Seamless Processing for Three Dimension Color Object Models

Chia-Lun Chen, Chao-Hua Wen, and Wen-Hsin Chan
Opto-Electronics & Systems Laboratories, Industrial Technology Research Institute
Hsinchu, Taiwan, R.O.C.

Abstract

3D object models are becoming widely used for computer graphic rendering or virtual environments. For rendering a 3D object, an object model is constructed from 3D spatial data by 3D system and then combined with several color images from different viewpoint using an optimal texture mapping. However, the discontinuity phenomena are generated somewhere between the merged images. A method of seamless processing for merging color textures of 3D object models is proposed. By transferring the color images from RGB color space to a different one, it separates the brightness from the color information. The seams are smoothed based on the brightness, and the result images are transferred back to the RGB color space. It keeps the color unchanged and maintains the detailed information of textures.

Introduction

3D (three-dimension) object models are becoming widely used in computer graphics and virtual environments. To construct an image based 3D model of an object, a 3D geometry model is obtained (such as created by laser scanning system) and incorporated with 2D images of different view angles by using texture mapping. A 3D graphics system allows a user to output or display images showing any side or surface of the object from any viewpoint. However, the seaming boundaries generated by merging 2D images are usually conspicuous. To solve this problem, some texture mapping methods were proposed to merge the images to a good alignment as possible, and some methods were used also to blend the images to smooth the boundary edges. Nevertheless, the textures around the seaming area are always blurred, and the real color of the images is deviated also.

It was realized that the seaming problem is caused by the non-uniformity of illumination when capturing the object images. Therefore, the images with uniform lightness are expected. A method being used often is to take the images under uniform light source, but it is difficult to set up the apparatus and is time consuming to make a 3D object model. Besides, if the light source is given when scanning a 3D object, all surface normals of the meshes are known, and then shading model could be used to remove the lighting effect if the material of the object is obtained. However, the measurement of the material of the object is not easy. Therefore, if the brightness of the images could be processed simply by software, that would be more convenient and speedy.

The growing application of 3D object rendering urges the need of quick construction of 3D object models. The quality of the object model is demanding too. To fulfill this requirement, we propose a different approach to solve the seaming problem. To keep the original color of the images while solving the seaming problem, the method reported here implements the color transform from RGB color space to another one in order to separate the brightness from the color information, so that we can process the brightness independently. The brightness difference at a region relative to two images is computed as calibration parameter to preserve the detailed texture. The brightness of the adjacent images is compensated to minimize the processing area. The seaming boundaries are then processed to balance all images.

The related works are reviewed in section 2. The proposed seamless processing algorithm is described in the section 3. In section 4, the experimental results are reported. Finally, the conclusions are given in section 5.

Related Works

While making 3D object models, seams are generated when combining two or more images due to the lighting effect. This is particularly apparent for color images. Usually, we do not want to destroy the original textures and colors in the images. However, the seaming edge occurs at the warped images mapped to a 3D object model, the different display coordinates ((x, y, z) in 3D) make it very hard to use the usual planar image processing methods to solve the color difference problem occurring at the seams.

In order to achieve seamless image processing, some methods were proposed including hardware and software. Binh Pham and Glen Pringle solved this problem by computing a relation mapping from the registered overlapping regions of two adjacent images in an image sequence.[1] They selected representative colors in the overlapping regions of two adjacent images and use multiple regression to minimize color differences to obtain a correction matrix. Then the matrix was used to correct the color in one image so that it matches the adjacent images. In a word, they forced the color of two adjacent images to be consistent, so that the real color of the images was lost. The errors were large if two images differ significantly. Moreover, it cannot be applied to all images for solving the discontinuity problem effectively due to the difference of selected reference colors in their correspondent images.

Since a seam phenomenon is caused by lighting effect, one can take object images under uniform illumination setup or processes the images to remove the non-uniformity lightness. Rushmeier and Bernardini proposed the method for computing normals and colors from multiple sets of photometric data, which are consistent with each other and an underlying lower resolution mesh.[2] They used five light sources at different positions. The color images were taken with the five light sources turned on in sequence. The five images were then synthesized into a single corrected RGB image with a weighted average, thereby solving the inhomogeneous problem introduced from the lighting effect. Thus the discontinuous phenomena around seams had been diminished. However, the neighborhood average method still has to be used to solve the remaining discontinuity, which results in the blurs of images causing the loss of the reality. In addition, it is time consuming for repeating the process of taking five pictures at each view angles.

