Efficient Dense Matching for Textured Scenes Using Region Growing

Maxime Lhuillier

Efficient Dense Matching for Textured Scenes Using Region Growing

Maxime Lhuillier
GRAVIR-IMAG and INRIA Rhône-Alpes
655, Avenue de l'Europe, 38330 Montbonnot, France
Maxime.Lhuillier@inrialpes.fr

Abstract

We present a simple and efficient dense matching method based on region growing techniques, which can be applied to a wide range of globally textured images like many outdoor scenes. Our method can deal with non-rigid scenes and large camera motions. First a few highly distinctive features like points or areas are extracted and matched. These initial matches are then used in a correlation-based region growing step which propagates the matches in textured and more ambiguous regions of the images. The implementation of the algorithm is also given and is demonstrated on real image pairs.

Keywords: Dense Matching, Region Growing, Correlation.

1 Introduction

Many algorithms have been proposed for dense matching. One popular approach is based on correlation, however this kind of algorithms is generally limited to relatively small disparity, hence small camera motion. For stereo images [DA89], [Kos93] whose epipolar geometry is known *a priori*, the search space can be reduced to a 1D along epipolar lines. Image rectification is usually used to accelerate the dense matching process, but does not allow zooming in/out of the camera.

Another approach is optical flow [BFB94], which handles non-rigid scenes but is limited to smaller displacements. Differential techniques give accurate estimation of displacements for smooth images, but fail for textured images and at depth discontinuities. Area-based matching techniques are fast, but do not perform well for sub-pixel displacements or dilations. Phase-based methods produce accurate results overall, but involve a large number of filters.

Occlusions are one of the major sources for wrong matches. Most of the recent stereo and optical flow work consists of incremental improvements to existing methods, to increase speed, accuracy or reliability. Only a few authors directly treat large occlusion stereo [IB94]. Usually, coarse to fine (e.g. [HA89]) or hierarchical (e.g. [MT94]) matching strategies seems to be necessary to deal with large disparity range.

Our algorithm uses mainly region growing techniques. Region growing is a classic approach for segmentation [HS85], [Mon87], and finding shapes [Bra93]. In its simplest sense, region growing is the process of merging neighboring points (or collections of points) into larger regions based on homogeneity properties. Further, a correct use of regions can help matching [HJ95], [SA95], although their boundaries are unstables.

An explicit region growing method is introduced in the photogrammetry domain by [OC89] with the "Gotcha" (Gruen-Otto-Chau) ALSC (Adaptative Least Square Correlation) algorithm. It starts with approximate patch matches between two SPOT satellite images and refines them. Their recovered distortion parameters are used to predict approximate matches for new patches in the neighborhood of the first match. Then, these patch matches are refined and so on. Complements for building extraction are discussed in the same domain by [KM96]: a pyramidal algorithm to produce seed matches and extraction of linear elements to remove possible blunders are proposed.

Our main assumption is that the scene is globally textured like many outdoor scenes. Non-rigid scenes, large disparities (e.g. a quarter of view size) and camera zooming in/out are allowed. The computation time is independent of any disparity bound.

Our algorithm has two main steps. The first step extracts and matches a sparse set of highly distinctive features: *seed points* and *seed areas*. Seed points are points of interest and are matched by correlation. If the scene is rigid, a robust technique to match points of interest through the recovery of the unknown epipolar geometry could be used. Seed areas complete these matches in the most uniform colored areas. We extract and match them by simultaneously matching and region growing in the most uniform colored areas of the images.

The second step uses these initial matches to seed a dense matching propagation, using a best first matching strategy. This extends the matches to include the textured areas of the image. If the scene is rigid, we can use the epipolar geometry obtained in the first step to constrain the propagation in the second step. Our pixel-to-pixels propagation deals with fine texture details, and stops just at the occlusion borders if they are enough textured. The result is a dense pixelic matching, but it needs less calculations than patch-to-patches propagation and distortion parameter estimations.

