Mathematical Problems of Computer Science 44, 42--50, 2015.

# 3D Scanner from Two Web Cameras

Aram V. Gevorgyan[1] and Hakob G. Sarukhanyan[2]

[1]Russian-Armenian University,
[2]Institute for Informatics Automation Problems of NAS RA
e-mail: aramgv@gmail.com, hakop@ipia.sci.am

**Abstract**

In this paper we present an approach how to build 3D scanner using two ordinary web cameras. This approach is based on stereovision and is beneficial from the other 3D scanners that it doesn't need laser, structured light or other expensive sensors. We put the object in front of the cameras at a distance about 40-50 cm and rotate it at small degrees. For every rotation we compute the disparity map from two cameras and then get the depth map in the form of point cloud. Then we filter out our object from the scene and then merge object point clouds from different views to get a full 3D model.

**Keywords**: Stereovision, Depth map, Registration, ICP, 3D model.

## 1. Introduction

The acquisition of 3d models from real objects is a very actual problem and has many applications in cinema, video games, education software and others. Device that can automatically acquire 3D is called 3D scanner. There are different types of 3D scanners: contact 3D scanners that need physical touch to get 3D information, or non-contact, that usually use laser or structured lighting. But most of these devices are expensive. In this article, we represent a system for getting 3D model from 2 web cameras using stereovision. This approach is beneficial from the others that it's enough to have two general web cameras and no other additional equipment. Stereovision is a technique that gives 3D information ( $X,Y,Z$ coordinates) of the scene using 2 or more cameras. It usually consists of the following steps: camera calibration, rectification, stereo correspondence (obtaining disparity map), triangulation (obtaining depth map from disparity map). We will speak about these steps in details in Section 2. Image with its depth (Z coordinate) is usually called 2.5D. From the left and the right camera two images we get point cloud of the object from one scene of view, but it's not enough to have an object 3D model. To build the object 3D model we need to take object images from different scenes and then merge these point clouds into one common, this process is known as registration and is covered in Section 2.

# 2. Related Work

## 2.1 Calibration

Camera calibration is the process of relating the ideal model of the camera to the actual physical device and of determining the position and orientation of the camera with respect to a world reference system. Camera internal parameters include focal length, image format and principal point. External parameters contain information about camera position and orientation in a world, rotation and translation vectors. If we know the internal and the external parameters we can say that the camera is calibrated. Cameras calibration is needed for rectification and depth calculation. Camera calibration is usually done by using chessboard pattern. More detailed calibrations is described in [1],[2], [3].

## 2.1 Rectification

If the two cameras aligned to be coplanar, the two images have the same image plane, so the corresponding points have the same row coordinates and the stereo correspondence problem is reduced from 2D to 1D. But in real world it is impossible to place two cameras ideally coplanar so rectification is needed. Rectification is a process of reprojecting image planes onto the common plane parallel to the line between optical centers that is called baseline.

Rectification is done using epipolar geometry [4]. Epipolar geometry is the intrinsic projective geometry used in stereovision. It is independent of scene structure, and only depends on the cameras' internal parameters and relative pose. Main concepts of epipolar geometry:

The epipole is the point of intersection of the baseline with the image plane.

An epipolar plane is the *plane* defined by a 3D point (or equivalently image point) and the optical centres.

An epipolar line is the intersection of an epipolar plane with the image plane.

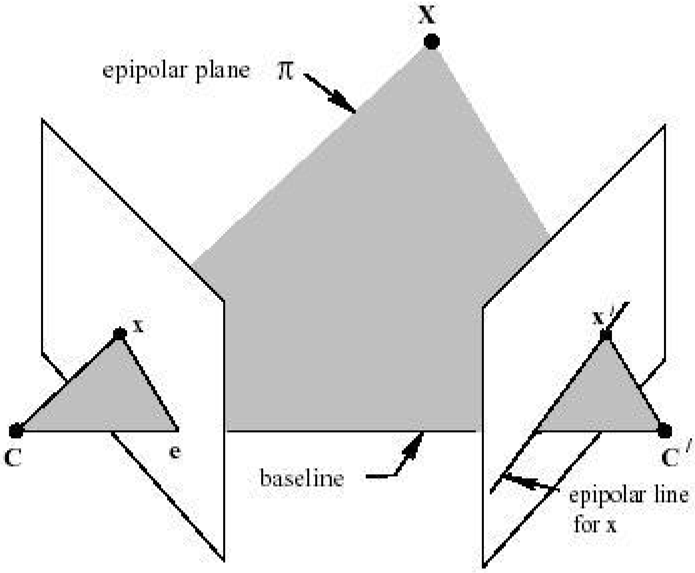


Fig. 1. The epipolar constraint.

