

Using Multiple Disparity Hypotheses for Improved Indoor Stereo

Cristian Dima¹
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15217
USA

Simon Lacroix²
LAAS/CNRS
7, Ave du Colonel Roche
31077 Toulouse Cedex 4
France

Abstract

This paper describes the design and implementation of an algorithm for improving the performance of stereo vision in environments presenting repetitive patterns or regions with relatively weak texture. The proposed algorithm makes use of the common assumption that the disparities corresponding to continuous surfaces in the world vary smoothly; we are using this assumption to alleviate the correspondence problem for pixels that cannot be reliably matched by the stereo algorithm. Our approach can be described as a reliability based filtering of the disparity image followed by a recursive propagation step. It can be applied to the output of almost any “standard” stereo algorithm with minimal modifications, and is computationally efficient.

1 Introduction

Stereo vision is now a commonly used technique for extracting information about the 3-D world in which mobile robots operate. The small size and price of modern cameras, the wealth of information contained in a good disparity map and the ever increasing computational power available on board mobile robots make stereo a preferred tool for obstacle detection and avoidance, localization, map building and visual odometry.

In principle, any stereo algorithm consists of two stages: (a) establishing correspondences between features in the two images and (b) using the shift in the position of each matched feature together with information about the stereo rig to recover the 3D position of the feature in the world. Solving the correspondence problem is generally harder than the reconstruction step, due to the inherent ambiguity in the matching process. The features to match can either be single pixels, or primitives extracted in the images, such as line segments, curves or regions.

Pixel based stereovision algorithms are very efficient in

natural, textured environments (*e.g.* outdoor natural scenes), where they provide dense 3D information. But for indoor environments, the pixel matching process is particularly difficult. The repetitive patterns and low-textured surfaces that abound in artificial environments are some of the worst possible scenarios for pixel based stereo matching (figure 1). In this paper we describe an algorithm that uses assumptions commonly made by stereo vision researchers in order to alleviate the pixel correspondence problem in such cases.

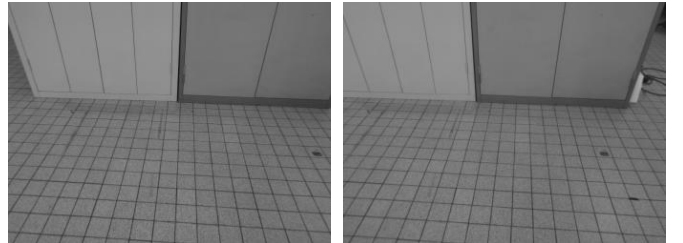


Figure 1: *An example of challenging image pair for a pixel based stereo vision algorithm*

After a brief overview of existing approaches in section 2, section 3 details our algorithm and some implementation aspects. We address efficiency issues and discuss the possible failure modes. Some illustrative results of our experiments are presented in section 4, and conclusions are drawn in section 5.

2 Related work

Numerous researchers tried to improve the performance of stereo vision by addressing problems posed by depth discontinuities and occlusion (*e.g.* [1, 2, 3]). Others have tried to use various diffusion techniques and assumptions such as the slow variations of disparity in order to share support between neighboring pixels [4].

An excellent recent survey and a taxonomy of the various stereo algorithms is presented in [5]. According to this taxonomy, our algorithm lies somewhere between the local methods and the global optimization ones, and fits into the category of seed-and-grow approaches. It is close in spirit to the methods presented in [6] and [7], and to the

¹cdima@ri.cmu.edu

²Simon.Lacroix@laas.fr

PMF algorithm [8]: we are using matches that are considered reliable to guide the search for correspondences at pixels that are ambiguous. However, the resemblance with [7], in which the authors do not perform dense stereo but only try to match edge features, stops at the idea of giving higher priority to matches that are more reliable.

The PMF algorithm [8] relies on the concept of disparity gradient limit, and is also a sparse matching algorithm using edge primitives. On the basis of psychophysical phenomena, the authors suggest that human vision uses a disparity gradient limit to help the stereo matching. Their approach relies on the definition of such a limit; some points on edge strings are then selected as seeds, and the authors measure the support given by other matches within a certain radius that respect the disparity gradient constraint. The support measures are then used to select matches in the order of the match score. The formula used is so that points that are close to each other cannot have very different disparities.