The two steps of the dense matching algorithm are respectively described in Sections 2 and 3. Results on real image pairs are presented in Section 4.

2   Initial Matching

In this section, we show how to produce a set of initial candidate matches. We first justify the choice of seed points and seed areas. Secondly, we explain how to compare seed areas, and finally describe their matching and their region growing-based extraction.

2.1   Which features to choose ?

Matching points of interest is now a robust process for rigid scenes, as demonstrated for example in [ZDFL94]. First, these points are extracted and matched with correlation. Because of noise and nearly repetitive patterns, a relaxation step and next a robust estimate of epipolar geometry seem to be necessary to produce reliable matches. However, a set of candidate point matches obtained by simple correlation is sufficient to seed concurrent propagations.

Matching edge segments is well adapted for polyhedral and low textured scenes. It is difficult to extract and match salient matches of edge segments in our case, because of our assumption of textured scenes.

Finally, it is known that segmentation is an unstable process. Nevertheless, we need only to extract some initial seed area matches. A process will be described below which

produces some reliable matches of the most uniform colored areas of the images. The simultaneous use of shape and mean color comparisons between some isolated and uniform colored regions in the globally textured images is sufficient to produce concurrent seed area matches.

Therefore, we use both seed areas and seed points. Such seed features are only matched in weak distortion areas between the two images. Dense matching propagation will extend matching to more difficult and distorted regions to match. If the scene is rigid, the epipolar geometry is recovered while matching seed points.

2.2 How to Compare Two Seed Areas ?

First, seed areas will not usually be distinguishable if their areas are too small. On the other hand, areas which are too large are subject to significant perspective and segmentation distortions. So we limit the minimum and maximum sizes of our seed areas. In practice, it turns out that the same interval of allowed values is sufficient for many different types of images. The range interval is 100 to 2000 pixels for all our tests.

Two areas A and B are compared very simply by their mean color and their shape :

$$c(A, B) = \frac{|t(A) - B| + |B - t(A)|}{|A| + |B|}$$

where $|A|$ is the area of A, "-" indicates set difference, and t is the translation from A's centroid to B's. Areas are easily discriminated by their colors and forms, and c allows for a little perspective distortion or initial segmentation error. Other measures could be used, such as the generalized Hausdorff measure [HJ95] or moments [BS97], but the two simple measures above have proved adequate in our experiments.

2.3 Extract Candidate Matches for Seed Areas

This extraction is an alternate sequence of region growing and matching steps for seed areas. At the beginning, each pixel forms a separate region.

During a growing step, for each connected 2×2 block of pixels in the images, their regions are merged if their color difference is less than a threshold (see examples in Figure 1). The threshold is the same for all blocks, but increases between growing steps.

During a matching step, each region of the first image is compared to each region of the second using the above criteria: candidate matches are accepted if both of their areas are within the thresholds and their mean color and shape differences are small.
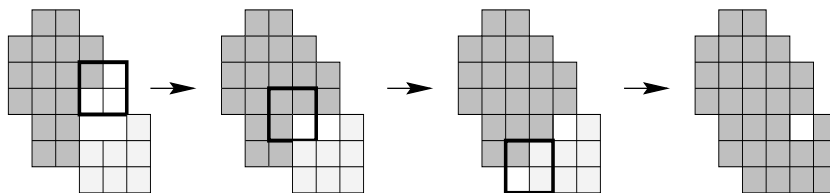

Figure 1: Tree successive pixel or region merges using most uniform colored 2×2 blocks of pixels. The big square is the selected block for a merge process.

Growing and matching steps are run several times at different color uniformity levels for two reasons. Firstly, region growing is not strictly identical in the two images because of noise and perspective distortion: successive comparisons are necessary to ensure good matches. Secondly, it allows the same interval of tested thresholds to handle many different types of views.

