our work a 3D head model is automatically reconstructed from two input images. For this purpose we created the new parameterized head model based on the hierarchical tree of facial features (Figure 4).

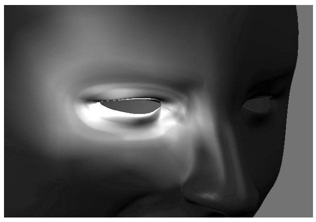

Figure 2: Example of vertex weights for the left eye. White color means weight value 1 and Black means 0.

All of these features have defined their own area of vertices. This assignment is done by vertex weights for each region as we can see in Figure 2. 3D positions of facial features are calculated from automatically detected 2D positions from input images. Facial features are organized in the hierarchical tree, where root node is the head center and child nodes are main facial features. Child nodes of the facial feature are its properties or other more detailed features. For example Left eyes nodes define left, right, top and bottom border coordinates in front image. The reconstruction system can easily adjust head parameters with the hierarchical tree of facial features, because in each facial feature position estimation all subnodes are precalculated to help find a real feature position. Another advantage of this approach is a possibility of simple user interaction and manual adjusting of the head model.

The reference mesh (Figure 3) of the parameterized model automatically controlled by the hierarchical tree contains 32672 triangles to achieve high resolution output model with better precision and smoothness.

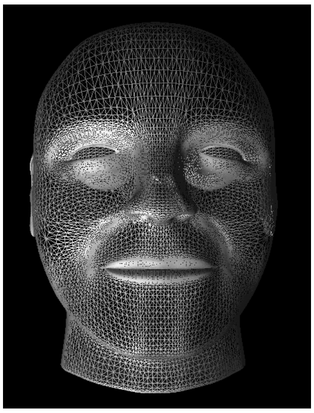

Figure 3: Reference mesh of parameterized head model

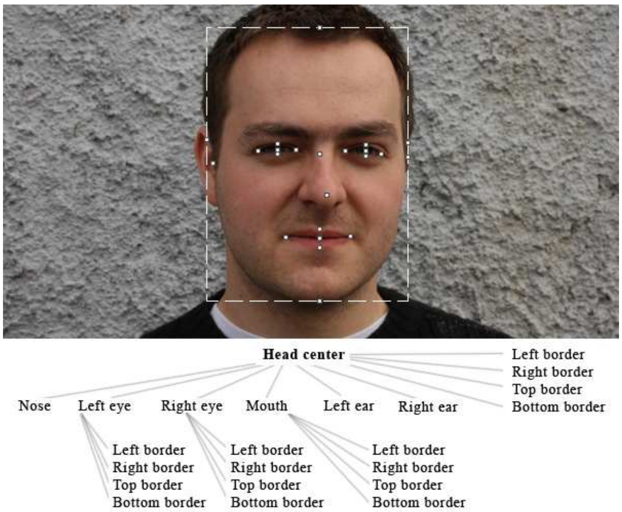

Figure 4: The hierarchical tree of facial features and theirs attributes. In bottom image are nodes in the tree and in top image is their visualization.

3.1 Head Model Reconstruction

Our automatic 3D head modeling algorithm with a parameterized model based on a hierarchical tree of facial features generates a three dimensional representation from two 2D images in two stages. In the first stage head position, facial features positions, and head attributes are detected from frontal and profile images using computer vision and image processing techniques. In the second stage is the parameterized model transformed to shape values detected from input images.

3.2 Face Recognition and Parameters Detection

Frontal and profile head photographs are expected as an input for the reconstruction system, therefore face and its properties detection should be performed separately for both images. To recognize the head and gather its properties from images two main techniques are used.

Haar Cascade Classifiers based on an extended set of Haar like features [19] are employed to detect face and facial features in the front image. For this purpose classifiers trained to detect Front head region [19], Eyes positions, Nose position [9] and Mouth position [9], and facial features proportions are used. The results of the face region and facial features detection can be seen in Figure 5.

