A VIEW SYNTHESIS USING DISPARITY IMAGE

[1] *Chi-Yin Wang (王琦茵)*, [2] *Chiou-Shann Fuh (傅楸善)*

[1] Dept. of Communication Engineering,
National Taiwan University, Taipei
[2] Dept. of Computer Science,
National Taiwan University, Taipei
E-mail: r01942140@ntu.edu.tw

ABSTRACT

Binocular cues are perceived by rendering proper viewpoint images obtained at slightly different view angles. Since the number of viewpoints of the multi-view video is limited, 3D display devices should generate arbitrary viewpoint images using available adjacent view images. In this project, we will use the disparity image to create new views of a specific subject starting from two pictures taken from given point of views, and try to build a synthetic image as taken from a virtual camera placed in a different point. The proposed algorithm utilizes a Sparse Census Transform (SCT) and color image segmentation based on mean shift to obtain an illumination-invariant disparity image.

Keywords *Sparse Census Transform; View Synthesis; Disparity Image;*

1. INTRODUCTION

Human beings have the ability to see 3D images with two eyes. Right and left eyes provide two different images which are called stereo vision images. The disparity image refers to the slight pixel difference or motion between a pair of stereo vision images. The objects that are close to eyes will appear to jump a significant distance while objects further away will move very little between these two different images. That motion is the disparity. In a pair of stereo vision images, we can measure the motion in pixels for every point and make an intensity image out of the measurements.

In this paper, we propose a method to generate the view synthesis from the disparity image. The rest of this paper is organized as follows. In Section 2, we describe the disparity method which utilizes a Sparse Census Transform (SCT) and color segmentation. In Section 3, we describe the view synthesis method using disparity data and define the boundary noise and propose the removal method. Finally, this paper is completed with our results and conclusions presented in Sections 4 and 5.

2. PROPOSED METHOD

2.1. Stereo Matching Algorithm

The Sparse Census Transform (SCT), an area-based algorithm, is used for acquiring the disparity image. The Census Transform (CT) is a fast algorithm for some real-time stereo vision applications. The CT is defined as an ordered set of comparisons of pixel intensities in a certain mask. Therefore, it consists of a comparison function ξ, which is used to compare the center pixel intensity p_1 with the pixel intensity p_2 in the mask, as shown in Fig. 1(a). The formula can be given by

$$\xi(p_1, p_2) = \begin{cases} 1, & p_1 > p_2 \\ 0, & p_1 \le p_2 \end{cases}$$

The CT creates a bit string for each pixel in the image I, as shown in Fig. 2. The formula is expressed as

$$I_{Census} = \bigoplus_{i=-m}^{m} \bigoplus_{j=-n}^{n} \left(\xi\left(I(u,v), I(u+i, v+j) \right) \right)$$

where the $\oplus$ symbol denotes a chained bit string, and the Census mask size is *(2m+1)*× *(2n+1)* pixels. The processing time is directly proportional to the length of bit string.

The proposed algorithm uses the Sparse Census Transform (SCT) to implement the stereo matching. Fig. 1 shows the difference between the Sparse Census mask and Census mask. The SCT can under-sample the image and minimize processing time. If the SCT and the CT both have the same number of bit strings, the SCT will perform better than the CT. The computation time of the

Census-based algorithm is directly proportional to the Census mask size.

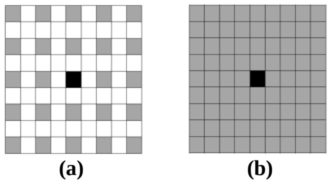

Fig. 1. Census mask.
(a) Sparse Census with *9×9* mask size.
(b) Census with *9×9* mask size.

Fig. 2.(a) shows the Sparse Census example with 9×9 mask size. Fig. 2(b) shows the length of each pixel's Sparse Census Transform which is $5\times 5\text{-}1=24$ bits. Fig. 2(c) shows an input image size is $m\times n$ pixels, and the result of the Sparse Census Transform with 9×9 mask is a 3D image of size $m\times n\times 24$.