Besides the hardware solution, common technique such as linear or nonlinear weighted averaging performed by software is used to deal with seams at image boundaries. However, they all will blur the images at edges causing the loss of detailed textures. Therefore, they are not ideal for the merges of several images to form 3D images. A method on the U.S. Patent No. 6,271,847 warped the images and then blended the warped images used a weighted pyramid feathering scheme.[3] The process was employed that extended the traditional Laplacian pyramid blending algorithm to handle alpha masked images and took advantage of weight maps. The use of a weight map associated with each pixel input image to indicate how much each pixel should contribute to the final, blended texture map, made it possible to create seamless texture maps. The method works well as shown in its experiment results. However, it is likely to distort the color because it processes images within the RGB color space.

Some methods were proposed to solve the mismatched color in 2D panoramic image. They probably processed the alignment and matched color between images. However, those approaches were applied in two-dimension. It is more complicated in three-dimension for an object model.

Seamless Processing Algorithm

For rendering a 3D object, an object model is constructed from 3D spatial data by 3D scanning system and then combined with several color images from different viewpoint using an optimal texture mapping. However, the discontinuity phenomena are generated somewhere between the merged images. The discontinuity phenomena that happen as a result of the brightness difference at the merging boundaries are caused by inhomogeneous light sources. Figure 1 shows the brightness distribution versus horizontal position around the seaming boundary.

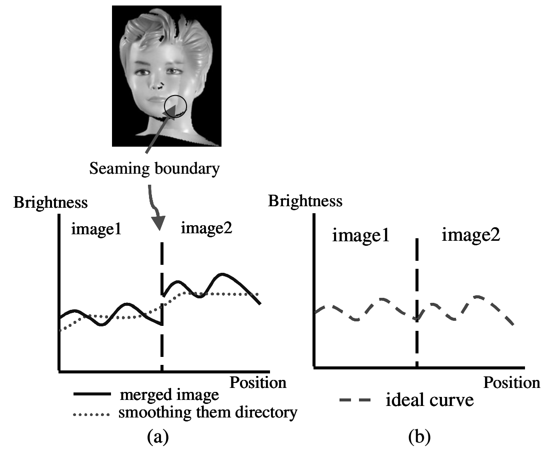


Figure 1. (a) Brightness distribution versus horizontal position around the seaming boundary of merged image (b) the ideal distribution curve

Ideally, both images have to be calibrated to have a continuous brightness variation (see Figure 1(b)) to obtain an optimized image merge without loosing the detailed contrast. However, people usually blend the image to smooth the curve shown as the dot line in Figure 1(a).

To keep the original colors and detailed textures in the images, the reported method separates the brightness information and uses it to correct, process images and then rebuild a 3D model. Also, the brightness difference is used to calibrate the image so that the detailed texture would not be destroyed.

The proposed method processes the seaming problem for two adjacent images in sequence. The scheme adopts a two-step correction method. First, the brightness of the whole image is calibrated. Secondly, the brightness differences at the overlapped boundaries are corrected. The procedure is depicted in Figure 2.

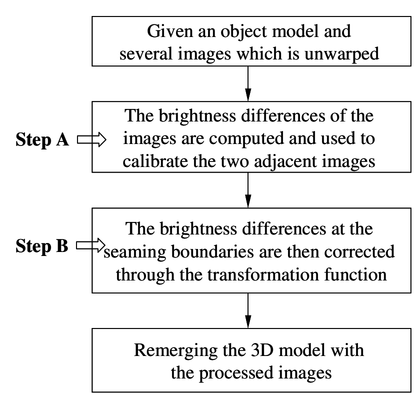


Figure 2. The procedure of the seamless processing algorithm

In step A, the brightness of the whole image is adjusted to reduce the processing width of the subsequent boundary brightness calibration. Since the average brightness difference of the two whole images is not apparent, we compute the average brightness difference from the overlapping regions of two adjacent images. The step 1 is performed as the procedure shown in Figure 3.