3 Dense Matching Propagation

We have described the first step of our algorithm in the previous section. A method to obtain seed point matches was cited. An algorithm was also proposed to obtain seed area matches. The second step is now described. We first justify the choice of the dense matching propagation strategy; then we give the principle of the algorithm; finally the algorithm and some implementation details are given. For clarity, the exact link between the first and the second step is explicated in the last part of this section.

3.1 Why Dense Matching Propagation ?

Our goal is dense matching for textured, non differentiable and noisy images. We choose a correlation-based method because it is simple and fast. Correlation is less sensitive to geometric distortion if small windows are used. However, matching with small windows can be ambiguous with nearly periodic textures such as those of outdoor scenes. Thus, a strength constraint is needed for reliable matching. We use the *continuity constraint*: except for some pixels on the object boundaries, the disparity must vary smoothly. Dense matching propagation is a simple and effective way to use this constraint: the propagation moves continuously from less ambiguous matches to more ambiguous ones.

3.2 Principle

A disparity map Map stores the region of correct pixel matches. The algorithm consists of growing this region. Let $Start$ be a set of active pixel matches near its boundaries. At each step we remove the best match (a, A) from set $Start$. Match (a, A) is the seed for a local propagation: new matches in the neighborhood of (a, A) (see Figure 2) are added simultaneously to set $Start$ and map Map. These new matches (b, B) are added only if neither pixel b nor B are already matched in Map.
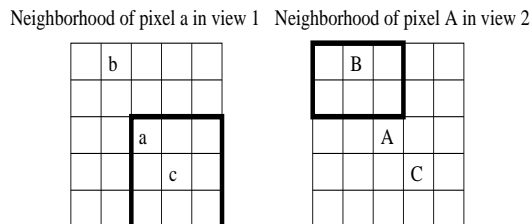

Figure 2: Definition of a match neighborhood. The neighborhood of a match (a,A) is a set of matches included in the two 5×5-neighborhood of a and A. Possible correspondents of b (resp. C) are in the black frame centered at B (resp. c).

Notice that:

- The set $Start$ is always included in the region of correct matches in Map and initial content of $Start$.

- The unicity constraint is guaranteed in Map by our choice of new matches. Thus, the number of local propagations and the size of the set $Start$ are bounded by the sum of the size of $Start$'s initial content and the area of the image.

- Choosing only one match in the neighborhood of (a, A) is inadequate. It does not produce a real 2D propagation, because the size of the set $Start$ can not increase and so can not contain the whole boundaries of region in Map in progress.

- The risk of bad propagation is reduced by the choice of the best match (a, A) of the set $Start$.

Furthermore, the more textured the image, the lower the risk of bad propagation. We reduce the risk by forbidding local propagation in regions which are too smooth.

Propagation is begun by initializing the set $Start$ as mentioned in Section 3.4. Propagation is stopped by image borders, too smooth regions and already matched areas. Occlusion contours stop it too, if they separate two different textures. If yes, they are included in borders of a finished propagation in one of the image.

3.3 Implementation and Algorithm

Disparity map Map is injective. We use a heap [AHU74] for the set $Start$ to store the potential seeds for local propagations and to select the best at each step. The complexity of propagation is then $O(nlog(n))$, where n is the area of the image. Notice that it is independent of any disparity bound.

If a is a pixel, let $\mathcal{N}_x(a)$ be the $x \times x$ window centered at pixel a. Let $s(a)$ be some estimate of the color roughness in $\mathcal{N}_3(a)$ and s_0 be a lower threshold. We use $s()$ to forbid propagation into insufficiently textured areas $\{a, s(a) < s_0\}$. The more perspective distortion is important between the two views, the upper this threshold should be.

Let $d(a, b)$ be a measure of the image intensity/color difference between $\mathcal{N}_3(a)$ and $\mathcal{N}_3(b)$, and value d_0 be an upper threshold. The ratio $r(a, b) = \frac{min(s(a), s(b))}{d(a,b)}$ is used as a measure of reliability for pixel match (a, b). Matches with the best (the uppest) reliabilities are first considered.

