VNU Journal of Science, Mathematics - Physics 23 (2007) 9-14

A process of building 3D models from images

Bui The Duy, Ma Thi Chau*

College of Technology, VNU
144 Xuan Thuy, Cau Giay, Hanoi, Vietnam

Received 9 July 2007; received in revised form 5 September 2007

Abstract. Recently, a number of new technologies to capture 3D data have been developed. The application potential of 3D models is enormous, such as, in education, entertainment, medicine, etc. In this paper, we present our work toward creating 3D model of free form objects from pair of images. We use the basic process of building 3D models proposed in ***Multiple View Geometry in computer vision*** by Richard Hartley and Andrew Zisserman which includes three main phases: Preprocessing, Matching, Depth Recovery.

1. Introduction

Nowadays, 3D model building is getting more and more attention from the research community. The rising attention is partly because of the technique's promising applications in such areas as architectural design, game produce, movie-postprocessing and so on. In order to have 3D models, the traditions are normally used, in which technicians use specialized equipments to get 3D information. The method costs a lot of expenses. In other approach, technicians use prior knowledge of objects to build the objects' 3D models manually and then apply the texture on these models. However, the methods require enormous manual effort. On the other hand, 3D models' qualities do not really meet the demand of reality, because subjective factors can affect the result. Recently, many researchers have been trying to find out robust as well as efficient methods to reconstruct 3D models. A new approach is investigated to reduce the human effort is to build 3D models automatically from images [1].

In this paper, we introduce our work of creating 3D model automatically from pair of images. Among many proposed methods we chose the framework proposed in [1] because of its completeness and practicality. The primary process described in [1] includes three main phases: Preprocessing, Matching, Depth Recovery. By combining and testing lots of related techniques and algorithms, we have introduced an effectively completed process which uses two images of an object as input and then automatically makes out the object's 3D model as output. The whole process consists of six steps in details: SUSAN corner extraction, SUSAN corner matching, F matrix computing, Polar rectification, dense matching, and triangulation and texturing. The approach is a promising feasible solution.

Section 2 gives an overview of the 3D model reconstruction and relevant techniques. We then propose our process by associating selected techniques in Section 3. We then show the experiments that we have done in Section 4.

* Corresponding author. Tel: 84-4-7547812
E-mail: chaumt@vnu.edu.vn

2. The 3D reconstruction process

The basic principle used in reconstructing 3D information is triangulation one [2]. In most techniques, a triangle is created between the object and two sensors. So, constructing 3D information needs at least two slightly different 2D images.

We follow the 3D reconstruction process introduced in [2], which is illustrated in Figure 1. The process consists of three main phases: Preprocessing, Matching, Depth Recovery. These steps will now be discussed in more details.

