

An Image Registration Algorithm Based on Cylindrical Prototype Model

Joong-Jae Lee, Gye-Young Kim, and Hyung-Il Choi

Soongsil University, 1-1, Sangdo-5 Dong, Dong-Jak Ku, Seoul, Korea
ljhhop@vision.soongsil.ac.kr, {gykim,hic}@computing.soongsil.ac.kr

Abstract. We propose an image registration algorithm based on cylindrical prototype model to generate a face texture for a realistic 3D face model. This is a block matching algorithm which aligns 2D images of a 3D cylindrical model. While matching blocks it doesn't use same sized blocks but variable sized blocks with considering a curvature of 3D model. And we make a texture of aligned images using an image mosaic technique. For this purpose, we stitch them with assigning linear weights according to the overlapped region and using the cross-dissolve technique.

1 Introduction

The realistic 3D modeling provides user with more friendly interface and is also very useful in virtual reality, broadcasting, video teleconferencing, interface agent and so on. We can obtain the 3D model of human through the hardware equipment like 3D scanner [1]. It is very easy and simple solution, but it is quite an expense. A number of techniques have been developed for resolving this problem. One way is to use front and side views of face, which creates a model as texture mapping after modifying a generic model [2][3]. However, it only uses two face images so it is less realistic at other views except front and side views. Another way is to use face images of four or eight views, which produces a face texture using panoramic techniques and mapping [4]. Through it, we may have the natural result. But it requires images photographed in exact angle, because it uses only image stitching in panoramic techniques. Mapping to model with the images that weren't photographed in accurate angles makes an unnatural texture, and also we wouldn't create a realistic 3D model. To resolve such situations, we propose the image registration algorithm based on cylindrical prototype model. It enables us to align images precisely using the correlation between them, even though exact view angles would not be met when a face is photographed. And the natural face texture is generated by stitching the aligned images with the cross-dissolving. For this purpose, we adjust weights according to the overlapped region. So, more realistic 3D face model can be created.

2 Image registration with variable sized blocks

The process of block matching is to partition each of two images to same sized blocks. And then it finds a candidate block, within a search area in another

image, which is most similar to the source block in one image, according to a predetermined criterion [5]. But the curvature of 3D cylindrical model causes surface patches of same size to be projected on 2D image as blocks of different size. While matching blocks, the proposed algorithm doesn't use same sized blocks but variable sized blocks with considering a curvature of 3D model. We denote block widths of 2D image of 3D cylindrical prototype model as in (1)

$$\begin{cases} BW = k_n \\ k_0 = 2R \sin\left(\frac{\theta_{Div}}{2}\right) \\ k_n = k_0 \cos\left[\left(|n| - \frac{1}{2}\right) \times \theta_{Div}\right] \end{cases} \quad (1)$$

where $0 < \theta_{Div} < \frac{\pi}{4}, 1 \leq |n| \leq \frac{\pi}{2\theta}$

where θ_{Div} denotes the division angle partitioning a 3D model into same portions, R denotes a radius of model. And k_0 denotes the base length of an inscribed triangle when the 3D model is divided by θ_{Div} , k_n except k_0 is the size of each block projected on 2D image. In (1), n represents column index of blocks projected on the same column of 2D image. It is designed to be positive for right hand side block and negative for left hand side block. It gets larger magnitude as its block lies closer to the left and right border of 2D image. Fig. 1 shows us the result of producing variable sized blocks. BW becomes small when the absolute value of index n grows.

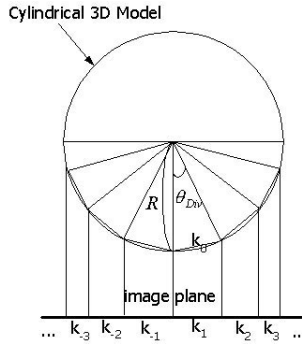


Fig. 1. Variable sized blocks depending on the curvature of cylindrical model

Fig. 2 shows us the different block widths of the proposed approach against the same block widths of the typical approach. In particular, we can see the variable sized blocks depending on the curvature of model. We partition an image to variable sized blocks to perform block matching. That is, we changeably adjust size of candidate blocks to be matched against the source block.

