

# ACHIEVING EFFICIENT DENSE MATCHING FOR UNCALIBRATED IMAGES

*B. Boufama & K. Ghanem*

SCHOOL OF COMPUTER SCIENCE  
University of Windsor  
Windsor, Ontario  
Canada N9B 3P4

## ABSTRACT

This paper presents a new method to achieve fast dense matching in a pair of uncalibrated images. Classical area-based dense matching methods suffer from the high computational time resulting from intensive correlation calculations during the search/selection process. In contrast to conventional methods that are based on similarity and correlation techniques, this method is based on enforcing known geometric constraints and uses correlations only on a very small number of points. In particular, this paper proposes a hybrid matching technique that segment the image into two sets: the edge and the nonedge regions. For the edge regions, where discontinuities usually occur, the correlation-based classical matching method is used whereas, for nonedge regions, a segment mapping is used to achieve a correlation-free pixel matching. This segment mapping implicitly enforces all the four well known constraints in stereo matching: epipolar, continuity, uniqueness and, order constraints. The experiments on real images validated our method and showed drastic CPU-time reduction compared to classical methods.

**keywords:** Dense matching, geometrical constraints, correlations, stereo vision

## 1. INTRODUCTION

A dense disparity map is required in many stereo applications, such as 3D reconstruction and novel view synthesis. Although numerous works have been carried out over the last two decades[1, 2, 3, 4], dense matching remains one of the bottlenecks in stereo vision. This is due to the large amount of computation involved in dense matching. Furthermore, matching success is affected by occlusions and image noise. Area-based correlation methods are a natural choice for dense matching of uncalibrated images. Different correlation functions have been proposed to improve the performance of area-based matching[5].

---

This work was supported in part by a grant from the National Science and Engineering Research Council of Canada

Due to the inherent complication and noise sensitivity of the pixel matching, classical approaches were either limited to a sparse matching or made some assumptions about the observed scenes[6]. Sparse matching methods, also called feature-based methods, extract some selected features (e.g., corners, lines) in each image then attempt to establish the matching[7, 8]. The main drawback of these methods is the sparseness of the recovered information. On the other hand, dense matching aims at matching each and every pixel in one image with its correspondence pixel in the other image. Most of these methods are based on correlation techniques[5] to evaluate region similarities. However, because correlations are calculated intensively in the search/selection process, these methods are too slow. Geometrical constraints, such as epipolar geometry, unicity, order and, continuity, can be used to reduce the search and therefore, improve the execution time[9]. The satisfaction of the continuity/discontinuity constraints was addressed by several researchers. In particular, some energy-based methods[3, 10] solve the correspondence problem as a minimization problem that preserve discontinuities at object boundaries. However, the process is iterative and is far from straightforward to implement. Furthermore, the CPU-time was not addressed as part of the problem.

The method we present in this paper addresses the dense matching problem in the case of uncalibrated images. We make use of the calculated epipolar geometry and the continuity constraints. The latter is used on homogeneous region while it is relaxed at region boundaries. Unlike energy-based methods, our method aims at decreasing CPU-time while it remains simple and straightforward. Images are segmented into two sets of regions, the edge regions and the nonedge regions. The continuity constraint is enforced for the latter and relaxed for the former. More importantly, correlation is not used at all inside the nonedge areas. Instead, a segment-to-segment matching is first established then, the pixel-to-pixel correspondence is carried out algebraically.

## 2. PROPOSED DENSE MATCHING METHOD

An ideal solution to the dense matching problem would be one that gives a closed-form function, mapping the entire image  $I$  to  $I'$ . This function would map point  $p$  in the image  $I$  to  $p'$  in the image  $I'$ . Unfortunately, it is impossible to find such a function.

Because disparity variation is smooth across a continuous surface but is usually abrupt across physical boundaries, these two regions are treated differently. In other words, an image can be segmented into two sets of pixels: edge and nonedge pixels respectively. Then, one can apply different matching strategies for these two sets.

Our approach can be summarized as follow. Using a basic edge detector, edges are first extracted from the left image which is then divided into two sets of regions: edge and non-edge regions. Then, all edge pixels from the left image are matched to their corresponding pixels in the right image using classical correlation based method. Finally, non-edge regions are matched using our new segment-based method that maps line segments from left image to the right one.