Epipolar constraint: We must search the correspondent point of the point in its correspondent epipolar line in the other image.

## 2.2 Stereo Correspondence

Stereo correspondence is a problem of finding matching pixels in the left and right images corresponding to the same points on the 3D surface. If images are rectified - all epipolar lines are parallel and the corresponding points lie on the same row in both images. Suppose $(x, y)$ are coordinates of the point in the left image and $(x', y')$ in the right image, where y= $y'$ because images are rectified. After finding the corresponding points we can compute the disparity map, where disparity for each pixel is $x - x'$. The disparity measures the displacement of a point between the two images. Correspondence problem is complicated by occlusion fact, some pixels are visible only in one of the images.

There are plenty of stereo matching algorithms. They usually can be divided into two classes: local methods and global methods.

In local methods disparity computation at a given point depends only on intensity values within a finite window. In contrast to local methods, in global methods disparity computation is done for all reference image pixels at once. This is achieved by minimizing energy function that consists of data term and smoothness term. Local methods are fast enough but their accuracy is not good and may be inacceptable. While global methods have good accuracy but they are time consuming.

Most commonly used stereo correspondence algorithms are: Block Matching, Semi-Global Matching [5] and Graph Cuts. Block Matching is a local method, Graph Cut is a global method that gives very good results but works slow and Semi-Global matching gives almost the same results as global methods but it works extremely faster. A good survey of stereo correspondence algorithms is in [6].

## 2.3 Triangulation

If we know the geometric arrangement of the cameras, then we can calculate the depth map from the disparity map by triangulation.

Let us consider the simple case: Suppose we have 2 cameras with parallel optical axes.

Let the baseline be perpendicular to the optical axes of the cameras and parallel to x-axis.The distance between the cameras is d and the cameras have equal focal length f.

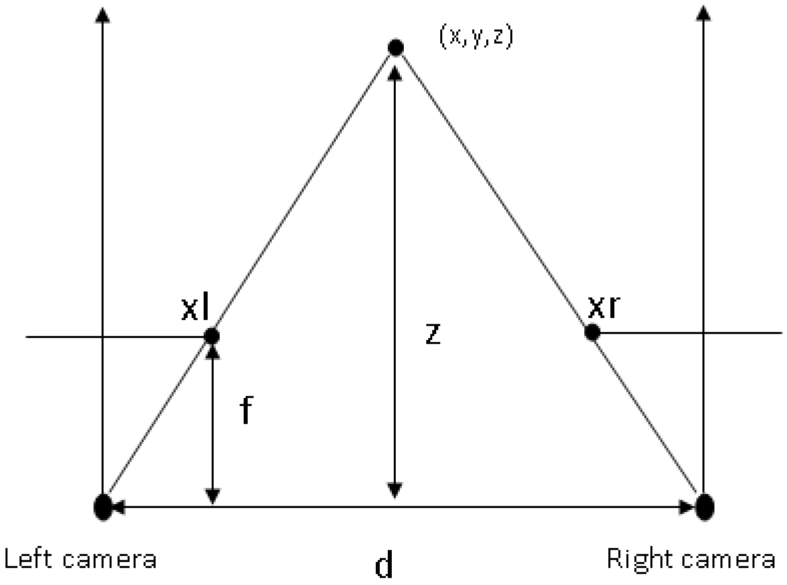


Fig. 2. A simplified stereo imaging system.

Consider $(\mathrm{x}, \mathrm{y}, \mathrm{z})$ are coordinates of a point in three-dimensional world, where $z$ is a depth. Let this point have (xl, yl) and $(xr, yr)$ coordinates in the left and right image planes of the respective cameras.

Then from the similar triangles we have:

$$\frac{z}{d} = \frac{f}{xl - xr} \Rightarrow z = \frac{df}{xl - xr},$$

where $\mathrm{xl} - \mathrm{xr}$ is disparity. Thus, the depth at various scene points may be recovered by knowing the disparities of the corresponding image points.
Also we can compute:

$$x = \frac{d\left(xl + xr\right)}{2\left(xl - xr\right)}, \quad y = \frac{d\left(yl + yr\right)}{2\left(xl - xr\right)}.$$

As we can see depth is inversely proportional to disparity, so when disparity is near 0, small disparity differences make for large depth differences, and when disparity is large, small disparity differences do not change the depth by much. So the depth is computed more accurately for objects near the camera.