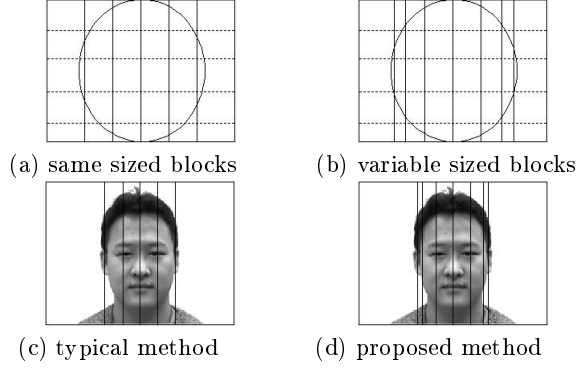
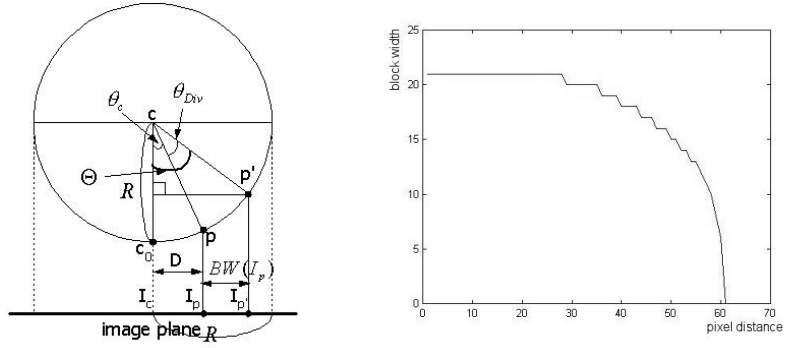


Fig. 2. Comparison of block size as block partitioning with front face view

Fig. 3(a) illustrates how to determine a block size of $BW(I_p)$ and the closed form of its equation is shown in (2). In (2), $BW(I_p)$ denotes the length between I_p and $I_{p'}$. It is obtained by looking for $I_{p'}$ which is projected on 2D image from p' on 3D model. Here, p' is separated from p on 3D model by division angle θ_{Div} .



(a) determination of block width (b) result ($R = 62, \theta_{Div} = 20^\circ$)

Fig. 3. Variable adjustment of block width

$$\begin{aligned}
 BW(I_p) &= |I_{p'} - I_p| = |I_{p'} - I_c| - |I_p - I_c| \\
 &= R \sin \Theta - D \\
 &= R \sin(\theta_{Div} + \theta_c) - D \\
 \text{where } \theta_c &= \sin^{-1}\left(\frac{D}{R}\right), \quad D = |I_p - I_c|
 \end{aligned} \tag{2}$$

Fig. 3(b) shows the block size produced when R is 62 pixels and θ_{Div} is 20° , we can note that it is adjusted variable depending on the curvature of 3D model. In fact, the block size gets smaller, as it becomes more distant from center. We accomplish the block matching using the color and texture feature that can reflect correlation between blocks. As for color feature, we use YIQ color model that is more stable than RGB color model in alterations of luminance [6]. Besides, Gabor Wavelet kernel is used to obtain the texture characteristics of a block [7]. We define a matching metric as in (3). In (3), $MF(i, j; u, v)$ denotes matching similarity considering the color and texture feature and it has a value between 0 and 1. The argument (i, j) denotes the position of source block, and (u, v) denotes the displacement between the source block and its matching candidate. The α and β are weighting factors that control the importance of related terms. We assign 0.4 to α and 0.6 to β experimentally.