### 2.1. Matching of edge pixels

Because speed rather than accuracy is important for us, we have used a simple Sobel filter to extract our edges. Although those edges are not one-pixel thin, they accurately locate the boundaries in the image(see Figure 1).

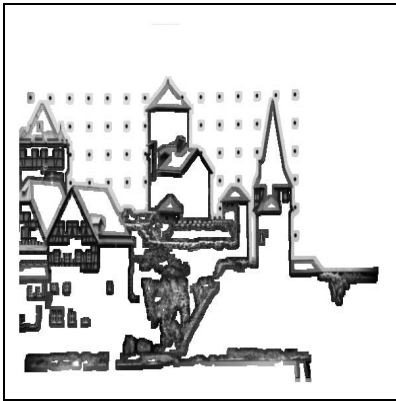


Fig. 1. Extracted edges from the Castle scene(see Figure 3)

These edge pixels are matched with their corresponding right image pixels using correlations and the epipolar constraint. Note that edge pixels only represents a fraction of the total pixels in the image. Therefore, the CPU time for their matching is not significant compared to the CPU time for the whole image.

### 2.2. Matching of nonedge pixels

Nonedge pixels are obtained by simply removing the edge pixels from the original image. The new image, without its edge pixels, is then considered as made of a set of line segments. A line segment is defined as the sequence of pixels along an epipolar line delimited by two edge areas. Figure 2 shows an example of four line segments along an epipolar line, where the white areas represent the edge pixels. Consider the pixels  $A, B, C, D, E$  and  $G$ . It can be easily seen that this portion of the epipolar line generates three line segments; namely the segments  $(AB)$ ,  $(CD)$  and  $(EG)$ .

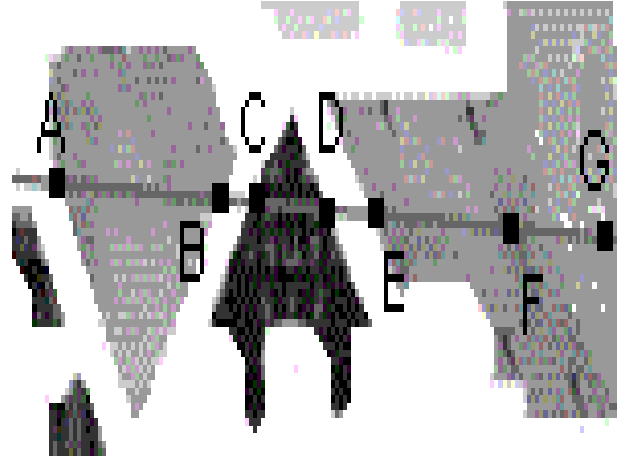


Fig. 2. Example of line segments in an image

Since every pixel from non-edge regions belongs to a certain line segment, the matching of non-edge regions of the image can be viewed as the matching of all line segments on all the epipolar lines. Obviously the matches of the pixels of a non-edge segment must lie on the same corresponding epipolar line in the other image. Furthermore, since non-edge regions consist of smooth surfaces without abrupt disparity changes, we can assume that local continuity is present in each line segment.

### 2.3. Matching of line segments

Consider the line segment  $AB$  from Figure 2 and assume that edge pixels have been matched in a previous step using correlation and epipolar constraint. Therefore, the pixels  $A$  and  $B$  have been already matched to their corresponding pixels  $A'$  and  $B'$  in the right image. In other words, we already have a matched pair of line segments:

$$(AB) \longleftrightarrow (A'B')$$

Now how can we achieve the pixel-to-pixel matching of all the pixels belonging to  $(AB)$ ? One can use correlations again by enforcing the continuity/order/uniquity constraints.

However, the CPU time will be again significant making the matching process costly. Note that enforcing these constraints has been already tried in previous works.

Knowing that in our case a line segment can only belong to a smooth surface, one can assert that all pixels belonging to such a segment will have very similar disparity. Therefore, we have used the following algebraic formula to achieve the pixel-to-pixel matching within each line segment :

$$p = A + d_p \longrightarrow p' = A' + \lambda d_p$$

where  $d_p$  is the pixel distance from  $p$  to  $A$ , in the left image, and the coefficient  $\lambda$  is equal to  $\frac{|A'B'|}{|AB|}$ .

