# 3D Vision System Using Calibrated Stereo Camera

**C. Mohandass[1], E. Pradeepraj[2], P. Tharunkumar[3], K. Udyakumar[4], S. Karthikeyan[5]**

*[5]Assistant Professor, Mechatronics Engineering, Maharaja Engineering College, Tamilnadu, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** *3D vision system can find the depth of an object in an image using stereo cameras. Which is consisting the two digital cameras they are placed in various methods of capturing image with same object but different sense. The vision algorithm in a matlab just extract the feature of an object in the image and try to match the same feature in another image. It will calculate the disparity between two images. The cameras used in the 3D vision system are calibrated by using the camera calibrator application in matlab we provide multiple set of images of checkerboard for calibrate the cameras. This is providing the camera parameters like focal length, center of camera. We can calculate the depth of the object using trigonometry formulas by create the triangle between the two cameras and object. We provide the servo system for moving the cameras independently and synchronously along x and y axis for various views of objects. Stereo vision in most important for doing tasks like tracking the moving object, estimate the path of moving objects, find location of objects along with the system.*

***Key Words***: Camera Calibration; Depth Estimation; Disparity; Digital Image Processing; Image Acquisition;

## 1. INTRODUCTION

Automatic 3D vision system deals with stereo cameras, camera calibration, image acquisition, image processing. System will have two parallel cameras, and servo control system. Two cameras and servo control system are connected to computer. It consists a processing software which will help us to processed the images acquired from camera and compute some information. We sing a MATLAB software for processing and controlling a system. By using stereo cameras, we can extract some depth information for two images. The servo system used to control the camera along x and y axis synchronously. System will use multiple algorithm for detecting the depth of an objects in the image. Servos are controlled by the controller corresponding to the computed data and predefined algorithms.

## 2. IMAGE ACQUISITION

The first stage of vision system is the image acquisition stage. After the image has been obtained, various methods of processing can be applied to the image perform the many different vision tasks required today. However, if the image has not been acquired satisfactorily then the intended tasks may not be achievable, even with aid of some form of image enhancement. Image acquisition tool box will use for acquire a single image form the camera.

### 2.1 WEBCAMLIST

Webcamlist returns a list of available vision compliant cameras connected to your system with model manufacturer. If the camera has a user defined name that name is displayed if you plug in different cameras during the same MATLAB session, then the camlist function returns an updated list of cameras.

### 2.2 WEBCAM

Webcam is the image acquisition object it returns an array containing all the video input objects that exist in memory. If only a single video input object exists in memory, it displays a detailed summary of the object.

### 2.3 PREVIEW

Preview creates a video preview window that displays live video data for video input object the window also displays the timestamp and video resolution of each frame, and current status of objects. The video preview window displays the video data at perfect magnification the size of the preview image is determined by the value of the video input object properly.

### 2.4 SNAPSHOT

Snapshot acquires the current frame as a single image form the camera and assigns it to the variable image if you can snapshot in a loop then it returns a new frame each time. The returned image is based on the pixel format of camera. snapshot uses the camera's default resolution or another resolution that you specify using the height and width properties if available.

### 2.5 VIDEO PLAYER

Video player = vision. VideoPlayer returns a video player object, for displaying video frames. Each call to the step method displays the next video frame. It configures the video player properties, specified as one or more name value pair arguments.

## 3. IMAGE PROCESSING

Image processing is done with matlab. The captured image is processed for extracting the required information. Which is Centroid, Bounding Boxes, Pixels Values, Area, and another data reduction process also carried out in image processing that are Data Reduction, Image Enhancement, Noise Reduction.

### 3.1 COLOR IMAGE

A true color image is an image in which each pixel is specified by three values one each for the red, blue, and green components of the pixel's color. MATLAB store true color images as an m-by-n-by-3 data array that defines red, green, and blue color components for each individual pixel. True color image doesn't use a color map. The color of each pixel is determined by the combination of the red, green, and blue intensities stored in each color plane at the pixel's location. Graphics file formats store true color images as 24-bit images, where the red, green, and blue components are 8 bits each. This yields a potential of 16 million colors. The precision with which a real-life image can be replicated has led to the commonly used term true color image. A true color array can be of class unit8, unit16, single, or double. In a true color array of class single or double. Each color components are a value between 0 and 1. A pixel whose color components are (0, 0, 0) is displayed as black, and pixel whose color components are (1, 1, 1) is displayed as white. The three-color components for each pixel are stored along the third dimension of the data array. For example, the red, green, blue color components of the pixel (10, 5) are stored in RGB (10, 5, 1), RGB (10, 5, 2), and RGB (10, 5, 3), respectively. To determine the color of the pixel at (2, 3), you would look at the RGB triplet stored in (2, 3 ,1: 3), suppose (2, 3, 1) contains the value 0.5176.

