Stereo Vision
Computer Depth Perception

Paul Munro
159.731 Machine Vision
Computer Science
Massey University, Albany
New Zealand
Email: paulonthego@yahoo.com

Antony P. Gerdelan
159.731 Machine Vision
Computer Science
Massey University, Albany
New Zealand
Email: gerdelan@gmail.com

Abstract

Stereo Vision is an area of study in the field of Machine Vision that attempts to recreate the human vision system by using two or more 2D views of the same scene to derive 3D depth information about the scene.
Depth information can be used to track moving objects in 3D space, gather distance information for scene features, or to construct a 3D spatial model of a scene.
As an emerging technology, Stereo Vision algorithms are constantly being revised and developed, and as we will discuss in this paper, many alternative approaches exist for implementation of a Stereo Vision system.
This paper introduces Stereo Vision as an area of current international research, presents some interesting applications of Stereo Vision, and discusses implementation of a Stereo Vision system.

Introduction

The binocular (*two-eyed*) human vision system captures two different views of a scene. The human brain processes each view and matches similarities. Most of the information captured in each a particular view is congruent with the information captured in the other, however, some information is not (refer to *figure 1*). The differences allow the human brain to build depth information.

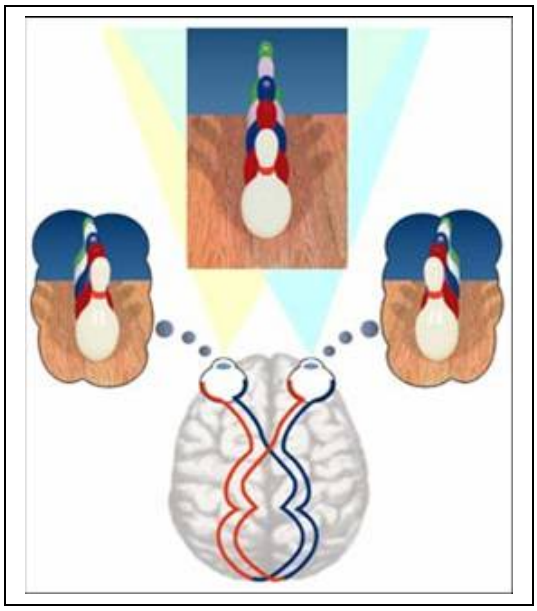


Figure 1 – Two views of a scene as captured by the brain

The ability of a machine to capture 3D information from the real world in a similar fashion to a human being is of great interest to science and industry. Research is being conducted in Stereo Vision to unlock the visual real world environment for intelligent machine participation.

The manufacturing industry has maintained an interest in automating production roles, and researches are being done into Stereo Vision to automate spatially-

perceptive manufacturing processes in the automotive, aircraft, and shipbuilding industries [20].

Stereo Vision systems are being developed to inspect infrastructure in human-inaccessible tunnels and pipes, and long sections of road and bridges [19].

The medical procedures involved with anthropometry and plastic surgery may be augmented by new systems capable of capturing and reproducing 3D information about the human body.

Stereo Vision is of particular interest to robotics. Several major projects have been undertaken by European automobile manufacturers to automate driving and navigation of road vehicles. Robots are being developed to build and flesh-out accurate three-dimensional maps of both indoor and outdoor areas.

APPLICATIONS

The applications of Stereo Vision detailed in this paper focus on two interesting, and related areas of machine vision research – the collection of 3D environment information for input to decision making navigation systems of autonomous vehicles, and the 3D reconstruction of real world environments by moving cameras (often also mounted on autonomous vehicles). Other research, with Stereo Vision systems mounted in fixed positions, exist to extend capabilities of security monitoring systems, and improve human face-recognition and tracking algorithms [14]. Research is also being done to create human-wearable Stereo Vision systems to assist the blind [15]. These technologies, however, will not be discussed in this section.

Autonomous Road Vehicle Navigation

Autonomous vehicles (or robots) employing Stereo Vision techniques must operate in *real time*, and thus are driving research for faster and more efficient Stereo Vision

algorithms, whilst retaining enough accuracy to build a navigable 3D map, track moving obstacles, and eliminate a reasonable volume of noise.

