International Journal For Technological Research In Engineering
Volume 1, Issue 9, May-2014 *ISSN (Online): 2347 - 4718*

STEREO MATCHING ALGORITHMS FOR 3D RECONSTRUCTION FROM 2D IMAGES

Prashanth RD[1] (M. Tech), Honnaraju B[2] (Assistant Professor), Deepu R[3]
Department of Computer Science
Maharaja Institute of Technology, Mysore, India.

Abstract: Stereo matching is one of the most active research areas in computer vision. It has provoked a great deal of research into computer vision systems with two inputs that exploit the knowledge of their own relative geometry to derive depth information from the two views they receive. 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. In this paper, we present a novel algorithm used for stereo matching based on pixel-based matching in the cost computation, fixed window in the cost aggregation, and trivial assignment in the disparity computation, then we will present an algorithm used for the disparity refinement of the results obtained by the first algorithm In order to establish a software implementation and a collection of data sets to show the results of the algorithms, we have found a flexible Matlab implementation of the algorithms that enables to analyze algorithm. Finally, we will make a comparison between the algorithm we have used and other algorithms used for stereo matching.

I. INTRODUCTION

The 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. The differences allow the human brain to build depth information. In stereo matching, if two calibrated cameras observe the same scene point p (refer to figure 1), its 3D coordinates can be computed as the intersection of two such rays. This is the basic principle of stereo vision that typically consists of three steps:

- Camera calibration.
- Establishing point correspondences between pairs of points from the left and the right images.
- Reconstruction of 3D coordinates of the points in the scene.

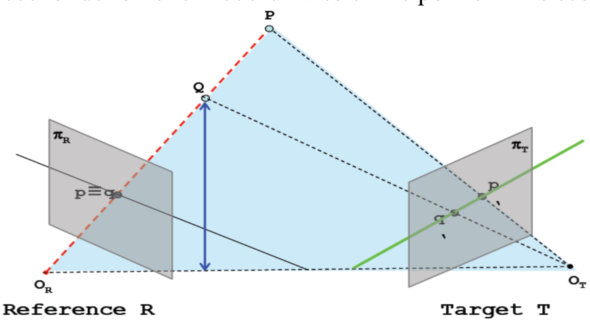

Fig. 1: Stereo Camera

If the two cameras are calibrated such that they will be perfectly aligned and with the same focal length, then the depth can be easily calculated as shown in the equations below (refer to figure 2):

By considering similar triangles (PO_RO_T and Ppp'):

$$\frac{B}{Z} = \frac{(B + x_T) - x_R}{Z - f}$$

$$B * (Z - f) = Z * (B + x_T - x_R)$$
$$B * f = Z * (x_R - x_T)$$
$$Z = \frac{B * f}{x_R - x_T}$$

Let $d = x_R - x_T$ is the disparity

And $B * f = constant$ for the pair of cameras

Then

$$Z = \frac{constant}{d}$$

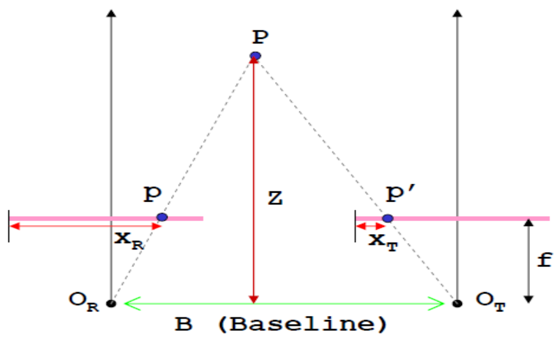

Fig. 2: Aligned Stereo Cameras with the same focal length

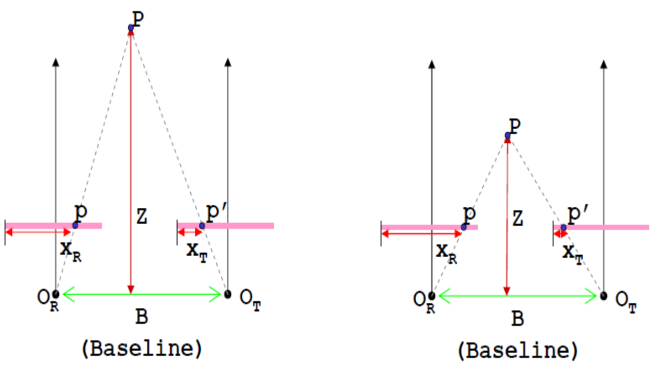

Fig. 3: Disparity and Depth Relationship

As we have seen in figure 3 that the depth of a pixel in a reference image (left image) can be determined knowing its disparity from its corresponding pixel in a target image (right image), so in the next section we will introduce the methodology of calculation of the disparity of a given pixel in a reference image from its corresponding pixel a target image.