In [6] the author proposes a two step algorithm for obtaining dense matching for uncalibrated image pairs with possibly large camera motion. A set of highly distinctive features (seed points and areas) are extracted by using a RANSAC-style algorithm for seed points and an alternate sequence of region growing and matching for the seed areas. The matched seeds are ordered based on a reliability measure and they are used for a propagation step which consists in an exhaustive search for candidate matches within a 5×5 neighborhood around each paired seed. While the idea of ordering the matches by reliability is good, the concept of area-based matching is questionable; furthermore, the presented algorithm cannot be used to obtain sub-pixel precision in matching and requires thresholds that are dependent on the (unknown) amount of distortion between the two views.

The interpretation of the disparity map as a 3-D surface is an aspect that is fundamental to our method. We will show that this interpretation (used in [9, 10, 5] and numerous other papers) makes the mathematical apparatus that we use intuitive.

3 Algorithm Description

3.1 Overview

Most of the dense matching algorithms start by using knowledge of the camera parameters to warp the input images so that for each pixel (x, y) in image I_1 the corresponding pixel in image I_2 is located on the same line at coordinates $(x + d_{xy}, y)$. Given the characteristics of the stereo rig and the range of 3-D world distances from the camera that are of interest, a search interval for d_{xy} is determined; a similarity measure between areas centered at pixels $I_1(x, y)$ and $I_2(x + d_{xy}, y)$ is computed for each valid d_{xy} , resulting in what is usually called a *correlation curve*. The word “correlation” does not neces-

sarily mean normalized cross-correlation (ZNCC) is used as the matching measure: other measures like SSD, SAD or the non-parametric measures introduced in [11] are often used. However, our method is not dependent on the chosen measure.

When the correlation curve has only one sharp peak with a score that is significantly higher than any other local maximum, the matching is trivial. Unfortunately, for pixels on surfaces with repetitive patterns or very weak texture the correlation curves are more confusing (see figure 2).

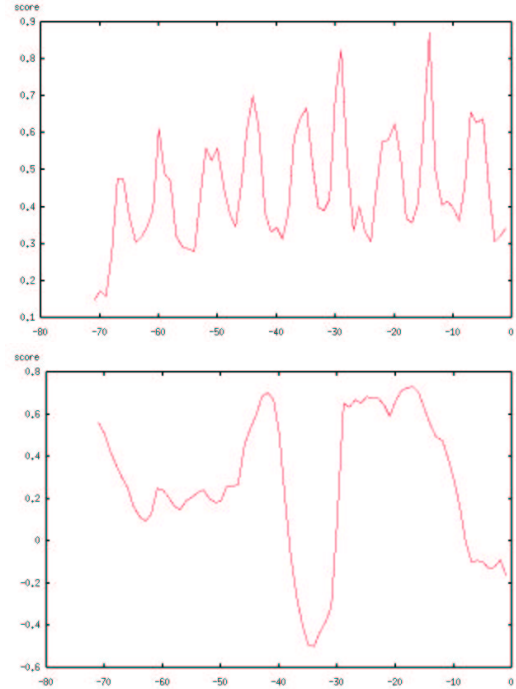


Figure 2: *Top: Correlation curve for a pixel on a repetitive pattern (the matching score is here ZNCC). In this particular case the correct disparity is -29 (a local maximum) and not -14 where the global maximum of the correlation curve is located. Bottom: Correlation curve for a pixel on a low-textured area. Note the “flatness” of the peak at the maximum (disparity = -17). The true disparity could be anywhere between -20.5 and -16.5.*

For the type of correlation curve corresponding to repetitive patterns, it is generally the case that one of the peaks is correct but not necessarily the one with the highest score, mainly because of the noise in the images. Similarly, when a relatively flat peak coming from a pixel in a low-textured area occurs, it is usually the case that the correct disparity is somewhere within a small interval around the maximum of the matching score.

Our algorithm deals with these two problems according to the following steps (detailed in the next sections):

1. A disparity map is computed, in which for every pixel we store not only the global maximum of the

correlation curve, but also the other top five local maxima and some measures of the reliability of the global maximum. These local maxima are referred to as *disparity hypotheses*.

