

# 3D- Model Reconstruction from Two Stereo Pair Images

Krunal N Patel

Assistant Professor, Information Technology Department., MBIT Engineering College, New V V Nagar, Gujarat, India

## ABSTRACT

### Article Info

Volume 3, Issue 2

Page Number: 295-305

### Publication Issue :

January-February-2018

### Article History

Accepted : 10 Jan 2018

Published : 08 Feb 2018

This paper considers the general problem of surface reconstruction from two stereo pair images with prominence on combined source of information about the surface from images data. The aim of is to construct a realistic, freeform 3D model of an object from a two stereo pair images. In this field we can easily construct the 3D model from multiple view and different known camera parameters but from minimal efforts with two images it is very difficult without much detail. This is a most ill posed under-constrained problem and relies on cues like shading, texture, occluding contour etc. Hence, we are suggesting the method which is used to reconstruct free form 3D model of objects represents in the two stereo pair images

Keywords: 3D Reconstruction, modelling, SVR, contour, Silhouette, Texture, Stereo Matching.

## I. INTRODUCTION

The term image matching means automatically correspondence establishment, between primitives extracted from two or more (digital) images depicting at least partly the same physical objects in space. Thus, the 3D information of the objects can be computed. In photogrammetry and remote sensing, image matching techniques have been employed for automatic relative and absolute orientation, point transfer in aerial triangulation image registration and automatic generation. To try to give an extensive overview about the image matching techniques is a difficult task by itself. This section gives a brief overview of the image matching techniques, in terms

of type of matching primitives, the employed constraints and the matching strategies [1].

Image matching belongs to the class of inverse problems, which are well known to be ill-posed. A problem is ill-posed, if no guarantee can be given that a solution exists, is unique, and is stable with respect to small variations in the input data. Image matching is ill-posed, because for a given point in one image, its correspondences on other image may not exist due to occlusion, because there may be more than one possible match due to repetitive texture patterns or a semi-transparent object surface, and because the solution may be unstable with respect to image noise or poor textures. To find the solution of an ill-posed

problem one usually has to cope with an optimization function exhibiting many local extrema, and thus a small pull-in range. Therefore, stringent requirements may exist for initial values for unknown parameters to be determined [2]. Moreover, the search space for these parameters must be constrained and reduced. The following assumptions, which usually hold true when dealing with photogrammetric and remote sensing imagery, can be introduced to convert the ill-posed image matching problem to well-posed problems.

- The various images have been acquired using one and the same or at least similar spectral bands, so that the image intensities of a physical object exhibit similar values.
- The illumination together with possible atmospheric effects is constant through the time interval for image acquisition. This assumption might not be correct in the cases of cross-track stereoscopic satellite images.
- The scene projected in the images is rigid.
- The object surface is piecewise smooth and opaque.
- The object surface exhibits a more or less diffuse reflection function.
- Initial values such as the range of the terrain height are possibly known.

In different applications, some assumptions of the list above are violated and, different combination of these assumptions can be introduced. It is the mixture of necessary assumptions and how to deal with deviations from these assumptions which make the design of a good image matching algorithm difficult and has led to the development of different algorithms in the past.

Basically, to design an image matching algorithm has to take the following questions into account:

- (a) Which kinds of primitives are selected for matching?
- (b) Which kind of constraint is used to restrict the search space?
- (c) How to implement and combine the constraints through optimization procedures?
- (d) Which strategy is employed in order to control the matching algorithms?

Different answers for these questions lead to different classes of the image matching algorithms and approaches.

