Middle-East Journal of Scientific Research 24 (9): 2825-2831, 2016
ISSN 1990-9233
© IDOSI Publications, 2016
DOI: 10.5829/idosi.mejsr.2016.24.09.23872

Benchmark and Result Analysis of 3D Reconstruction with Structured Light Scanning

Ms. B. Bhuvaneshwari and A. Rajeswari

Department of Electronics and Communication Engineering,
Coimbatore Institute of Technology, India

Abstract: Computer vision based 3D Reconstruction is gaining importance because of its better visualization and accurate depth measurement. Over the years, various passive and active 3D reconstruction methods were devised to exploit the explicit information about the scene or objects in the scene. Active methods have the advantage of producing accurate 3D models at high speed. In this paper, an active method of 3D reconstruction is performed by structured light scanning using a hardware setup consisting of two cameras and a projector and is implemented using OpenCV and Python. The results obtained are compared with the standard results obtained using Matlab and Visual C++ and the benchmark results are tabulated and the best solution for implementing 3D reconstruction is determined. The benchmark results include the CPU, RAM and the Processor through which the minimum system requirement is calculated.

Key words: Structured light scanning · 3D reconstruction · Triangulation · 3D model

INTRODUCTION

3D reconstruction is one of the challenging problems in computer vision. It has its applications in robot navigation, industry quality inspection, object recognition, medical imaging, space exploration, documentation and conservation of historic buildings, monuments or archaeological sites and object measurements. 3D reconstruction can be obtained by passive methods that require images from uncontrolled illumination and by active methods that require a controlled environment for acquiring the images. Passive methods include stereo vision, shape from motion, shape from defocus, shape from silhouettes, shape from photo-consistency and Simultaneous Localization and Mapping (SLAM). Active methods include Time-of-Flight and Structured Light where the surfaces are reconstructed from the geometric relationships existing between the source of light and sensor devices [1]. In Active methods, 3D data acquisition is done by laser range scanners or by structured light projecting systems. A pattern is projected on the surface of the object and 3D information is extracted by simple triangulation technique. The main advantages of the active techniques are its speed and its ability to obtain accurate 3D models [2].

The theoretical explanation of the Structured light technique is given in [3]. In general, structured light scanners in the literature assume an angled configuration in which the normals from the centre of the projector and the centre of the camera sensor meet at the calibration plane with a number of methods being proposed for reliable pattern detection [4-6]. Salvi *et al*. [7] have proposed a classification of structured patterns based on their coding strategy. Light patterns projected on the surface of the objects include stripes [8-10], IR random dots [11] and circular color spots. Imaging of wider surface areas are obtained by using multiple projectors [12]. Shape reconstruction techniques with a structured light with temporal and spatial coding are summarized in [13]. Several methods for high-speed capturing were proposed by using a DLP projector and a high speed camera [14, 15].

In this paper, the concept of structured light is used and it is based on projecting a known pattern which gets deformed based on the shape of the object. This deformation of the patterns allows the vision system to calculate the depth and surface information of the object. This process is implemented in python language using OpenCV library files in Pycharm platform. The process starts with camera calibration and then capturing multiple

Corresponding Author: Ms. B. Bhuvaneshwari, Department of Electronics and Communication Engineering, Coimbatore Institute of Technology, India.