To detect a head shape in the front image a Skin tone based image segmentation [14, 12, 17, 18] is used. The image encoding is converted into YCrCb color space and for all pixels Cr and Cb values are evaluated. If Cr and Cb coordinates of the pixel are in a skin tone area (Figure 6a), the pixel is segmented as skin-tone colored. By this technique all pixels are divided into skin tone pixels and non skin tone pixels as we can see in Figure 6. Then we can analyze skin tone colored pixels and acquire information about the head shape.

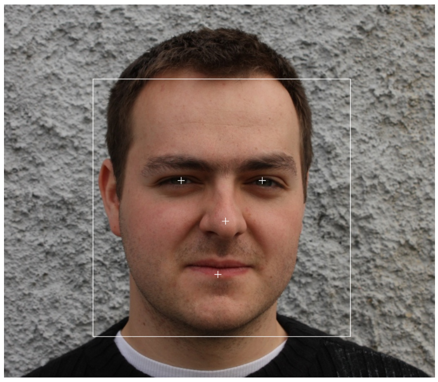

Figure 5: Face region and facial features detection by Haar Cascade Classifiers. Facial features are marked with a cross and face region with a rectangle.

Figure 6: Skin tone based segmentation. (a) Adjusted skin-tone area in CrCb color plane. Black pixels give us the information about Cr and Cb value considered as skin-colored. (b) Input image, and (c) segmentation result.

A color space selection is an important choice in Skin tone based image segmentation technique. We examined three color spaces Normalized RGB, Lab and YCrCb to select one, which will give us the best segmentation results. We chose chrominance plane from each color space to have the most chrominance information stored in two selected channels. Skin tone areas in tested color spaces can be seen in Figure 7. Finally we chose YCrCb color space, because it gives us the most covered facial area by skin tone segmented pixels, and we manually adjusted skin tone area in CrCb plane (Figure 6a) to achieve better segmentation result. Skin tone area is stored in 255x255 binary image in our reconstruction system, because Cr and Cb values can reach maximum value 255.

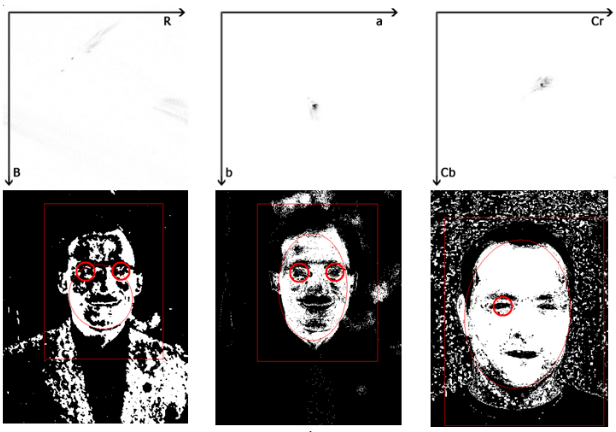

Figure 7: Skin tone areas (top) and segmentation result (down) in Normalized RGB (a) Lab (b) and YCrCb (c) color spaces. Skin tone areas were created by selecting skin areas in images from training set and adding skin pixel values into the skin tone area.

Figure 8: Profile face and facial features detection. From input image (a) the binary image is created by the skin tone based segmentation (b). The segmented image is filtered by a median filter and moving probes are sent from head center to each direction (c) to find borders. Then a head bounding box (d) is calculated. Finally a front head contour is obtained by sending moving probes from left border to right and gathered face curve is analyzed.

To detect parameters from the profile image (Figure 8) we use the skin tone based segmentation and then we apply a median filter to smooth the segmentation result and reduce noise. After median filtering we calculate head center by a linear interpolation of skin tone colored pixels' coordinates. We send moving probes from head center to each direction. These probes stop moving when they reach non skin tone colored pixel. After all probes stop, we calculate head bounding box by getting minimum and maximum x and y coordinate of probes. Then we obtain front head contour by sending moving probes from left to right border. Probes stop if they reach skin tone colored pixel. After all probes stop, we smooth out the result curve by a median filter and we can analyze it to obtain the information about tip of the nose position, eyes and mouth position. To analyze back head contour we send moving probes from right border to left.

