ABHIYANTRIKI
An International Journal of Engineering & Technology
(A Peer Reviewed & Indexed Journal)
Vol. 2, No. 11 (November, 2015) http://www.aijet.in/ eISSN: 2394-627X

Stereo Vision for Mobile Robots: A Review

Akhil Appu Shetty
Research Scholar
Deptt. of Instrumentation & Control Engineering
MIT, Manipal University, Manipal, India

C. Gurudas Nayak*
Associate Professor
Deptt. of Instrumentation & Control Engineering
MIT, Manipal University, Manipal, India

V. I. George
Professor
Deptt. of Instrumentation & Control Engineering
MIT, Manipal University, Manipal, India

Abstract

This paper discusses the concept of stereo vision with an application to mobile robots. The main centre of concentration is local stereo matching algorithms due to their speed of execution. Some of the work done on stereo matching in uneven lighting is discussed as this is one of the most important hurdles when it come to using vision for outdoor navigation. A methodology for carrying out the above mentioned research is also proposed.

Keywords*: Stereo Vision, Disparity Map, Stereo Matching, Mobile Robots.*

****Author for correspondence*** cg.nayak@manipal.edu

1. Introduction

Mobile robots can be used in many aspects of everyday life. Some museums use robots to patrol their galleries at night, monitoring air quality and humidity levels. Mobile robots also work in homes and businesses. Hospitals may use robots to transport medications. Some explore other planets or inhospitable areas on Earth, collecting geological samples. Others are useful in seeking out landmines in former battlefields.

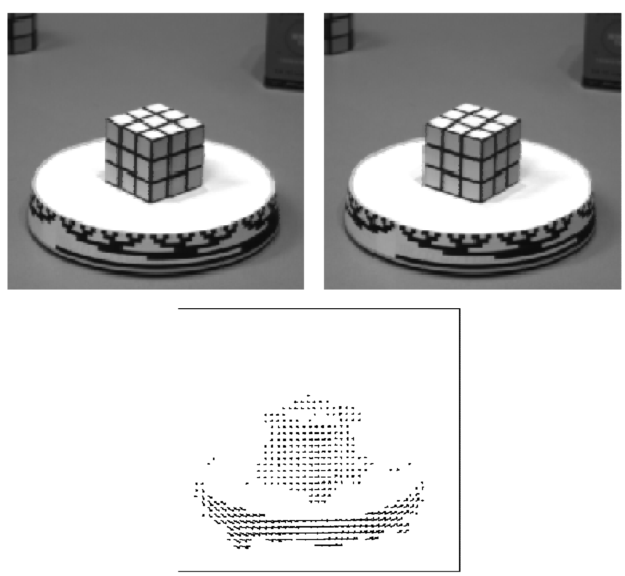

Fig. 1: Images captured in consecutive frames and optical flow representation of the movement of the objects [1]

Many of these applications demand autonomous systems which can work in unknown environments. The ability to perceive the environment is an important part in the design of mobile robots, especially in an unknown, unstructured environment. Generally sonar and radar can be used for this purpose, but the data acquired from them is not very dense. Another option that can be used is lasers. But results provided by them are computationally complex. The best solution would be to use cameras which are available at a much smaller price as compared to lasers and provide dense information about its surroundings. There are generally two methods to use cameras for navigation: optical flow and stereo vision.

Optical flow uses a single camera to capture images. Each captured frame is compared with the previous frame to detect changes in position of objects which is used to decide parameters like speed of object, direction of movement, etc. The disadvantage this method has is that the information obtained through optical flow is relative to the observer. Figure 1 shows the optical flow representation of the movement of the objects in the images captured consecutively.

For this reason, cameras are used as distance measuring sensors in the form of a stereo vision setup, which compares two images taken simultaneously from the two cameras and compares them, as it grants the robot the ability to detect and avoid obstacles, since it is a critical functionality deemed necessary for a moving platform. A stereo vision setup can be represented diagrammatically as shown in figure 2. The depth information can be extracted using the equation given below:

$$Z=f*B/d$$

Relating the above equation to figure 2, ‘f’ is the focal length of the cameras, ‘B’ is the baseline separating the two cameras, ‘Z’ is the distance of the point ‘P’ from the cameras (the depth), and ‘d’ is the disparity which can be obtained from the figure as

$$d= x- x'$$

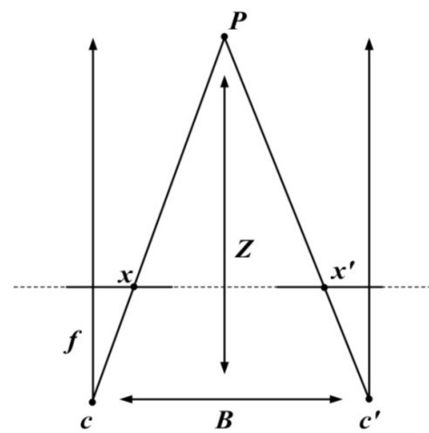

Fig. 2: Diagrammatic representation of a stereo vision setup [2]

A local stereo matching method will be used which considers a pixel in one image (e.g. the left one) and simply searches for the best match of this pixel in the other image. A local intensity function is used as a similarity measure for finding the matching cost between two pixels. A point in 3D space, when projected on two images will have the same intensity. The degree of similarity between two point sets can be calculated using a correlation measure. The most commonly used local correspondence method is block matching. It estimates the disparity at a point in one image by comparing a small surrounding region with a series of small regions from the other image.