Obviously the coefficient  $\lambda$  represents the stretching or shrinking factor, for a particular segment, between two images. It will be often very close to 1. The above matching formula might not be very accurate for very long line segment. In this situation, we can split the line segment into two subsegments and therefore, we will end up with a different  $\lambda$  for each subsegment. As shown on Figure 2, the segment  $EG$  is broken down into two subsegments,  $EF$  and  $FG$ .

Because most of the image pixels will be matched using the above formula, the CPU-time for the dense matching will be drastically reduced. In particular, there will be no search for candidates and no correlation calculations. This new way of doing the matching has several advantages. Not only it is much faster than traditional methods, it will work on smooth surfaces (with no texture) where correlation-based algorithms will fail.

### 3. EXPERIMENTAL RESULTS

Two pairs of images have been used here for testing our new matching method (see Figure 3 and Figure 5). The result of any dense matching algorithm would be difficult to evaluate, given that the number of matched pixels runs usually in the hundreds of thousands. For these experiments, we have shown the results by reconstructing the right image using only the calculated disparities and the left image as inputs. In theory, the reconstructed image should look very similar to the right image. In particular, mismatches will be easily spotted as they will not be coherent with the image.

For each pair of images, we have used two correlation window sizes,  $5 \times 5$  and  $7 \times 7$  respectively. Although correlation-based matching is only used for edge pixels, it affects the whole quality of the dense matching. As it can be seen from the results, using a  $7 \times 7$  correlation window yields better results. Of course, this comes with a CPU-time cost. In our experiments, the dense matching of the castle scene had a CPU-time of 0.515s and 0.938s for correlation windows of sizes  $5 \times 5$  and  $7 \times 7$  respectively. The CPU-time jumped from half second to almost one second. This implies that

although edge pixels only represent a fraction of the total image pixels, they represent a significant share of the total CPU-time for the dense matching. This is because we are using a classical correlation-based search/select algorithm for these edge pixels. Nevertheless, the results we have obtained are excellent in term of both quality and CPU-time.

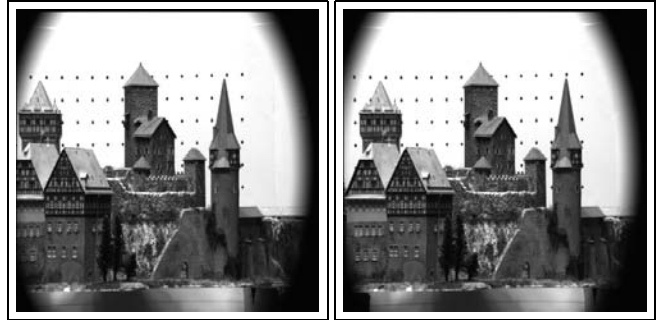


Fig. 3. A pair of the castle scene : resolution  $384 \times 576$

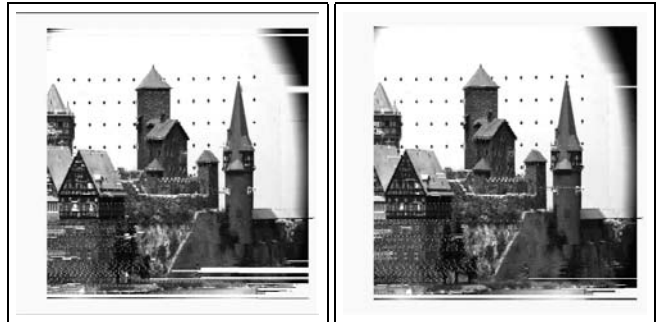
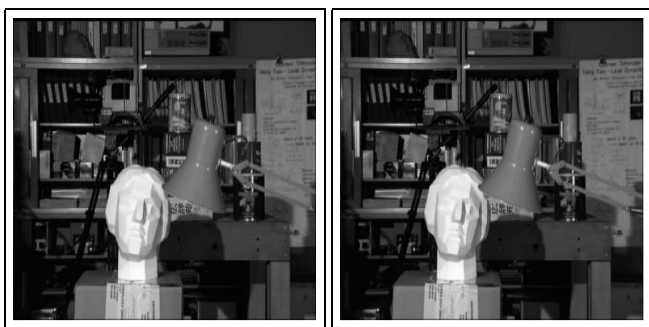