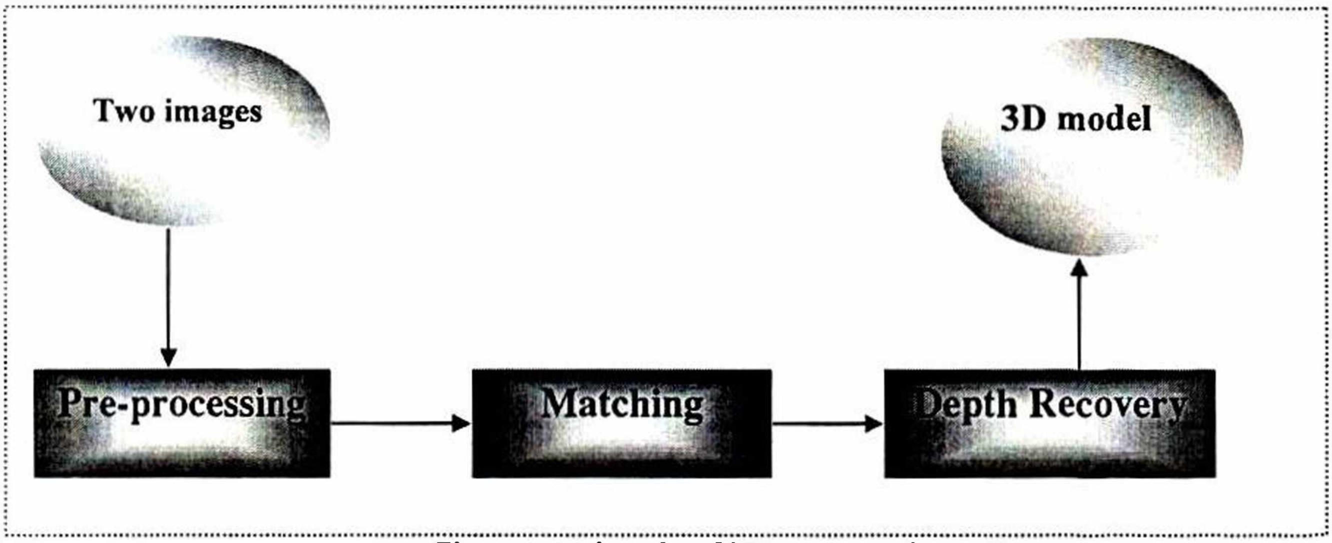


Figure 1. Main tasks of 3D reconstruction.

2.1. Preprocessing

The first step involves in relating two different images. In order to determine the geometric relationship between images, it requires number of corresponding feature points. Feature points are strongly different from its neighbors in the image so it can be matched uniquely with a corresponding point in another image. There are many kinds of feature points and methods of feature extraction published [3]. These corresponding feature points are then used to determine the geometry constraints between two images, which are mathematically expressed by the fundamental matrix.

2.2. Matching

At this step, input images are rectified according to the fundamental matrix computed by first step. Among the 3 main steps of the 3D reconstruction the matching step is extremely important. The above feature matching is only spare matching. But we need all image points are matched for having a real model. Image pairs are rectified so that epipolar lines coinciding with the image scan lines which reduces the correspondence search to a matching of the image points along each image scan-line. In rectification, pair of images is re-sampled so as to make imposing the two view geometry constraints simple. As a result, most image points in the first images are corresponding to image points in the second one.

2.3. Depth recovery

At this stage, by dense disparity matching determined in the second step, 3D information of all image points is computed. Triangulation principle and optimal triangulation method [2] are used to estimates the depth of all image points or raw 3D model. After that, one of original images is used to texture the raw model to have final 3D model.

3. A proposed process

In this section we motivate and present our completed process of 3D model building and its relation to others. The whole process is shown in figure 2.

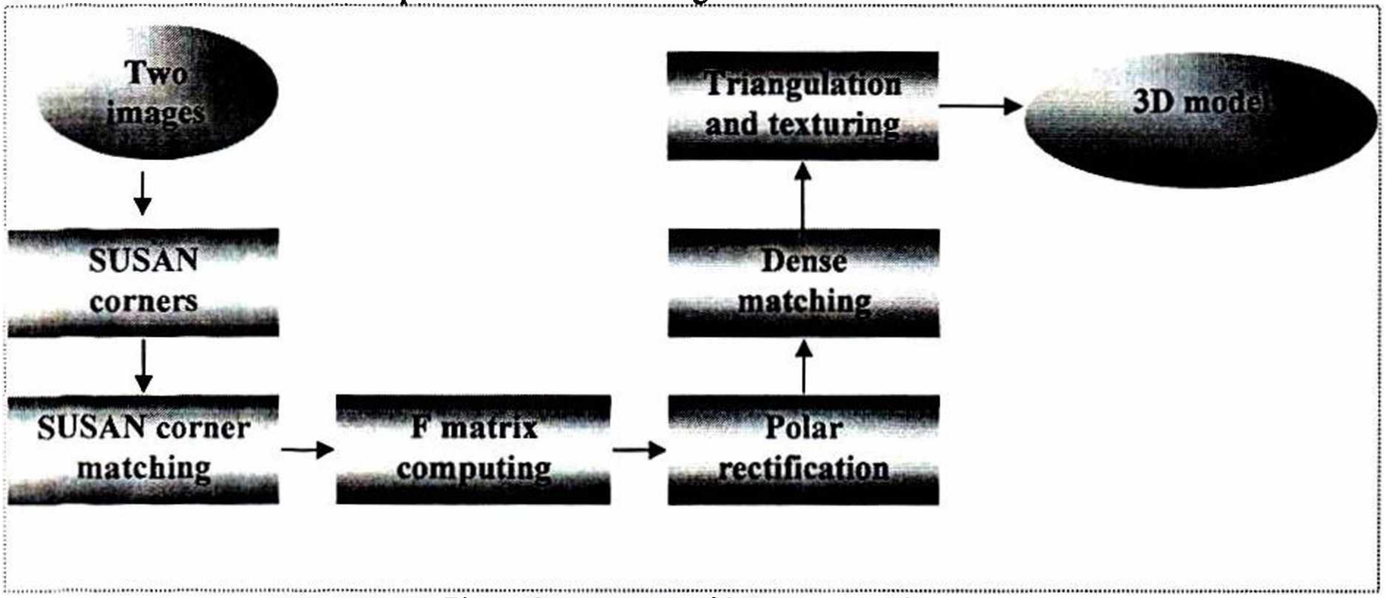


Figure 2. A process of 3D reconstruction.