### 3.2 BINARY IMAGES

Binary image has a very specific meaning in MATLAB. A binary image is a logical array of 0s and 1s. Thus, an array of 0s and 1s whose values are of data class, say, unit8, is not considered a binary image in MATLAB. A numeric array is converted to binary using function logical. Thus, if A is a numeric array consisting of 0s and 1s. we create a logical array B using the statement,

```
B = logical (A)
```

If a contains elements other than 0s and 1s. The logical function converts all nonzero quantities to logical 1s and all entries with values 0 to logical 0s. Using relational and logical operations also results in logical arrays. To test if an array is of class logical, we use the following function,

```
Islogical (C)
```

If c is a logical array this function returns a 1. Otherwise it returns a 0. Logical arrays can be converted to numeric arrays using the class conversion functions.

### 3.3 CONTRAST ADJUSTMENT

You can adjust the intensity values in an image using the imadjust function, where you specify the range of intensity values in the output image. For example, this code increases the contrast in a low-contrast grayscale image by remapping the data values to fill the entire intensity range [0, 255].

```
I = imread('image(1).jpg');
J = imadjust(I);
Imshow(J);
Figure, imhist(j,64);
```

This figure displays the adjusted image and its histogram. Notice the increased contrast in the image, and that the histogram now fills the entire range. You can decrease the contrast of an image by narrowing the range of the data.

### 3.4 SEGMENTATION

Image segmentation is the process of partitioning an image into parts or regions. This division into parts is often based on the characteristics of the pixels in the image. For example, one way to find regions in an image is to look for abrupt discontinuities in pixel values, which typically indicate edges. These edges can define regions. Another method is to divide the image into regions based on color values.

```
Level = graythresh(I);
```

level = graythresh(I) computes a global threshold (level) that can be used to convert an intensity image to a binary image with im2bw. level is a normalized intensity value that lies in the range [0, 1]. The graythresh function uses Otsu's method, which chooses the threshold to minimize the interclass variance of the black and white pixels.

### 3.5 EDGE DETECTION

In an image, an edge is a curve that follows a path of rapid change in image intensity. Edges are often associated with the boundaries of objects in a scene. Edge detection is used to identify the edges in an image. To find edges, you can use the edge function. This function looks for places in the image where the intensity changes rapidly, using one of these two criteria. Places where the first derivative of the intensity is larger in magnitude than some threshold. Places where the second derivative of the intensity has a zero crossing. edge provides a number of derivative estimators, each of which implements one of the definitions above. For some of these estimators, you can specify whether the operation should be sensitive to horizontal edges, vertical edges, or both. edge returns a binary image containing 1's where edges are found and 0's elsewhere. The Canny method differs from the other edge-detection methods in that it uses two different thresholds (to detect strong and weak edges), and includes the weak edges in the output only if they are connected to strong edges.

## 3.6 CORNER DETECTION

Corners are the most reliable feature you can use to find the correspondence between images. The following diagram shows three pixels one inside the object, one on the edge of the object, and one on the corner. If a pixel is inside an object, its surroundings (solid square) correspond to the surroundings of its neighbor (dotted square). This is true for neighboring pixels in all directions. If a pixel is on the edge of an object, its surroundings differ from the surroundings of its neighbors in one direction, but correspond to the surroundings of its neighbors in the other (perpendicular) direction. A corner pixel has surroundings different from all of its neighbors in all directions. The corner function identifies corners in an image. Two methods are available. The Harris corner detection method (The default) and Shi and Tomasi's minimum eigenvalue method. Both methods use algorithms that depend on the eigenvalues of the summation of the squared difference matrix (SSD). The eigenvalues of an SSD matrix represent the differences between the surroundings of a pixel and the surroundings of its neighbors. The larger the difference between the surroundings of a pixel and those of its neighbors, the larger the eigenvalues. The larger the eigenvalues, the more likely that a pixel appears at a corner.