Fig. 4. These two images are the reconstruction of the right image from the castle scene. The left one is obtained with a correlation window of  $5 \times 5$  while the right one is obtained with a correlation window of  $7 \times 7$

The reconstructed images look very similar to the original ones showing that our dense matching method yields correct results. Furthermore, the CPU-time is below one second for large images. For the smaller head scene images the CPU-time was only 0.344s and 0.610s for correlation window of sizes  $5 \times 5$  and  $7 \times 7$  respectively. The white spots represent pixels that were not matched. Most of these pixels can be easily interpolated if needed.

### 4. CONCLUSION

This article has proposed a new method for a fast pixel correspondence across images in the general case of uncalibrated images. Unlike most existing methods that rely entirely on correlation scores for dense matching, the proposed method uses correlations only when matching edge



**Fig. 5.** A pair of the head scene : resolution  $340 \times 455$



**Fig. 6.** These two images are the reconstruction of the right image from the head scene. The left one is obtained with a correlation window of  $5 \times 5$  while the right one is obtained with a correlation window of  $7 \times 7$

pixels. Because edge pixels usually represent physical boundaries in the scene, their disparities might not vary smoothly across images. Therefore, an efficient matching algorithm should use two different strategies, one for edge pixels and another one for the other pixels. The proposed method follows this philosophy. First, an image is segmented into two sets of pixels: edge and nonedge. Then, edge pixels are matched using correlation scores and the epipolar constraint. Because these pixels are located on some sort of boundaries, they usually have enough texture in their vicinity making the possibility of a mismatch very unlikely. Once these edge pixels have been matched, a segment mapping is used to achieve a correlation-free matching for the remaining pixels. Such a segment mapping implicitly enforces the epipolar, continuity, uniqueness and, order constraints. As a consequence, the CPU-time for the matching is drastically reduced. As it was shown in the experiments, dense matching can be achieved in a  $1/3$  of a second. Furthermore, the matching of homogeneous regions, with less texture, is easily achieved using our method than when using correlation-base technique.

## 5. REFERENCES

- [1] T. Kanade, "Development of a video rate stereo machine," in *Proceedings of the ARPA Image Understanding Workshop*, November 1994, pp. 549–558.
- [2] J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, no. 1, pp. 43–77, 1994.
- [3] L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert, "Dense disparity map estimation respecting image discontinuities: A pde and scale-space based approach," Technical Report 3874, INRIA, 2000.
- [4] B. Boufama and D. O'Connell, "Identification and matching of planes in a pair of uncalibrated images," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, no. 7, pp. 1127–1143, 2003.
- [5] P. Aschwanden and W. Guggenbühl, "Experimental results from a comparative study on correlation-type registration algorithms," in *Robust Computer Vision*, Förstner and Ruedel, Eds., pp. 268–282. Wichmann, 1992.
- [6] H. Chabbi, L. Theiler, and S. Paris, "Automatic detection of planar contours from uncalibrated images," in *Proceedings of the 16th International Conference on Pattern Recognition, Quebec City, Canada*, August 2002, pp. 541–544.
- [7] Z. Zhang, R. Deriche, O.D. Faugeras, and Q.T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Artificial Intelligence*, vol. 78, no. 1-2, pp. 87–119, 1994.
- [8] J. H. McIntosh and K. M. Mutch, "Matching straight lines," *Computer Vision, Graphics and Image Processing*, vol. 43, no. 3, 1998.
- [9] B. Boufama and K. Jin, "Towards a fast and reliable dense matching algorithm," in *International Conference on Vision Interface, Calgary, Canada*, May 2002, pp. 178–185.
- [10] L. Robert and R. Deriche, "Dense depth map reconstruction: A minimization and regularization approach which preserves discontinuities," in *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, April 1996, pp. 439–451.