3.1. SUSAN corner extraction

Feature can be classified as feature area, feature line or feature point. SUSAN (Smallest Univalue Segment Assimilating Nucleus) corners are feature points which are easily computed and effective in matching. To extract Susan corners, we use a circular mask. Its center is called nucleus. USAN (Univalue Segment Assimilating Nucleus) area is defined as an area including interested pixels which have the same brightness as nucleus's brightness. The shape of USAN areas conveys important information about the structure of the image in the region around the nucleus [4]. An algorithm proposed in [4] uses the information by comparing the brightness difference between the nucleus and its neighbors (pixels within the same circular mask) to extract SUSAN corners.

3.2. SUSAN corner matching

Given a point $c_1(u_1,v_1)$ (a SUSAN corner found in 3.1) in the first image, we use a correlation window of size $(2n+1) \times (2m+1)$, centered at this point. We then select a rectangular search area of size $(2d_u+1)\text{x}(2d_v+1)$ around this point in the second image (called $c_2(u_2,v_2)$), and perform a correlation operation on a given window between c_1 and c_2 lying within the search area in the second image. The correlation score, $S(c_1,c_2)$, is defined as:

$$S(c_1,c_2) = \frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m}\left[I_1(u_1+i,v_1+j)-\overline{I_1(u_1,v_1)}\right]\times\left[I_2(u_2+i,v_2+j)-\overline{I_2(u_2,v_2)}\right]}{(2n+1)(2m+1)\sqrt{\sigma^2(I_1)\times\sigma^2(I_2)}}$$

where as,

$$\overline{I_k(u,v)} = \sum_{i=-n}^{n}\sum_{j=-m}^{m} I_k(u+i, v+j)/(2n+1)(2m+1)\,,\ k=1,2.$$

$\sigma(I_k)$ is the standard deviation of the image I_k in the neighbourhood $(2n+1) \times (2m+1)$ of (u,v), which is given by:

$$\sigma(I_k) = \sqrt{\frac{\sum_{i=-n}^{n}\sum_{j=-m}^{m} I_k^2(u,v)}{(2n+1)(2m+1)} - \overline{I_k(u,v)}}$$

The score ranges from 1 down to -1 for two correlation windows which are similar or not. A constraint on the correlation score is then applied in order to select the most consistent matches: for a given pair of points to be considered as a candidate match, the correlation score must be higher than a given threshold. For each point in the first image, we thus have a set of candidate matches from the second one and vice versa. So we use some techniques known as relaxation techniques [5, 6] to resolve the matching ambiguities. The idea is to allow the candidate matches to reorganize themselves by propagating some constraints, such as continuity and uniqueness, through the neighborhood.

3.3. Fundamental matrix

Fundamental matrix 3 × 3 F expresses mathematically the geometry constraints between two images. Hartley [2] has pointed out RANSAC algorithm, a simple method, to compute F matrix. This matrix can be found by solving 8 linear equations. So, N samples of feature matching couples are used not only to compute F matrix but also to refine it.

3.4. Polar rectification

Rectification is an important step aim to save time and cost in matching by reducing the size of search area. Polar rectification transforms input images from Deccacter co-ordinate (x,y) into polar co-ordinate (r,θ) [7] (figure 3). We use rectified images as input of matching step. As a result of rectification, in matching, instead of searching corresponding point in the whole second image, we only search it in a specific scanline.