2. We perform a grouping operation on the disparity map that results in blobs such that the pixels in each blob have disparities that vary continuously. During the grouping process we aggregate the confidence in each pixel so that at the end we have a measure of the average “match quality” for each blob. Finally, we set thresholds on these quality measures to filter out the unreliable blobs. The pixels belonging to the blobs that are filtered will be considered as *unset*, *i.e.* without a reliable disparity value assigned to them.
3. Start a recursive propagation process, which consists of (a) detecting the unset pixels that have enough reliable disparities in a small neighborhood around them, (b) getting an estimate of the disparity for the unset pixels on the basis of their reliable neighbors, and (c) using this estimate and the stored disparity hypotheses to select a valid disparity for the unset pixels.

3.2 Computing the Disparity Hypotheses

The algorithm used to compute the disparity hypotheses for each pixel is a slightly modified version of the stereo correlation algorithm described in [12]:

1. The stereo pair is corrected for distortion and rectified using the calibration information.
2. For each pixel $I_1(x, y)$ in the left image the following steps are performed:
 - we scan along a given disparity interval on the corresponding line in the right image, I_2 , obtaining the correlation curve.
 - we keep track of the top five local maxima found during this scan.
 - for the global maximum, we check 3 criteria defined on the correlation curve to declare it as a “valid” one:
 - the matching score at the global maximum
 - the difference between the score at the global maximum and the score of the second-best local maximum.
 - the sharpness of the peak, defined by the score difference between the global maximum at disparity d_{max} and the scores at disparities $d_{max} - 1$ and $d_{max} + 1$.
 - if the global maximum found in image I_2 meets these criteria, an *inverse* search is initiated. The inverse search consists in centering a window around pixel $I_2(x + d_{max}, y)$ in the right

image and computing the match score over a disparity range in the left image I_1 ; the global maximum for the inverse search is checked with the same criteria as the global maximum of the direct search. In case d_{max} is the correct disparity for the pixel $I_1(x, y)$, one would expect the best score for the inverse search to occur at $I_1(x, y)$. If this is true, d_{max} is considered to be the correct disparity for pixel $I_1(x, y)$; if not, the pixel is considered as *unset*: there is no reliable disparity estimate for it, and it will be analyzed later.

3.3 Disparity Grouping and Filtering

Once we have a disparity image with reliability measures for each pixel, we perform a grouping step which results in a set of blobs with similar disparities, using a classical blob coloring algorithm. As a result, each blob corresponds to a region of the image where the disparity varies continuously. We will see that a continuous disparity corresponds to a continuous 3-D world surface.

The purpose of the grouping step is to aggregate the per pixel confidence measures in order to reduce the effect of the noise in the disparity image. The direct and inverse searches are unfortunately unable to completely eliminate the bad matches, and having stricter thresholds during the match process would certainly eliminate some of the good matches. However, aggregating the per pixel confidence measures into a more stable per blob confidence measure allows us to do better filtering in two ways:

- if a good match only *marginally* satisfies the thresholds for the confidence per pixel, any increase in these thresholds would eliminate it. However, since a correct match is hardly due to noise, it is usually neighboring other good matches that originate from the same 3-D world surface. During the grouping step the correct match will be included into a bigger blob corresponding to correct matches *regardless* of the margin by which it satisfies the thresholds. When the confidence levels in each pixel are averaged over the whole blob, the average will be only slightly decreased by these correct but low-scoring matches.
- if a bad match passes our pixel confidence tests, one would expect that in general it will not pass them by a large margin. Furthermore, even if a bad match with a great score occurs, it is unlikely that it will be accompanied by numerous other bad matches with very close disparities and high scores. As a result, when we form groups we would expect the blobs corresponding to bad matches to be either very small or have small confidence values.

The only implicit assumption we make here is that on

average good matches will score better in our confidence tests than bad matches.

The confidence measures per blob that we compute during the grouping process are:

- the average matching score for the direct and inverse search
- the average difference between the top two maxima for the direct and inverse search
- the average sharpness of the peaks

For the filtering stage we only have three thresholds, since there is no reason to differentiate the direct and inverse searches. We also have an inferior limit on how small a blob we can accept as valid. This limit is extremely low (on the order of 10-20 pixels), to avoid the elimination of small objects that have disparities different than their surroundings in the image. This threshold on the minimal size of the blobs is necessary because for really small samples of pixels the average is not stable enough.

Deciding what disparity blobs to eliminate can be accomplished by any function of the blob confidence measures; however, simple thresholds on the blob statistics proved to be reliable in our experiments. The development of more complex decision functions for filtering is left for future research.