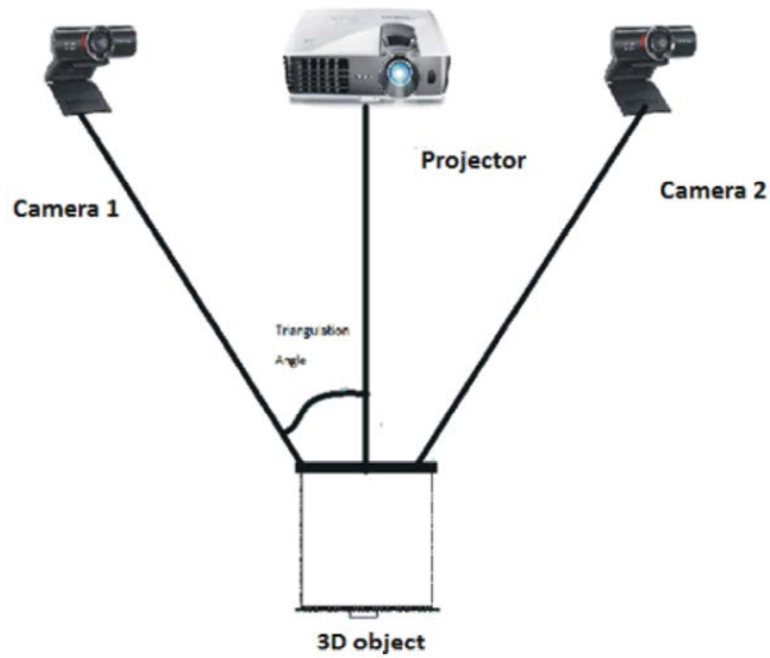

Fig. 1: Structured Light Scanning

images by the two cameras placed at the left and at the right of the object on which the fringe pattern is made to fall by the projector. The images are then processed based on thresholds set. The points in 3D co-ordinate system are derived from the 2D points using triangulation process. The triangulation used here is a derived triangulation process called 3D line scanning triangulation which is more accurate and less time consuming than the traditional triangulation process. This process finally gives a fine 3D model with more information about the object.

3D Reconstruction Using Structured Light

Structured Light Method: Structured light 3D scanning systems work by projecting a known light pattern onto a 3D scene and subsequently imaging it using an appropriate camera. The light pattern will be distorted by the objects in the scene and therefore the pattern observed by the cameras are different to the one originally projected. By determining the shift of each pixel in the pattern the depth of any given point in the cameras, field of view can be determined. The principle behind structured light cameras is therefore very similar in principle to stereo vision cameras as highlighted above. The same principle of disparity is used and the shift in the reference pattern is used to calculate the disparity map instead of the actual images of the scene. This is a 2.5D scanner and to obtain an exact 3D model, the object should be rotated and imaged at different angles. Typical applications for structured light scanners include, gesture recognition, robot navigation etc.

Figure 1 shows the diagrammatic overview of such a system.

The proposed method is based on structured light concept with stereo cameras. Structured light is a known pattern produced by projecting a narrow band of light onto an object, simply grid of horizontal or vertical bars. The pattern consists of vertical stripes and horizontal stripes of various sizes. The stripes are then projected onto the scene and subsequently comes into contact with objects in the scene. The stripes therefore are distorted by the objects surface and the stripes imaged by the camera are shifted from what would be expected. It is this shift, or disparity, that is used as the basis upon which the calculation of the 3D characteristics of the scene or object is made.

The fringes used are of very small width or of small size and therefore the structured light scanners suffer from the same problem as laser scanners and so the depth is calculated over a limited area in one scan. In order to scan an entire object, structured light scanners typically project a range of different light patterns in different orientations in order to successfully determine the depth at each point in the scene. For this reason, scanning with structured light systems is very slow and is therefore suited only for scanning static objects.