## 3.7 GEOMETRIC TRANSFORMATION

If you know the transformation matrix for the geometric transformation you want to perform, you can create one of the geometric transformation objects directly, passing the transformation matrix as a parameter. For example, you can use a 3-by-3 matrix to specify any of the affine transformations. For affine transformations, the last column must contain 0 0 1 ([zeros(N,1); 1]). The following table lists affine transformations with the transformation matrix used to define them. You can combine multiple affine transformations into a single matrix.

## 4. CAMERA CALIBRATION

Camera resectioning is the process of estimation the parameters of a pinhole model approximating the camera that produced a given photography or video. Usually the pinhole camera parameters are represented in a camera matrix this process is often called camera calibration.

## 4.1 INTRINSIC PARAMETERS

The intrinsic matrix K contain 5 intrinsic parameters. These parameters encompass focal length, image sensor format, and principal point. The parameters represent focal length in terms of pixels where Mx and My are the scale factors relating pixels to distance and F is the focal length in terms of distance. The skew coefficient between the x and y axis, and is often o. Uo and Vo represent the principal point, which would be ideally in the center of the image. Nonlinear intrinsic parameters such as lens distortion are also important although they cannot be included in the linear camera model described by the intrinsic parameter matrix. Many modern camera calibration algorithms estimate these intrinsic parameters as well in the form of nonlinear optimization techniques.

## 4.2 EXTRINSIC PARAMETERS

When a camera is used, light from the environment is focused on an image plane and captured. This process reduces the dimensions of the data taken in by the camera from three to two dimension. Each pixel on the image plane therefore corresponds to a shaft of light from the original scene. Camera resectioning determines which incoming light is associated with each pixel on the resulting image. In an ideal pinhole camera, a simple projection matrix is enough to do this. With more complex camera systems, errors resulting from misaligned lenses and deformations in their structures. Can result in more complex distortions in the final image. The camera projection matrix is derived from the intrinsic and extrinsic parameters of the camera, and is often represented by the series of transformation. Camera resectioning is often used in the application of stereo vision where the camera projection matrices of two cameras are used to calculate the 3D world coordinates of a point viewed by both cameras.

## 5. STEREO VISION

Stereo vision is an area of computer vision focusing on the extraction is an area of 3D information from digital images. The most researched aspect of this field is stereo matching given two or more images so that their 2D positions can be converted into 3D depths, producing as result of 3D estimation of the scene. As many other breakthrough ideas in computer science stereovision is strongly related to a biological concept, namely when a scene is viewed by someone with both eyes and normal binocular vision. By aid of stereoscopic vision, we can perceive the surrounding in relation with our bodies and detect objects that are moving towards or away from us.

## 5.1 DISPARITY ESTIMATION

The whole concept of stereo matching is based on finding correspondences between the input images. In this exercise, correspondence between two points is determined by inspecting the pixel neighborhood N around both points. The pairing that has the lowest sum of absolute differences is selected as a corresponding point pair. In practice, a matching block is located for each pixel in an image. The relative difference in the location of the points on the image planes is the disparity of that point. Due to the assumption of being constrained into a one-dimensional search space, these disparities can be represented as a 2D disparity map which is the same size as the image. Disparity of a point is closely related to the depth of the point. This is essentially a block matching scheme familiar from video compression, although in this application the search space can be constrained to be horizontal only. The matching block size is one of the most important parameters that affect the outcome of the estimation. Smaller blocks can match finer detail, but are more prone to errors, while large blocks are more robust, but destroy detail. In this assignment, a symmetric, square block of radius r has pixels.

## 5.2 LOCAL FEATURES

Local features and their descriptors are the building blocks of many computer vision algorithms. Their applications include image registration, object detection and classification, tracking, and motion estimation. Using local features enables these algorithms to better handle scale changes, rotation, and occlusion. The Computer Vision System Toolbox provides the FAST, Harris, and Shi & Tomasi corner detectors, and the SURF and MSER blob detectors. The toolbox includes the SURF, FREAK, BRISK, and HOG descriptors. You can mix and match the detectors and the descriptors depending on the requirements of your application. Local features refer to a pattern or distinct structure found in an image, such as a point, edge, or small image patch. They are usually associated with an image patch that differs from its immediate surroundings by texture, color, or intensity. What the feature actually represents does not matter, just that it is distinct from its surroundings. Examples of local features are blobs, corners, and edge pixels.

