

Efficient Surface Interpolation with Occlusion Detection

Houman Rastgar¹ Boubaker Boufama² Saida Bouakaz³

¹Department of Electrical and Computer Engineering, University of Windsor

²Department of Computer Science, University of Windsor, boufama@uwindsor.ca

³Department of Computer Science, Universite' Claude Bernard Lyon I

Abstract

In this paper we present a novel dense matching algorithm that relies on sparse stereo data in order to build a dense disparity map. The algorithm uses a recursive updating scheme to estimate the dense stereo data using various interpolation techniques. The major problem of classical template matching techniques is their reliance on a fixed template shape and poor performance around untextured regions. In this paper we attempt to alleviate the problem of template matching techniques by using an adaptive window shape and also by avoiding searching in homogenous image regions that are difficult to match by templates. The outcome is an algorithm that performs at least ten times faster than template matching, and yet it achieves higher accuracy. Moreover, our algorithm preserves depth discontinuities and assigns disparities at occluded regions.

Keywords: Stereo Matching, Computer Vision, Sparse Disparity Estimation.

1. Introduction

The ability to estimate depth disparity maps of stereo image pairs with accuracy, reliability, and efficiency depends to a great extent on the type of matching approaches used and the computational strategy employed for the matching [1-2]. In this paper we present an efficient recursive method for estimating dense disparity maps by utilizing several important properties of the disparity function.

We start out by dividing the image into two parts, the feature points which we refer to as Ground Control Points (GCPs) and the regular points. The feature points can be found by any simple sparse matching methods. Following this, epipolar lines are divided into feature delimited line segments and interpolated from the depth of their delimiting feature points that were calculated previously. The interpolation technique estimates the depth of the regular points by a rule-based scheme where the disparity of the

delimiting features and some additional points on the line are checked against several conditions. The aim of this step is to infer information regarding the surface that is to be interpolated. When a condition is met the corresponding interpolation technique to that surface model is used to estimate the disparity of the regular points. If none is satisfied, then the segment is divided into two and each one is processed by the same scheme until the algorithm converges. The motivation behind this scheme is presented later on.

It is important to note that unlike energy minimization-based methods [3-4], our method aims at decreasing CPU-time while it remains simple and straightforward. We have tested our algorithm on several rectified stereo images and results are presented. The results show that our algorithm increases accuracy while at the same time significantly reducing CPU-time.

2. Proposed Algorithm

As mentioned before, our algorithm starts out by computing a sparse disparity map for the stereo pairs. The results of this stage are a set of highly reliable disparities or Ground Control Points (GCP) [5] which are the disparities of intensity edges in our algorithm. This is used later on in our conditional interpolation technique in order to build a dense disparity map. Since the accuracy of the algorithm will ultimately depend on the results of this initial stage, it is imperative that this initial disparity map be computed as accurately as possible. As it is mentioned later on, the Canny edge detector is chosen to extract the GCPs and a simple correlation technique can be used to match these points. The results are usually obtained in a very short time with high accuracy since the number of these edge points is merely a fraction of the whole image.

Following this we start our interpolation strategy. This stage processes line segments that are delimited by our pre-computed GCP pixels. Typically a single epipolar line is divided into a few such line segments and each segment is processed individually.

2.1. Ground Control Points

The purpose of using GCPs in our algorithm is to use their disparity in order to interpolate every other pixel in the image. The logic is behind this is that the depth discontinuities in a disparity map contain the most critical points. Thus if the disparity of these points are known the rest of the depth map can be easily obtained from this initial data.

In order to detect depth discontinuities, we resort to using intensity discontinuities, i.e. edge points [6-7]. The logic behind this is related to the concept of image formation; a depth discontinuity produces a discontinuity in the reflectance at that point which leads to a sudden change of intensity.

In order to see this better, note the images in Fig. 1. Right image represents the edge intensities extracted using the Canny [8] edge detector and left image represents the actual depth discontinuities. It is obvious that there is a correlation between depth discontinuities and intensity edges. However, one can see that the intensity edges include most depth discontinuities, as well as many other points (such as color changes as well as textures and etc.) [9]. However, in spite of this, the number of edge points remains only a small fraction of the total image and a good initial guess at where the depth discontinuities could be located.