Autonomous vehicles are primarily interested in identifying, classifying, and tracking the movement of obstacles, in order to plan paths of movement that avoid collision. Betrozzi *et al* [13] give us a breakdown of the common obstacle recognition model (refer to *figure* 2).

Road vehicle vision systems and algorithms, if they are to be broadly applicable, must be robust enough to handle changes to:

- Road
- Traffic
- Illumination
- Weather.

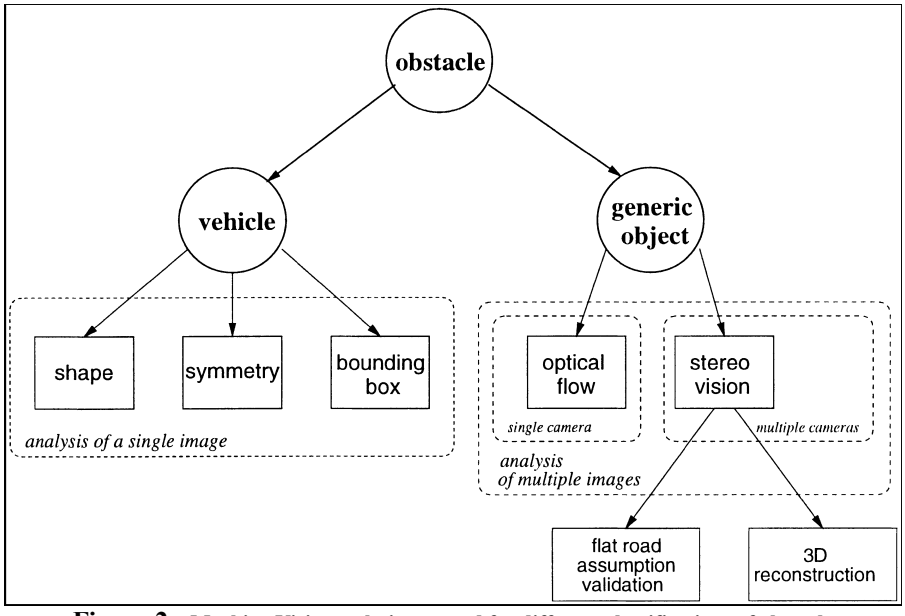


Figure 2 - Machine Vision techniques used for different classifications of obstacle

We can see that different subsets of Machine Vision algorithms are used for analysing different types of obstacles. Betrozzi [13] gives us the abstract model used in autonomous road vehicles in *figure 2*. Moving obstacles are positioned in the conceptual 3D space created by the Stereo Vision system, and given simplified dimensions by surrounding the object with a *bounding box*. The autonomous vehicle then requires minimum calculation to plan a path that will avoid collision with the

obstacle. Other obstacles are tracked using *optical flow*, and compared to the velocity vector of the car to determine if they are moving or stationary obstacles. Additionally, we can see that road vehicles perform some simple 3D reconstruction of the road ahead to assist forward-thinking path planning features of the navigation system.

Figure 3 – The VaMP prototype autonomous road vehicle

Many researches with autonomous road vehicles have been conducted by various universities, automobile manufacturers and military contractors, however very few have produced more than ponderous results. Some autonomous road vehicles (or *Smart Cars*), have successfully driven over long distances, and produced very promising experimental results, driving in real traffic conditions at high speed. The following experimental vehicles are amongst the most successful:

- The VaMP prototype (*figure* 3) was driven from Munich, Germany to Odense, Denmark [16] in 1995
- The RALPH system was tested using the NavLab 5 Stereo Vision sytem over a journey from Pittsburgh, PA to San Diego, CA in 1995.
- The ARGO experimental vehicle was driven by the GOLD system for nearly 2000 km throughout Italy during the *MilleMiglia in Automatico Tour* [17] in 1998.