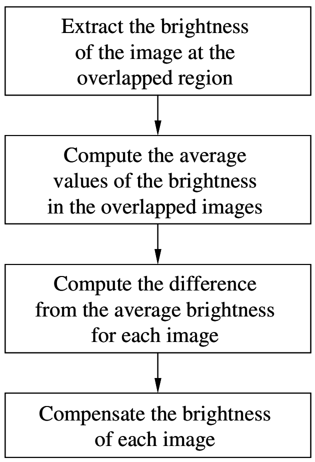


Figure 3. The detailed procedure of step 1

However, since the texture has been warped, merged and mapped onto a 3D model, the only information we know is spatial data of the overlapping region. That is, we should know how to locate the overlapping region of a 3D model onto each 2D image and extract the color data (R, G, B) in the area. The geometric model used in this paper is build with polygon meshes. We obtain the polygon meshes data at the overlapped region. The position coordinates of the vertices P1, P2, P3 of a mesh in the image are computed from their 3D information. The vertex coordinates are then used to draw a relative triangular mesh region in the 2D images, shown as Figure 4.

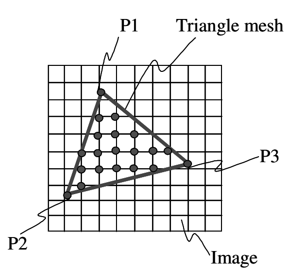


Figure 4. A schematic view of the location of a 3D triangle mesh located onto 2D image

The positions of all the points inside the relative meshes are computed for obtaining the color information. The color data are then transformed to a color space coordinate system that can single out the brightness information. This can prevent to change the real color of images. The color space here can be the CIELAB, CIELUV, and YCrCb coordinate systems. Finally, the brightness at the overlapped region is extracted. The system then computes to obtain an average brightness $\overline{L}_k$ of the overlapping region for image *k*.

$$\overline{L}_k = \frac{1}{A} \sum_{j=1}^{N} \sum_{i \in \text{mesh } j} L_{kji} \quad (1)$$

i denotes the point within mesh *j*, and *A* is count of the total points.

Then the average brightness difference σ_L of the two regions is

$$\sigma_L = \overline{L}_1 - \overline{L}_2 \quad (2)$$

The corrected brightness L'_{ki} of point *i* in image *k* is computed by

$$L'_{ki} = L_{ki} - \frac{1}{2}\sigma_L \quad (3)$$

The image data is then transformed to (R, G, B) color space and saved. This gives a preliminary brightness calibration for the whole image (see Figure 5).

Step B is the boundary calibration, which mainly calibrates the boundary brightness. The flowchart is shown as Figure 6.

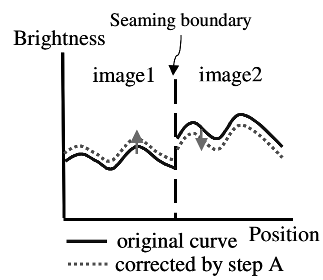


Figure 5. Brightness distribution around the seaming boundary of the image corrected by step A

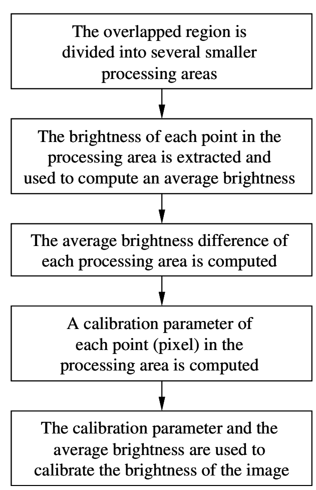


Figure 6. A flowchart of calibrating the brightness in the overlapped region

First, the overlapped region that has been processed with the overall image brightness calibration by step 1 is divided into several smaller processing areas as shown in Figure 7(a). The purpose of division is to enhance the image processing effect.