II. METHODOLOGY

In this section, we describe the algorithm we used in the stereo matching. Basically there exist two different (not mutually exclusive) strategies for stereo matching.

- Local algorithms: In order to increase the SNR (reduce ambiguity) the matching costs are aggregated over a support window will be discussed later.
- Global (and Semi-global[2]) approaches:
- Many algorithms search for the disparity assignment that minimizes a certain cost function over the whole [1] stereo pair

$E(d) = E_{data}(d) - E_{smooth(d)}$

We used a Novel algorithm which consists of four main steps as follows:

- Matching cost computation.
- Cost aggregation.
- Disparity computation.
- Disparity refinement.

A. Matching cost computation

First we have to introduce the correspondence problem that tries to figure out which parts of an image correspond to which parts of another image as shown in figure 4

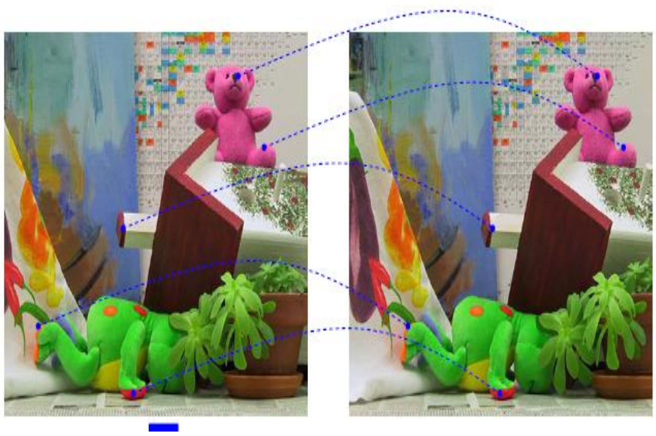

Fig. 4: The Correspondence Problem

We used the Absolute differences Pixel-based matching costs where the matching energy function is equal to the absolute difference between the pixel in the reference image and the pixel in the target image as shown in Fig 5

$e(x, y, d) = |I_R(x, y) - I_t(x + d, y)|$ Equation (1)

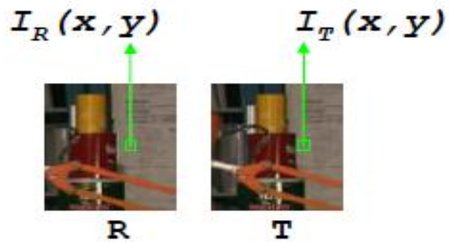

Fig. 5: Pixel Based Matching Cost

By applying Equation (1) using different disparities from 1 to D_{max} (maximum disparity) this will result in what is known as the disparity space image DSI.

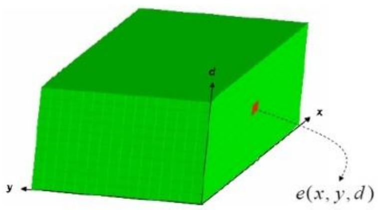

Fig. 6: DSI

Where DSI is a 3D matrix where each element $e(x, y, d)$ of the DSI represents the cost of the correspondence between $I_R(x, y)$ and $I_t(x + d, y)$

B. Cost aggregation

It is used in order to increase the SNR (reduce ambiguity) the matching costs are aggregated over a support window. In the proposed algorithm we aggregate matching costs of DSI horizontally then vertically then we used the simplest Fixed Window (FW) cost aggregation strategy, as shown in Figure9

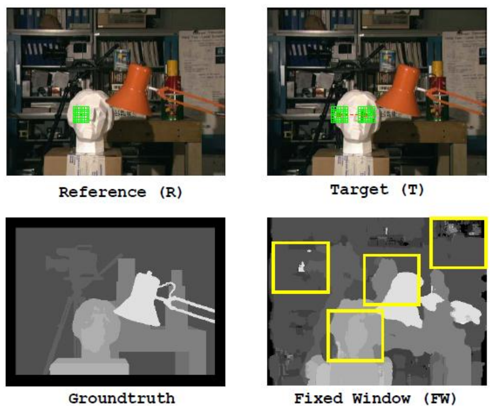

Fig. 9: Fixed Window Cost Aggregation.