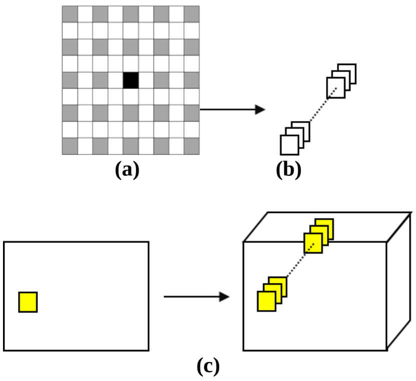

Fig. 2. The example of the Sparse Census Transform.
(a) Sparse Census Transform with 9×9 mask size.
(b) The bit string for each pixel.
(c) Sparse Census Transform of the image.

For the calculation of the matching costs of each disparity level d (from d_{start} to d_{end}), the cost function is defined as the Hamming distance over the bit strings. It can be written in

$$DSI(x,y) = Hamming\left(SCT_{Left}(x,y), SCT_{Right}(x,y)\right)$$
$$\forall d \in \{d_{Start}, d_{Start+1}, \cdots, d_{end}\}$$

where SCT_{Left} and SCT_{Right} are SCT results of right and left images respectively. The calculated costs are stored in a 3D Disparity Space Image (DSI), as shown in Fig. 3. The lowest cost is at disparity level d_{min}. The level values are used as integer disparity values. When the cost function in every pixel is obtained, it can be noticed that there is not only one disparity value with the lowest cost occasionally. We have to choose the correct disparity value from the disparity levels with the same lowest cost. When disparity levels *A* and *B* are both with the lowest cost, the costs of their neighbors have to be considered. Here, *y(A)* means the cost value at the disparity level *A* for a certain pixel. The correct disparity value is chosen by

$$d_{\min} = \begin{cases} level\ A, & \sum_{i=-1}^{1} y(A+i) < \sum_{i=-1}^{1} y(B+i) \\ level\ B, & else \end{cases}$$

The true disparities are usually not integer. To calculate the so-called subpixel disparities, a parabolic fitting is used. The best integer disparity and its neighbors are used to span the parabola, and its minimum gives the disparity in subpixel accuracy. The subpixel disparity for every pixel is estimated by using a parabola interpolation which passes through dmin and its neighbors. Then the subpixel disparity for one pixel can be calculated as follows

$$d_{sub} = d_{\min} + \frac{y(d_{\min}+1) - y(d_{\min}-1)}{2\left(2y(d_{\min}) - y(d_{\min}-1) - y(d_{\min}+1)\right)}$$

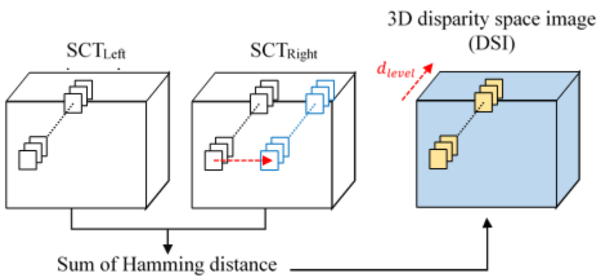

Fig. 3. Diagram for creating a 3D disparity space image (DSI). The DSI is a 3D image of size *width*×*height*×*disparity* range.

2.2 Image Processing

In this paper, a mode filter is used to smooth and denoise the disparity image. The median filter is a well-known image processing filter. It provides a way to smooth an image while preserving edges. A similar non-linear filter with slightly different properties is the mode filter which uses a moving window to smooth an image. The value of the central pixel in the window is replaced by the value having the maximum number of occurrence inside the window, so the impulse noise and the incorrect matching points can be removed.

To preserve the edge information and denoise the disparity image, we apply the mean-shift color segmentation method to optimize the disparity image. We utilize the color segmentation algorithm on the stereo vision images to let the color image be made up of some labeling regions. For each region, the corresponding disparity values from the disparity image are found and assigned the median values of the pixels in the region. That means a disparity plane is assigned to each region in order to get a clean disparity image.