As the output of this stage we have a disparity map with multiple hypotheses per pixel, in which the unreliable pixels are labeled as such¹. Furthermore, the multiple disparity hypotheses per pixel are only relevant for the unreliable pixels; for the others the global maximum is accepted as a valid disparity estimate.

3.4 Local Planar Approximation of the Disparity Surface

As shown in figure 2, pixels on repetitive patterns have correlation curves that present multiple local maxima with similar scores. In this case, because of foreshortening and the different viewing angle of the two cameras the global maximum does not necessarily correspond to the correct disparity. However, it is almost always the case that one of the local maxima is the correct one; if we can somehow get a rough estimate of the correct disparity, we can narrow the search for a maximum to its neighborhood and select the correct peak even if it is not the global maximum of the correlation curve. Our method to obtain an estimate of the disparity value at an unreliable pixel is to locally approximate the *disparity surface* with a plane.

¹Any concept of disparity blob is here abandoned, its only purpose being to make the reliability based filtering more robust

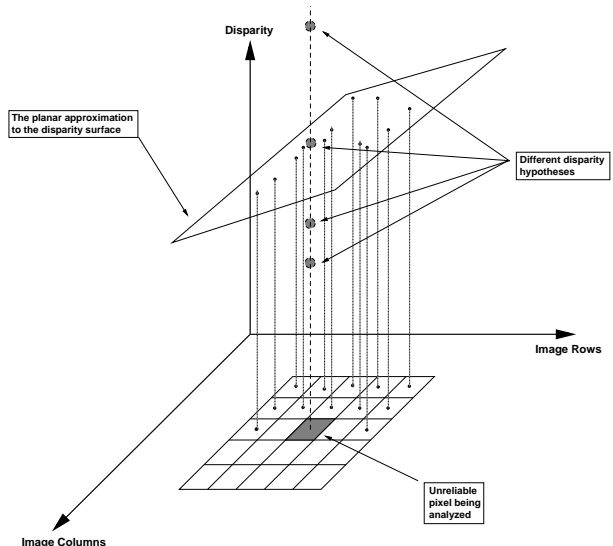


Figure 3: Using the local planar approximation of the disparity map disambiguate the match.

To have an intuition on this approximation process, it helps to think of the disparity map as an elevation map. A disparity map in which some of the pixels are labeled as unreliable corresponds to a surface with holes. If around any missing (*i.e.* unreliable) value we have enough good estimates of the disparity, we can then use the local planar approximation and get a rough estimate on the missing disparity; while this value will obviously not be good enough to be kept as a final disparity estimate, it will be sufficient to disambiguate the match in the case of a repetitive pattern (figure 3).

We would like to stress the fact that fitting a plane to the disparity surface is a principled operation based on the fact that disparity varies slowly along surfaces (see [9] for an excellent analysis and justification of this assumption). The same type of local approximation of the disparity surface is used in [10], where the authors also prove its equivalence to imposing an affine constraint on the transformation between two patches in the left and right image.

The first step of the process of fitting a plane through the valid disparities in a small neighborhood around a pixel of interest consists in making sure that the estimation of the plane parameters would not be degenerate. While in theory we could fit a plane through as little as three non-collinear points, in practice the noise in the reliable disparities found would make such an estimate numerically unstable (not to mention the need to check the points collinearity). In our algorithm we require that at least a third of the pixels in a 5x5 neighborhood (*i.e.* 8 out of 24) have valid disparities before fitting a plane. This results in an over-determined system of equations that, solved using a least-squares fit (which also allows to bypass the collinearity test, as it is impossible to have 8

collinear points in a 5x5 neighborhood).

Once the parameters of the fitting plane are found, we can obtain the disparity estimate at the unreliable pixel and step into the disparity propagation algorithm:

1. The disparity image is scanned once, to count the number of reliable pixels in the 5x5 neighborhood of all the unset pixels. If the result of the count is greater than a third of the pixels in the neighborhood we, insert the pixel in a linked list of “treatable” pixels.
2. We scan the linked list and for each element we attempt to select one of the multiple disparity hypotheses stored in the pixel structure. A plane is fit to the disparity surface to obtain an estimate of the disparity at the pixel being treated. At this point, we are in one of the following three cases:
 - (a) *The estimate is within 1 pixel of one of the hypotheses.* In this case we have high confidence in the fact that we identified a correct local maximum and subsequently the correct disparity. We set this hypothesis as the unique estimate at this location in the image and we relabel the pixel as *valid* (meaning that it can be used for further propagation steps).
 - (b) *The estimate is between 1 and 3 pixels from one of the hypotheses, and this hypothesis is a flat peak.* In this case we will consider the estimate correct, but since this pixel does not provide any real information about its neighbors we will *not* use it for further propagation steps. We call such pixels *set*.
 - (c) *The estimate is more than 3 pixels away from any disparity hypothesis that we have.* In this case we will not do anything, since the planar approximation does not provide sufficient information for establishing a reliable match. The pixel will still be considered as invalid.