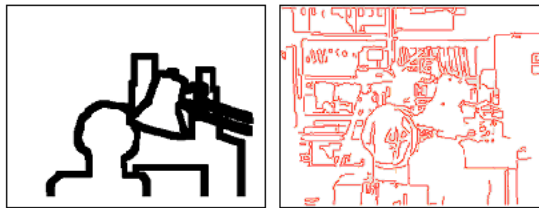


Fig. 1. Left image shows the depth discontinuities and the right image shows the intensity edges of the Tsukuba stereo pair.

2.2. Interpolation

After matching all the edge points we interpolate the disparity of every other point by assuming a surface model for every edge delimited line segment. We assume four different surface models based on the disparities of the delimiting GCPs as shown in Fig. 2. Each interpolation technique is related to a different type of surface. Type (A) interpolation is applied to slanted surfaces; whereas type (C) is applied to flat ones (parallel to the image plane). Interpolation types (B) and (D) are used in order to interpolate line segments where a depth discontinuity is suspected on the left or the right of the segment respectively.

It must be noted that it is almost impossible with the knowledge of the disparities of two delimiting

GCPs (black squares) to determine the type of the surface. Therefore more information is required and thus we make use of the intermediate points (red squares) as shown in Fig. 2. The intermediate points are found based on the disparities of their neighboring GCPs and they are matched using a simple template matching technique. Since the disparities of the neighboring GCPs of the intermediate points are known, some assumptions can be made when searching for the intermediate points' matches. For instance, we use an adaptive window shape in order to guarantee an accurate match for the intermediate point. The way our adaptive shaped window works is by noting the disparities of the bounding GCPs. In order to avoid matching a point incorrectly at a depth discontinuity we adjust the windows to three different shapes as shown in Fig. 3 [5]. The strategy for adjusting the window shape can be seen in the flowchart in Fig. 5. We are also able to narrow down the search neighborhood of an intermediate points' candidate matches by making use of the ordering constraint and the disparities of its neighboring GCPs as shown in Fig. 4. The reason why we have adopted a recursive algorithm is that it is entirely possible for two neighboring GCPs to contain a number of different types of such surfaces. Therefore the process has to be repeated until a surface model is positively identified. Basically every line segment is matched against these four surface models and if it is identified against any of them it is interpolated accordingly, otherwise it is broken into two segments and the process repeats until every pixel is interpolated using one of the four interpolation techniques.

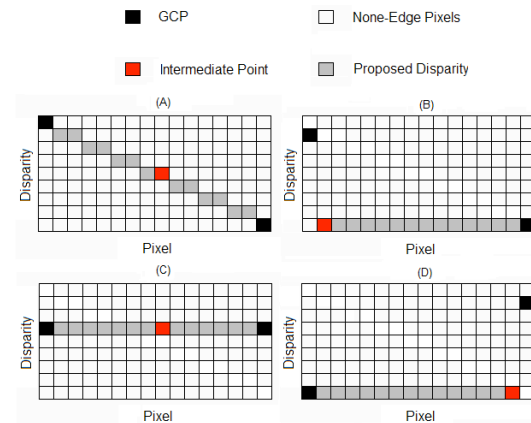


Fig. 2. Different interpolation techniques are adopted for different surface models. (A) and (B) are used to interpolate continuous surfaces (slanted and flat). (C) and (D) are used for depth discontinuities.

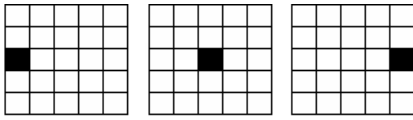


Fig. 3. Depending on the disparities of the neighboring GCPs, a different window shape is used for searching for the intermediate points.

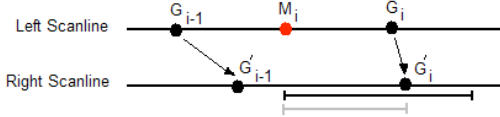


Fig. 4. Using the ordering constraint to reduce the search neighborhood (grey line shows the reduced search area).

Some terms need to be defined at this time regarding the flow chart in Fig. 5. G_i is the i th GCP point whereas M_i is the intermediate point corresponding to G_i . Arithmetic operations on these points, such as averaging shown in the flowchart operate on their x coordinates, therefore $(G_{10}+G_{11})/2$ denotes the point halfway between these points. In addition $D(G)$ denotes the disparity of a point G , also $P(G)$ denotes the same point in the previous scanline (i.e. $P(x,y) = (x,y-1)$).

