Abstract In the research of robot vision, perspective display often needs to be generated from omnidirectional images. In this paper, we represent 360-degree immersive panorama by SCVT (Spherical Centroidal Voronoi Tessellation) image, referred to spherical bubble, and propose a method of quickly generating the perspective display from spherical bubble according to users’ view direction and zoom-in/out operation. This process is done by employing the adjacent cues between neighboring cells and the pyramid data structure of spherical bubble. The experimental results are also presented to show the effectiveness of the proposed method.

Keywords: perspective display, 360-degree immersive panorama, spherical bubble

1. Introduction

In recent years, we have seen a growing interest in omnidirectional image sensors with wide field of view for robotic systems. They have been popularly used in many applications such as motion estimation [1], visual odometry [2], and navigation [3] of a mobile robot. Though omnidirectional cameras enable robots to gain more valuable information from the environment, the acquired images are not friendly to humans because of large distortion. In order to overcome the deficiency, perspective display, which is much similar to that of humans’ vision, often needs to be generated, for example, in the application like robot control, humanoid robot, especially for constructing the interfaces of robot that focused on providing users with the most current information as if they can see by themselves. In addition, some algorithms based on conventional cameras may be adapted directly to the perspective images.

Generally, an omnidirectional image can be regarded as bubbles, which mean a 360-degree panorama without considering the specific visual sensors. Thus, the problem of getting perspective display from omnidirectional cameras can be redefined as getting that from bubbles. It is a basic problem for the research of robot vision. Such processing is usually done by hardware in practice for efficiency.

Considering the isotropy of bubbles, a natural representation of bubbles is spherical map, or spherical image. Moreover, the cells, i.e., pixels, of spherical bubbles should be as uniform as possible so that the isotropy of the sphere around any cell point is preserved as well as possible.

In general, a spherical image is based on the successive spherical model (Fig. 1(a)). However, it has high cost of mapping the spherical point to the omnidirectional image pixel because of mass non-linear calculation. It is not suitable for the usage of robot in our case. Here, we represent bubbles, by SCVT (Spherical Centroidal Voronoi Tessellation) images which are called spherical bubbles in this paper and propose a method to generate perspective display swiftly more than before.

The rest of this paper is organized as follows. The related research is introduced in section 2. The method of fast generation of perspective displays is described in section 3. The experimental result is presented in section 4. Finally, we draw a conclusion in the last section.

Fig.1 Perspective display model: (a) Perspective display based on the successive spherical model, (b) Perspective display based on SCVT map

2. Related Research

2.1 Research about bubbles

Bubbles are widely used in many fields. A famous one refers to the street-level images in Google Street View [4]. Adding street-level images to traditional line-segment maps enhances the reality of on-line maps dramatically. Users are able to visit cities on the internet by navigating between bubbles.
Bubbles are also regarded as environment maps in computer graphics. There are some forms of bubbles with different features and different applications. A common one is called a cubic environment map [5], which is stitched from the images captured by multiple cameras or lenses with overlapping field of view [6]. A cubic environment map consists of six perspective images which correspond to the six planes of a cube with the view point at the center of the cube [7], as shown in Fig. 2.

Let the distance of the center of the cube from the square plane be 1. Then, the distance from the center to the corner point becomes \( \sqrt{3} \). The sampling rates for the directions, which is defined as the ratio of the solid angle between the maximum (the central pixel of the square plane) and the minimum (the pixel at the corner of the square plane), differ from a factor of \( \sqrt{3} \).

Fig.2 Sampling rate of cubic environment map for the directions

Besides the cubic map, spherical environment map (which is different from the SCVT map proposed in this paper) [8], paraboloid map [9] and latitude-longitude map [10] have been proposed.

The spherical environment map [8] is based on the simple analogy of a small, perfectly mirroring ball; the image that an orthographic camera sees when looking at this ball is the environment map. However, for the spherical environment map, the sampling rate of this map is maximal for directions opposing the viewing direction, and goes towards zero for directions close to the viewing direction. Moreover, there is a singularity in the viewing direction because all points where the viewing vector is tangential to the sphere show the same point of the environment.