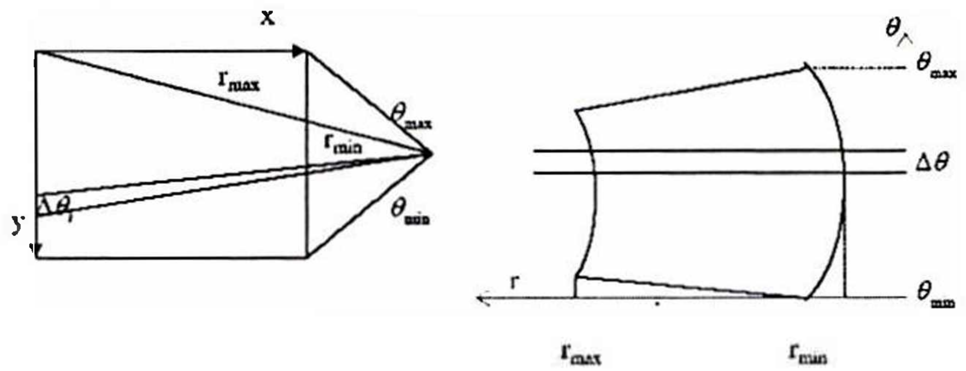


Figure 3. Co-ordinate transformation.

3.5. Dense matching

Each pixel (x,y) in the first image we put a correlation window such as (x,y) is the position of window's center. We find out (x',y') matching with (x,y) by changing another window on scanline of

(x,y) in the second image. Disparity of the two window determine if (x,y) and (x',y') are matching pair. The disparity is calculated by SAD (Sum of Absolute Differences) as follow:

$$c(x,y,d)=\frac{\sum_{i,j}\left|I_1(x+i,y+j)-I_2\left(x'+d+i,y'+j\right)\right|}{\sqrt{\sum_{i,j}I_1(x+i,y+j)^2}\times\sqrt{\sum_{i,j}I_2\left(x'+d+i,y'+j\right)^2}}$$

where as I_k is the mean of the k^{th} window's grey intensities

Nishihara [8] has suggested some correlation window's sizes to increase matching accuracy.

3.6. Triangulation and texturing

For each 3D to 2D correspondence (X, x), we have projection equation $x = PX$, where as x and x' are image points. X is related point in 3-space. P and P' are camera matrices [2]. $AX = 0$ is a result of combining the two equations. Singular Value Decomposition [2] is an effective way to compute X.

Fortunately, between (P, P') and fundamental matrix has a great constraint [2] we can easily compute one from other and in turn. We can have unique F matrix from P and P'. However, pair of P and P' is not unique one from a specific matrix F. We choose P and P' as follow

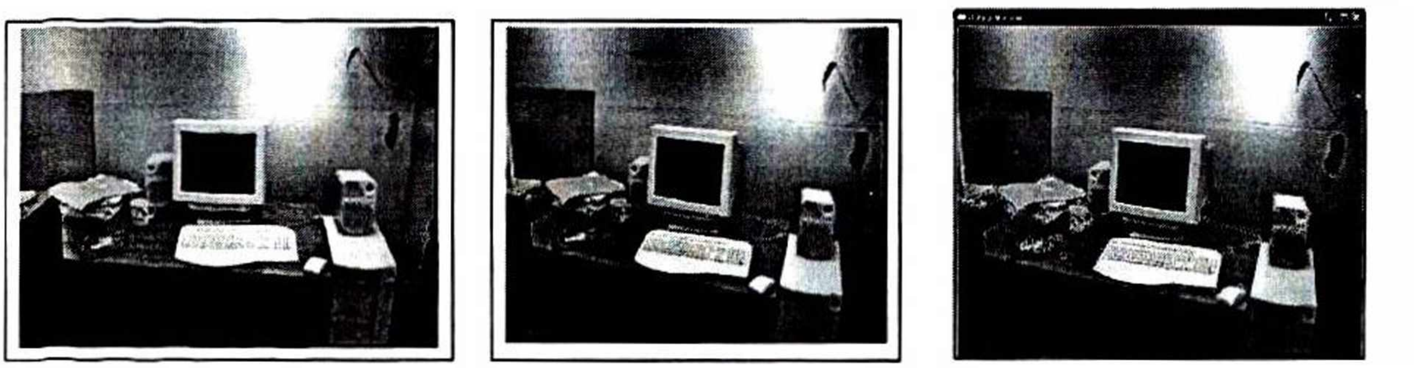
$$P=[I|0] \text{ and } P'=[[e']_x F + e'v^T|\lambda e']$$

where as v is a three-dimension vector and λ is a non-zero constant.

In reality there are many matching points between the two images. Therefore, it was necessary to compute an algorithm that is going to choose a corresponding point from the second image with the highest confident level.