## 5.3 FEATURES DETECTION

Feature detection selects regions of an image that have unique content, such as corners or blobs. Use feature detection to find points of interest that you can use for further processing. These points do not necessarily correspond to physical structures, such as the corners of a table. The key to feature detection is to find features that remain locally invariant so that you can detect them even in the presence of rotation or scale change.

## 5.4 FEATURES EXTRACTION

Feature extraction is typically done on regions centered around detected features. Descriptors are computed from a local image neighborhood. They are used to characterize and compare the extracted features. This process generally involves extensive image processing. You can compare processed patches to other patches, regardless of changes in scale or orientation. The main idea is to create a descriptor that remains invariant despite patch transformations that may involve changes in rotation or scale.

## 5.5 CASCADE OBJECT DETECTOR

The Computer Vision System Toolbox cascade object detector can detect object categories whose aspect ratio does not vary significantly. Objects whose aspect ratio remains approximately fixed include faces, stop signs, or cars viewed from one side. The vision. CascadeObjectDetector System object detects objects in images by sliding a window over the image. The detector then uses a cascade classifier to decide whether the window contains the object of interest. The size of the window varies to detect objects at different scales, but its aspect ratio remains fixed. The detector is very sensitive to out of-plane rotation, because the aspect ratio changes for most 3D objects. Thus, you need to train a detector for each orientation of the object. Training a single detector to handle all orientations will not work.

## 5.6 POSITIVE SAMPLES

You can specify positive samples in two ways. One way is to specify rectangular regions in a larger image. The regions contain the objects of interest. The other approach is to crop out the object of interest from the image and save it as a separate image. Then, you can specify the region to be the entire image. You can also generate more positive samples from existing ones by adding rotation or noise, or by varying brightness or contrast.

## 5.7 NEGATIVE SAMPLES

Negative samples are not specified explicitly. Instead, the trainCascadeObjectDetector function automatically generates negative samples from user-supplied negative images that do not contain objects of interest. Before training each new stage, the function runs the detector consisting of the stages already trained on the negative images. If any objects are detected from these, they must be false positives. These false positives are used as negative samples. In this way, each new stage of the cascade is trained to correct mistakes made by previous stages.

## 5.8 FORE GROUND DETECTOR

The foreground detector system object compares a color or grayscale video frame to a background model to determine whether individual pixels are part of the background or the foreground. It then computes a foreground mask. By using background subtraction, you can detect foreground objects in an image taken form a stationary camera.

## 5.9 DEPTH ESTIMATION

While the disparity map estimated from the stereo pair already distinguishes between objects at different distances, disparity is not the same as depth.

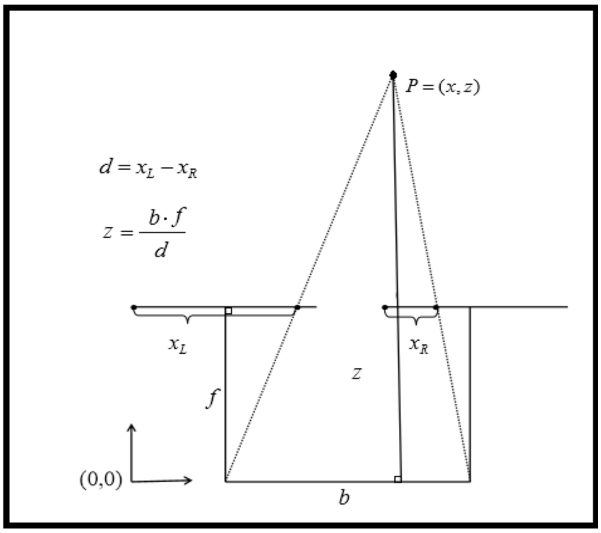

**Fig -1**: Triangulation

The relation depends on the camera configuration. For efficient stereo matching, the concept of epipolar lines is essential. The horizontal translation in the camera positions between the stereo pair should only create differences in the horizontal direction. This allows for the matching to happen only in one direction, along an epipolar line, greatly reducing the search space. However, this is rarely the case in a realistic capture setting. Camera misalignment makes it impossible to reliably search only on the epipolar lines. This is why a software rectification step is performed after capture. In short, the misalignment of the image planes can be corrected by rotating them in a 3-dimensional space.