$$\begin{aligned} \mathbf{d}(i, j) &= (u^*, v^*) \text{ where } MF(i, j; u, v) \text{ is maximized} \\ MF(i, j; u, v) &= 1 - [\alpha \cdot M_{color}(i, j; u, v) + \beta \cdot M_{texture}(i, j; u, v)] \\ \text{where } 0 < \alpha, \beta < 1, \alpha + \beta &= 1, 0 \leq MF(i, j; u, v) \leq 1 \end{aligned} \quad (3)$$

In (3), the displacement of (u^*, v^*) between the source block and its best matching candidate is used for aligning two images. We compute the matching similarity each of candidate blocks against the source block, and choose the block that has the highest similarity as the best matched block. We then determine the relative location of the matched block to the source block as the displacement vector (u^*, v^*) . The u^* denotes a displacement in x axis and v^* denotes a displacement in y axis.

3 Image Mosaiking

We utilize an image mosaiking for creating a texture of 3D model. It is implemented by cross-dissolving aligned images. Since the cross-dissolving helps to smoothly blend the images together, it is mainly used for morphing in computer vision and graphics [8]. Our approach can be formalized as in (4). In (4), T denotes a texture image, I_1 and I_2 depicts aligned 2D images. P_i denotes i th pixel, w_1 and w_2 denotes weights for I_1 and I_2 to stitch them.

$$T = \begin{cases} I_1, & \text{if } Visible(P_i, I_1) \text{ and } \neg Visible(P_i, I_2) \\ I_2, & \text{if } \neg Visible(P_i, I_1) \text{ and } Visible(P_i, I_2) \\ w_1 \cdot I_1 + w_2 \cdot I_2, & \text{if } Visible(P_i, I_1) \text{ and } Visible(P_i, I_2) \end{cases} \quad (4)$$

Especially for overlapped region, images are stitched by assigning weights w_1 and w_2 to I_1 and I_2 , respectively, according to rules defined in (5). Fig. 4 shows three types of overlapping occurred when aligning two images with displacement of D .

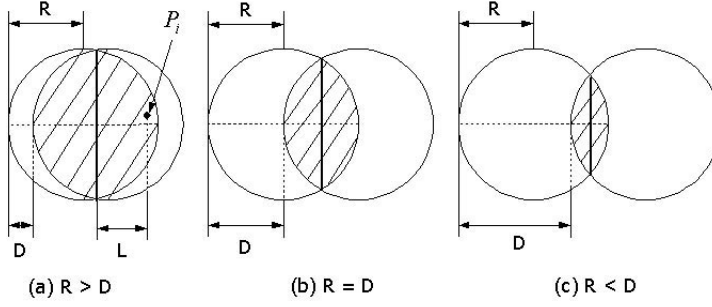


Fig. 4. Three types of overlaps occurring when image stitching

Weights are determined as in (5) by considering the amount of overlap and displacement. In (5), R is radius of a model, D is a displacement between two aligned images and L denotes the distance from the center of an overlapped region to any pixel P_i as denoted in Fig. 4(a).

$$w_1 = \begin{cases} \frac{2L}{2R-D}, & \text{if } D > R \text{ or } D < R \\ \frac{2L}{R}, & \text{if } D = R \end{cases} \quad (5)$$

$$w_2 = 1 - w_1$$

where $0 < w_1, w_2 \leq 1, w_1 + w_2 = 1$

4 Experimental results and discussions

To evaluate the proposed approach, we have used face images photographed in each of four and eight directions. Fig. 5 shows the result of generated texture from those images. To compare the performance of our approach with typical method, we experiment with face images having photographic errors of about 10° of viewing angle as shown in Fig. 6.



Fig. 5. Results of generated texture with face images

As we may notice, both approaches may obtain satisfactory results for a texture of nonoverlapping regions. However, for a texture of overlapping regions, elliptical dotted areas, our method shows superior results as in Fig.7. In typical approach, an eye region is blurred and two ears are revealed much like the ghosts phenomenon. Our approach, however, shows the perfect alignment and does not make any problems in terms of texture.