The paraboloid map consists of two paraboloids [9]. Although the paraboloid map can be reused for any given viewing direction, that is, it is view-independent, the sampling rate for the directions is still as great as 4. The latitude-longitude map, which is generated by dividing a sphere along the latitude and longitude [10], is proposed. Since the latitude-longitude map is severely over-sampled around the poles, the sampling rates for the directions differ greatly.

2.2 Research about SCVT

A SCVT image is known as its quasi-uniform property [11], which can be obtained by subdividing the icosahedrons iteratively, as shown in Fig. 3. Table 1 shows the corresponding sampling rate (the ratio of largest cell to smallest cell) of the SCVT maps for the directions. Although the sampling rate for the directions varies with the subdivision levels, it has a limit value, about 1.36. Therefore, spherical bubbles are a better representation for the isotropy with respect to direction, in comparison with the conventional environmental maps.

Perspective displays are generated from spherical bubbles according to users’ view direction and zoom-in/out operation. SCVT image can be represented as 2D array \( S(i,j) \), in computer [12], [13], as shown in Fig. 4(a). To generate perspective displays, the pixel of perspective displays \( P(x_p, y_p, f_p) \), must be determined from that of the SCVT images. While the mapping from pixel \((i, j)\), of SCVT
images to spherical coordinate, $(\theta, \phi)$, can be carried out quickly by using a look up table, $T((i, j), (\theta, \phi))$; the inverse mapping is troublesome. Fig. 4(b) shows the angle change of the pixels in a row of the SCVT image array of Fig. 4(a), where the curves correspond to the rows of SCVT image array. It indicates that the mapping relation between spherical angle $(\theta, \phi)$ and array number $(i, j)$ is nonlinear. Therefore, search is necessary in the inverse map as follows.

\[
\begin{bmatrix}
x_p \\
y_p \\
f_p
\end{bmatrix} \xrightarrow{\text{Search}} \begin{bmatrix} i_s(\theta) \\ j_s(\phi) \end{bmatrix}
\]

In the related research [12] and [13], given a spherical polar coordinate $(\theta, \phi)$, an initial position is first estimated in 2D array $S(i, j)$; then, the corresponding cell point is found by iteratively searching a local maximum among the estimated cell and its neighboring cells. That is to say, finding the corresponding pixel of perspective displays from spherical bubbles according to the spherical coordinates involves two search processes. One is to find the approximate location according to the average intervals of azimuth angle and polar angle between neighboring pixels, called average search, and the other is to find the nearest pixels according to the neighboring relations among pixels, called neighboring search.

Since users want an instant response when cameras move with a robot, the generation of perspective displays should be carried out as speedily as possible. We accelerate this processing by employing the adjacent cues between neighboring cells and the pyramid data structure of spherical bubble. This will be much helpful that users can get different sequential views from the camera in a moving robot by transformation between bubbles.

The SCVT map has the distinguished advantage over the conventional bubbles. Although SCVT maps have been proposed in related research [12] and [13], these studies focused on the algorithm of finding neighboring pixels. In comparison with them, in this paper we use SCVT map to represent bubbles. This paper has the following characteristics.

1. Generate perspective view from spherical bubble by employing the neighboring relation among pixels. As mentioned above, finding the corresponding pixel of perspective display from spherical bubble according to the spherical coordinates involves two search processes: average search and neighboring search. If the resolution of the perspective display and that of the spherical bubble are approximately the same, the neighbors of a pixel in the perspective display should correspond to the neighbors of the corresponding pixel in the spherical bubble. Thus, in this case, to generate the perspective display, we can omit the average search, and only carry out neighboring search except for the first one.

2. Use the pyramidal data structure of spherical bubble to cope with the change of resolution of perspective display. To generate a spherical bubble with approximately the same resolution as the perspective display, the pyramidal data structure of SCVT image from the original spherical bubble is used. To generate perspective display, its resolution is first computed. Then, the corresponding layer of the SCVT image is selected from the pyramidal data structure.