3. VIEW SYNTHESIS USING DISPARITY IMAGE

In this section, we explain the procedure of the view interpolation method. After getting a clean disparity image, we use the disparity images and images provided by eyes to get the view synthesis. The disparity value describes the distance between the camera and objects in the scene. Using this geometrical information, we can map corresponding pixels between different viewpoints. When we change the viewpoint, some background regions disappear or appear because of foreground objects. To make sure that background objects would not be occluded by foreground objects and result from holes, before the procedure of the view synthesis, we generate the level images of each disparity, where the level images are obtained by extracting the objects from reference view images with the same disparity value, and the unselected regions are transparent. Fig. 4. shows the procedure of the level images. The level imges consist of objects with different disparities in each level. Within the level images, the hole problems can be reduced and even can be ignored. Then the level images for each disparity level can be calculated as follows

$$L(d) = \begin{cases} I_{Left}(d) & \text{, if } I_{Left}(d) \bullet I_{Left}(d) = 1 \\ I_{Left}(d) + I_{Right}(d) & \text{, if } I_{Left}(d) \bullet I_{Left}(d) = 0 \end{cases}$$
$$\forall d \in \{d_{Start}, d_{Start+1}, \cdots, d_{end}\}$$

where the $\bullet$ symbol implements logical conjunction; and I_{Left} and I_{Right} are left and right images respectively; and the disparity level d are used as integer disparity values by the rounding operation.

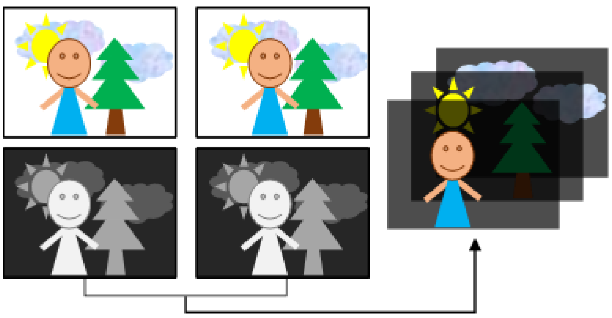

Fig. 4. The example of the level images.
The number of the level images is the maximum of *disparity.*

Once we obtain the level images, we can map all level images corresponding to the target view by exploiting the reference disparity image. Therefore, the level image will shift less if the disparity level d is small and will shift more obviously if the disparity level d is large. We superimpose all level images onto themselves in accordance with the disparity level to synthesize a virtual image.

This process induces a few extremely small holes, but these holes are imperceptible to our eyes. These small holes are filled using the average of our two reference view images. This trick reduces effects of tiny holes generated by the rounding operation during viewpoint shifting.

4. EXPERIMENTAL RESULT

The proposed method has been simulated at Matlab7 on system with 2.20GHz i3-2330M processor with 4.00GB RAM machine. The average computation time is about 40-45 seconds. Fig.5. shows the results of the disparity images using Sparse Census Transform, the mode filter and the mean-shift color segmentation method. Fig.6. shows the calculation results of the structural similarity (SSIM) method between our disparity image and the ground truth image, where the darker regions in the SSIM result images are the pixels with different values and white regions are the pixels with almost no difference. The value of the SSIM result will approximate to 1 if two images are totally the same. Fig. 7. shows the synthesized images and their holes areas between three different viewpoints. We can find that the holes in white color of each synthesized image are not obvious, so they are easy to be removed.

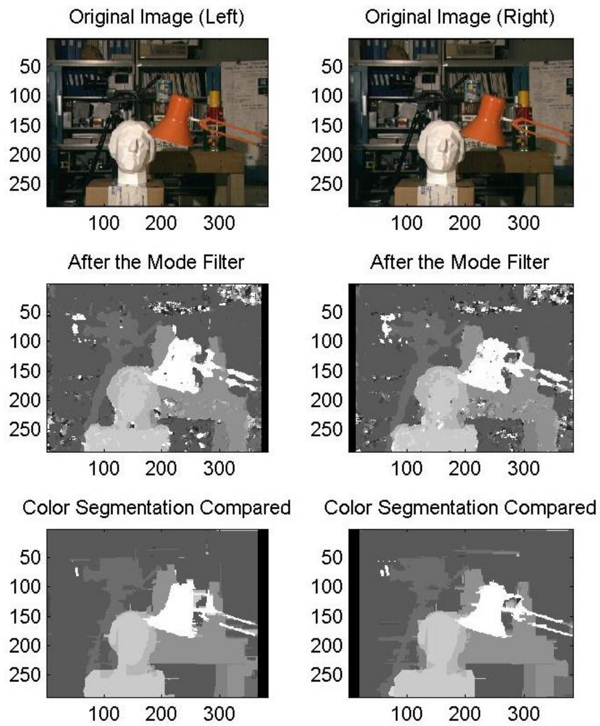

Fig. 5. Simulation results of Middlebury stereo datasets.
Left and right images are respectively in the left and rigth column.
From top to down: original view images, the disparity images of the *9×9* Sparse Census Transform and *3×3* mode filter, the disparity images (the mean-shift color segmentation method are applied for the Sparse Census Transform)