## II. Matching Primitives

In order to compute the correspondences among different images, the essential step is to define the primitives to be matched. On these primitives the matching similarity measure is computed and evaluated. Having computed the similarity measures, the matching problem is solved and consequently, the disparity map can be reconstructed. Ideally, we would like to find the correspondences of every individual pixel in an image. However, it is obvious that the information content in the intensity value itself is too low for unambiguous matching. Therefore, coherent collections of pixels or features are matched. Normally two main classes of matching primitives can be differentiated: features and image intensity patterns. The features are typical the easily distinguishable primitives in the input images and the matching can be performed only on these features. The methods using features belong to a group called **feature-based matching** (FBM)[3]. On the other hand, the methods processing image intensity patterns perform matching over a square or rectangular window of pixels and usually establish correspondences in (all) pixels. They belong to a group called **area-based matching** (ABM).

### A) Area-Based Matching (ABM)

ABM is usually based on local image windows. ABM is justified by the continuity assumption, which asserts that at the level of resolution, at which image matching is feasible, most of the image depicts portions of a continuous surface; therefore, adjacent pixels in an image window will generally represent contiguous points in object space. Examples of ABM are the **cross-correlation** method, and the **least squares matching** [4]. In these methods, a small window, composed of gray values, serves as matching primitive [4]. The window center, possibly weighted with respect to the gray value gradient can be used for the definition of the location of a point to be matched.

Cross-correlation is an algorithm for computing correspondences based on the similarity of gray levels within an image window. A point is given in the reference image, and its matches are searched for in the search image. For that purpose, the reference image patch is moved in the search image, at each searching site in the search image, a similarity measure of the gray levels is calculated. Based on the similarity measure, the correspondences are determined. However, the algorithm will fail if the images are not similar, because the inherent problem of the cross-correlation lies in that the similarity measures, based on gray values, are not invariant with respect to rotation and scale. The problem with using standard squared windows also arises at surface discontinuities or at surfaces with big disparity variations. One possible method is to wrap the image window according to the estimated orientation of the image and surface to reduce the effect of projective distortion. Other approaches use adaptive windows, shift-able windows, or windows based on image segmentation. These approaches could improve the matching results but the problem caused by occlusions still remains.

Least Squares Matching (LSM) is the most accurate image matching technique, the location of the match can be estimated with an accuracy of up to 0.01 pixels. Being same as the cross-correlation method, it is based on the similarity of gray levels (in LSM, similarity is represented by the sum-of-squared differences). However, it is very sensitive with respect to the quality of the approximations; they have to be known with an accuracy of a few pixels. For that reason, LSM is often used to improve accuracy as a final step following the application of another matching technique, e.g. cross-correlation, for establishing the approximations. Just as the cross-correlation method, LSM will fail if the two image patches are not similar (due to the perspective distortions, the image rotation or different image scale and the insufficient modeling of the object surface); it is especially confronted with problems if there are occlusions due to surface discontinuities. LSM can be expanded to more than two images. In addition, geometrical constraints can also be included in the mathematical model. ABM can generate very dense matching results and this is very important for DSM generation. Disadvantages of ABM are the sensitivity of the gray values to change in radiometry due to factors such as illumination changes, the large searching distance for matching including various local extrema and the large data volume which must be handled. Blunders can occur in areas of occlusions, poor and repetitive texture.

### B) Feature-Based Matching (FBM)

FBM does not use the gray levels themselves as the description of the images but rather an abstract image representation derived from a feature extraction process. FBM normally comprises of two stages. Firstly, the features, together with their attributes, are extracted in each image individually. Secondly, corresponding features from different images have to be found under certain assumptions regarding the

local geometry of the object to be reconstructed and the geometric or radiometric constraints [5]. As compared to gray value windows, features are in general more invariant with respect to geometric and radiometric variations.

The features can be points of interest, edge pixels, line segments, contours and regions. Each feature is characterized by a set of attributes. For example, the interest values for points, the image gray values within the flanking region of edges or around points, the edge orientation and strength (gradient across the edge) for edge pixels and segments, the length and curvature of edges and lines, the size and the average brightness of regions. Feature extraction schemes are often computationally expensive and require a number of free parameters and thresholds that must be chosen a priori. The result of feature extraction is a list containing the features and their attributes for each image. The second stage, i.e. the correspondence stage often consists of two phases. In the first phase, a preliminary list of candidates is built up based on a similarity measure and very often constraints for restricting the search space. Usually, the epipolar constraint is used.