B.1 Advantages of the FW algorithm

- Easy to implement.
- Fast, thanks to incremental calculation schemes.
- Runs in real-time on standard processors (SIMD).
- Has limited memory requirements.

- Hardware implementations (FPGA) run in real-time with limited power consumption (<1W).

Other approaches used in **Cost aggregation:**

- Using Shift-able Windows
- Using Multiple Windows
- Using Variable Windows:
- Segmentation

C. Disparity computation

This step aims at finding the best disparity assignment (e.g. the best path/surface within the DSI) that minimizes a cost function over the whole stereo pair. As mentioned above, differences between two images gives depth information. The key step to obtaining accurate depth information is therefore finding a detailed and accurate disparity map. Disparity maps can be visualized in grayscale. Close objects result in a large disparity value. This is translated into light grayscale values. Objects further away will appear darker. But generally in global stereo matching algorithms the energy function has two terms as the following:

$$E(d) = E_{data}(d) - E_{smooth(d)}$$

- The data term Edata measure how well the assignment fits to the stereo pair (in terms of overall matching cost). Several approaches rely on simple pixel-based cost functions but effective support aggregation strategies have been successfully adopted
- The smoothness/regularization Esmooth term explicitly enforces piecewise assumptions (continuity) about the scene. This term penalizes disparity variations and large variations are allowed only at (unknown) depth borders. Plausibility of depth border is often related to edges.

So finding the best assignment that minimizes the energy function a NP-hard problem

Relevant approaches are:

- Graph Cuts
- Belief Propagation
- Cooperative optimization

D. Disparity refinement

Most stereo correspondence algorithms compute a set of disparity estimates in some discretized space. For applications such as robot navigation or people tracking, these may be perfectly adequate. However for image-based rendering, such quantized maps lead to very unappealing view synthesis results (the scene appears to be made up of many thin shearing layers). To remedy this situation, sub-pixel disparity estimates can be computed in a variety of ways, including iterative gradient descent and fitting a curve to the matching costs at discrete disparity levels this provides an easy way to increase the resolution of a stereo algorithm with little additional computation. However, to work well, the intensities being matched must vary smoothly, and the regions over which these estimates are computed must be on the same (correct) surface. We used Sub-pixel interpolation where the sub-pixel disparity is obtained interpolating three matching costs with a second degree function as shown in figure17

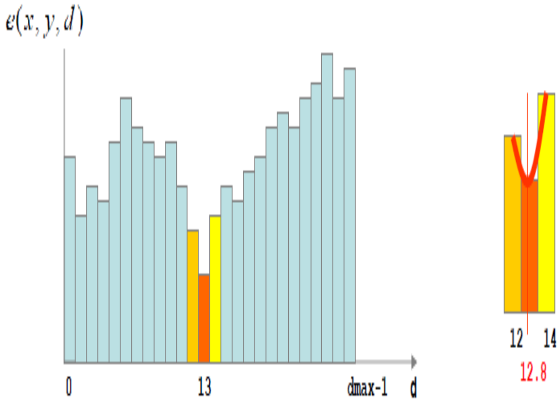

Fig. 17: Sub-Pixel Interpolation

This method is computationally inexpensive and reasonably accurate.

E. Other methods

Not all binocular stereo correspondence algorithms can be described in terms of our basic local algorithm. Here we briefly mention some additional algorithms that are not covered by our paper. A univalued representation of the disparity map is not essential. Multi-valued representations, which can represent several depth values along each line of sight, have been extensively studied recently, especially for large multi-view data set. Another way to represent a scene with more complexity is to use multiple layers, each of which can be represented by a plane plus residual parallax. Finally, deformable surfaces of various kinds have also been used to perform 3D shape reconstruction from multiple images.

III. EXPERIMENTAL RESULTS

In this section, we describe the experiments used to evaluate the stereo algorithms. Using the implementation framework we have found, we use the middle bury dataset for evaluation of stereo matching algorithms.