Flow of the Proposed Algorithm

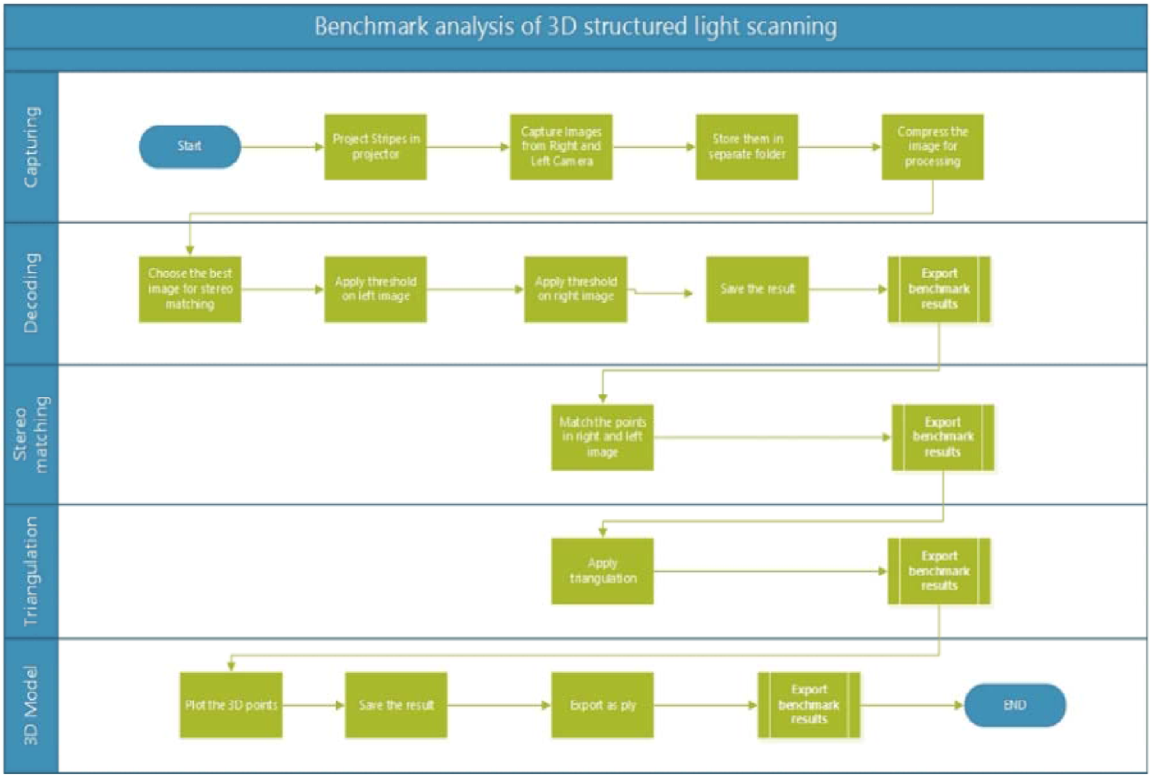

Fig. 2: Cross Functional Flowchart of the proposed algorithm

Data Capture and Camera Calibration: The scanning apparatus consists of two cameras and a projector. The cameras and projector should be arranged to ensure that no camera ray and projector plane meet at small incidence angles. Camera calibration is performed to calculate the intrinsic and extrinsic parameters of an optical system. Camera calibration is done by different methods and in case of performing single shot measurements using stereo sensors, it is done preliminary to the measurement and is regarded for the measurement performing time, until a calibration update is performed. Self calibration is used to realize the preliminary calibration of the sensor.

Once the projector and the cameras are arranged, the lighting needs to be considered. Structured light is fairly strong against ambient illumination, but it will help to turn off as many lights as possible.

Decoding of Captured Images: The pixels intensity values between the first two projections are compared to determine the pixels in the shadow regions. Pixels whose intensity values in the black and white image projection are larger than the threshold are considered as valid pixels. If the difference is smaller than the threshold, the pixels are considered to be in the shadow region and are neglected. The threshold value is adjusted based on the data captured. To provide a mapping between the pixels in the camera, the encoded projected patterns are decoded into their corresponding decimal number [16].

Triangulation: 3D scanning methods are mostly based on triangulation. This works on the basic trigonometric principle of taking three measurements of a triangle and using those to recover the remaining measurements. The images of the object are taken from two perspectives to get two angle measurements and the distance between the two cameras is known. These measurement are used to calculate the distance of the object. Instead of using multiple image sensors, one of the sensor can be replaced with a laser pointer.

3D point in space is computed through the intersection of mapped pixels by Triangulation. A Ray is a single point defined with its direction from the camera's centre of projection and intersects through image plane and extends outwards the scene. Ray's direction vector includes camera's centre of projection and pixel point through which the ray passes. Two rays P & Q are formed

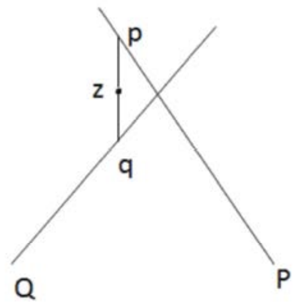

Fig. 3: Ray Intersection [17]

as shown in the Figure 3 with a pixel chosen from both the images. The segment pq is the shortest line perpendicular to PQ ray. Due to ill-pose problem in triangulation, intersection point is computed from two rays with the segment line being perpendicular to P and Q ray. The intersection point is considered to be the mid point(z) of the shortest segment [17].