Betrozzi et al. tell us from experience with the ARGO vehicle [13] that the Stereo Vision systems of autonomous road vehicles must meet a number of different

challenges, and so several subsystems (refer to *figure 4*), with specially designed algorithms, must be created to provide:

- lane markings detection
- traffic signs recognition
- obstacle identification
- filter out shadows on the road
- adapt if other road vehicles that obscure visibility

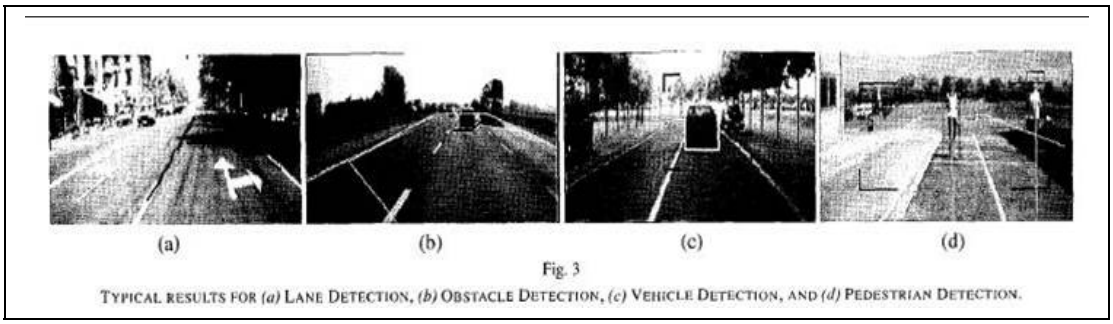


Figure 3 – Several of the Stereo Vision sub-systems used in the ARGO road vehicle

All-terrain Robot Navigation

Until now NASA moon rovers have been *tele-operated* from Earth. There is an enormous time delay between communications to and from Earth, which presents a major problem to NASA. Autonomous robot rovers are being designed at Carnegie Mellon University to fully automate small rover journeys over lunar terrain using a Stereo Vision-based approach [17]. The CMU team has adopted a very simple navigation algorithm for their lunar rover:

1. Analyse images of area directly in front of rover
2. Identify obstacles
3. Identify terrain type (e.g loose, hard rock)
4. Determine 3D positions, and put in a small grid
5. Choose the best of 8 possible paths through grid

The Stereo Vision system of the CMU lunar rover analyses terrain immediately ahead of the vehicle, and tries to identify obstacle features. Each obstacle triangulated in 3D

space, and awarded a *terrain roughness* score (by an undisclosed filtering/classification system). The obstacles are then plotted on a simple grid, representing the area in front of the rover (refer to *figure 5*). The darker squares in *figure 5* represent those obstacles with higher *terrain roughness* ratings.

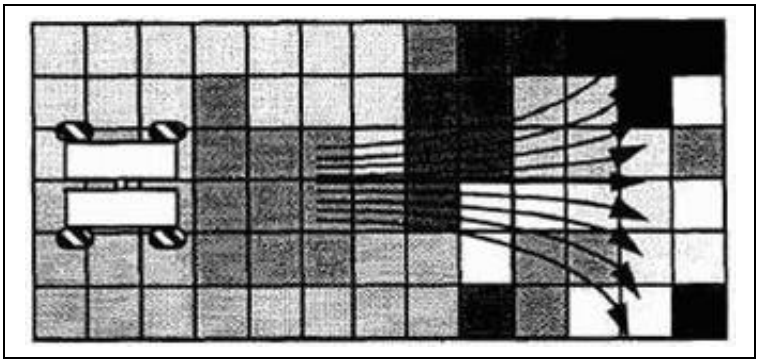


Figure 5 – Plotting obstacles on a grid map of the area ahead of a lunar rover

Again, with reference to *figure 5,* the rover navigation system sums the scores of every cell in the grid passed, for each of nine hard-coded paths. The system then selects the path with the lowest total *score* through the grid. Presumably an additional system would be required to ensure the long-term path of the rover was valid, and that the rover would be able to reverse or turn on the spot, should all possible paths be obstructed.