Having the depth map we can generate the point cloud. Point cloud is a set of data points in some coordinate system.

More about how to compute the depth is in [7], and how to compute the object distance from web cameras – in [8].

## 2.4 Registration

Registration is a process of merging point clouds from different scenes into one common model. In other words, registration transforms multiple 3D point clouds into the same coordinate system so as to align overlapping components of these sets. The most popular and commonly used algorithm of registration is the Iterative Closest Point (ICP). There are many variants of the algorithm, but all these variants affect one of six stages of the algorithm:

1. Selection subsets of two point clouds
2. Matching corresponding pairs
3. Weighting the corresponding pairs appropriately
4. Rejecting certain pairs based on looking at each pair individually or considering the entire set of pairs
5. Compute an error metric based on the point pairs
6. Minimizing the error metric

More about ICP and its variants are in [9], [10], [11].

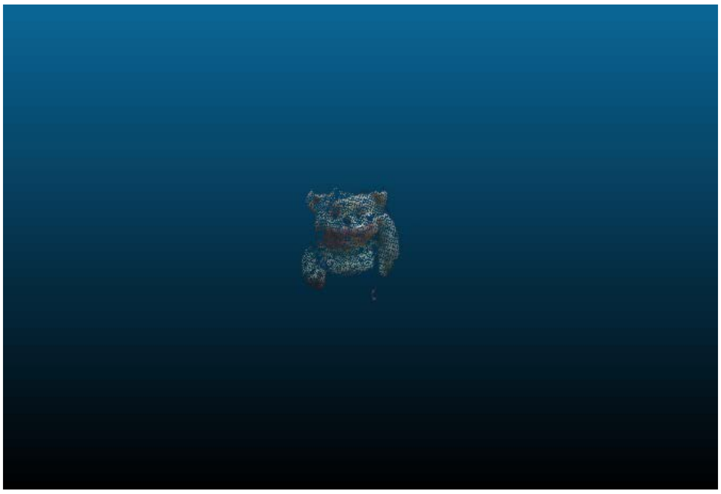
## 3. Realization

For our realization we use OpenCV - the biggest image processing library [12], [13] and Point Cloud Library [14] - 2D/3D images and point clouds processing library.

For data capture we use two Logitech HD Pro Webcam C920 web cameras. We calibrate our cameras with 8x6 chessboard pattern printed on A2 paper and tightly attached on the glass. It is very important to have a flat and difficult stretching surface for calibration pattern. We also tried with A4 paper, but it didn't give the sufficient results. Also illumination is very important for calibration.

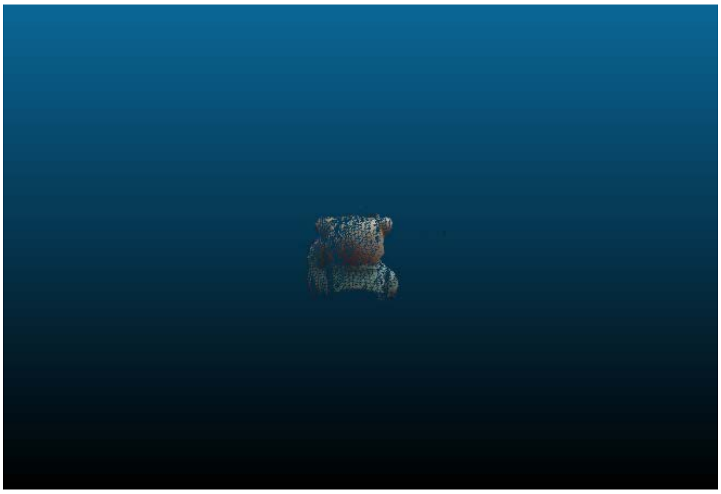
As it was mentioned above, Stereo Block Matching (SBM) is one of the best stereo correspondence algorithms combining speed and accuracy, so we choose SBM for computing disparity map.

We place an object at about 40-50 cm from the cameras and start rotating it about 1-3 degree angle. We capture the left and right camera images for every rotation and compute disparity maps for them. For every disparity map, having cameras parameters, by triangulation we compute the depth map in the form of a point cloud. Then we segment our object from the point cloud by its depth values, to have point cloud only of the object with no background.