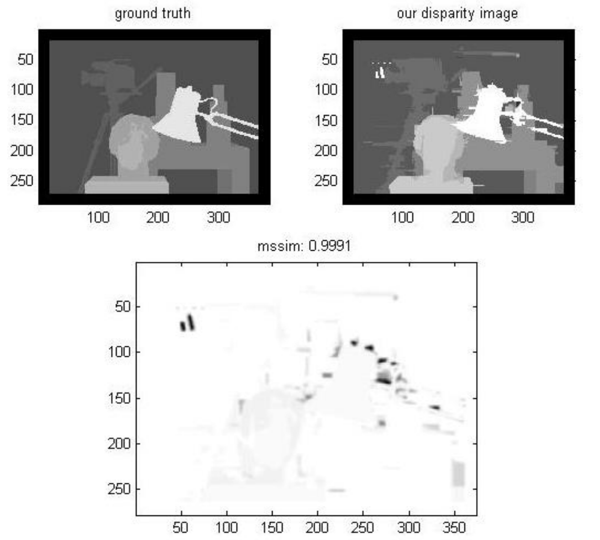

Fig. 6. The calculation results of the structural similarity (SSIM) method between the disparity image and the ground truth image.

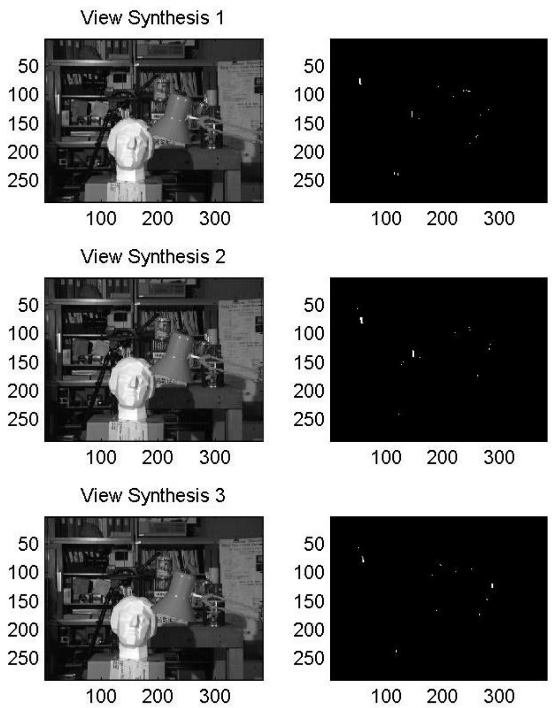

Fig. 7. First column shows the synthesized images filled using the average of two reference view images. Second column shows their holes areas in white color.

5. CONCLUSION

This paper has described the procedure of the view synthesis method which generates intermediate view images using disparity information. The results of the disparity images and the mean-shift color segmentation method have a great influence on the size of hole regions, and result from the quality of view synthesis images. Following these steps, we can choose a set of proper parameters to get the good synthesized images.

6. REFERENCES

[1] M. Humenberger, C. Zinner, M. Webera, W. Kubingera, and M. Vincze, "A Fast Stereo Matching Algorithm Suitable for Embedded Real-Time Systems," *Computer Vision and Image Understanding*, Vol. 114, Issue 11, pp. 1180-1202, 2010.

[2] S.-C. Pei and Y.-Y. Wang, "Census-based Vision for Auditory Depth Images and Speech Navigation of Visually Impaired Users," *IEEE Transactions on Consumer Electronics*, Vol. 57, Issue 4, pp. 1883-1890, 2011.

[3] A. Klaus, M. Sormann, and K. Karner, "Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure," *Proceedings of International Conference on Pattern Recognition*, Hong Kong, Vol. 3, pp. 15-18, 2006.