Method

Triangulation

The same method that is used in navigation and surveying is used to calculate depth. Basic triangulation uses the known distance of two separated points looking at the same scene point. From these parameters the distance to the scene point can be calculated. This same basic idea is used in stereo vision to find depth information from two images. *Figure 6* below graphically shows the geometry.

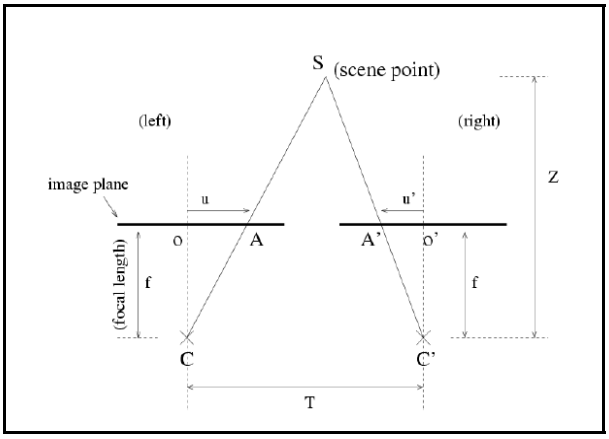


Figure 6 – Triangulation in a Stereo Vision system

In the above arrangement, two cameras (*C,C'*) see the same feature point (*S*). The location of the point in the two image planes is denoted by *A* and *A'*. When the cameras are separated by a distance T, the location of A and A' from the cameras normal axis will differ (denoted by *U,U'*). Using these differences, the distance (*Z*) to the point can be calculated from the following formula:

$$Z = f\frac{T}{U - U'} \quad (1)$$

In order to calculate depth however, the difference of *U* and *U'* need to be established. The image analysis techniques used to find the differences in the images are the focus of the next section.

Disparity

As mentioned above, differences between two images gives depth information. These differences are known as disparities. The key step to obtaining accurate depth information is therefore finding a detailed and accurate disparity map. Disparity maps can be visualised in greyscale. Close objects result in a large disparity value. This is translated into light greyscale values. Objects further away will appear darker.

Obtaining depth information is achieved through a process of four steps. Firstly the cameras need to be calibrated. After calibrating the cameras the assumption is made that the differences in the images are on the same horizontal or *epipolar* line [4]. The secondly step is the decision as to which method is going to be used to find the differences between the two images. Once this decision is made, an algorithm to obtain the disparity map needs to be designed or decided on. The third step is to implement the algorithm to obtain the disparity information. The final step is to use the disparity information, along with the camera calibration set in step one, to obtain a detailed three dimensional view of the world.

This report focuses on the basic ideas behind the algorithms used to obtain disparity information. The other steps are relatively straight forward in their operation and implementation. It should be noted that even within the algorithms described below; there is ongoing research and therefore many different implementations.

There are many algorithms used to find the disparity between the left and right images. Additionally, there is a large amount of ongoing research into finding quicker and more accurate algorithms. However there are two commonly used algorithms that are currently used to find disparity. The first method is *feature-based* [1]. The second method is an *area-based* statistical method. Because they are widely used, we will focus on these two methods in this report.

Feature Based Disparity

This method looks at features in one image and tries to find the corresponding feature in the other. The features can be edges, lines, circles and curves. Nasrabadi [2] applies a curve segment based matching algorithm. Curve segments are used as the building block in the matching process. Curve segments are extracted from the edge points detected. The centre of each extracted curve is used as the feature in the matching process.

Medioni and Nevatia [3] uses segments of connected edge points as matching primitives. Stereo correspondence is achieved through minimising the differential disparity measure for global matching, by taking into account things such as end points and segment orientation.