As input images to parameters detection could be used photographs in an optional resolution and quality, but we recommend to use images with resolution 512x512 and higher, where more information about facial features is included and high resolution texture could be created.

3.3 3D Model Update

Updating model parameters is a key point in the 3D head reconstruction with the parameterized model. In this step new vertex coordinates are calculated to shape head from input images. Head color texture and normal map texture are created from input images and they are mapped to reconstructed model geometry to reach more realistic result in rendering.

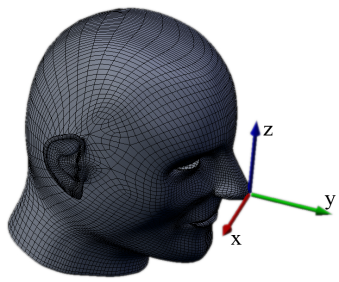

Figure 9: The parameterized head model with a reference coordinate system aligned to the tip of the nose.

All features in the parameterized model based on the hierarchical tree of facial features have defined vertex weights to transform correct vertex due to the features coordinates in the hierarchical tree calculated in the head recognition phase. These weights were precalculated in a training process (Figure 2). Each facial feature has also defined its own local coordinate system, which center is stored in our parameterized model and usually it is in the center of facial feature. Rotation of local feature coordinate system is same as in global coordinate system.

Firstly we need to define a model coordinate system origin to perform image space to model space position calculation. In our model a tip of the nose was selected as a reference point (Figure 9) of a 3D model coordinate system. The first step in updating model phase is global scale transformation application. We don't change head model width in global transformations. According to model width and head aspect ratio in images we calculate vertical head scale along z coordinate and depth head scale along y coordinate as

$$V_s = \frac{FH_{2D}W_{3D}}{FW_{2D}H_{3D}} \tag{1}$$

and

$$D_s = \frac{PW_{2D}H_{3D}}{PH_{2D}D_{3D}}, \quad (2)$$

where V_s is a vertical scale coefficient along z axis, D_s is a depth scale coefficient along y axis, FW_{2D} and FH_{2D} are head sizes in front image, PW_{2D} and PH_{2D} are head sizes in profile image and W_{3D} is 3D head model width, H_{3D} is 3D head model height and D_{3D} is 3D head model depth in model space. After calculating V_s and D_s we scale all vertex positions in the 3D head model with those coefficients. Global scale is performed in coordinate system with origin in head center to preserve head center position.

After performing the global scale we translate head model to have the tip of the nose in the origin (0,0,0). Then we can make local head transformations in local features coordinate systems on vertices with weights defined per each facial feature in the hierarchical tree. We calculate new features positions and sizes according to the location in front and profile image. Then we calculate a final vertex position by the linear interpolation between new features positions and their old locations.

$$Vertex_i = (1-p).oldVertex_i + p.newVertex_i \quad (3)$$

According to equation 3 we calculate the position of i^{th} vertex $Vertex_i$ from old i^{th} vertex position $oldVertex_i$ and new vertex position $newVertex_i$ for each vertex and each facial feature in the 3D model. We set parameter p in linear interpolation as vertex weight defined for each facial feature. We calculate new vertex coordinate from old coordinate by scaling and translating according to facial feature transformations. Final coordinates of each vertex are obtained by applying equation 3 to it for each facial feature transformation (scale, translate), where vertex weight for current vertex and facial feature is set as parameter p. The vertex position $newVertex_i$ is calculated by concrete facial feature transformation.

After performing all local transformations according to facial features positions and sizes in images we have adjusted 3D head geometry, which shapes a head from input images. Then we create a head texture from detected head image areas and map this texture on the reconstructed geometry. Front and profile images are joined together in texture creation phase. The first step is scaling a smaller image up to have same head height as in the bigger one. Then is front head image rectangle from frontal image joined with profile head image rectangle. A profile rectangle is connected to front rectangle from both left and right sides. Horizontally flipped profile image is joined from the left side. X joint coordinate is in front image left eye left border position and right eye right position and eye center in profile image. Blending between images is calculated by the linear interpolation in connection area. After image junction, known texture part is cloned into background texture part. For example the hair texture is cloned into the space above head.