Using the above techniques, perspective display can be generated from spherical bubble with lower computation cost.

3. Generating Perspective Display from SCVT Map

3.1 Generation of perspective display by using neighboring relation

Assume that the resolution of the generated perspective display is approximately the same as that of the spherical bubble. Thus, if two pixels are adjacent to each other in perspective displays, they should also be adjacent in SCVT maps. That is, neighboring relation between pixels is preserved for both perspective display and spherical bubble. In this case, the perspective display can be generated simply from

Journal of Signal Processing, Vol. 18, No. 2, March 2014 113
spherical bubbles as follows.

1. For the first pixel at the top-left corner of the perspective display, compute its corresponding nearest pixels in the spherical bubble by average search and neighboring search.

2. For the next pixel to be generated, compute its corresponding nearest pixels in spherical bubble by starting from the known neighboring location merely with neighboring search.

Thus, the average search is necessarily carried out only for the first pixel.

3.2 Generation of perspective display by using pyramidal data structure of spherical bubble

In practice, the resolution of perspective display is changeable with users’ zoom-in/out operation. This means that the resolution of perspective display may be different from the original spherical bubble obtained from the captured images. Here, the pyramidal data structure of spherical bubble is used to cope with this problem so that the above proposed method of generating the perspective display using neighboring relation can be applied. In addition, the quality of perspective image is also promoted during the process of up-sampling. The detail information is given below.

The pyramidal data structure of spherical bubble corresponds to the subdivision of the initial icosahedron. The arrays of cell points for 0-level and 1-level subdivisions corresponding to Fig. 3(a) and (b) are shown in Fig. 5(a) and (b), respectively. The array of cell points for 2-level subdivision corresponding to Fig. 3(c) is shown in Fig. 4(a). The pyramidal data structure of spherical bubble is generated by the down-sampling or up-sampling of the original spherical bubble.

\[ i_L = 2, \quad j_L = 2 \]

To avoid the aliasing problem, the down-sampling is carried out by averaging the neighboring cells. Here, the neighboring cell search is carried out by the algorithm [13].

Fig. 6 shows the sketch of the down-sampling with averaging the neighboring cells. The hexagonal cells with red lines indicate those of \( S_{L-1}(i_{L-1}, j_{L-1}) \) while the cells with black line indicate those of \( S_{L}(i_{L-1}, j_{L-1}) \). Each cell of \( S_{L}(i_{L-1}, j_{L-1}) \) contains the entire corresponding cell of \( S_{L}(2i_{L-1}, 2j_{L-1}) \) computed in terms of Eqs. (2) and about half of the neighboring cells of \( S_{L}(2i_{L-1}, 2j_{L-1}) \). Thus, the pixel value of \( S_{L-1}(i_{L-1}, j_{L-1}) \) can be computed as follows.

\[
V^{L-1} = \frac{1}{N} \sum_{j=1}^{N} V_j^L + V_j^L
\]

Where \( V^{L-1} \) and \( V^L \) are the pixel value of \( S_{L-1}(i_{L-1}, j_{L-1}) \) and \( S_{L}(2i_{L-1}, 2j_{L-1}) \), respectively. \( V_j^L \) indicates the pixel value of the neighboring cells of \( S_{L}(2i_{L-1}, 2j_{L-1}) \). \( N \) is the number of the neighboring cells. \( N = 6 \), except for the twelve cell points of the icosahedron, where \( N = 5 \).
The pyramidal data structure of spherical bubble is generated respectively. The array of cell points for 2-level subdivisions corresponding to the subdivision of the initial icosahedron. The detail information is given below.

For the first pixel at the top-left corner of the th-level SCVT array shown in Eqs.(2), we use four neighboring cells of the pixel value of , is determined as follows. Each cell of th-level subdivision array is the number of neighboring relation between pixels, as mentioned in Section 3.1.