### III. Constraints

Matching constraints can be derived based on assumptions about the real world. For example, the object surface should be piecewise smooth or the matched primitives should exhibit similar attributes in different images. The epipolar constraint and the similarity constraint are two main examples of unary constraints. In this section, several commonly used constraints will be described

#### A) Geometry Constraint

Very often the images are taken with a system which can be modeled by the pinhole camera model is the central perspective projection, which can be

represented by the so-called collinearity equations. This model provides for a very powerful constraint called epipolar geometry constraint. The epipolar constraint is vital in reducing matching ambiguity problems and computational costs. If the image orientation parameters are known precisely, only candidates along the epipolar line are possible matches[12]. If the orientation parameters are only known approximately, the search space is still restricted to a band centered at the epipolar line, its width depending on the quality of the approximations. Most of the matching algorithms developed in computer vision explicitly or implicitly use this constraint.

#### B) Similarity Constraints

Typically, matching candidates that satisfy the epipolar geometry constraints are subjected to further checking using similarity measures computed from the attributes of matched primitives. The definition of similarity measures for a good match obviously plays an important part in each matching algorithm. For ABM the similarity between gray value windows is defined as a function of the differences between the corresponding gray values. The most common similarity measures are for instance sum-of-squared differences (SSD), sum-of-absolute differences (SAD) and normalized cross-correlation (NCC)[13]. Based on these similarity measures, the correspondences are determined. There exist different methods for determining the matches based on similarity measures, such as winner-takes-all and stable matching. Other methods relate to defining appropriate image window, both in its size and shape, in order to increase the discriminability power of the similarity measures.

#### C) Compatibility Constraint

Compatibility constraints belong to high order constraints and they express the fact that a pair of matches are mutually compatible, and are usually derived from the assumptions about the underlying physical principle of the real world. Examples of these constraints such as the **surface uniqueness constraint**, the **surface smoothness constraint**, and **ordering constraint** can be introduced into the matching algorithms. These constraints can be seen as the prior knowledge about our surrounding environments and regularization terms in order to solve the ill-posed matching problems.

Topological constraints attempt to use the fact that the relationship of the three-dimensional structure in different images should be identical and that these constraints the projection of this structure in different viewpoints. The ordering constraint, which requires the order of two matched primitives in one image should be kept in other images, is an example of topological constraint. In general, the topological constraints assume that the relative positions of the primitives remain similar between images. This is less true of features widely separated in three-dimensional space and where occlusions often result in local violations of this constraint.

#### IV. Matching Strategy

An image matching algorithm may consist of a number of steps. Each of the individual modules that can be employed for each step has both advantages and disadvantages. In the matching strategy the individual steps carried out within the algorithm are determined. A good matching strategy is decisive for a successful solution in more complicated situations. Some of the aspects of a good strategy such as combining multiple matching primitives, hierarchy and redundancy are discussed in the following.

The performance of the image matching procedure depends on the quality of the images and the quantity

of information being carried out by them. Compared to the traditional scanned 8-bit/pixel images, digital imagery from linear array sensors has better radiometric performance (e.g. higher dynamic range and signal-to-noise ratio). Nowadays, most of the linear array sensors have the ability to provide more than 8-bit/pixel digital images. Using of these images results in a major improvement for image matching in terms of reducing the number of mismatches for “homogeneous” areas and especially for dark shadow areas. However, the following radiometric problems still have to be considered:

- The original digital images usually show poor image contrast.
- Images may suffer from blur problem caused by stability incompleteness of the stabilizer, e.g. CCD line jitter and motion blur, and deficiencies of the lens system. The problem is more serious on the forward and backward viewing images.
- Image noise is an unavoidable problem in digital images while many features extraction and image matching algorithms are sensitive to image noise.
- Radiometric problems caused by the variations in the sensor view angle, the sun angle and shadowing, the seasons and the atmospheric conditions should also be considered.