When the texture creation is finished, we need to calculate normal map texture. Grayscale copy of color texture is used as a surface height map, because it contains main head surface details information. For each pixel in grayscale image it is calculated its x direction and y direction intensity difference with the next pixel in current direction. This difference is scaled with surface height scale constant to decrease differences. We found empirically the height scale constant 0.01 in our experiments. X and Y differences are brought into account as derivations of image function in their directions. From these derivations we can create direction vectors and calculate image function normal as their cross product. Then we normalize obtained normal and transform its coordinates into RGB color space for storing in normal map texture.

The head model uv coordinates also need to be adjusted to correctly shape head in new created texture. For this purpose uv coordinates of vertices are recalculated for each facial feature. Scale and translation are performed on all vertices with the origin in concrete facial feature center and with linear falloff.

3.4 Head Rendering

When the 3D head model is reconstructed we would like to display it to user and enable him or her to interact with the model and optionally change parameters to reach better precision. For making the reconstructed 3D head model rendering more realistic we can use local lighting illumination calculation. Very simple approximation of skin illumination is a local Phong lighting model [24], because it is not so computational expensive and gives us a better performance with slower computers. To enhance details in rendering we can use a Normal mapping [13] technique, which simulates local object normals with a normal map texture calculated in the reconstruction phase. The Normal mapping enhances details, but causes very hard and rough surface, because it does not calculate light transport under the skin surface.

When we want to reach a realistic skin lighting, we should simulate a subsurface scattering effect caused by light scattering under the skin. We use a texture-space diffusion technique described by Simon Green in the GPU Gems book [11]. In the first step the 3D head model is rendered without the texture to texture space according to uv coordinates and lighting is calculated by a Phong model or advanced lighting calculations like the normal mapping. In the second step the result is blurred with a predefined convolution to simulate light transport under the skin. The Uniform blur only approximates subsurface light transport, because a distance between two pixels is in the texture space different than in the 3D world space. In the third step the model is rendered with the color texture and lighting computed in previous steps. The Light transport of different color channels could be simulated variously to create inner material light coloring, like soft red color from blood under the skin. Rendering techniques

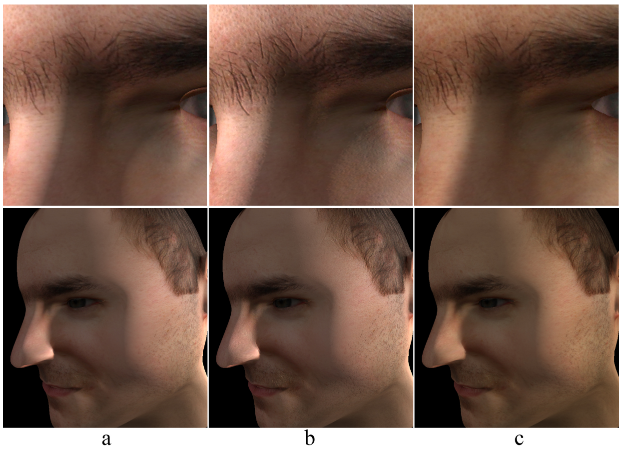

Figure 10: Head rendering result with techniques (a) Phong illumination model, (b) normal mapping, (c) texture-space diffusion. Detail rendering (top) and whole head (bottom).

results comparison could be seen in Figure 10.

4 Results

The proposed modeling system was implemented in C++ programming language using OpenCV [2] library in detection phase and OpenGL [3] library to render the reconstructed head model. Glsl shading language [4] was used to implement shaders calculating Phong model, normal mapping and texture-space diffusion lighting simulation.

Lighting calculation	fps
No shader	65
Phong lighting model	64
Normal mapping	42
Texture-space diffusion	26

Table 1: Frame rates in rendering the reconstructed 3D head model with different rendering techniques.

Method	Quality	Reconstruction time
Morphable Model	89%	50 minutes
CyberExtruder	82%	4 seconds
Our method	57%	4 seconds

Table 2: Reconstruction quality evaluation and speed comparison. Our solution, morphable model [8] and CyberExtruder [16] solution were tested.