**a)**

**b)**



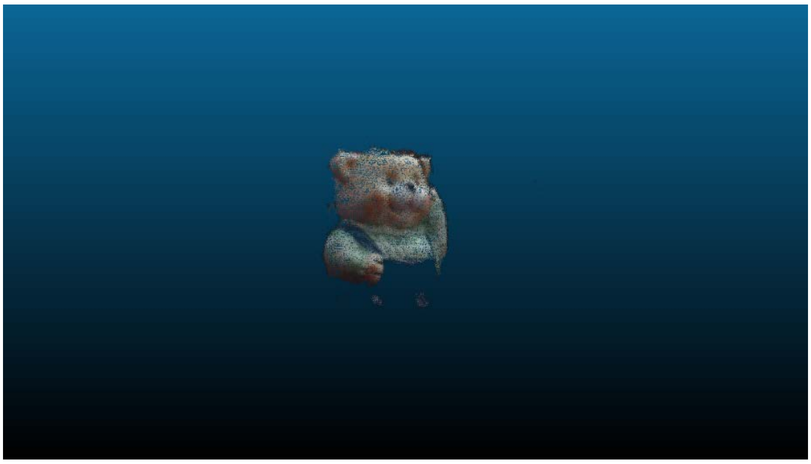
**c)**

Fig. 3. Point clouds of the object from different positions.

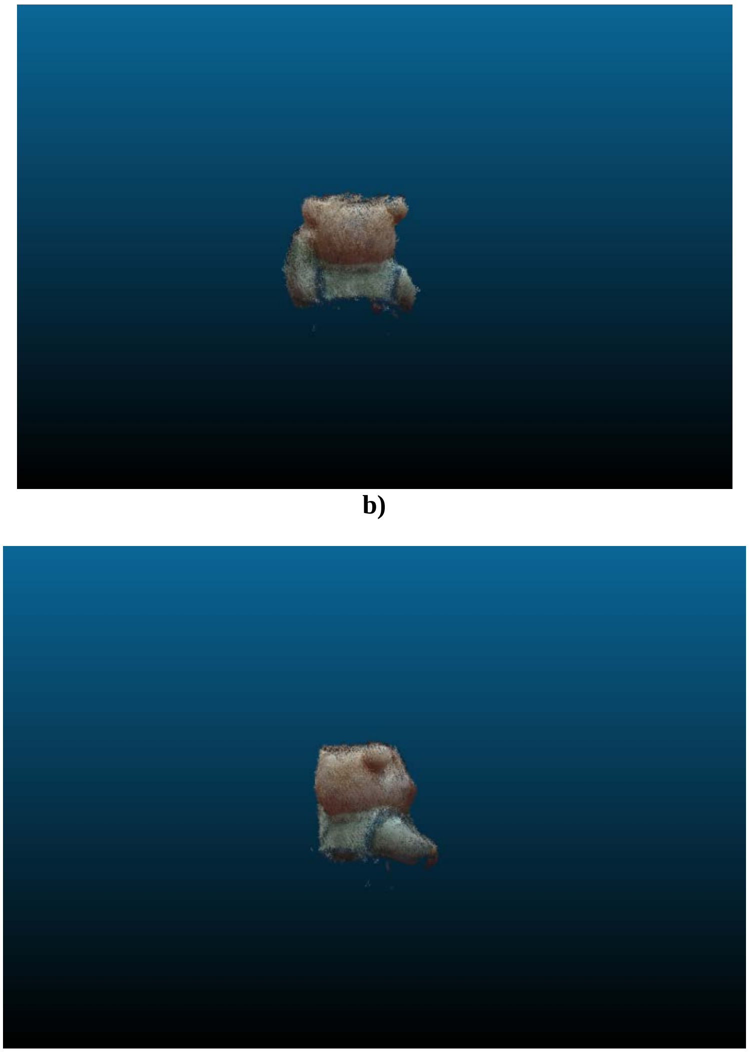
Then the process of registration starts. At first we take the first point cloud as a global model. Then at each next step we take the next consecutive point cloud and merge it to the global model, transforming it to the global models coordinates and adding to it.

Merging of two point clouds consists of the following steps:

1. Finding correspondence between point clouds
2. Rejecting bad correspondences
3. Finding transformation matrix based on correspondences and transform point cloud
4. If convergence criteria is reached stop, else go to step 1



**a)**



b)

c)

Fig. 4. Constructed 3D model.