2. Review of Literature

In [3], a method to use disparity maps, obtained through a stereo vision system, to detect obstacle free paths in real time is proposed by the authors. This system can be used to guide a visually impaired person in their surroundings. A fuzzy logic system is used that assigns a certainty to be a part of a free path to each group of pixels. Real outdoor images, presenting complex lighting conditions and scene ambiguities, are used. The stereo cameras will determine the location, direction and speed per frame of objects relative to the user in medium to long ranges (up to 25m). A stereo matching algorithm based on dynamic programming is used. The authors notice that the disparity values of obstacle free areas decrease slightly and linearly from bottom of the map to the top. The authors claim that even in non uniform lighting environments, their algorithm provides an obstacle free area to walk through. In [4], the authors claim that deciding the support region for local stereo algorithms is a challenging task. Here the authors propose an anchor based diagonal shape adaptive support region construction for every pixel which lies along the diagonal structure of the support region for stereo matching. The support region was constructed based on diagonal arm length, which is used for controlling the maximum size of the support and the confidence level which controls gray level similarity in the support region. Diagonal arms are selected based on the number of pixels which have similar gray levels as that of the anchor pixel (the pixel around which the support region is created). The maximum left and right span of each diagonal pixel is computed based on same gray level intensity. The performance of the algorithm is tested using Middlebury data sets. The proposed algorithm was compared with SAD, SSD and NCC algorithms. The authors claim that the proposed algorithm demonstrated superior characteristics in preserving image details and the disparity map generated is better among the mentioned methods.

In [5], the authors present a method which uses a triangular window, rather than the usual square or rectangle shaped windows, whose vertices are decided using a corner detection algorithm on the image. In the algorithm, these triangles are matched with respect to area and length of corresponding sides. The authors state that their algorithm is run in MATLAB. Middlebury data sets were used and the execution time was recorded between to be from 114s to 135s. The authors in [6] propose a new illumination invariant dissimilarity measure which can substitute the present intensity based measure sand is also rapidly computable. This method can be used by any stereo algorithm and hence improve the overall performance of the algorithm. The system makes use of HSL colour space rather than RGB. The authors also incorporate the gestalt theory on pixels during cost aggregation to assign right significance weights. The only disadvantage that the authors state is the in ideal lighting conditions, their algorithm might not perform very well when compared to the intensity based ones. Window size of 15X15 was used as this size was found to suppress noise better. Two identical algorithms were developed and tested. These algorithms used a gestalt based adaptive support weight aggregation scheme. One of the methods used the luminosity compensated dissimilarity measure (LCDM) and the other used the absolute differences (AD) as a dissimilarity measure. The authors claim that using this method, details of the image is preserved and it is able to compensate for luminosity non uniformities. It is also easy to be computed at high frame rates. This algorithm can be embedded in the first stage of a stereo algorithm. The data set used was from Middlebury stereo data sets. The implementation was carried out in MATLAB. But it was not fast enough to be used in real time. A C++ version could be reasonably fast.

3. Methodology

The methodology for this research is divided into three phases. In the first phase an accurate disparity map of the immediate environment is generated, indicating the nearby obstacles. Trials with different window sizes will be conducted to decide which the best size for this project is. In phase II, the disparity map obtained will be refined and obstacle points will be identified from it. In the next phase, the obstacle region will be segmented and this information will be fed to the path planning algorithm which will then generate the required control signals.

a) ***Phase I***: Phase I is explained in figure 3. Out of the two images obtained, one of the images is subjected to intensity variation to represent outdoor uneven lighting. This helps decide the threshold of variation for which my algorithm would work. The left and the right images are sent to the stereo matching block. A window based stereo matching algorithm is used. This algorithm itself is divided into three stages: matching cost estimation, cost aggregation and disparity generation. Windows of different sizes are used to find out which window size provides the best and the fastest disparity map. The computed disparity maps are compared to the ground truth to find the best match.