The system was tested on AMD Athlon X2 1.9 Ghz, 2GB RAM computer with NVIDIA GeForce 7150 graphic card. Detection in a front image takes 1.54 seconds in average and detection in a profile image takes 1.8 seconds in average in this hardware configuration. The 3D head model updating process takes 4 seconds on average. The Reconstructed 3D head model was rendered at interactive framerates reported in Table 1. In each tested rendering technique we used the textured model.

Quality of reconstruction was measured by statistics of subjective human evaluation. Respondents were asked to evaluate the quality of 3D head reconstruction from concrete front and profile images in percentage. Three different approaches were compared by this evaluation (Table 2). Final approach quality was figured out as an average of respondents answers. Quality of our solution could be improved in future work by adding more head attributes to parameterized model and by enhancing detection precision.

Example of front and profile source photographs, reconstructed head geometry and textured model with Phong illumination could be seen in Figure 11.

We create a 3D head model from 2D images to use reconstruction result in various applications and therefore we need to export model and texture to common data format usable in broad scale of programs. Many data formats to store 3D information are available today, but most of them are strongly application dependent and their binary formats are difficult to read and do not allow all required information storing.

Therefore, for 3D data storage we chose Collada format [7, 5], which was created to enable work with one content in many digital content creation tools, it was accepted as industry standard by Khronos Group and it is used in OpenGL ES and several other real-time APIs. Collada provides us easy readable XML based data format to store required geometric and radiometric information.

The basic model information like reconstructed geometry, global transformations and visual effects should be stored in the output file. Collada output file contains relative links defining file names and paths to textures stored in jpg file format.

5 Conclusion and Future Work

We created the automatic image based 3D head modeling system, which is capable of reconstruct the 3D head model from 2D images. This system creates 3D head model automatically, but it allows user to change reconstructed model to reach better precision. The 3D head modeling system renders the reconstruction result with advanced illumination techniques simulating skin illumination. Finally, user can export reconstructed high polygonal model for use in other applications.

New parameterized head model based on the hierarchical tree of facial features, which defines vertex weights for each feature to set transformation intensity, was developed in this work. This model provides easy feature position and attributes estimation for modeling system and user. Our parameterized model consists of 32672 triangles and allows modeling system to reconstruct high resolution

a b c d

Figure 11: Example of front (a) and profile (b) source photograph, reconstructed head geometry (c) and textured model(d).

model.

We introduce a novel method for detecting head parameters from image with use Haar Cascade Classifiers and skin-tone based image segmentation in this work. This method gives us correct results to achieve good precision with updating a parameterized model.

To load parameterized head geometry and render the reconstructed model we created a Collada rendering engine, which could be used in many other works to easily load and render digital content with GPU.

Many extensions could further improve our modeling system, because the application is object based and allows updating or changing each reconstruction part for other purposes. More facial features controllers could be added and better head parameters detectors could be implemented to enhance the reconstruction precision and speed.

In a future work we are planning to improve the reconstruction precision with optical flow based techniques used in the morphable model approach [8]. Reconstruction precision could be improved also by adding more head attributes to the parameterized model. We are aiming to develop a hair reconstruction technique, which could update the modeling system to give full head and hair reconstruction result. Lower precision head model could be created by decimating high precision reconstructed model, but a better way is to estimate it during reconstruction and make model details selection part of the modeling system. We are planning to use OpenCL [1] library to move computation to GPU.

6 Acknowledgments

I would like thank doc. RNDr. Andrej Ferko, PhD. for his helpful guidance and support in this work, Jörgen Ahlberg for providing his parameterized face model report and Michal Šukola, Tomáš Matúš and others for providing their photographs to development and presentation purposes. I thank to anonymous reviewers for their many valuable comments and suggestions. This work has in part been funded by Slovak Ministry of Education VEGA No. 1/0763/09.

References

[1] Opencl. [online]. [quot. 2010-02-16]. Available on www: < http://www.khronos.org/opencl/>.

[2] Opencv. [online]. [quot. 2009-11-18]. Available on www: <http://opencv.willowgarage.com/wiki/Welcome>.