In the case (a), one more step is performed before removing the pixel from the list: the count of the *unset* pixels within a 5x5 neighborhood of the pixel is incremented: the ones that then become “treatable” are added to the linked list. This step is crucial for the computational efficiency of our algorithm; after the initial disparity image scan, there is no need to ever scan the whole image again: the propagation will recursively occur.

3. The propagation step simply stops when the list of treatable pixels becomes empty. This is guaranteed to occur since each element of the list is deleted once it is analyzed and can not be subsequently added. The number of pixels that can become valid is bounded by the number of total pixels in the image and thus so is the number of iterations.

3.5 Computational Efficiency Issues

Since our algorithm can work with almost any pixel stereo implementation, we separate the complexity analysis for producing the initial disparity map from the analysis of our algorithm.

In general runtime for stereo correlation algorithms depend heavily on the size of the windows being used for computing the match measure, on the range of disparities being searched and on the size of the stereo pair. However, through careful implementation the dependency on the size of the correlation window can be almost annihilated, meaning that with relatively low overhead one can achieve almost constant runtime in the size of the correlation window [12]. Our only change to the algorithm of [12] is to record the top five local maxima instead of the global maximum for each correlation curve, and to record more confidence information per pixel. As a result this only added $O(1)$ operations per pixel.

The implementation of the grouping step requires only one pass through the disparity image, so its runtime is $O(n)$ in the number of image pixels. We have not presented the specific details of the implementation, but it is enough to mention that the grouping and filtering step require at least one order of magnitude less time than the correlation or propagation steps.

The propagation complexity is the following: the initial pass through the image (in which we count the reliable neighbors of each unreliable pixel and we construct the linked list) is $O(n)$ in the number of pixels in the image. We have shown that the number of treatable pixels added to the list and analyzed is bounded by the number of pixels in the image, so the propagation step is also $O(n)$. As a result, the whole propagation process requires a time that is linear in the number of image pixels.

4 Results

4.1 Technical Setup

The stereo pairs on which we present results in this paper and in its online version were captured using a stereo rig with a 300 mm baseline. We have used Digital Interface XCD-X700 digital cameras with Computar lenses (4.8mm focal, 1:1.8 stop). The system was calibrated using a combination of the Matlab implementation of the OpenCV camera calibration routines and the Calife system developed at LAAS-CNRS. Calife was also used for rectifying and sub-sampling the images.

In our experiments we had the option to choose the matching metric. We performed tests using the zero mean normalized cross-correlation (ZNCC), the Census non-parametric image transform [11], and various small variations of the two. Our system generally performed better with the census image transform: this is what we used to generate the following results.

4.2 Experimental Results

A large number of tests were performed to thoroughly evaluate the performance of our algorithm in a variety of real-world conditions. We have collected more than 200 stereo pairs in various locations of the LAAS building, making sure to encounter cases that are well known to cause poor performance in stereo algorithms: specular reflections and shadows, repetitive patterns and regions with low texture (walls, furniture, etc.) and scenes presenting depth discontinuities and occlusions.

The main objective of our research was to address what we have identified as the worst problems of indoor stereo, *i.e.* the repetitive patterns and the low-textured regions. We did not tackle other problems that represent open research questions in themselves, like using correlation windows with adaptive sizes for optimal behavior at occlusion boundaries. Our experiments prove that our algorithm handles extremely well the cases it was designed to handle and it does not perform worse than the original stereo algorithm on the others.

Figure 4 presents the typical performance of our algorithm on a repetitive pattern; the scene is also presenting a relatively strong foreshortening effect.