Let $(r_a, g_a, b_a), 0 \leq r_a, g_a, b_a < 1$ be the color of a pixel a. We use the following definitions for all our tests:

$$n(a, b) = .299|r_a - r_b| + .587|g_a - g_b| + .114|b_a - b_b|$$

$$d(a, b) = \frac{1}{9} \sum_{\delta \in \{-1,0,1\} \times \{-1,0,1\}} n(a + \delta, b + \delta)$$

$$d_0 = 0.07$$

$$s(a) = max\{n(a, b), b - a \in \{(1, 0), (-1, 0), (0, 1), (0, -1)\}\}$$

$$s_0 = 0.04$$

$$r(a, b) = \frac{min\{s(a), s(b)\}}{d(a, b)}$$

The algorithm is

// First, initialize $Start$ as mentioned in the next paragraph.
//Next, propagate:
$Local \leftarrow \emptyset$
while $Start \neq \emptyset$ do
. pull from $Start$ the match (a, b) which maximizes reliability $r(a, b)$
. let $Local$ be a empty heap of pixel matches
. // Store in $Local$ the potential matches of local propagation from match (a, b):
. for each (c, d) in $\{(c, d), c \in \mathcal{N}_5(a), d \in \mathcal{N}_5(b), (d - b) - (c - a) \in \{-1, 0, 1\} \times \{-1, 0, 1\}\}$ do
. . if $s(c) > s_0$ and $s(d) > s_0$ and $d(c, d) < d_0$ // and possibles others constraints
. . then store match (c, d) in the heap $Local$
. . end if
. end do
. // Store in $Start$ and Map consistent matches of $Local$ with Map:
. while $Local \neq \emptyset$ do
. . pull from $Local$ the match (c, d) which maximizes $r(c, d)$
. . if c and d are not already matched in the disparity map Map
. . then store match (c, d) in the disparity map Map and heap $Start$
. . end if
. end do
end do

If the scene is rigid, we add the epipolar constraint for match (c, d) in the line
    if $s(c) > s_0$ and $s(d) > s_0$ and $d(c, d) < d_0$.

3.4   Link between the First and Second Step

The two steps were described in the previous sections. The first step produces candidate matches of seed points and seed areas, which are accurate up to a few pixels. The second step needs candidate pixel matches with pixel accuracy. We combine the two steps using the following strategy: all candidate matches are starts of simultaneous and concurrent propagations.

If a seed point match is accurate about some pixels, it is a good trick to convert it to a set of concurrent, candidate pixel matches of its neighborhood. Best candidates will be first selected, and a single good one is sufficient to provoke an avalanche of correct matches in the second step. Bad candidates are then discarded if one of their pixel is already matched.

Seed area match (A, B) is converted to concurrent, candidate pixel matches in set $Start$ with the simple process below, where t is the translation vector which maps A's centroid to B's.

For each pixel a of A's boundaries, store candidate match $(a, t(a))$ in the set $Start$.
For each pixel b of B's boundaries, store candidate match $(-t(b), b)$ in the set $Start$.

Although region boundary is unstable, these candidate matches have proved adequate in our experiments to start a matching avalanche effect.

4 Experimental Results

Results on real image pairs are now discussed. It should be stressed that the same parameter values introduced in the various steps of the algorithm were used for all of the tests.

4.1 Visualizing Arbitrary Dense Matching with a Checker-Board

Since we test on non stereo pairs, disparity can not always be interpreted as depth, and it might also be large. Depth map and displacement field are not well adapted to display the results. We designed a global way to visualize dense matches for arbitrary images as follows. Pixels of the first image are colored with a gray-black checker-board. For each matched pixel of this image, we color the corresponding pixel of the second image with the same color. This makes it easy to visualize the match of each square and its distortion. A best way for color displays consists of blending a red-blue checked board with the original images.