In order to reduce the effects of the above-mentioned radiometric problems and optimize the images for subsequent feature extraction and image matching, a pre-processing method, which combines an adaptive smoothing filter and the Wallis filter, is developed. The method mainly consists of 3 processing stages. In the first stage, the noise characteristics of the images

are analyzed quantitatively in both homogeneous and non-homogeneous image regions. Secondly, an adaptive smoothing filter is applied to reduce the noise level and, in the meanwhile, to sharpen edges and preserve even fine detail such as corners and line end-points. The filter requires as input an estimate of the noise, which has already been estimated in the first stage. Finally, the Wallis filter is applied to strongly enhance and sharpen the already existing texture patterns.

#### A) Multiple Matching Primitives

As described in the previous section, both ABM and FBM techniques have advantages and disadvantages, therefore a combination of both usually could improve the results over either technique alone. The edge information can be used to improve the disparity map produced using cross-correlation. In this method, the area-based matching provides a dense disparity map, and the feature-base matching provides an accurate location of discontinuities. These are then combined to produce a dense disparity map with improved accuracy at depth discontinuities[18].

Different matching strategies are adopted in each match step. Adaptive windows with variable size and shapes are also used to consider the local information of the pixels.

#### A) Redundancy

With the conventional methods the human operator uses his/her insight and intelligence to select the points that well represent the terrain surface and measures them. In this case, the implicit philosophy behind is that for economic reason as few points as possible are to be measured which will be sufficient to represent the terrain. Nowadays image-matching techniques are employed to collect the points automatically for DSM generation. The computational costs for collecting a

very large number of points is relatively low. The points can be matched with potentially sub-pixel accuracy. The philosophy of image matching is “redundancy” so that intelligent setting of points by skilled human operator is replaced by redundant capture of points. Matching with multiple images also provides a high redundancy of feature observations (matches) and this will increase both the precision and reliability of the matching process. In addition, higher measurement accuracy through the intersection of more than two imaging rays can be achieved.

#### B) Occlusion Problem

The occlusion problem in image matching refers to the fact that some points in the scene are visible in one image but not in other images, due to the scene and imaging geometry. Occlusions often appear near surface discontinuities, and they increase with increasing baseline between the sensor stations, especially when the surface discontinuities are vertical to the baseline. In fact, occlusion is one of the main culprits to prevent from obtaining correct surface models and it contributes a main source of mismatches in almost all of the existing matching methods. Whether a point in the reference image near a surface discontinuity is occluded in the other images, strongly depends on the number of images and configuration.

From the standpoint of the correct match, matching with multiple images (more than 2) can give a clue to resolve the occlusion problem. When an area is occluded in an image from a camera station, images from another different camera station have a chance to observe this area and thus it can provide a correct match.

There are algorithms for handling occlusion problems: methods that detect occlusion, methods that reduce sensitivity to occlusion, and methods that model the

occlusion geometry. The first group of methods attempts to handle occlusion regions by detecting them either before or after the matching. These occlusion regions are then either interpolated based on neighboring matching results. Methods using the two-way consistence check belong to this group. Normally they compute two sets of matching results, one based on the correspondence from the left image to the right and the other based on the correspondence from the right image to the left. Inconsistent matches are assumed to represent occlusion regions in the scene. The second class of methods attempts to use some robust methods to reduce the sensitivity of matching to occlusion and other image differences. Methods of detecting occlusion regions and recovering three-dimensional information in those regions by exploiting multiple images (more than two) belong to the third group of methods, since these methods attempt to recover the occlusion regions by directly modeling the occlusion regions from multiple images.