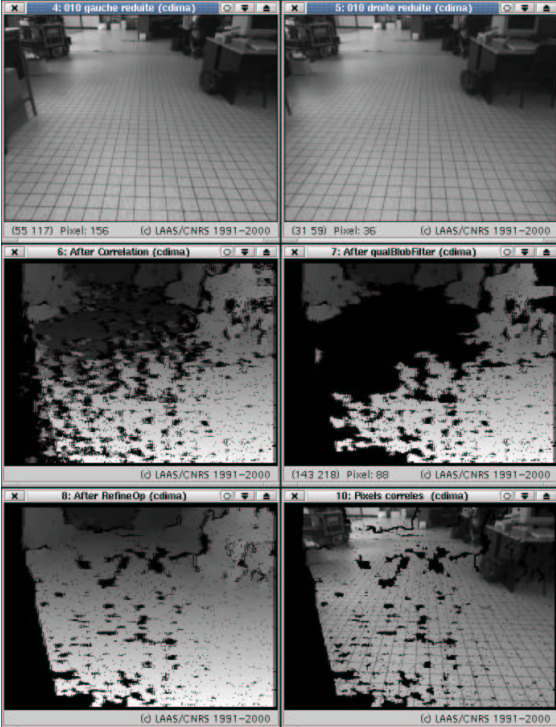


Figure 4: Top row: difficult stereo pair. Note the unfavorable alignment of the tiles with the optical axis. Middle row: the output of the standard stereo-correlation algorithm (left) and the results of the quality-based filtering step (right). In the unfiltered output there is a large amount of wrong disparity estimates in the upper-left corner. Bottom row: result of the propagation step (left) and original left image in which only the matched pixels are present (right).

There are also theoretical limitations on how much lack of texture a stereo algorithm or even a human can handle. Our method has better performance than a standard algorithm in low-texture conditions, and it breaks down nicely by stopping the propagation process before producing erroneous matches: figure 5 shows its result on a scene presenting a uniformly painted closet that makes the correspondence problem challenging.

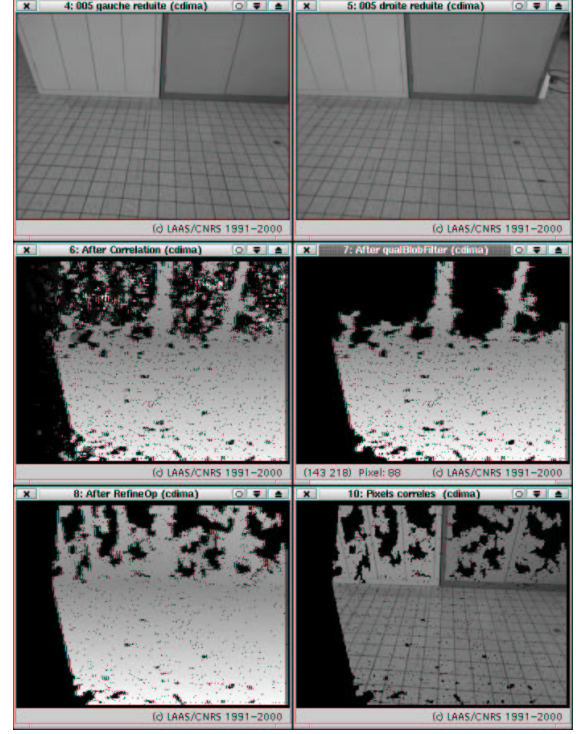


Figure 5: Results of our algorithm on a stereo pair presenting low-textured areas.

In our experiments we wanted to check if our method is robust enough to handle variations in the environment without requiring the readjustment of the various thresholds on a per scene basis. For this purpose we have taken a sequence of 40 image pairs in various positions with respect to the lighting sources in our lab; as a result we have an image series with various degrees of shadow and specular reflections effects. We have processed all this sequence without changing any threshold and the system proved to be very reliable. This image series and the results of our algorithm on each stereo pair in the sequence are available in the online version of this paper².

4.3 Failure Modes

Throughout our experiments we have identified two types of failure modes:

Filtering Errors: they can occur if blobs of erroneous disparities pass our confidence tests; their effect could be serious given that the propagation step will use bad data and will most likely enlarge the blobs. However,

²URL http://www.ri.cmu.edu/~cdima/research/LAAS_2001/

the use of numerous confidence measures and averaging them makes it so that even the simple decision rule based on thresholds that we use is very reliable. In case even more reliability is needed (for a dependable system for example) our method could easily be modified to work with more complex decision rules.