3.3 The process of generation of perspective display

Step1. Compute the resolution of the perspective display to be generated. The resolution of the perspective display is measured as the pixels per unit solid angle by mapping the perspective display to a unit sphere.

Step2. Select the level whose resolution is closest to the perspective display, from the pyramidal data structure of spherical bubble.

Step3. Generate the perspective display by using neighboring relation between pixels, as mentioned in Section 3.1.

4. Experiment

While the mapping from the pixel of perspective display to the closest pixels of spherical bubble is carried out by both the average search and neighboring search in the conventional method in [12], [13], the proposed method employs the neighboring relation combined with the pyramidal data structure of spherical bubble so that the mapping can be achieved only by neighboring search. In this section, we present the experimental results to show the performance of the proposed method in comparison with the conventional method. In our experiments, all the perspective images are generated by both two methods using closest pixel without any interpolation, unless it is expressly stated.

4.1 Performance of computational speed

We use the spherical image in Fig. 8(a) to test our method for computational processing speed. The spherical image is represented by a compact rectangular 2D array, SCVT as described in the research [13], which is generated from a pair of fisheye image as shown in Fig. 8 (b). The detailed information on the format can be found in the reference [13].

The original SCVT image is 640x256 pixels, corresponding to 7th-level subdivision of an icosahedron. The pyramidal data structure of the SCVT image generated by the proposed down-sampling and up-sampling algorithm is shown in Fig. 9.

At first the average of the solid angle per pixel, which stands for the resolution of image, is computed. Then, the corresponding level of the pyramidal data structure of the SCVT image is selected according to the computed average of the solid angle per pixel. Finally, the perspective display is generated from the selected level of the pyramidal data structure of the SCVT image. The algorithms are tested by a desktop PC with a 2.8-GHz Intel Pentium D central processing unit and 512-MB memory.
Since the resolution of perspective display is determined by the view size and the view field (see Fig. 1(b)), we can carry out the experiments with the both conditions varying, respectively.

First, the image size of the perspective display to be generated is fixed as 100×100 pixels. The field of view of the perspective display (perspective angle) is changed. Table 2 shows the computational time of some cases, which correspond to different level of subdivision of spherical bubble by the proposed method. Though only 7th-level subdivision of the original SCVT data structure is used in the conventional method, the speed varies heavily because average search is carried on for every pixel, of which the cost of computation is sensitive to the perspective angle. However, the proposed method performs well. The computational time is approximately shortened to the half. What is more, it holds a good property of computational stability, since the cost of doing neighboring search is almost the same for each cell. We also list the processing time of generating perspective image mapping from fisheye image directly in the reference [6] based on the widely used successive spherical model. It needs to solve the non-linear equations, and executes much slowly.

Then, we take another experiment by changing the image size with the perspective angle fixed to 40°. The computational time of some cases are shown in Table 3. As the image size is enlarged, computational cost of either method increases. However, our method is superior to other two algorithms.

Obviously, the results shows that the algorithms based on discrete spherical model, SCVT perform better than that based on successive spherical model greatly. Compared with conventional method, the computational time of the proposed method is much shorter, less than the half at best. Therefore, it can generate perspective images fast copings with changing resolution. In other words, it means we can get perspective display instantly from spherical bubble according to users’ view direction and zoom-in/out operation.

4.2 Performance of image quality

In order to test our algorithm in respect of image quality, an ordinary image in Fig. 10 is employed. We map it back to discrete spherical model to obtain a SCVT image, which is indicated in Fig. 11(a). Fig. 11(b) gives us an intuitionistic spherical view of the SCVT format by CG.

![Fig.9 The pyramidal data structure of the SCVT image generated referring to 8th, 7th, 6th, 5th, 4th level of subdivision of spherical bubble](image)

![Fig.10 The original image used in the experiment in section 4.2](image)
Let Fig. 11(a) correspond to 7th-level subdivision of an icosahedron. Then, we generate the pyramidal data structure by the proposed method, and resize the original image with interpolation to make the resolution correspond to different level of the pyramidal data structure, with perspective angles fixed. Regarding the resized images as the reference ones, we can evaluate the qualities of perspective displays generated by the conventional method and our new method. Note that it may bring in some absolute deviation between the reference images and the SCVT images because of resized operation and the process of discrete division. However, it is still meaningful to compare two methods based on the same standard.