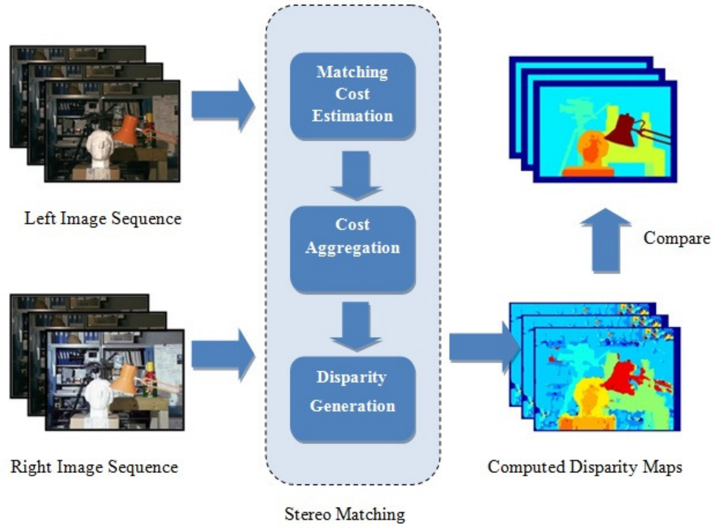

Fig. 3: Phase I

b) ***Phase II***: Once the disparity map is computed, it is refined to remove any noise induced during the stereo matching algorithm, as shown in figure 4. These disparity maps are then subjected to triangulation and the range for obstacle detection is estimated. The next stage involves the clustering of 3D points related to the obstacles in the range of distance decided in the previous step. From this, information about the obstacles in the immediate environment can be extracted.

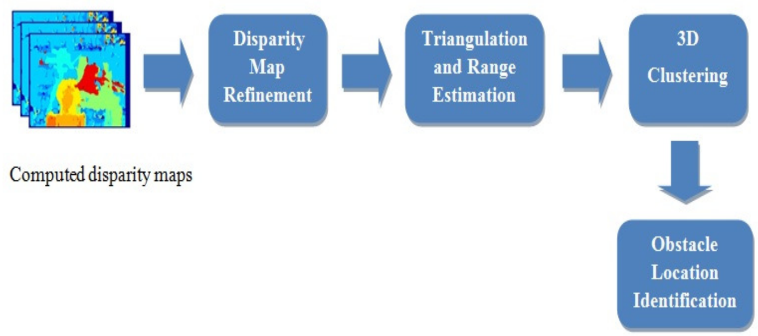

Fig. 4: Phase II

c) ***Phase III***: In the third phase, represented by figure 5, the obstacle region is segmented from the background, with the help of the data obtained from the previous stage. Once the obstacles are segmented, this information is used for decision making and path planning so that the robot can use this information to avoid the obstacles present in the immediate environment. Various data sets such as is Malaga stereo and laser urban data set, KITTI vision bench mark suite and the New College dataset will be used for

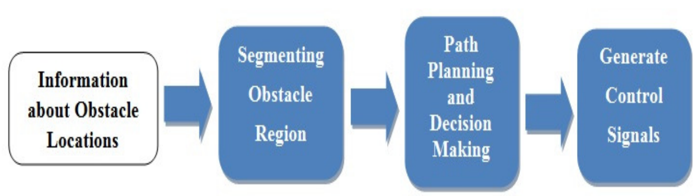

Fig. 5: Phase III

testing the proposed algorithm. Further datasets will be selected for variations of illumination and number of objects. The software used in this research will be OpenCV in the C++ framework & MATLAB with Image Processing Toolkit which is in-line with the techniques used in the proposed approach.

4. Conclusion

From the above discussion it is evident that the area of stereo matching, especially in uneven lighting, is an area which still needs improvement if we are to use it in real time environment.

References

[1] http://robotics.eecs.berkeley.edu/~sastry/ee20/vision3/node2.html [Last accessed 11/07/ 2015].
[2] https://commons.wikimedia.org/wiki/File:Stereo-camera-model.jpg [Last accessed 11/07/ 2015].
[3] Ortigosa, N., & Morillas, S. “Fuzzy free path detection from disparity maps by using least-squares fitting to a plane”, Journal of Intelligent Robotic Systems. Vol. 75, Issue 2, pp. 313-330, 2014.
[4] Raghavendra, U., Krishnamoorthi, M., & Karunakar, A. K. “Anchor diagonal based shape adaptive local support region for efficient stereo matching”, Signal, Image and Video Processing. Vol. 9, Issue 4, pp. 893-901, 2015.
[5] Zhang, X. W. Qu, Ma, S. J., Yang, Z. R., & Kong, L. F. “A dense stereo matching algorithm based on triangulation,” Journal of Computational Information Systems. Vol. 8, Issue 1, pp. 283-292, 2012.
[6] Nalpantidis, L., & Gasteratos, A. “Stereo vision for robotic applications in the presence of non ideal lighting conditions”, Journal of Image and Vision Computing. Vol. 28, Issue 6, pp. 940-951, 2010.