4.2 Results

We show in Figure 3 that a single seed point match is sufficient for textured scenes to start a dense matching propagation (1.5s on a Ultra SPARC 300Mhz). Our visual matching checker board suggests that the majority of the matches are good.

Large occluded areas occur in the second scene (cf. Figure 4). The same detection and correlation as [ZDFL94] are used without relaxation or epipolar constraint, to produce seed point matches. Each isolated, textured region should contain at least one good seed point match to be matched. User can manually add some seed point matches to improve the final result.

The third scene is a rivulet (cf. Figure 5), with a blurred region near the right bottom corner. Colored Image dimensions are 512×768. 151 seed point (resp. 408 seed area) matches are extracted and matched in 5s (resp. 14s). Automatic seed point and seed area matches are start of concurrent propagations, which produce 238460 matches in 19s. Epipolar constraint limits bad propagations. Further, there is only seed areas to start the propagation in the blurred region.

We show finally extractions of seed area matches alone and the resulting propagations in Figure 6. Our method has been tested successful on many others textured image pairs.
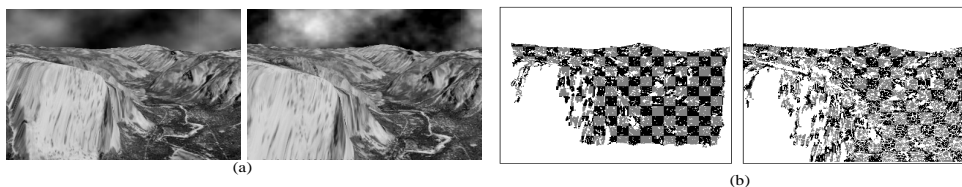

Figure 3: Efficiency of unconstrained propagation for a textured, camera zooming image pair. We manually set a single seed point match near the center of the textured Yosemite 2-16 image pair (a), with 1-2 pixel accuracy and show the resulting propagation (b). Only textured areas are matched.
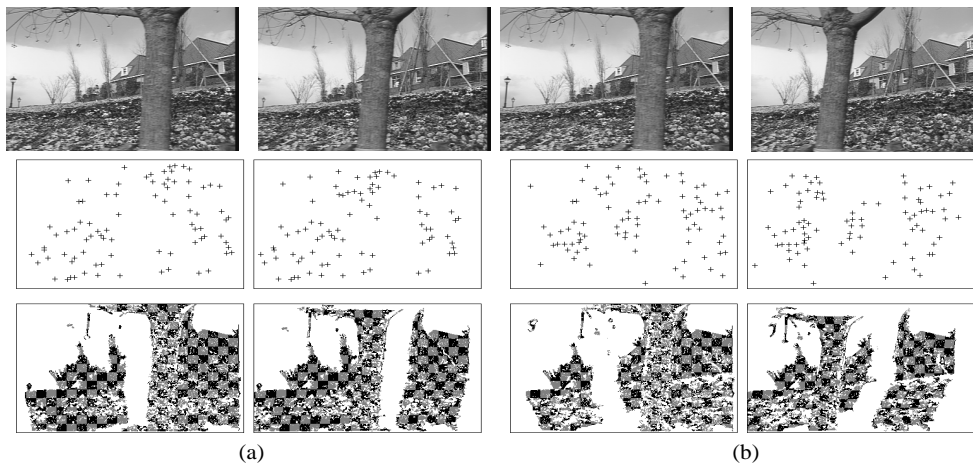
Figure 4: Minimal set of seed matches (a), large displacements and some user help (b). Automatic point seeding (a) produces more than one good match for each textured, isolated region. Thus, the resulting propagation (unconstrained) fills each of them (a,b). User can interact: we manually add to automatic seed matches a single one on the trunk bottom (b) to match it correctly.
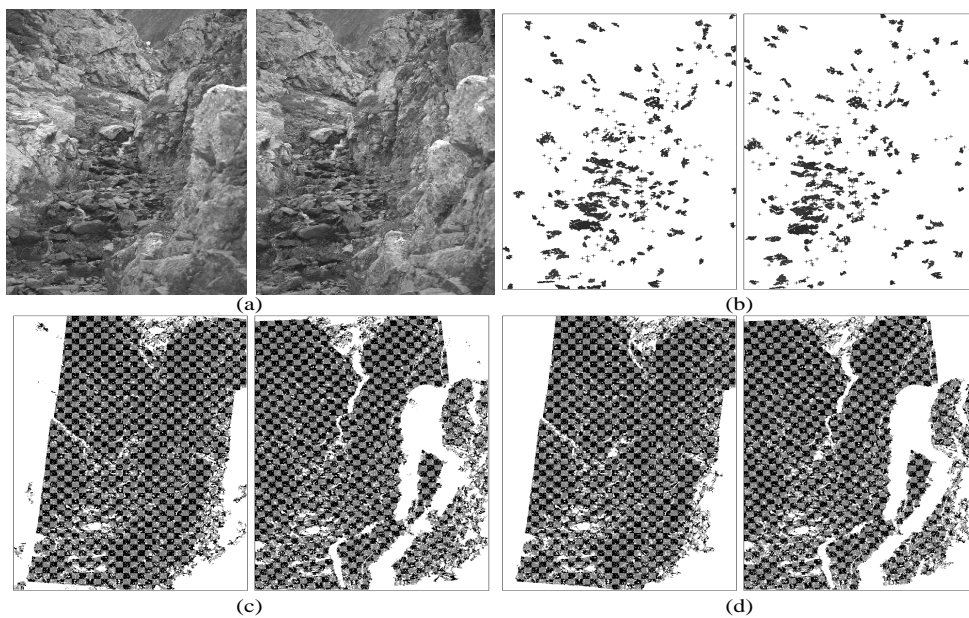