## 5.10 CORRESPONDENCE

Correspondence, or feature matching, is common to most depth sensing methods. We discuss correspondence along the lines of feature descriptor methods and triangulation as applied to stereo, multi view stereo, and structured light. Subpixel accuracy is a goal in most depth sensing methods. It's popular to correlate two patches or intensity templates by fitting the phase, similar to the intensity correlation methods for stereo systems, the image pairs are rectified prior to feature matching so that the features are expected to be found along same line at about the same scale. Multiview stereo systems are similar to stereo however the rectification stage may not be as accurate, since motion between frames can include scaling, translation, and rotation. Since scale and rotation may have significant correspondence problems between frames, other approaches to feature description have been applied to MVS, with better results. A few notable feature descriptor methods applied to multi view and wide baseline stereo include the MSER method. Which uses a blob like patch, and the SUSAN method. Which defines the feature based on an object region or segmentation with a known centroid or nucleus around which the feature exists.

## 5.11 DEPTH GRADUALITY

As show in fig 2, sample Cartesian depth computations cannot resolve the depth field into a linear uniform grain size. In fact, the depth field granularity increases exponentially with the distance from the sensor, while the ability to resolve depth at long ranges is much less accurate.

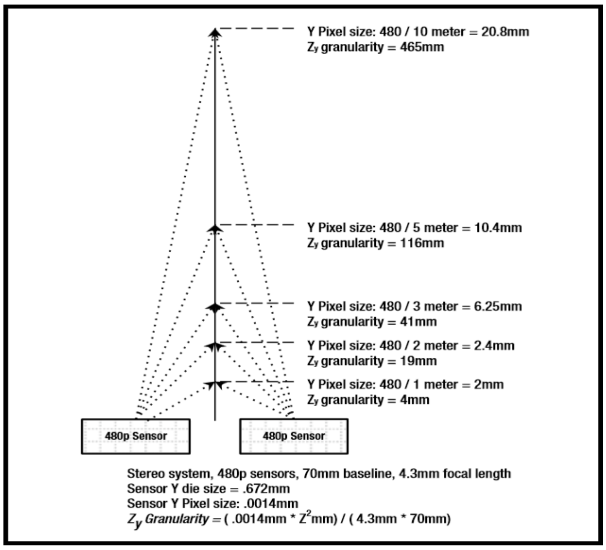
Y Pixel size: 480 / 10 meter = 20.8mm
$Z_y$ granularity = 465mm
Y Pixel size: 480 / 5 meter = 10.4mm
$Z_y$ granularity = 116mm
Y Pixel size: 480 / 3 meter = 6.25mm
$Z_y$ granularity = 41mm
Y Pixel size: 480 / 2 meter = 2.4mm
$Z_y$ granularity = 19mm
Y Pixel size: 480 / 1 meter = 2mm
$Z_y$ granularity = 4mm
480p Sensor
480p Sensor
Stereo system, 480p sensors, 70mm baseline, 4.3mm focal length
Sensor Y die size = .672mm
Sensor Y Pixel size: .0014mm
$Z_y$ Granularity = ( .0014mm * $Z^2$mm) / ( 4.3mm * 70mm)

**Fig -1**: Depth Graduality of Stereo Camera System

## 5.12 MONOCULAR DEPTH PROCESSING

Monocular, or single sensor depth sensing, creates a depth map from pair of images frames using the motion form frame to create the stereo disparity. The assumptions for stereo processing with a calibrated fixed geometry between stereo pairs assumption for stereo processing with a calibrated fixes geometry between stereo pairs do not hold for monocular methods, since each time the camera moves the camera pose must be recomputed. Camera pose is 6 degree of freedom equation, including x, y, z linear motion along each axis and roll, pitch and yaw rotational motion about each axis. In monocular depth sensing methods, the camera pose must be computed for each frame as the basis for comparing two frames and computing disparity.