Consider two rays P & Q which pass through points a and b with direction vectors and respectively. The line segment p and q are defined with s and t scalar values as shown in Eq. (1) and (2).

$$p = a + s * \vec{u} \tag{1}$$

$$q = b + t * \vec{v} \tag{2}$$

The segment pq is perpendicular to lines P and Q which results in the dot product of their vector is zero and hence the scalar value becomes;

$$s = \frac{\vec{w}.\vec{u} * \vec{v}.\vec{v} - \vec{v}.\vec{u} * \vec{w}.\vec{v}}{\vec{v}.\vec{u} * \vec{v}.\vec{u} - \vec{v}.\vec{v} * \vec{u}.\vec{u}} \tag{3}$$

$$t = \frac{\vec{v}.\vec{u} * \vec{w}.\vec{u} - \vec{u}.\vec{u} * \vec{w}.\vec{v}}{\vec{v}.\vec{u} * \vec{v}.\vec{u} - \vec{v}.\vec{v} * \vec{u}.\vec{u}} \tag{4}$$

The intersection point is the average of two end-points of the segment which is taken as the final intersection point so as to remove duplicates generated by geometry. This method of linear triangulation is best suited for simplest 3D reconstruction.

Experiment and Result Analysis: The experimental setup consists of two cameras and a projector. Fringe patterns are projected on the object and a total of 84 images are acquired and store them in separate directories created for right and left camera. The images in which the fringes on the object is more distinguishable from the surrounding is chosen as the best pair of image. Then stereo matching is done and matched points between the two images were found. Finally the triangulation process gives the 3D model of the object. Triangulation used is 3D line scanning triangulation which is much faster than the traditional triangulation method and much more accurate. The 3D model is viewed in Meshlab software.

The sample images acquired using two cameras after projecting the fringe pattern and the obtained 3D results are shown in Figure 4 and 5.

Benchmark results are compared with the Matlab, Visual C++, Java and Python version of the structured light scanning to determine which is more productive. Programming using Matlab, Visual C++ and Java are not demonstrated in this paper as our main objective is to prove that the programming using the Open CV in Python provide more productive results. To prove this, the application designed in each programming language is used and the python version is designed to analyze the benchmark on CPU, RAM and Processor utilization of each module and their total time to produce the processed 3D reconstructed images.

Figure 6 displays the time taken by each process in seconds. It is inferred that the time taken to implement the overall process is less in Python when compared to other platforms [18].

Figure 7 displays the CPU, GPU and RAM consumption of each platform in their respective metrics. It is inferred that Python consumes much less resources than any other platform for 3D structured light scanning. Compared to Matlab, Python code is compact and readable. Python data structures are superior to Matlab data structures. Python provides more control over the organization of one's code and better namespace management. Python makes it easy to maintain multiple versions of shared libraries. Python offers more choice in graphics packages and toolsets. Python comes with extensive standard libraries and has a powerful datatypes such as lists, sets and dictionaries Python is an interpreted language with elegant syntax and that makes it a very good option for scripting and rapid application development in many areas.