The well-known MSE (Mean Square Error) and SNR (Signal to Noise Ratio) are used as evaluation criterion. SNR is expressed in the way of logarithmic decibel scale, which is defined in the reference [15] as follow:

$$ SNR = 10 \log_{10} \frac{S}{N} $$

(5)

Here, $S$ is the square sum of all pixels’ value in the perspective image generated, calculated by the formula:

$$ S = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} (r(i,j))^2 $$

(6)

Where $r(i,j)$ is the value of pixel $(i,j)$. And $N$ is the square err of all the corresponding pixels’ value between the generated image and the reference one.

$$ N = \sum_{i=0}^{h-1} \sum_{j=0}^{w-1} (r(i,j) - t(i,j))^2 $$

(7)

Where $t(i,j)$ refers to the value of pixel $(i,j)$ in the reference one. For the subjective evaluation, we also give SSIM results.

First, we calculate the resolution of the resized reference image, and test 8th level of subdivision of spherical bubble for our proposed method, referring to up-sampling. Fig.12 shows the results of perspective displays generated by the two algorithms. Obviously, the quality of (b) has better performance than (a). Table 4 lists the MSE, SNR and SSIM in comparison with the resized reference one. R, G, B in the second line indicates the RGB channels of image, respectively. From the table, we can see the quality of perspective display is improved apparently because of up-sampling operation.

Then, we turn to the case of 6th level of subdivision of the spherical bubble, referring to down-sampling. Though the computational time by our method is 33ms, greatly reduced from 78ms by the conventional method, the perspective image

---

Table 3 Comparison of computational time under the condition of the fixed perspective angle

| image size | level of pyramidal data structure | conventional method | proposed method | method based on successive spherical model |
|------------|----------------------------------|---------------------|----------------|------------------------------------------|
| 300×300    | 8th Level                        | 608 ms              | 312 ms         | 2388 ms                                   |
| 200×200    | 7th Level                        | 255 ms              | 125 ms         | 748 ms                                    |
| 100×100    | 6th Level                        | 62 ms               | 31 ms          | 203 ms                                    |

---

Fig. 12 The perspective displays generated using the original SCVT image data in Fig.11(a): (a) Conventional algorithm, (b) Proposed algorithm with 8th level of subdivision
gets deteriorated as shown in Fig.13. In such a case, if the quality is required, we can just use the original subdivision of icosahedron 7-th level instead of 6-th level. Then the image with the same quality as Fig. 13(a) can be obtained. The processing time is sacrificed comparing to 7-th level, however, it is still much faster than the conventional method, shortened from 78ms to 38ms. A good balance between the processing speed and the image quality can be made by adopting the pyramidal data structure skillfully according to our requirement.

![Fig.13 The perspective displays generated using the original SCVT image data in Fig.11: (a) Conventional algorithm, (b) Proposed algorithm with 6-th level of subdivision](image)

The results of experiments above show that our method performs well. The computational cost of generating perspective image is cut down and the quality is improved by up-sampling operation. For the pyramidal data structure of SCVT image, even though it is possible to do more discrete division in theory, we usually take it as far as 9-th level division in practice. The size of 9-th level SCVT image is 2560\times 1024, and it has enough resolution for the full view sensor often used.

### Table 4 Comparison of image quality with reference image

| Method            | MSE   | SNR(dB) | SSIM |
|-------------------|-------|---------|------|
|                   | B     | G      | R    | B    | G     | R    | B    | G     | R    |
| Conventional Method | 180.73| 180.17 | 185.76| 19.63| 19.59 | 19.72| 83.3%| 83.8% | 83.5%|
| Proposed Method    | 120.90| 121.54 | 129.01| 21.38| 21.30 | 21.30| 86.1%| 86.5% | 86.2%|

5. Conclusions

Spherical bubble, represented as SCVT image, is quasi-uniform, and has distinctive advantage in the sampling rate for the direction over other maps. Omnidirectional images can be regarded as spherical bubbles for processing. When omnidirectional cameras are used in robotic systems, it is necessary to frequently generate perspective display with changeable resolution from spherical bubble according to the users’ view direction and zoom-in/out operations.