## V. Implementation Methodology for Stereo Pair Images

### SAD (Sum of Absolute Difference) Algorithm Implementation

The most common approach in both stereo disparity calculations and motion compensation is to slide a block taken from one image over a second image. This approach is known as the **Block Matching Algorithm**. At each possible offset, a square sense error is computed. Finding the position where the sub-images are most similar (and the minimum error occurs) is equivalent to computing the disparity.

Disparities typically have a small dynamic range (often < 10 pixels) compared to the actual distances to objects. Therefore, measuring disparities to integral pixel values results in very low depth resolution. The

**left view right views** focal point imaging surface (gets reversed image) distant object nearby object positive negative zero disparity solution is to measure disparities to subpixel resolution, with half-pixel accuracy being common and quarter-pixel used in some systems.

Finding corresponding points for every pixel in an image is an extremely computationally expensive task. Consider a straightforward implementation: for every pixel in the left image, a surrounding block of pixels (often 16x16 or 32x32) is slid across a row from the right image (which is the same height as the block from the left, but the width of the whole image.) At each position, the square-sense error (or other error metric) is computed, involving a large number of additions and multiplications.

Various optimizations intended to reduce the amount of computation have been proposed. Rather than searching an entire row, a subset of it is usually selected based on an estimate of the maximum disparity likely to be seen in the data. The search range can also be dynamically adjusted by exploiting the fact that nearby points are likely to have similar disparities.

Another class of optimizations relies on the observation that the error function as a function of horizontal offset (from which disparity is determined) is typically quite smooth, with a single and dramatic minimum.

Methods can be used for Correlation are:

1. SAD (Sum of Absolute Differences)
2. SSD (Sum of Squared Differences)
3. NCC (Normalized Cross Correlation)
4. ZNCC (Zero mean Normalized Cross Correlation)

### A) Disparity Computation Using SAD

Most common dissimilarity measures are squared intensity differences (SD) and absolute intensity differences (AD).

A SAD block finds the similarity between two input images by performing the sum of absolute differences. The greater the similarity between two matrixes the lower the SAD value that result.

$$C_{SAD}(x, y, d) = \sum_{(i,j) \in N(x,y)} |I_1(i, j) - I_2(i + d, j)|$$

Where, d is disparity,  $I_1$  and  $I_2$  are intensities of image 1 and image 2[12].

In this the SAD method is implemented the implementation contains the comparison of two images (Left and Right) pixel by pixel and taking absolute differences of relative pixel of left image with right image.

### B) Mean Shift Segmentation

This is an advanced and versatile technique for clustering-based segmentation. The mean shift algorithm seeks a *mode* or local maximum of density of a given distribution. This algorithm works as follows:

- Choose a search window (width and location)
- Compute the mean of the data in the search window
- Center the search window at the new mean location
- Repeat until convergence
- Initialize windows at individual pixel locations
- Perform mean shift for each window until convergence
- Merge windows that end up near the same “peak” or mode

### C) Disparity map and Depth map:

“Stereo disparity is the difference in position between correspondence points in two images”

- Disparity is inversely proportional to scene depth
- Brighter pixel means bigger disparity (closer to the viewer)

- The relation between Depth and Disparity:

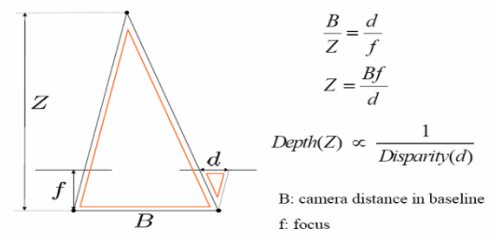


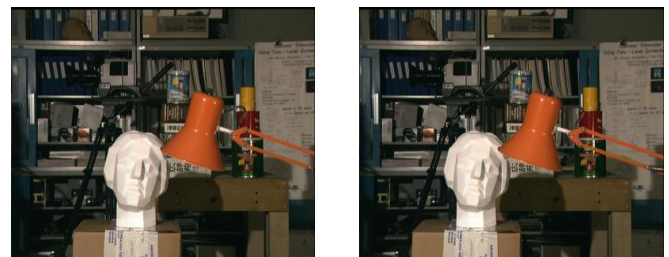
Figure 1: Relationship between Disparity and Depth