Also the conditional flowchart (b) is an expanded version of the shaded decision block in flowchart (a). In addition, the intermediate point LM_i is the intermediate point to the left of the initial intermediate point M_i . Likewise, the intermediate point RM_i is the intermediate point to the right of the initial intermediate point. These additional intermediate points are situated immediately to either the left or right of an existing GCP and are used to infer information regarding depth discontinuities. The actual location of these points is defined in the flowchart (see Fig. 5). Also the fourth block in the general flowchart states that if the GCPs of a segment and its intermediate point have disparities equal to their neighbors in the previous scanline, the disparities of the current line segment will be copied from the previous line segment. This is a very likely scenario which helps reduce computation time and also it enforces the inter-scanline consistency.

At this point it should also be mentioned that our matching technique attempts to find disparities for the left image. However, at times it becomes necessary to reverse this process and estimate the disparity for certain points by finding a match for a point in the right image as shown in the flowchart. The reason for this is that for pixels neighboring suspected depth discontinuities there is a high likelihood of occlusion. Therefore we find a point immediately to the left (or right, depending) of a discontinuity and assign that pixel's disparity to all of its neighboring points. As a

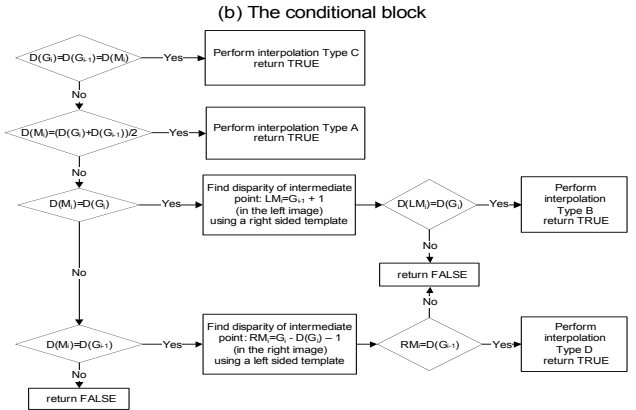
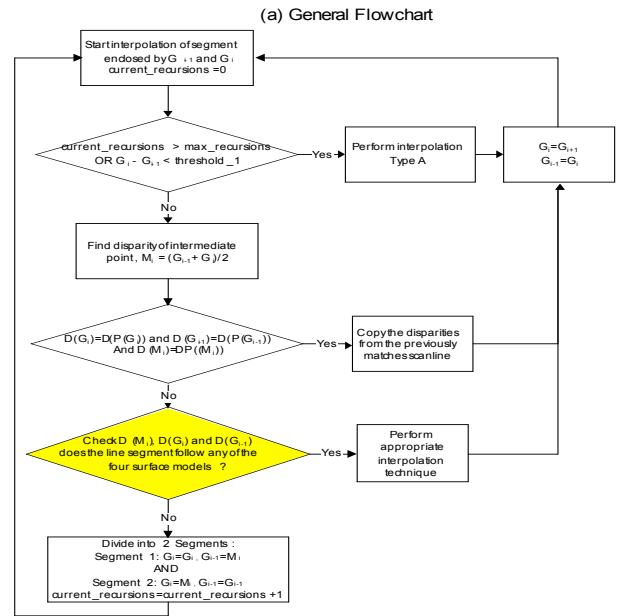


Fig. 5. The flow chart of our recursive algorithm. Top part shows the overall algorithm and the bottom shows the expanded conditional statement in the top part. The conditional block is where the surface type is decided and interpolation is carried out.

result we are able to find disparities at occluded points even though there are no matches for them.

3. Experimental Results

We have tested our algorithm on a set of stereo images [10]. The results show our algorithm outperforms a typical template matching technique.

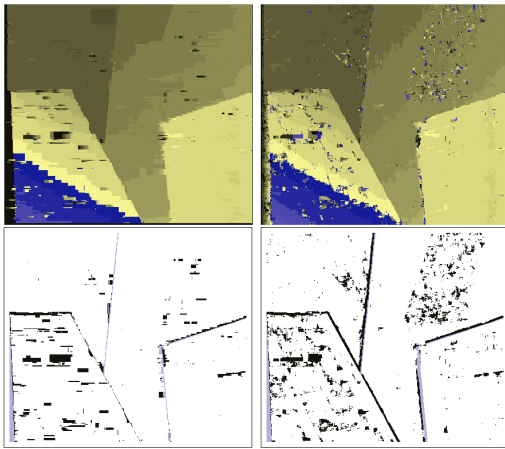


Figure 6. Comparison of our method with NCC. Top shows disparity map and bottom shows error pixels. Left shows our algorithm and right shows NCC.