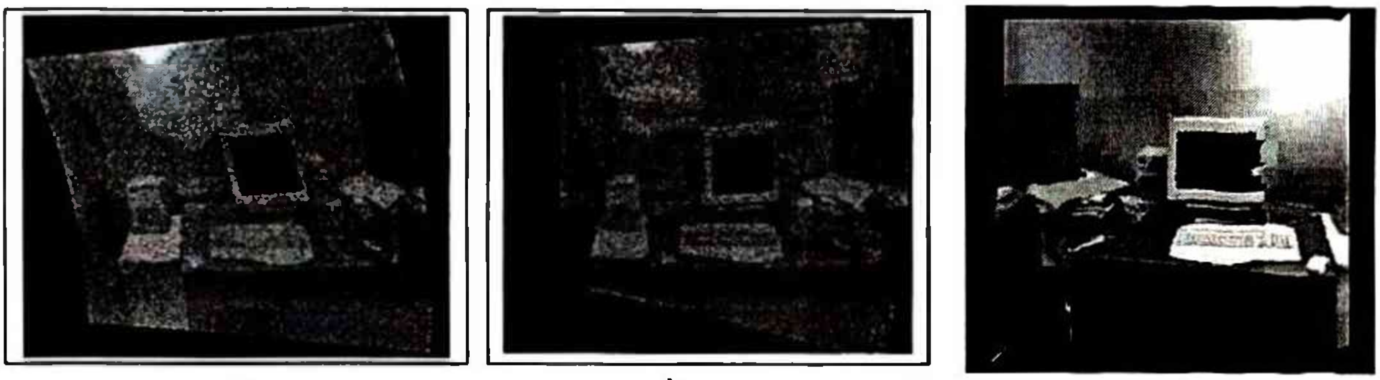
4. Experiments and discussion

In this section we give the results of our technique on synthetic and real data. The synthetic experiment setup is based on some related work. We have two input images (figure 4 a, b). Figure 4c shows Susan corners computed get from two original images. Pair of rectified images are presented in Figure 5a, b, and figure 5c is the picture of the 3D resultant model.



a, b, c,

Figure 4. a,b Two original 480x640 images; c, Susan corners.



a, b, c,

Figure 5. a,b Pair of rectified images; c, 3D resultant model.

The process involved to two input images. Two images suitable for the initialization process are selected so that they are not too close to each other on the one hand and there are sufficient features matched between these two images on the other hand. However, there are still some inexact areas in the 3D model because of occlusion and the simplicity of the used algorithms [6, 9]. The result can be refined each time a new view (image) is added. In future, to improve the quality we will try to use more sophisticated algorithms as well as increase the amount of images.

5. Conclusion

We presented in this paper our work toward the creating of a 3D model from two images. Using a building process in thee steps, we have generated a 3D model of a free-from view with a fair overall quality. In the future we want to improve the reconstruction process more in order to have a more detailed and accurate 3D model.

References

[1] R. Sablatnig, M. Kampel, *Computing relative disparity maps from stereo images*, ERASMUS Intensive Program, Pavia, Italy, 2001.

[2] R. Hartley, A. Zisserman, *Multiple View Geometry in computer vision*, Cambridge University press, 2000.

[3] C. Harris, M. Stephens, *A combined corner and edge detector*, Fourth Alvey Vision Conference (1988) 147.

[4] S.M. Smith, J.M. Brady, *SUSAN - a new approach to low level image processing*, Springer Netherlands, 2004.

[5] Oliver Faugeras Bernard, Real time correlation-based stereo: algorithm, implementation and application. *Technical Report 2013*, INRIA, Institut National de Recherche en Informatique et en Automatique, 1993.

[6] T. Kanade, M. Okutomi, A Stereo Matching Algorithm with an adaptive window: Theory and Experiment, Pattern Analysis and Machine Intelligence, *IEEE Transactions* 16, 1994.

[7] R.I. Hartley, Theory and practice of projective rectification, *Technical Report 2538*, INRIA, Institut National de Recherche en Informatique et en Automatique, 1995.

[8] H.K. Nishihara. PRISM, A Practical Real-Time Imaging Stereo matcher, *Technical Report A.I. Memo 780*, MIT, Cambridge, MA, 1984.

[9] U.R. Dhond, J.K. Aggarwal, "Structure from Stereo – A Review", *IEEE Tran. Man and Cybernetics* 19 (1989) 1489.