Another problem during filtering can occur if somehow erroneous matches have disparities that are continuous with good matches nearby. In this case the bad pixels will be added to a good blob, and if this one is extremely large it could be the case that its average reliability measures are not sufficiently decreased by the bad pixels.

This latter failure occurred only *once* over all our tests (more than 200 stereo pairs). We have unsuccessfully tried to make this error re-occur by taking other stereo pairs in the same position and illumination conditions, and we concluded that its probability of occurrence is negligible. If high dependability needs to be guaranteed, one could change our grouping step so that the confidence measures are not averaged over whole blobs but only within some distance from the pixel being evaluated.

Propagation Errors: Our method imposes very strong requirements before any unreliable pixel is relabeled as good: we ask that the disparity estimate is within one pixel of a peak of the correlation curve. Our experiments have shown that the tolerance of one pixel is not tight, any value between 0.5 and 2 pixels working almost as well. We have also tried imposing some additional constraints such as having a minimal acceptable score of a selected local maximum or having a maximal value for the residual of the plane fit. While both these constraints make sense, we have simply found that the algorithm works just as well without them: it does not try to propagate in the occluded areas of the image and it handles well depth discontinuities. We have only seen propagation errors in cases where we expected to see them: on repetitive patterns at very large distances from the camera (close to 0 disparity) where we have a sudden change in the orientation of the 3-D world surface but not enough information in the image data. This failure mode was expected given the focal length of the lenses we used and the resolution of our images; we consider it an unimportant failure mode for any real application since any depth reconstruction at that range is inherently unreliable. For concrete examples of this failure mode we refer our reader to the online version of this work.

5 Conclusions and Future Work

We have presented an algorithm that improves the performance of stereo vision in indoor environments. The algorithm is extremely simple, performs well and is computationally efficient. Given that we did not address the issue of using adaptive window sizes and that depth discontinuities are frequent indoors we think this should be

the next addition to our algorithm in order to have a performant indoor stereo system. Implementing the various approaches that we proposed for increasing the reliability of our algorithm even further is another direction to take for obtaining a deployable system.

Acknowledgments: During this research project Cristian Dima has been funded by the Centre National de Recherche Scientifique. We would also like to thank Florent Lamiraux for all the support provided.

References

- [1] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *IEEE TPAMI*, pp. 16(9):920–932, 1994.
- [2] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *CVPR*. Udine Univ. (italie), June 1999, pp. 858–863.
- [3] Daniel Scharstein and Richard Szeliski, "Stereo matching with non-linear diffusion," *IJCV*, vol. 28, no. 2, 1998.
- [4] Abdol-Reza Mansouri, Amar Mitiche, and Janusz Konrad, "Selective image diffusion: application to disparity estimation," in *IEEE Int. Conf. Image Processing*, Oct. 1998, vol. 3, pp. 284–288.
- [5] Daniel Scharstein and Richard Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *IJCV*, 2002.
- [6] Maxime Lhullier, "Efficient dense matching for textured scenes using region growing," in *British Machine Vision Conference*, 1998.
- [7] Osafumi Nakayama, Akashi Yamaguchi, Yoshiaki Shirai, and Minoru Asada, "A multistage stereo method giving priority to reliable matching," in *ICRA*, Nice, France, May 1992, pp. 1753–1758.
- [8] Stephen B. Pollard, John E. W. Mayhew, and John P. Frisby, "Pmf: A stereo correspondence algorithm using a disparity gradient limit," *Perception*, vol. 14, pp. 449–470, 1985.
- [9] Charles V. Stewart, Robin Y. Flatland, and Kishore Bubna, "Geometric constraints and stereo disparity computation," *IJCV*, vol. 20, no. 3, pp. 143–168, 1996.
- [10] K. Palaniappan, A.F. Hasler, Y. Huang, and X. Zhuang, "Robust stereo analysis," in *Proc. IEEE Int. Symp. on Computer Vision*, 1995, pp. 175–181.
- [11] Ramin Zabih and John Woodfill, "Non-parametric local transforms for computing visual correspondence," in *ECCV*, 1994, vol. II, pp. 151–158.
- [12] O. Faugeras and T. Vieville et al., "Real-time correlation-based stereo: algorithm, implementations and applications," Tech. Rep. RR-2013, INRIA - Sophia Antipolis, August 1993.