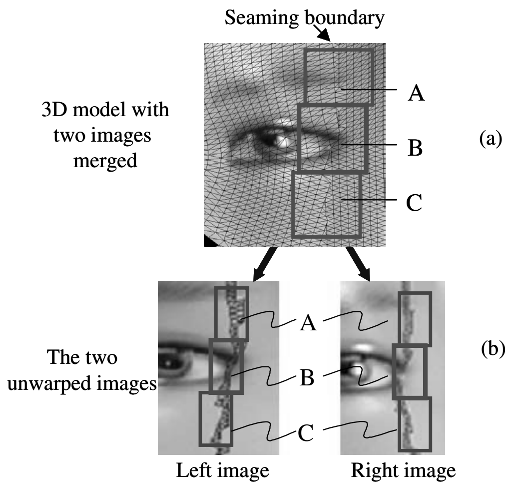


Figure 7. (a) A schematic view of processing a smaller area, (b) Areas mapped to the two images

The spatial data of the meshes in those areas is obtained. The information is then computed to get the mapping area on the two images as shown in Figure 7(b).

Second, the color information of each point in the small processing area is obtained. The brightness is extracted by transforming the color information (R, G, B) to another color space as described in the above. Hence we can calibrate the brightness independently without changing the color.

Besides, in order to keep the detailed texture, we cannot smooth the brightness directly. The only way to get a seamless merged image is to add or subtract a weighted parameter to the brightness of each point separately to get a smooth distribution on image. This will keep the contrast of the brightness variation unlike the one as a result of using neighborhood average method. Therefore, the detailed texture will not be destroyed.

Next, the average brightness $L_{k\ avg}$ of small processing region m in image k is given by

$$L_{k\,avg} = \frac{1}{A} \sum_{j=1}^{n} \sum_{i \in \text{mesh } j} L_{kji} \tag{4}$$

Where the L_{kji} is brightness of point i within mesh j in image k. The n is number of the meshes within the small region m and A is count of the total points. The mark 1, 2 is referred to the image 1 and image 2 respectively.

Compute the average brightness $L_{1\ avg}$ and $L_{2\ avg}$ for image 1 and image 2 individually. Then the average brightness difference σ_m of small processing region m between the image 1 and image 2 is given by

$$\sigma_m = L_{1avg} - L_{2avg} \tag{5}$$

Fourth, the calibration parameter $f(x,y)W$ of each point i (pixel) in the processing area is computed by

$$f(x, y) = 1 - \frac{d(x, y)}{D} \tag{6}$$

$$W = c_1\sigma_{m-1} + c_2\sigma_m + c_3\sigma_{m+1} \tag{7}$$

$f(x,y)$ is a position function of $d(x,y)$, where $d(x,y)$ is the distance from the processing pixel i to the seaming boundary, and the boundary and D is the width of the processing area. The processing width D is not large because we perform the brightness calibration for the whole image in the beginning in step A.

W is a weighting dependent on the average brightness difference σ. The parameters σ_m, σ_{m-1}, σ_{m+1} are the average brightness differences of the processing area B and the adjacent processing areas A and C respectively (see processing areas A, B, and C in Figure 7). The c1, c2, c3 are the corresponding position parameters. In words, the closer the point is to the adjacent areas or boundary, the less calibration there is.

Finally, the final brightness is obtained by subtracting the calibration parameter from the brightness obtained from the first calibration. The calibrated brightness L_i'' for point i within small processing area m is given by

$$L_i'' = L_i' - f(x, y)W \tag{8}$$

where the L_i' is the calibrated brightness by step A.

Since the calibration parameter and the average brightness are used to calibrate the brightness of the image, so that the brightness of the image border at the seaming boundary and of the seams are the same as that of the adjacent image. At the same time, we only use the brightness to perform calibration; the colors and detailed textures in the original image can be maintained. Figure 8 shows the brightness distribution curve as a result after processing by step B.

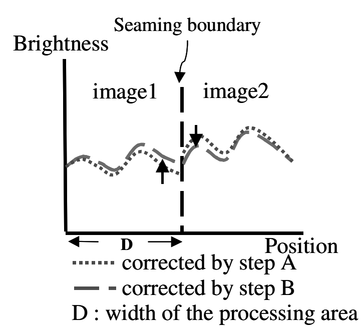


Figure 8. Brightness distribution around the seaming boundary of the image corrected by step B

Experimental Results

To demonstrate the performance of the proposed algorithm, a 3D object modeling system was used to implement the algorithm. The 3D modeling system we used here is developed by Industrial Technology Research Institute (ITRI). The system includes a laser scanning digitizer (Portable 3D scanner) with an in-house 3D surface modeling software (TriD).[4] The proposed algorithm was programmed as the procedure described in the previous section and was embodied in TriD system.