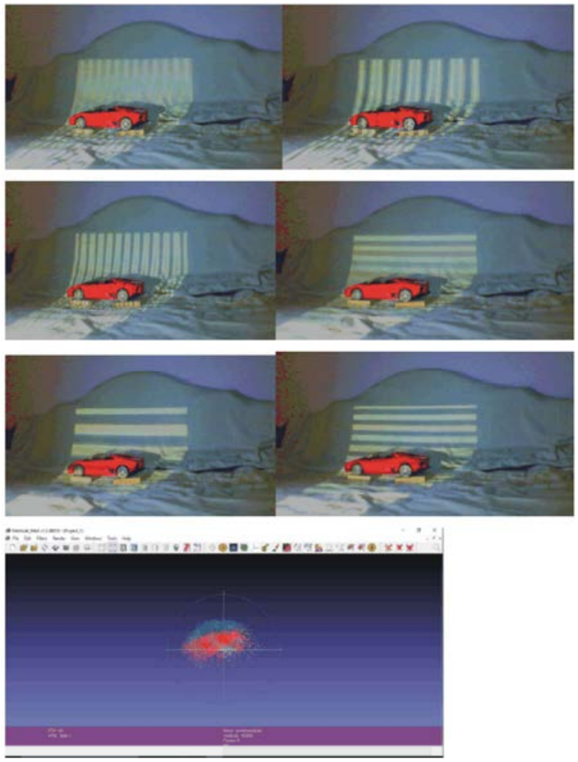

Fig. 4: Results for car image

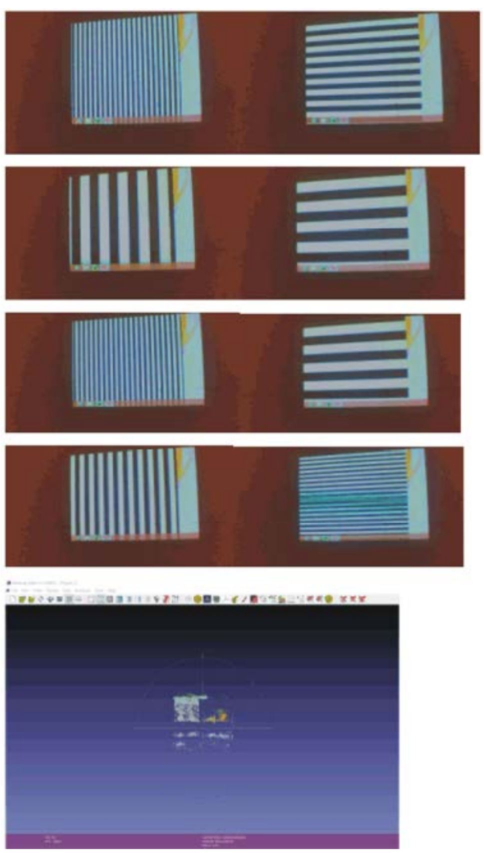

Fig. 5: Results for wall image

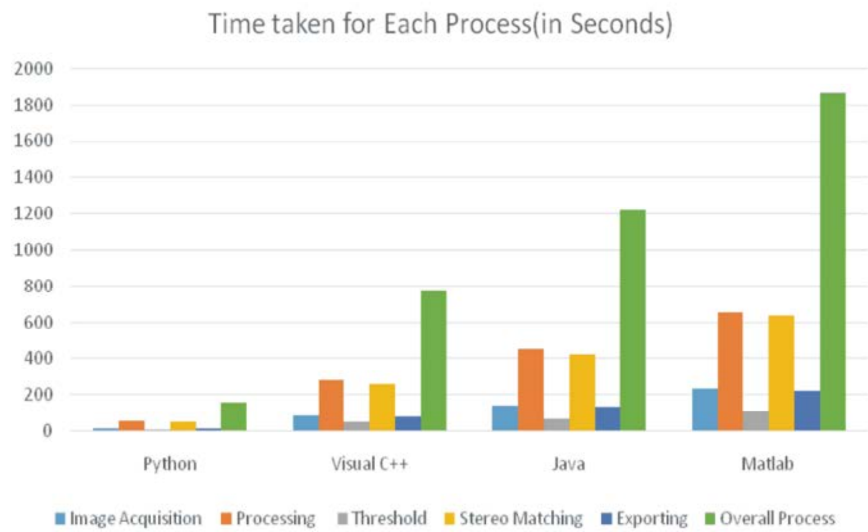

Fig. 6: Visual infographic representation of time taken by each process

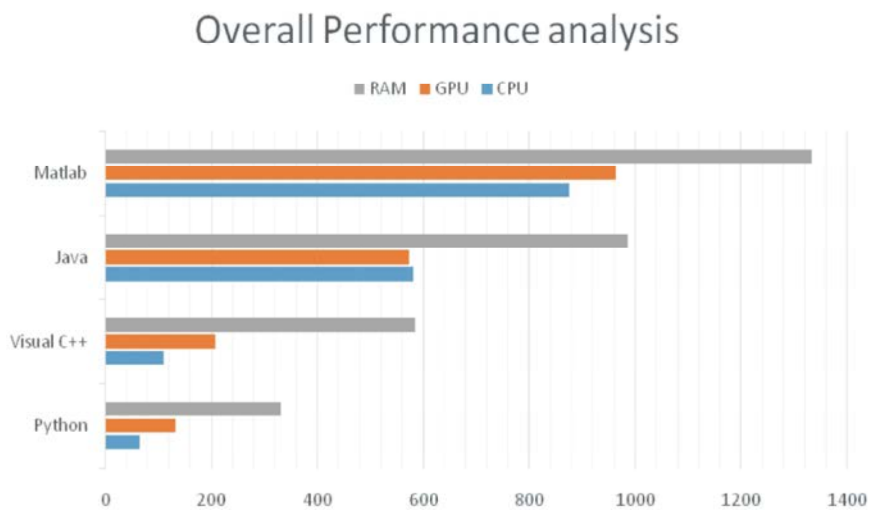

Fig. 7: Visual infographic representation of overall performance analysis

CONCLUSION

In this paper, an active method of 3D reconstruction using structured line method is implemented. From the experimental setup of structured light 3D scanner, the benchmark analysis of OpenCV Python has been recorded. The comparative results indicates that the Python 3D reconstruction is better in speed and performance than other platforms of the same category.

REFERENCES

1. Albitar, C., P. Graebling and C. Doignon, 2007. Robust structured light coding for 3D reconstruction, in: Int Conf Comp Vis (ICCV).
2. Chen, S. and Y. Li, 2008. Vision processing for realtime 3D data acquisition based on coded structured light, IEEE T Image Process, pp: 17.
3. Chen, C., Y. Hung, C. Chiang and J. Wu, 1997. Range data acquisition using color structured lighting and stereo vision, Image and Vision Computing, 15: 445-456.
4. Gorthi, S. and P. Rastogi, 2010. Fringe projection techniques: Whither we are? , Opt Laser Eng.
5. Kawasaki, H., R. Furukawa, R. Sagawa and Y. Yasushi, 2008. Dynamic scene shape reconstruction using a single structured light pattern, in: IEEE Int Conf Comp Vis and Pat Rec (CVPR).

6. Pavlidis, G., A. Koutsoudis, F. Arnaoutoglou, V. Tsioukas and C. Chamzas, 2007. Methods for 3D digitization of cultural heritage , Journal of Cultural Heritage.