## VI. Results

### A. GENERATING A 3D VIEWING MODEL

3D Plot of calculated disparity is achieved by this way. This deals with generating the final 3D view using the given stereo pair of the images. The disparity calculated in the above step can be used directly to plot on a 3D mesh to get an estimate of the relative distances of various objects in the image. But, there still exists a major problem, i.e. the loss of original 3D intrinsic characteristics of the image in the 3D model.

This is the original stereo pair images taken from two slightly different angles.



Left Image

Right Image

Figure 2 : Original stereo Pair Images



The disparity has been calculated from that original stereo pair images.

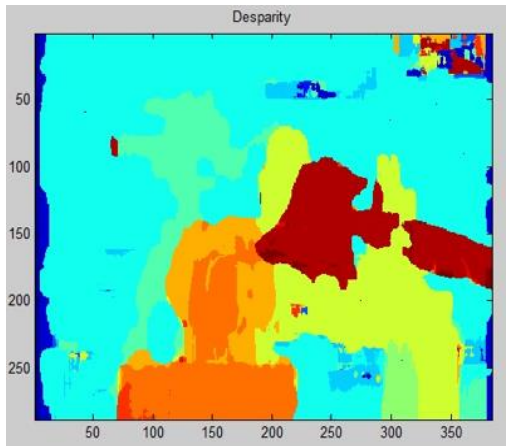


Figure 3 : Disparity Image

After that the depth has been calculated from that disparity.

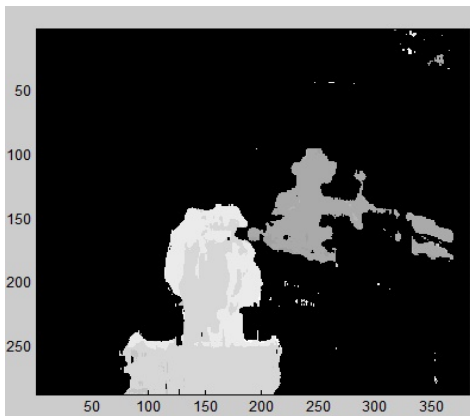


Figure 4: Depth

Then the 3D model has been generated as shown below. And the different views have been shown.

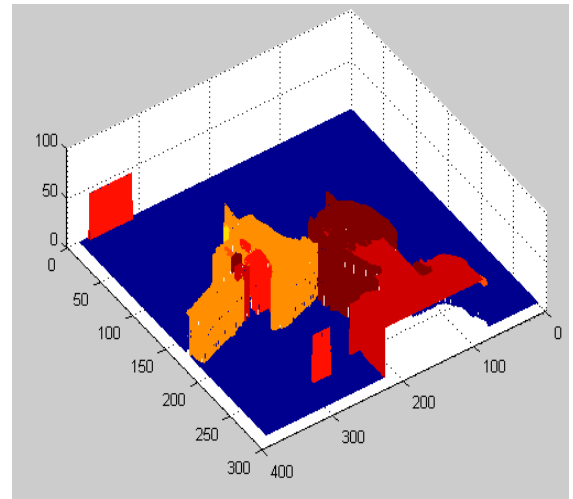
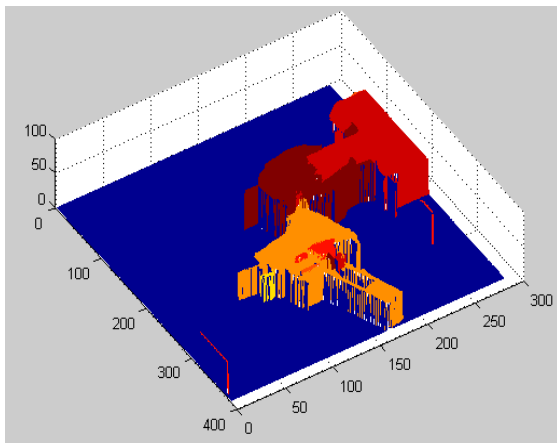