The modified 3D object modeling process is shown in Figure 9. And the circle lines the modified process out.

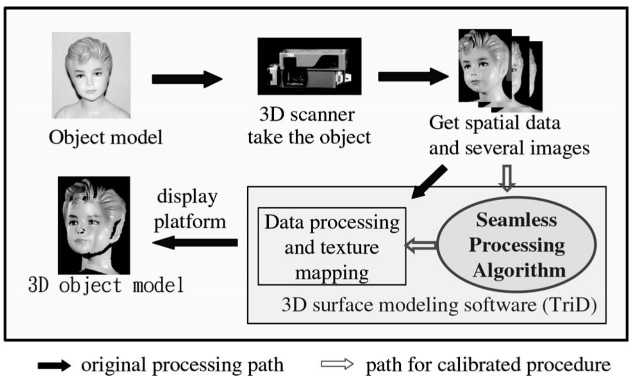


Figure 9. The process of building a 3D model

We used a 3D scanner to take an object model. Then we had a set of spatial data and several images taken in different viewpoint. The 3D software system, TriD, constructed object model from those 3D spatial data and used color data to render displays. Figure 10(a) shows the two adjacent viewpoint images are taken, include both spatial meshes and color data (in Fig.10(b)). The images were then merged and combined with the geometric shape using texture mapping as in Figure 10(c).

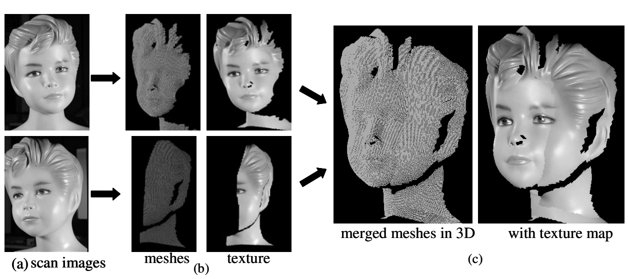


Figure 10. The process of rendering a 3D model

The 3D color object model was then built and displayed on the screen. However, the seaming boundaries appeared on the object surface evidently if the model processed without going through the calibrated procedure as the circle lined out in Figure 9.

We tried to solve this problem by using common technique. Figure 11 shows the result of blending two images at the overlapping area by using linear weighted average. The processing result was not good and the image was blurred also around the boundary.

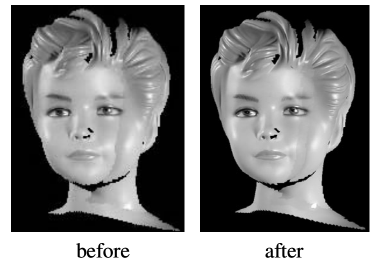


Figure 11. Processed by using linear weighted average.

We performed the seamless processing algorithm we proposed. Figure 12(a), (b) shows schematic views of the images before and after being processed. Figure 12(c), (d) is the brightness distribution curve of a horizontal line in image (a) and (b) respectively. In Figure 12(c), the curve-circled area is referred to the seaming boundary area. There is an obvious brightness variation occurring here. However, the curve is becoming smooth after the seamless process as shown in Figure 12(d). Figure 12(e) shows both the two curve before and after processing. The effect of the result is the same as in Figure 5 and Figure 8 as we expect. The brightness of the left image is reduced and the one of the right image is increased. The phenomenon is the effect of step A. By step B, the curve at circled area becomes smooth.