7. Salvi, J., J. Pages and J. Batlle, 2004. Pattern codification strategies in structured light systems, Pattern Recognition, 37(4): 827-849.
8. Davies, C.J. and M.S. Nixon, 1998. A Hough transformation for detecting the location and orientation of three-dimensional surfaces via color encoded spots. IEEE Trans. SMC.
9. Scharstein, D. and R. Szeliski, 2003. High-accuracy stereo depth maps using structured light, Proc. CVPR, pp: 195-202.
10. Zhang, L., B. Curless and S.M. Seitz, 2003. Spacetime stereo: shape recovery for dynamic scenes, Proc. CVPR.
11. Ypsilos, I.A., Adrian Hilton and Simon Rowe, 2004. Video-rate capture of dynamic face shape and appearance. Proc. Int. Cont. Automatic Face and Gesture Recognition.
12. Zhang, L., N. Snavely, B. Curless and S.M. Seitz, 2004. Spacetime faces: high-resolution capture for modeling and animation. Proc. SIGGRAPH, pp: 548-558.
13. Batlle, J., E. Mouaddib and J. Salvi, 1998. Recent progress in coded structured light as a technique to solve the correspondence problem: a survey, Pattern Recognition, 31(7): 963-982.
14. Narasimhan, S., S. Koppal and S. Yamazaki, 2008. Temporal dithering of illumination for fast active vision, In European Conference on Computer Vision, 4: 830-844.
15. Weise, T., B. Leibe and L.V. Gool, 2007. Fast 3d scanning with automatic motion compensation, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), pp: 1-8.
16. Douglas Lanman, Gabriel Taubin, 2009. Build your own 3D scanner: 3D photography for beginners, SIGGRAPH 2009.
17. Kyriakos Herakleous and Charalambos Poullis, 2014. 3DUNDERWORLD-SLS: An Open-Source Structured-Light Scanning System for Rapid Geometry Acquisition.
18. Aruoba, S. and J. Fernández-Villaverde, 2014. A comparison of programming languages in economics, NBER Working Paper No. 20263.