### 5.13 SURFACE RECONSTRUCTION AND FUSION

A general method of creating surfaces from depth map information is surface reconstruction. Computer graphics methods can be used for rendering and displaying the surfaces. The basic idea is to combine several depth maps to construct a better surface model, including the RGB 2D image of the surface rendered as a texture map. By creating an iterative model of the 3D surface that integrates several depth maps form different viewpoints, the depth accuracy can be increased, occlusion can be reduced or eliminated, and a wider 3D scene viewpoint is created. The work of Curless and Levoy Present a method of fusing multiple range images or depth maps into a 3D volume structure. The algorithm renders all range images as iso surfaces into the volume by integrating several range images. Using a signed distance surfaces, the new surfaces are integrating into the volume for a cumulative best guess at where the actual surfaces exist. Of course, the resulting surface has several desirable properties, including reduced noise, reduced holes, reduced occlusion, multiple viewpoints, and better accuracy.

### 5.15 NOISE

Noise is another problem with depth sensors, and various causes include low illumination and, in some cases, motion noise, as well as inferior depth sensing algorithms or systems. Also, the depth maps are often very fuzzy, so image per processing may be required. To reduce apparent noise. Many prefer the bi lateral filter for depth map processing, since it respects local structure and preserves the edge transitions. In addition, other noise filters have been developed to remedy the weakness of the bi lateral filter, which are well suited to removing depth noise, including the Guided Filter, which can perform edge preserving noise filtering like the bi lateral filter, the edge avoiding wavelet method, and the domain transform filter.

### 5.16 DETECTION AND TRACKING

In the tracing mode you must tracing the points tracker. As you track the points. Some of them will be lost because occlusion. If the number of points being tracked falls below a threshold, that means that the face is no longer being tracked. You must the switch back to the detection mode to try re acquire the face.

## 6. CONCLUSION

Stereo vision is more important for identify the object motions and estimate the velocity of object move towards the system. Human vision system helps us to do lot of processes like playing games, driving so the robot have the sense of depth it is more useful. In future stereo vision take major role on the advance driver assistant and artificial intelligence. Our stereo vision system has various algorithm for accurate depth estimation. It can trach objects estimate their depth from camera. Also, system have ability to move to cameras along x and y axis synchronously. It is more useful in drones and capturing a 3d image. Stereo vision also used for making 3d movies in industrial robots it helps to navigate the welding tools with high accuracy and identify the failures on the work piece using inspection. Development of stereo vision system make the robot easily interact with various application. It reduces the complex sensor arrangement for various parameter sensing because it can identify the text, shape of objects, human faces.

## REFERENCES

[1] Stereopsis, Available from http://en.wikipedia.org/wiki/Stereopsis, Accessed: 15/04/2017.

[2] R. Szekely, Computer Vision: Algorithms and Applications, Springer, 2010.

[3] D. Marr, T. Poggio, “A Computational Theory of Human Stereo Vision”, Proceedings of the Royal Society of London. Series B, Biological Sciences, vol. 204, no. 1156, pp. 301-328, May 1979

[4] R. A. Lane, N. A. Thacker, Stereo vision research: An algorithm survey, Technical Report 94/16, University of Sheffield, 1994.

[5] M.J. McDonnell, “Box filtering techniques”, Computer Graphics and Image Processing, vol. 17, pp. 65–70, 1981.

[6] Z.F. Wang, Z.G. Zheng, “A Region Based Stereo Matching Algorithm Using Cooperative Optimization”, IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 2008.

[7] A. Klaus, M. Sormann, K. Kaner, “Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure”, ICPR, vol. 3, pp. 15-18, 2006.

[8] S.K. Gehrig, U. Franke, “Improving stereo sub-pixel accuracy for long range stereo”, IEEE International Conference on Computer Vision, pp. 1-7, 2007.

[9] V. Venkateswar, R. Chellappa, “Hierarchical Stereo and Motion Correspondence Using Feature Groupings. International Journal of Computer Vision, vol. 15, pp. 245-269, 1995.

[10] S. Birchfield, C. Tomasi, “Depth Discontinuities by Pixel-to-Pixel Stereo”, Proceedings of the International Conference on Computer Vision, vol. 1, pp. 489-495, 1999.