Fig. 6 shows a comparison of our algorithm on the Venus [10] image with a greedy correlation based (NCC) algorithm. The results show that our algorithm performs significantly better at object boundaries where occlusions occur. More importantly, our algorithm is able to assign correct disparities to occluded points using our interpolation strategy at depth discontinuities as shown in Fig. 7. As it can be seen from the figure, a regular greedy search for matches in an occluded region results in errors. In addition, our method generally computes the results several times faster than a typical template matching technique.

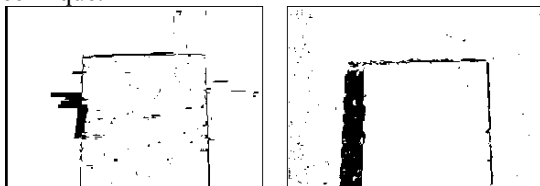


Fig. 7. Bad pixels (error > 1) for our algorithm and right shows NCC.

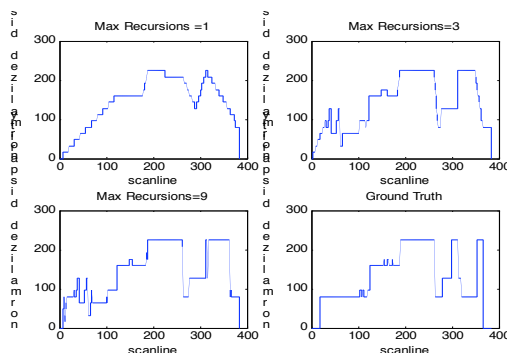


Fig. 8. A random scanline ($y=160$) for the Tsukuba stereo pair after various number of iterations. Bottom right contains the actual disparities.

In addition, the recursive nature of this algorithm does not incur a significant computation cost. Fig. 8 shows that after a few iterations the disparities converge to a good solution.

4. Conclusion

We have presented an iterative scheme for estimating dense disparities by using an initial set of edge points. These points are typically no more than 3% of the whole image and are extracted using a typical edge extractor. We outline an algorithm for detecting the location of depth discontinuities and for interpolating disparities. Our interpolation method attempts to find the type of surface that is to be interpolated and it applies an appropriate strategy correspondingly. Our algorithm performs 10 to 19 times faster for several stereo images; in fact we achieve dense disparities for the popular Tsukuba image set in less than a second on a Pentium 4 processor. Also the accuracy of our algorithm is higher for most cases than a typical NCC method. In addition we are able to obtain good results at object boundaries and also to assign disparities to occluded points.

5. References

- [1] D. Scharstein and R. Szeliski. „A Taxonomy and evaluation of Dense Two-Frame Stereo Correspondence Algorithms”. *IJCV* 47(1/2/3):7-42, April-June 2002.
- [2] M.Z. Brown, D. Burschka, and G.D. Hager, "Advances in Computational Stereo," *PAMI*, Vol. 25, No. 8, pp. 993-1008, 2003.
- [3] V. Kolmogorov and R. Zabih. "Computing visual correspondence with occlusions using graph cuts". *ICCV*, 508–515, 2001.
- [4] Y. Boykov, O. Veksler, and R. Zabih. "Fast approximate energy minimization via graph cuts". *PAMI*, 23(11): 1222-1239, 2001.
- [5] Aaron F. Bobick, Stephen S. Intille, "Large Occlusion Stereo", *IJCV*, v.33 n.3, p.181-200, Sept. 1999.
- [6] S. Birchfield. Depth and Motion Discontinuities. PhD thesis, Dept. of Electrical Engineering, Stanford University, June 1999.
- [7] T. Kanade and M. Okutomi. "A stereo matching algorithm with an adaptive window: Theory and experiment". *PAMI*, 16(9):920--932, September 1994.
- [8] J. Canny. "A Computational Approach to Edge Detection", *PAMI*, Vol. 8, No. 6, Nov. 1986.
- [9] Régis Vaillant, Olivier D. Faugeras, "Using Extremal Boundaries for 3-D Object Modeling", *PAMI*, 14(2): 157-173 (1992).
- [10] D. Scharstein and R. Szeliski. MiddleBury College Stereo Vision Page. <http://www.middlebury.edu/stereo/> accessed Febuary 7, 2006.