For each feature in the left hand image (Q_L) there needs to be a similar feature in the right hand image (Q_R). A measure of similarity is needed to associate the two features. This measure is given by the following formula adapted from Candocia and Adjouadi [1].

$$\Psi(L \rightarrow R) = \frac{1}{N_L} \sum_{q=1}^{h} \frac{1}{(D_q + 1)} \quad (2)$$

Where N_L = total number of features in the left image. $\frac{1}{N_L}$ is the weight associated with a matched feature. h is the minimum number of features found in either image, ie $h = \min(N_L, N_R)$. D_q is the minimum distance between a matched feature in the left and right images.

The features in the right image, within a constrained search area, with the highest similarity coefficient over a threshold are associated. These are then compared globally to other associated features to check for consistency. The difference in location of these features gives the disparity.

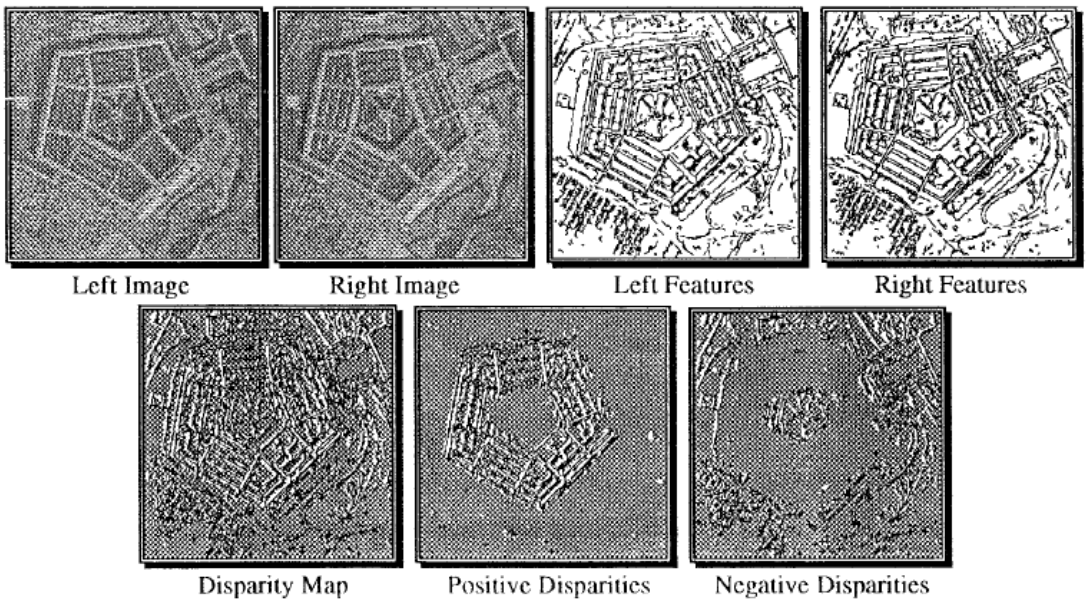


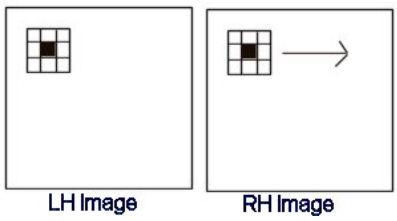
TABLE I
RESULTS OF THE STEREO FEATURE MATCHING TECHNIQUE

Scene	Features	Features Matched	% Matched	Features Corrected	% Corrected	Time
Pentagon	13491	10928	81.0%	1223	11.2%	4h 9m 36s
Fruit	7525	6159	84.9%	655	10.6%	3h 37m 40s
Renault	2459	2093	85.1%	226	10.8%	1h 48m 20s
Stereogram	6800	6257	92.0%	0	0	15m 56s

Results of the above feature based algorithm from Candocia and Adjouadi [1]

Area Based Disparity

There are two techniques that are used in this algorithm. In both these methods a window is placed on one image. The other image is scanned using the same size window. The pixels in each window are compared and operated on. These are then summed to give a coefficient for the centre pixel. These techniques have been developed by Okutomi and Kanade [5]