Figure 5: Rivulet image pair (a), automatic seed point and area matches (b), comparison between unconstrained (c) and epipolar constrained (d) propagation. Epipolar constraint limits bad propagations, especially in the blurred right down. In this blurred area, seed areas matches are necessary to start propagations.

5   Conclusion

A new method has been proposed for dense matching two textured images like many outdoors scenes. The algorithm has two main steps. First, we extract and match interest points. Seed area matches complete these matches in the most uniform colored areas of the images. Second, a correlation-based match propagation is started from these seeds, to produce a dense matching covering only sufficiently textured areas of the images.

We have successfully tested the algorithm on real image pairs with large displacements. Running time is acceptable and independent of any disparity bound. Rigidity constraint is not indispensable for enough textured areas, but limits bad propagations in low textured ones. If automatic seeding is insufficient, user can add simply new seed point matches to improve the resulting propagation. However, our method is not suitable for non textured images like indoor scenes and manufactured objects: the dense matching propagation is immediately stopped.
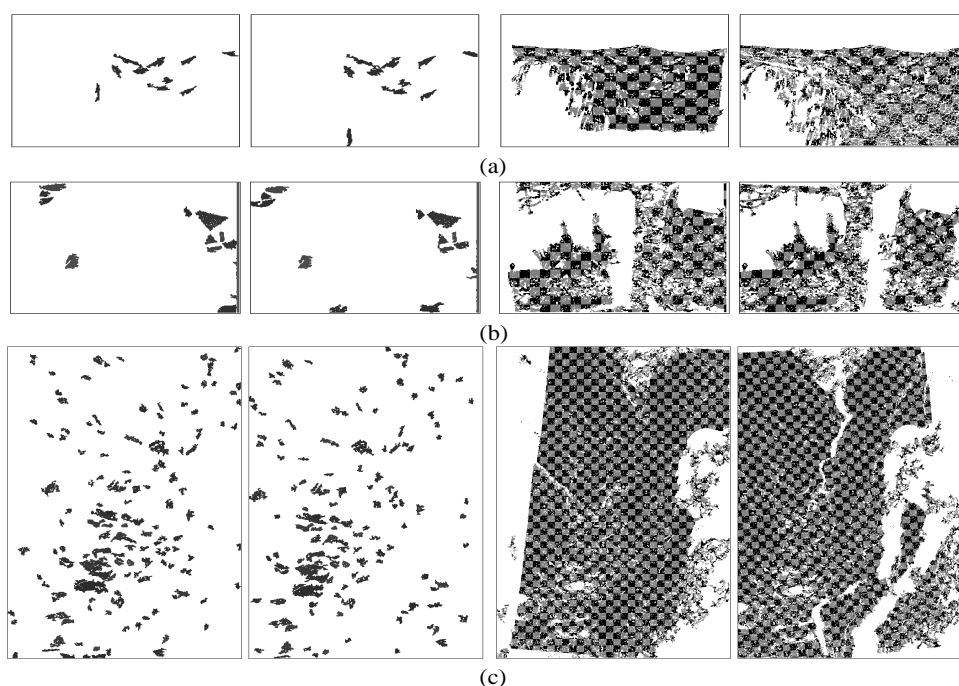