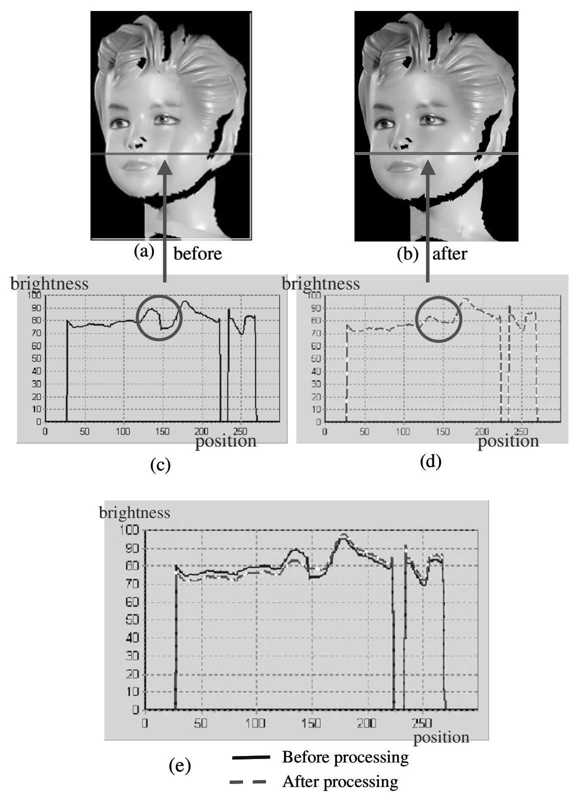
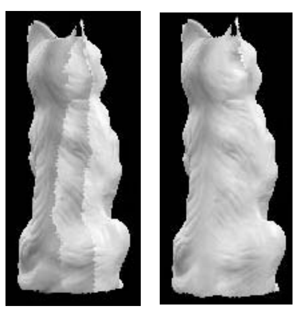


Figure 12. Processed by using seamless processing algorithm

Figure 13 shows another result of the girl head model after seamless processing. In Figure 14, the whole 3D cat model is built with merging of multiple images. There are two seaming boundaries in the cat model since the image is the last one to be merged on the model. It still shows a good performance doing the seamless processing. Since we did not take the top head image of the cat, there is no texture around the right ear of the cat. That is why it still shows like boundary effect around there.

before after

Figure 13. Processed by using seamless processing algorithm



before after

Figure 14. Merging of multiple 3D images using seamless processing algorithm

Conclusions

The experiment reported here demonstrates an evident improvement of the proposed algorithm. Moreover, the method has the following advantages:

(1) Under different color spaces, only the brightness is calibrated and the original colors of the image are not changed.

(2) The total brightness of two images is first calibrated to reduce the brightness difference at the seaming boundary. It can reduce the processing pixels and prevent too much distortion.

(3) The brightness difference at the overlapped region is taken as the input of a conversion function to calibrate the image near the overlapped region. This can keep the contrast of the image and so avoid a blurred image and effectively solve the discontinuity problem.

(4) The process is automatically executed by a program. Actually, the program is accomplished and compiled with the TriD graphic system. The method is simple, convenient and applicable to all images.

Although the above description uses 3D images as the example, the method can be applied to 2D images such as panoramic photography. Nevertheless, a method of image processing used for panoramic image may not be used directly for 3D models since it is more complicated in three-dimension. *Note: The proposed algorithm is patent pending.*

References

1. Binh Pham, Glen Pringle, Color Correction for an image Sequence, pp. 38-42, IEEE, May 1995.
2. H. Rushmeier, F. Bernardini, Computing Consistent Normals and Colors from Photometric Data, Proceedings of the Second International Conference on 3D Digital Imaging and Modeling - 3DIM'99, pages 99-108, Ottawa, Canada, October 4-8, 1999.
3. H-Y Shum, Richard S. Szelski, Inverse Texture Mapping Using Weighted Pyramid Blending and View-Dependent Weight Maps, US Patent 6271847 B1, August 7, 2001.
4. Wen-Jean Hsueh, 3D surface digitizing and modeling development at ITRI, Proceedings of SPIE, Vol. 4080, 2000.
5. M. Soucy, G. Godin, R. Baribeau, F. Blais, and M. Rioux, Sensors and Algorithms for The Construction of Digital 3-D Colour Models of Real Objects, IEEE, 1996.

Biography

Chia-Lun Chen obtained her master degree in electrical engineering from the University of Massachusetts Lowell in 1992. She is current working in Digital Color Department of Digital Imaging Technology Division at Opto-Electronics and Systems Laboratories of Industrial Technology Research Institute. Her interests include image quality, multi-display system calibration, multi-spectrum image, and color image processing.