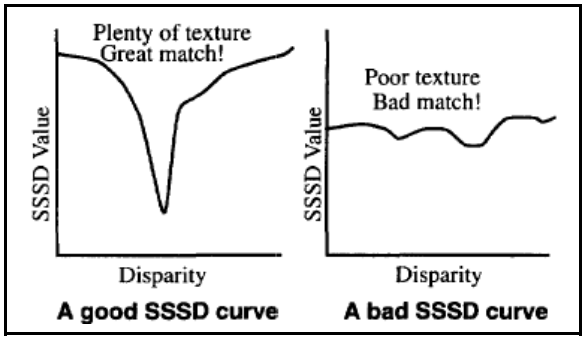
The first operation described is correlation. The output of the scanning window is convolved with the first and the location that gives the highest convolution coefficient is deemed to be the corresponding area. The correlation coefficient is given by:

$$C_{LR} = \sum_{[i,j] \in \text{Window}} L(i,j)R(i,j) \qquad (3)$$

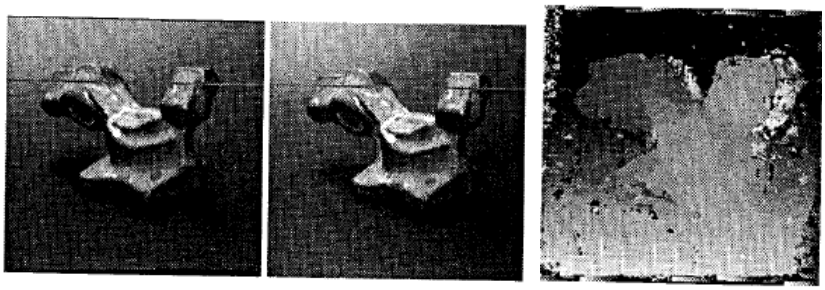
The second method uses the same window principle, but uses the sum of squared differences (*SSD*). This examines the pixel values in both windows and estimates the

disparity by calculating the SSD coefficients. In this method the SSD coefficient needs to be minimised. The formula for SSD is:

$$SSD = \sum_{[i,j] \in Window} [L(i, j) - R(i, j)]^2 \qquad (4)$$



SSD values over scan area taken from Ross [6]

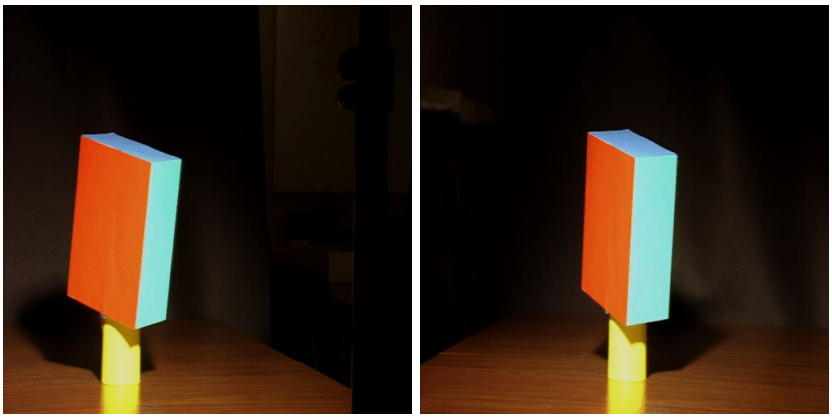


LH Image RH Image Disparity Map

Taken from Cochran and Medioni [7]

OpenCV Implementation

OpenCV [8] has a predefined library that includes a method of obtaining disparity maps for a given stereo pair. This was implemented using the code given in the appendix. The following result was obtained.



LH Image RH Image

Disparity Map

After trying many different options this is the best result that was obtained.

The disparity map shows streaking errors. This is attributed to the fact that OpenCV, misses important disparity jumps at the edges of objects and therefore assume the wrong disparities in later search stages [9].

Discussion

Each algorithm has distinct advantages and disadvantages. These generally relate to speed and accuracy. There are also a few considerations and assumptions that are made in all stereo vision algorithms

Advantages

The main advantage of the feature based algorithm is its speed. The process of finding features in both images, then calculating the disparity can easily be done in real time.