A. Resultant Stereo Match

The used image is Tsukuba figure 18

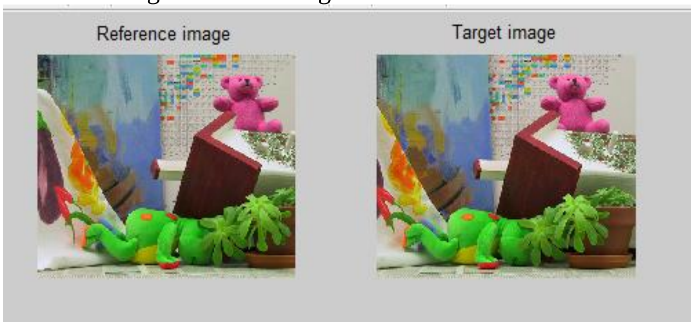

Fig. 18: Stereo Images

The resultant disparity map image without sub-pixel Interpolation is shown in figure 19

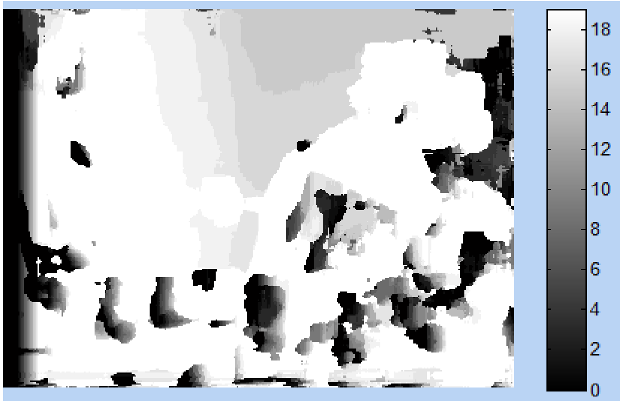

Fig. 19: Disparity Map without Sub-Pixel Interpolation

The resultant disparity map image with sub-pixel interpolation is shown in figure 20 where the scene is smoother

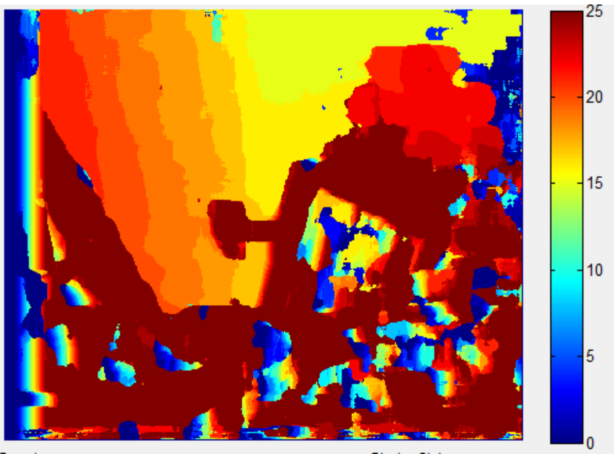

Fig. 20: Disparity Map with Sub-Pixel Interpolation

The resultant 3D Image with Azimuth and elevation variation is showed in figure21

Fig. 21: Reconstructed 3D Image

IV. CONCLUSION

In this paper we have introduced the methodology ofstereo matching, then we introduced a novel algorithm based on support window used in stereo matching, and we introduced other methods that could be used in stereo matching like:

- Area based algorithm in matching cost stage.
- Using shift-able windows, multiple windows, variable windows size, and segmentation in cost aggregation stage.
- Graph cuts, belief propagation, cooperative optimization in disparity computation stage.
- Iterative gradient descent in curve fitting stage.

We have shown the role of stereo matching in the formation of a completed 3D scene using the algorithm explained. Although that the algorithm we have used provide the best accuracy, and it is widely used in many applications, due to its fast speed, there are many applications where this algorithm is used such as:

1) 3D Tracking
   - people counting (building, bus, train)
   - Safety
   - Surveillance and security
2) 3D Graffiti detection
3) 3D Scanning
4) Space time stereo
5) 3D motion detection

Finally we would like to say that stereo matching is one of the most active research areas in computer vision, due to its importance in real-time application, and the biggest challenge in this area of research is to design an algorithm that find the best match in the minimum time.

REFERENCES

[1] Stefano Mattoccia, " Stereo Vision: Algorithms and Applications ", July 25 2010, pp.1-56

[2] Daniel Scharstein, Richard Szeliski, "A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms"

[3] Paul Munro, Antony P. Gerdelan, "Stereo Vision Computer Depth Perception", pp. 1-12

[4] Milan Sonka and Thomson, *"Image Processing*, Analysis, and Machine Vision, Third Edition",pp573-594

[5] "www.vision.deis.unibo.itsmattstereo.htm"

[6] "http://en.wikipedia.org/wiki/Sum_of_absolute_differences"