(a)

(b)

(c)

Figure 6: Seed area matches for Yosemite (a), Lausanne 0-1 (b), Rivulet (c) image pairs and the resulting (unconstrained) propagations.

References

[AHU74] A.V. Aho, J.E Hopcroft and J.D. Ullman. The design and analysis of computer algorithms. *Addison-Wesley, Reading, MA, USA*, 1974.

[BFB94] J. Barron, D. Fleet, and S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1): 43–77, 1994.

[Bra93] M. Brand. A short note on local region growing by pseudophysical simulation. *CVPR93*, pages 782–783.

[BS97] D. Bhattacharya and S. Sinha. Invariance of stereo images via the theory of complex moments. *Pattern Recognition*, 30(9): 1373–1387, 1997.

[DA89] U.R. Dhond and J.K. Aggarwal. Structure from stereo – a review. *IEEE Transactions on Systems, Man and Cybernetics*, 19(6): 1489–1510, 1989.

[HA89] W. Hoff and N. Ahuja. Surfaces from Stereo: Integrating Feature Matching, Disparity Estimation, and Contour Detection. *PAMI*, 11(2): 121–136, 1989.

[HJ95] D.P. Huttenlocher and E.W. Jaquith. Computing visual correspondence: Incorporating the probability of a false match. *ICCV95*, pages 515–522.

[HS85] R.M. Haralick and L.G. Shapiro. Survey, image segmentation techniques. *Computer Vision, Graphics, and Image Processing*, 29:100-132, 1985.

[IB94] S.S. Intille and A.F. Bobick. Disparity-space images and large occlusion stereo. In *ECCV94*, pages 179–186, 1994.

[KM96] T. Kim and J.P. Muller. Automated urban area building extraction from high resolution stereo imagery. *Image and Vision Computing*, 14:115-130, 1996.

[Kos93] A. Koschan. What is new in Computational Stereo Since 1989: A Survey on Current Stereo Papers. Technical Report 93-22, University of Berlin, 1993.

[Mon87] O. Monga. An optimal region growing algorithm for image segmentation. *IJPRAI*, 1(3): 351–375, 1987.

[OC89] G.P. Otto and T.K. Chau. A region-growing algorithm for matching of terrain images. *Image and Vision Computing*, 7(2):83-94, 1989.

[MT94] S.B. Marapane and M.M. Trivedi. Multi-Primitive Hierarchical (MPH) Stereo Analysis. *PAMI*, 16(3) : 227–240, 1994.

[SA95] S. Sull and N. Ahuja. Integrated Matching and Segmentation of multiples Features in Two Views. *CVIU*, 62(3): 279–297, 1995.

[ZDFL94] 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*, 78(1-2): 87–119, 1995.