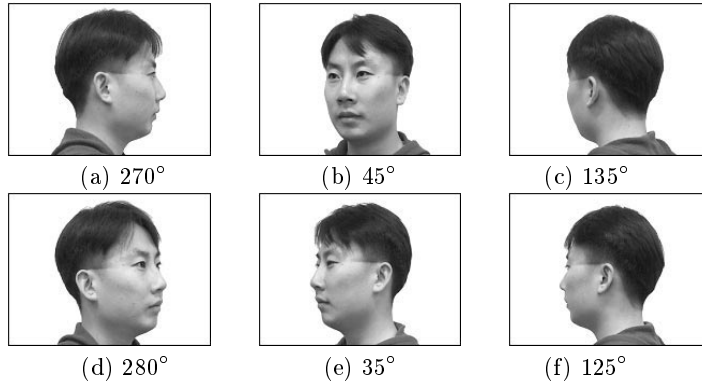


Fig. 6. Face images photographed in exact and inexact directions

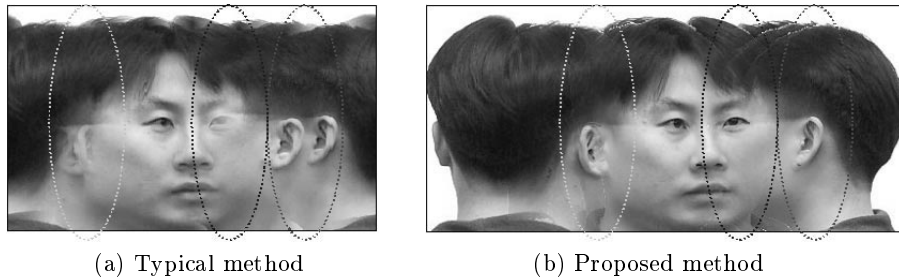


Fig. 7. Comparison of results produced with the photographic errors

In this paper, we have proposed an image registration algorithm based on cylindrical prototype model which could align the 2D photographic images of 3D cylindrical prototype model. We have showed that our approach can be used to obtain a face texture using face images. The main concept of our approach is to align input image using the correlation between images. More specifically, when block matching, it uses variable sized blocks with considering a curvature of 3D model instead of same sized blocks. The proposed approach can solve the

problems like the blurring effect and ghosts phenomenon, occurred when a texture is generated with images photographed at inaccurate angles.

Acknowledgement

This work was partially supported by the KOSEF through the AITrc and BK21 program (E-0075)

References

1. Yuencheng Lee, Demetri Terzopoulos, and Keith Waters.: Realistic Modeling for Facial Animation. Proceedings of SIGGRAPH 95. In *Computer Graphics* (1995) 55–62
2. Takaaki Akimoto, Yasuhito Suenaga.: 3D Facial Model Creation Using Generic Model and Side View of Face. In *IEICE TRANS. On INF and SYST.*, Vol E75-D, No.2 (1992): 191–197
3. Takaaki Akimoto, Yasuhito Suenaga.: Automatic Creation of 3D Facial Models. In *IEEE Computer Graphics and Applications* (1993) 16–22
4. Han Tae-Woo.: 3D face modeling system for realistic facial expression animation. MD Thesis, Department of Computer Science, Korea Advanced Institute of Science Technology (1998)
5. Kyoung Won Lim, Byung Cheol Song, and Jong Beom Ra.: Fast Hierarchical Block Matching algorithm Utilizing Spatial Motion Vector Correlation. Proceedings of *Visual Communications and Image processing*, Vol. 3024 (1997) 284–291
6. Yu-Ich Ohta, Takep Kanade, Toshiyuki Sakai.: Color Information for Region Segmentation. In *Comuter Graphics and Image Processing*, Vol. 13 (1980) 222–241
7. W. Y. Ma and B. S. Manjunath.: Texture Features and Learning Similarity. *IEEE International Conference on Computer Vision and Pattern Recognition*, San Francisco, CA (1996)
8. G. Wolberg.: Image morphing: a survey. *The Visual Computer*, 14(8/9) (1998) 360–372