Figure 5: 3D Model for the given two Stereo pair Images



Left Image

Right Image

Figure 6: Original stereo Pair Images

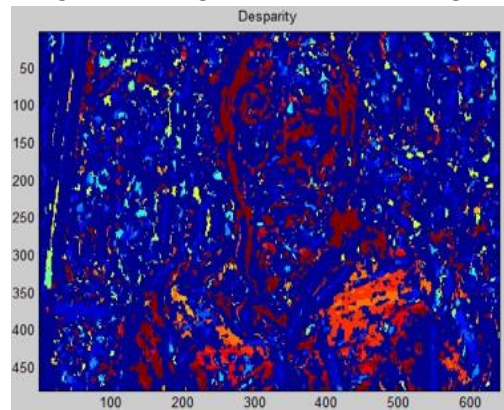


Figure 7: Disparity Image

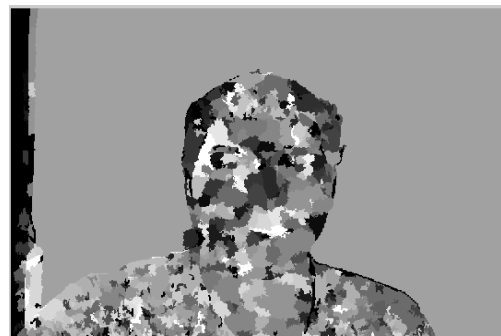
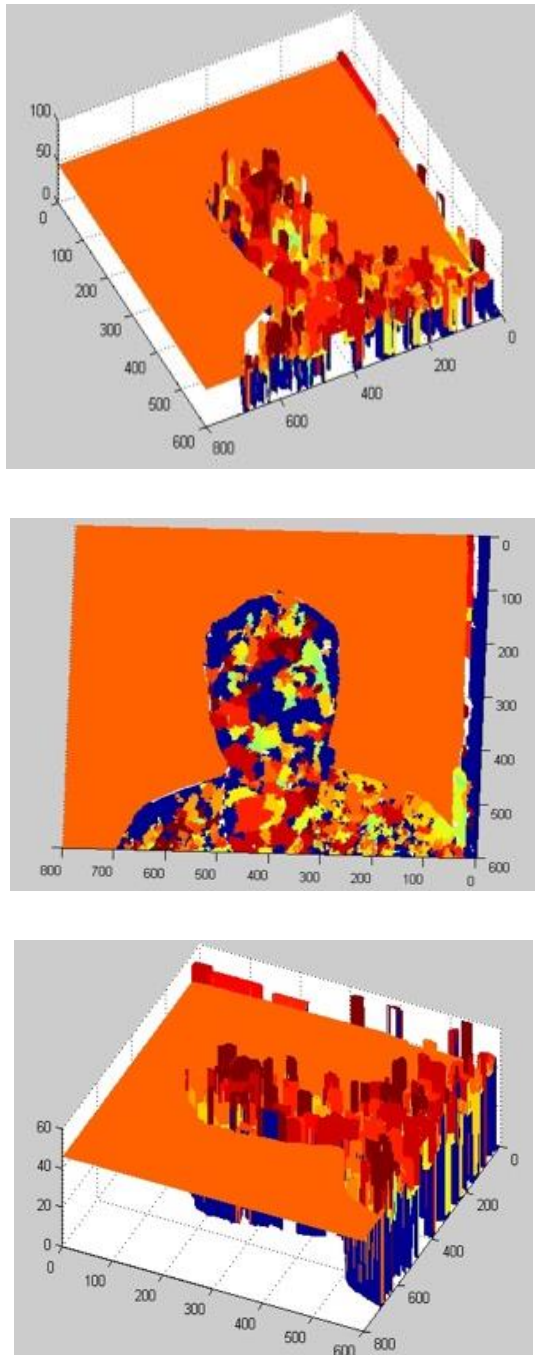