The area based method provides dense disparity maps. Other improvements can be obtained by pre-processing the images before implementing the algorithm. [12]

Disadvantages

While the feature based method is fast, the biggest disadvantage is it produces sparse disparity maps. Even with feature rich stereo pairs this method cannot produce maps as detailed as the area based method. For this reason most current stereo vision implementations use area based approaches.

The main problem with area based disparity is the time it takes to accurately calculate the disparity. The scanning of the image and the calculation of correlation coefficients turns out to be computationally expensive. There is balance between accuracy of the calculation and the speed in which it is done. New research shows the correlation calculation running in parallel on field programmable gate arrays (FPGA) to speed the process up. [11]

Another problem with this method is the selection window size. The window must be large enough to include a range of intensity variations and be small enough to

distinguish the differences between the images. Okutomi and Kanade [10] have developed an algorithm in the area of adaptive windowing. This is where the window size is adjusted according to the image texture to give better disparity estimates.

Assumptions/Limitations

There are two main assumptions that are made when using stereo vision algorithms. These are:

1. The Corresponding image regions are similar
2. A point in one image must be matched by only one point in the other.

These assumptions lead to the following problem areas; the first problem area is *occlusion*. This is when pixels or features in one image are missing in the other. There are many ways this is dealt with, however to reduce processing time, most algorithms consider this to be an unimportant case. Leaving this important feature out, produces errors in the disparity maps and therefore in the depth estimation. There is research specifically in this area to specifically identify occluded regions [12]. The other problem area is *repetition*. This is when features or areas are repeated along the epipolar line. This can lead to false match errors. There are algorithms to deal with these problems, but all require additional processor time.

Appendix

OpenCv code

```
#include "cv.h"
#include "highgui.h"
#include "cvaux.h"
#include "stdio.h"


int main (int argc, char ** argv){

      IplImage* srcLeft = cvLoadImage("Cube_left.jpg",1);
      IplImage* srcRight = cvLoadImage("Cube_right.jpg",1);
      IplImage* leftImage = cvCreateImage(cvGetSize(srcLeft),
IPL_DEPTH_8U, 1);
      IplImage* rightImage = cvCreateImage(cvGetSize(srcRight),
IPL_DEPTH_8U, 1);
      IplImage* depthImage = cvCreateImage(cvGetSize(srcRight),
IPL_DEPTH_8U, 1);

      cvCvtColor(srcLeft, leftImage, CV_BGR2GRAY);
      cvCvtColor(srcRight, rightImage, CV_BGR2GRAY);



      cvFindStereoCorrespondence( rightImage, leftImage,
CV_DISPARITY_BIRCHFIELD, depthImage, 255, 15, 8, 4, 8, 15 );

      cvNamedWindow("left", 1);
      cvNamedWindow("right", 1);
      cvNamedWindow("depth", 1);
      cvShowImage("left",leftImage);
      cvShowImage("right",rightImage);
      cvShowImage("depth",depthImage);
      cvWaitKey(0);

      cvSaveImage("depth_output.jpeg",depthImage);

      cvReleaseImage(&depthImage);
      cvReleaseImage(&rightImage);
      cvReleaseImage(&leftImage);
      cvReleaseImage(&srcLeft);
      cvReleaseImage(&srcRight);
      return 0;
}
```

References

[1] F. Candocia, and M. Adjouadi: "A similarity measure for stereo feature matching". IEEE Transaction on Image Processing, Vol. 6, pp. 1460-1464, 1997.

[2] N. Nasrabadi. "A Stereo vision technique using curve segments and relaxation matching," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14 no. 5, pp. 566–572, May 1992.

[3] G. Medioni, and R. Nevatia, "Segment-based stereo matching," *Comput. Vis., Graph., Image Process.*, vol. 31, pp. 2–18, July, 1985.

[4] P. Foggia, A . Limongiello,.and M. Vento, "A real-time stereo-vision system for moving object and obstacle detection in AVG and AMR applications,"
IEEE *Transactions on Image Process*, Vol. 6, No. 10, October 1997

[5] Okutomi, Masatoshi and Kanade, Takeo (1991) A Multiple-Baseline Stereo. *CVPR proceedings*, 1991.