[3] Opengl. [online]. [quot. 2009-11-16]. Available on www: <http://www.opengl.org/>.

[4] Opengl shading language. [online]. [quot. 2009-11-17]. Available on www: <http://www.opengl.org/documentation/glsl/>.

[5] Collada - digital asset exchange schema for interactive 3d. [online], 2009. [quot. 2009-12-10]. Available on www: <http://www.khronos.org/collada/>.

[6] J. Ahlberg. Candide-3 an updated parameterized face. report no. lith-isy-r-2326. Technical report, Dept. of Electrical Engineering, Linköping University, Sweden, 2001.

[7] R. Arnaud and M.C. Barnes. *COLLADA Sailing the Gulf of 3D Digital Content Creation*. A K Peters, Wellesley, 2006.

[8] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.

[9] M. Castrillón-Santana, O. Déniz-Suárez, L. Antón-Canalís, and J. Lorenzo-Navarro. Face and facial feature detection evaluation, performance evaluation of public domain haar detectors for face and facial feature detection. In *International Conference on Computer Vision Theory and Application*, 2008. [online], [quot 2010-01-10]. Available on www: <http://alereimondo.no-ip.org/OpenCV/uploads/37/CameraReadyPaper63.pdf>.

[10] N. Faggian, A. Paplinski, and J. Sherrah. Active appearance models for automatic fitting of 3d morphable models. In *AVSS '06: Proceedings of the IEEE International Conference on Video and Signal Based Surveillance*, page 90, Washington, DC, USA, 2006. IEEE Computer Society.

[11] S. Green. *GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics*, chapter Real-Time Approximations to Subsurface Scattering. Addison-Wesley Professional, Indiana, 2004.

[12] A. Hadid, M. Pietikäinene, and B. Martinkauppi. Color-based face detection using skin locus model and hierarchical filtering. In *ICPR '02: Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 4*, pages 196–200, Washington, DC, USA, 2002. IEEE Computer Society.

[13] W. Heidrich and H.P. Seidel. Realistic, hardware-accelerated shading and lighting. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 171–178, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.

[14] R.L. Hsu, M. Abdel-Mottaleb, and A.K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:696–706, 2002.

[15] Y. Hu, B. Yin, S. Cheng, and Ch. Gu. An improved morphable model for 3d face synthesis. In *Proceedings of International Conference on Machine Learning and Cybernetics*, pages 4362–4367, Washington, 2004. IEEE Computer Society.

[16] J.D. Ives. Method for generating a three-dimensional representation from a single two-dimensional image and its practical uses. [online]. Newark: CyberExtruder.com, Inc. [quot. 2009-12-11]. Available on www: <http://www.cyberextruder.com/>.

[17] Ch. Kim and J. Yi. An optimal chrominance plane in the rgb color space for skin color segmentation. In *International Journal of Information Technology*, volume 12, pages 73–81. World Scientific Publishing, 2006.

[18] P. Kuchi, P. Gabbur, P.S. Bhat, S.David, and S. Smieee. Human face detection and tracking using skin color modeling and connected component operators. In *IETE journal of research*, volume 48, pages 289–293, New Delhi, 2002. Institution of Electronics and Telecommunication Engineers.

[19] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *IEEE ICIP 2002*, pages 900–903, 2002.

[20] K. Lin, F. Wang, J. Yao, and Ch. Zhou. Human head modeling based on an improved generic model. In *FSKD '08: Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery*, pages 300–304, Washington, DC, USA, 2008. IEEE Computer Society.

[21] Z. Liu, Z. Zhang, C. Jacobs, and M. Cohen. Rapid modeling of animated faces from video. Technical report, Journal of Visualization and Compute Animation, 2000.

[22] M. Rydfalk. Candide, a parameterized face, report no. lith-isy-i-866. Technical report, Dept. of Electrical Engineering, Linköping University, Sweden, 1987.

[23] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Conference On Computer Vision And Pattern Recognition*, pages 511–518, 2001.

[24] J. Žára, B. Beneš, J. Sochor, and P. Felkel. *Moderní počítačová grafika*. Computer Press, Brno, 2004.