## 4. Conclusion and Future Work

As it was mentioned, our approach is based on stereovision and it is beneficial from the others that it was cheaper and more available. There are some other 3D scaners based on stereovision [15], [16], [17], but they need laser or sctructured light. Benefits of our aproach that it is enough to have only two ordianry web cameras. Our constructed 3D model is not ideal and its quality can be enhanced. For future work we plan to use color variant of ICP algorithm, that is good for the objects that don't have good geometric features. Due to many factors some holes appears at disparity map, so we also plan to try hole filling algorithms for disparity maps, that may enhance the registrations results.

## References

[1] P. Hillman, “White paper: Camera Calibration and Stereo Vision”, Lochrin Terrace, Edinburgh EH3 9QL, Tech. Rep, 2005.

[2] Z. Zhang, “A flexible new technique for camera calibration”, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* pp. 1330–1334, 2000.

[3] Z. Zhang, "Camera calibration with one-dimensional objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 892-899, 2004.

[4] Epipolar Geometry tutorial. [Online]. Available: http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT10/node3.html

[5] H. Hirschmuller, “Accurate and Efficient Stereo Processing by Semi Global Matching and Mutual Information.” *IEEE Computer Vision and Pattern Recognition*, vol. 2, pp. 807-814 2005.

[6] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms”, *International Journal of Computer Vision*, vol. 47, pp. 7-42, 2002.

[7] P. J.Bagga, “Real time depth computation using stereo imaging”, *Journal Electrical and Electronic Engineering*, vol. 1, pp. 51-54, 2013.

[8] M. A. Mahammed, A. I. Melhum and F. A. Kochery, “Object distance measurement by stereo vision”, International Journal of Science and Applied Information Technology, vol. 2, pp. 05-08, March 2013.

[9] P. J. Besl and H. D. Mckay, “A method for registration of 3-Dshapes”, *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 239–256, 1992.

[10] S. Rusinkiewicz and M. Levoy, “Efficient variants of the ICP algorithm”, 3-D Digital Imaging and Modeling, pp. 145–152, 2001.

[11] A. Johnson and S. Kang, “Registration and integration of textured 3D data”, In Proc. 3DIM ’97, Ottawa, pp. 234–241, 1997.

[12] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, 1st ed., O’Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, 2008.

[13] The OpenCV website. [Online]. Available: http://opencv.org/

[14] The Point Cloud Library website. [Online]. Available: http://pointclouds.org/

[15] Z. Lv and Z. Zhang, "Build 3D Scanner System based on Binocular Stereo Vision", Journal of Computers, vol. 7., No 2, February 2012.

[16] Tzung-Han Lin, "Resolution adjustable 3D scanner based on using stereo cameras", Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific, 2013.

[17] M. Pashaei, S. M. Mousavi, "Implementation of a low cost structured light scanner", Int. ch. Photogramm. Remote Sens. Spatial Inf. Sci., vol. XL-5/W2., pp.477-482, 2013.

**Submitted 04.09.2015, accepted 20.11.2015**

# 3D սկաներ երկու վեբ տեսախցիկից

Ա. Գևորգյան և Հ. Սարուխանյան

### Ամփոփում

Սույն հոդվածում ներկայացված է երկու սովորական տեսախցիկից 3D սկաներ կառուցելու եղանակ: Այն հիմնված է ստերեոտեսողության վրա և տարբերվում է մնացած 3D սկաներներից նրանով, որ չի պահանջում լազեր, համակարգված լույս կամ այլ թանկարժեք սենսորներ: Օբյեկտը տեղադրվում է տեսախցիկներից մոտ 40-50 սանտիմետր հեռավորության վրա և պտտվում փոքր անկյուններով: Ամեն պտույտի համար հաշվարկվում է անհավասարությունների քարտեզը, որից ստացվում է խորությունների քարտեզը՝ կետերի ամպի տեսքով: Այնուհետև առանձնացվում է օբյեկտը տեսարանից, միավորվում են տարբեր տեսարաններից կետերի ամպերը և ստացվում է ամբողջական 3D մոդել:

# 3D сканер из двух веб камер

А. Геворкян и А. Саруханян

### Аннотация

В данной статье представлен метод построения 3D сканера при помощи двух обычных веб камер. Этот метод основан на стереозрении и отличается от других 3D сканеров тем, что не требует лазера, структурированного света или других дорогих сенсоров. Объект помещается на расстоянии 40-50 сантиметров перед камерами и вращается на маленькие градусы. Для каждого вращения вычисляется карта смещений по двум камерам и строится карта глубины в виде облака точек. Далее мы отделяем объект от сцены и соединяем облака точек объекта из разных ракурсов, чтобы получить полную 3D модель.