[6] B. Ross "A Practical Stereo Vision System" *Computer Vision and Pattern Recognition, IEEE Computer Society Conference* June 1993 Page(s):148 – 153.

[7]S. Cochran, and G. Medioni, "Accurate surface description from binocular stereo" *IEEE Interpretation of 3D Scenes, 1989. Proceedings, Workshop* Nov. 1989 Page(s):16 – 23

[8] OpenCV *http://www.intel.com/technology/computing/opencv/index.htm*

[9] H. Sunyoto, W van der Mark , and D. Gavrila. "A Comparative Study of Fast Dense Stereo Vision Algorithms" *IEEE Intelligent Vehicles Symposium* June 2004

[10] Okutomi, Masatoshi and Kanade, Takeo "A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment". *IEEE Transactionson Pattern Analysis and Machine Intelligence, Vol. 16, No. 9*, Sept 1994

[11] Divyang K. Masrani, W. James MacLean, "A Real-Time Large Disparity Range Stereo-System using FPGAs," *icvs*, p. 13, Fourth IEEE International Conference on Computer Vision Systems (ICVS'06), 2006.

[12] A Cooperative Algorithm for Stereo Matching and Occlusion Detection, C. Zitnick and T. Kanade, tech. report CMU-RI-TR-99-35, Robotics Institute, Carnegie Mellon University, October, 1999.

[13] Bertozzi M., Broggi A., Fascioli A. "Vision-based intelligent vehicles: State of the art and perspectives", Dipartimento di Ingegneria dell'Informazione, Università di Parma, I-43100 Parma, Italy, and Dipartimento di Informatica e Sistemistica, Università di Pavia, I-27100 Pavia, Italy 1999.

[14] Matsumoto Y., and Zelinsky, A. "An Algorithm for Real-time Stereo Vision Implementation of Head Pose and Gaze Direction Measurement", Nara Institute of Science and Technology 8916-5 Takayamacho, Ikoma-city, Nara, Japan, and The Australian National University.

[15] Molton N., Se S., Brady J.M., Lee D., and Probert P. "A stereo vision-based aid for the visually impaired", Department of Engineering Science, University of Oxford, Oxford, OX1 3PJ, U.K, appears in *Image and Vision Computing* 16 (1998) 251–263.

[16] Maurer M., Behringer R., Thomanek F., Dickmanns E.D., "A compact vision system for road vehicle guidance", appears in *Proceedings of the 13th International Conference on Pattern Recognition*, Vienna, Austria, 1996.

[17] Broggi A., Bertozzi M., Fascioli A., "The 2000 km test of the ARGO vision-based autonomous vehicle", *IEEE Intelligent Systems* (1999) 55–64.

[18] Reid Simmons, Eric Krotkov, Lonnie Chrisman, Fabio Cozman, Richard Goodwin, Martial Hebert, Lalitesh Katragadlda, Sven Koenig, Gita Krishnaswamy, Yoshikazu Shinoda, and William Whittaker, Paul Klarer, "Experience with Rover Navigation for Lunar-Like Terrains", The Robotics's Institute, Carnegie Mellon University, Pittsburgh, PA 15213, and Sandia National Laboratories, Albuquerque, NM 87 185, appears in *IEEE Intelligent Systems*, 1995.

[19] S. W. Lawson and J.R.G. Pretlove, "Augmented reality for underground pipe inspection and maintenance", Mechatronic Systems and Robotics Research Group,
School of Mechanical and Materials Engineering, University of Surrey, Guildford, Surrey, GU2 5XH, United Kingdom.

[20] R. Bostelman, J. Albus, N. Dagalakis, A. Jacoff, "RoboCrane Project: An Advanced Concept for Large Scale Manufacturing", Intelligent Systems Division, National Institute of Standards and Technology, Gaithersburg, Maryland 20899