Figure 8: Depth



**Figure 9: 3D Model for the given two Stereo pair Images**

### VII. Conclusion

In this approach two views have been used instead of single view. From those two views the disparity has been found out. Then from that calculated disparity the depth related to that stereo pair images have been found out. After that plotting that depth on to the mesh, we have got the 3d model from that stereo pair images. And it works

on all the images. But the important conclusion is that the time taken for calculating the depth from single image is less compare to tow time taken in two views, and also minimal user interface and constraint required in single view than two views.

### VIII. REFERENCES

- [1]. H. Baker and T. Binford. Depth from edge and intensity based stereo. IJCAI, 1981.
- [2]. D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models for 3D object reconstruction. Proc. ICCV, pages 269–276, 1987.
- [3]. M.J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. IJCV, 19(1):57–91, 1996.
- [4]. D. Scharstein and R. Szeliski. Stereo matching with nonlinear diffusion. IJCV, 28(2):155–174, 1998.
- [5]. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In Proc.ICCV, pages 259–268, 1987.
- [6]. D. Liebowitz. Camera Calibration and Reconstruction of Geometry from Images.PhD thesis, University of Oxford, Dept. Engineering Science, Jun 2001.
- [7]. D. Liebowitz, A. Criminisi, and A. Zisserman. Creating architectural models from images. In Proc. EuroGraphics, volume 18, pages 39–50, Sep 1999.
- [8]. D. Liebowitz and A. Zisserman. Metric rectification for perspective images of planes. In Proc. CVPR, pages 482–488, Jun 1998.
- [9]. D. Liebowitz and A. Zisserman. Combining scene and auto-calibration constraints. In Proc. ICCV, Sep 1999.
- [10]. Andreas Klaus, Mario Sormann and Konrad Karner,Segment-Based Stereo Matching Using

Belief Propagation and a Self-Adapting Dissimilarity Measure VRVis Research Center 8010 Graz, Austria

- [11]. L. Zhang, G. Dugas-Phocion, J.S. Samson, and S.M. Seitz. Single view modeling of free-form scenes. In Proc. CVPR, pages I:990–997, 2001.
- [12]. Daniel Scharstein, Richard Szeliski A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms
- [13]. A. P. Witkin. Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17(1–3):17–45, Aug 1981.
- [14]. D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. *ACM Trans. Graph.*, 24(3):577–584, 2005.
- [15]. Mukta Prasad, Andrew Zisserman and Andrew Fitzgibbon : Single View Reconstruction of Curved Surfaces University of Oxford, U.K.
- [16]. O. Veksler. Stereo matching by compact windows via minimum ratio cycle. ICCV, 2001.
- [17]. R. D. Arnold. Automated stereo perception. Technical Report AIM-351, Artificial Intelligence Laboratory, Stanford University, 1983.
- [18]. Developing 3D Viewing Model from 2D Stereo Pair with its Occlusion Ratio, Himanshu Johari, Vishal Kaushik & Pawan Kumar Upadhyay.
- [19]. Daniel Scharstein, Richard Szeliski, a Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms.
- [20]. Andreas Klaus, Mario Sormann and Konrad Karner, Segment-Based Stereo Matching Using Belief Propagation and a Self-Adapting Dissimilarity Measure VRVis Research Center 8010 Graz, Austria.
- [21]. Daniel Scharstein, Richard Szeliski, A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms

**Cite this article as :**

Krunal N Patel, "3D- Model Reconstruction from Two Stereo Pair Images", *International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT)*, ISSN : 2456-3307, Volume 3 Issue 2, pp. 295-305, January-February 2018. Available at doi : <https://doi.org/10.32628/CSEIT217438> Journal URL : <https://ijsrcseit.com/CSEIT217438>