In this paper, the adjacent cues among neighboring pixels combined with the pyramidal data structure of spherical bubble are employed to cut down the computational cost of generating perspective display. In addition, image quality is also improved when up-sampling is carried on. The experimental results are presented to show the effectiveness of the proposed methods. In future, we will incorporate more outstanding interpolation schemes instead of linear interpolations during the process of generating pyramidal structure. Besides that, we also intend to use spherical bubble for real-time tasks in a mobile robot.

### References

[1] J.C. Bazín, C. Demonceaux, P. Vasseur and I.S. Kweon: Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment, International Journal of Robotics Research, Vol. 31, Issue 1, pp. 63-81, 2012.

[2] D. Gutierrez-Gomez, L. Puig and J. J. Guerrero: Full scaled 3D visual odometry from a single wearable omnidirectional camera, Proc. IROS, pp. 4276-4281, 2012.

[3] R. Guo, L. Han and X. Cheng: Omni-directional vision for robot navigation in substation environments, Proc. ROBIO, pp. 1272-1275, 2009.

[4] L. Vincent: Taking online maps down to street level, Computer, Vol.40, No.12, pp.118-120, 2007.

[5] J. Kopf, B. Chen, R. Szeliski and M. Cohen: Street slide: browsing street level imagery, Proc. SIGGRAPH ’2010, Vol.29, Issue 4, No.96, 2010.

[6] S. Li: Monitoring around a vehicle by a spherical image sensor, IEEE Trans. on ITS, Vol. 7, No.4, pp.541-550, 2006.

[7] D. Voorhies and J. Foran: Reflection vector shading hardware, Proc. SIGGRAPH’94, pp.163-166, 1994.

[8] R.S. Wright, N. Haemel, G. Sellers and B. Lipchak: OpenGL Supper Bible (5th Edition), 2010.
sensor often used 2560 division in practice. The size of 9 division perspective requirement. We require a pyramidal data structure to speed up the processing and improve the image quality. The original SCVT image data in Fig. 11 is cut down to a resolution of 9×1024, and it has enough resolution for the full view.

For the icosahedron subdivision algorithm, we usually take it as far as the 7th level instead of the 6th level, as shown in Fig. 11. In such a case, if the 7th level is considered, we can just use the original perspective. However, when we consider the 6th level, the necessary to frequently generate perspective display with omnidirectional images can be reduced.

In this paper, the adjacent cues among neighboring pixels during the process of seeing is employed to combine with the pyramidal data structure of spherical omnidirectional images. The computational cost of in/out operations is reduced from spherical bubble according to the uniform, and has distinctive advantage in the sampling rate for independent environment map.

Hanchao Jia received the B.S. degree from Beijing University of Chemical Technology, China in 2008, and the M.S. degree from Tottori University, in 2011, Japan. He is currently working toward the Ph.D degree at the Graduate School of Engineering, Tottori University, Tottori, Japan.

Shigang Li received the B.S. degree in electrical engineering from Tsinghua University, China, in 1985. He then received the M.S. and Ph.D. degrees from Osaka University, Japan in 1990 and 1993, respectively. After receiving his Ph.D., he worked at Osaka University as a Research Associate from 1993 to 1995. He became an Associate Professor at the Faculty of Information Sciences, Hiroshima City University, Japan in 1995. In 2001, he moved to the Faculty of Engineering at Iwate University, Japan. Since 2007, he has been a Professor at the Faculty of Engineering, Tottori University, Japan. His research interests include computer/robot vision, intelligent transportation system and mixed reality.

(Received October 30, 2013; revised January 13, 2014)