INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Real time correlation-based stereo: algorithm, implementations and applications

Olivier Faugeras - Bernard Hotz - Hervé Mathieu - Thierry Viéville
Zhengyou Zhang - Pascal Fua - Eric Théron - Laurent Moll
Gérard Berry - Jean Vuillemin - Patrice Bertin - Catherine Proy

N° 2013

août 1993

PROGRAMME 4

Rapport de recherche

ISSN 0249-6399

Real time correlation-based stereo: algorithm, implementations and applications

Olivier Faugeras - Bernard Hotz - Hervé Mathieu - Thierry Viéville
Zhengyou Zhang - Pascal Fua - Eric Théron - Laurent Moll
Gérard Berry - Jean Vuillemin - Patrice Bertin - Catherine Proy

Programme 4 — Robotique, image et vision
Projet Robotvis

Rapport de recherche n ° 2013 — août 1993 — 49 pages

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

Abstract: This paper describes some of the work on stereo that has been going on at INRIA in the last four years. The work has concentrated on obtaining dense, accurate, and reliable range maps of the environment at rates compatible with the real-time constraints of such applications as the navigation of mobile vehicles in man-made or natural environments.

The class of algorithms which has been selected among several is the class of correlation-based stereo algorithms because they are the only ones that can produce sufficiently dense range maps with an algorithmic structure which lends itself nicely to fast implementations because of the simplicity of the underlying computation. We describe the various improvements that we have brought to the original idea, including validation and characterization of the quality of the matches, a recursive implementation of the score computation which makes the method independent of the size of the correlation window, and a calibration method which does not require the use of a calibration pattern.

We then describe two implementations of this algorithm on two very different pieces of hardware. The first implementation is on a board with four Digital Signal Processors designed jointly with Matra MSII. This implementation can produce 64×64 range maps at rates varying between 200 and 400 ms, depending upon the range of disparities. The second implementation is on a board developed by DEC-PRL and can perform the cross-correlation of two 256×256 images in 140 ms.

The first implementation has been integrated in the navigation system of the INRIA cart and used to correct for inertial and odometric errors in navigation experiments both indoors and outdoors on road. This is the first application of our correlation-based algorithm which is described in the paper. The second application has been done jointly with people from the french national space agency (CNES) to study the possibility of using stereo on a future planetary rover for the construction of Digital Elevation Maps.

We have shown that real time stereo is possible today at low-cost and can be applied in real applications. The algorithm that has been described is not the most sophisticated available but we have made it robust and reliable thanks to a number of improvements. Even though each of these improvements is not earth-shattering from the pure research point of view, altogether they have allowed us to go beyond a very important threshold. This threshold measures the difference between a program that runs in the laboratory on a few images and one that works continuously for hours on a sequence of stereo pairs and produces results at such rates and of such quality that they can be used to guide a real vehicle or to produce Discrete Elevation Maps. We believe that this threshold has only been reached in a very small number of cases.

(Résumé : tsvp)

Stéréoscopie en temps réel à base de corrélations : algorithme, implémentations et applications

Résumé : Cet article fait la synthèse de travaux réalisés dans le domaine de la stréréoscopie à l'INRIA ces quatre dernières années. Ces travaux se sont concentrés sur l'obtention de cartes de profondeur denses, précises et fiables à des cadences compatibles avec les contraintes du temps réel liées à des applications comme la navigation d'un véhicule dans un environnement artificiel ou naturel.

La classe d'algorithmes qui a été sélectionnée parmi les différentes méthodes possibles de stéréoscopie est celle des algorithmes à base de corrélations. Seules ces méthodes peuvent produire des cartes de profondeurs suffisamment denses à partir d'une structure algorithmique qui se prête facilement à des implémentations rapides de par sa simplicité intrinsèque. Nous décrivons différentes améliorations apportées au schéma initial, en particulier la validation et la caractérisation de la qualité des appariements, une implémentation récursive des scores de calcul qui rend la méthode indépendante de la taille de la fenêtre de corrélation, et une méthode de calibration qui ne nécessite pas l'utilisation d'une grille de calibration.

Nous décrivons ensuite deux implémentations de cet algorithme sur deux systèmes matériels très différents. La première implémentation a été réalisée sur une carte industrielle dotée de quatre processeurs de traitement du signal (DSP) réalisée conjointement par l'INRIA et Matra-MSII. Cette implémentation permet de calculer des cartes de profondeurs de taille 64×64 à des cadences variant entre 200 et 400 msec, selon l'intervalle de disparité requis. La seconde implémentation a été réalisée sur une carte développée par DEC-PRL et peut réaliser le calcul d'une carte d'intercorrélation de deux fois 256×256 en 140msec.

La première implémentation a été intégrée au système de navigation du robot mobile de l'INRIA et utilisée pour corriger les erreurs des capteurs inertiels et odométriques lors d'expérimentations de navigation autonomone en intérieur et en extérieur, sur une route. Ce travail constitue la première application de l'algorithme décrit dans ce papier. La seconde application a été réalisée conjointement avec une équipe de l'agence nationale française spatiale (CNES) pour étudier les possibilités d'utilisation de la stéréoscopie lors de futures explorations planétaires nécessitant la construction de cartes numériques de terrain.

Nous avons ainsi démontré qu'il est possible aujourd'hui de réaliser à faible coût un module de vision stéréoscopique temps-réel et utilisable au sein d'applications effectives. Cet algorithme n'est pas le plus sophistiqué des méthodes actuellement disponibles, mais il a été rendu robuste et fiable grâce à de nombreuses améliorations. Bien que chacune de ces améliorations ne soient pas vraiment fondamentale au niveau théorique, ajoutées les unes aux autres, elles permettent de dépasser une frontière rédibitoire. Cette frontière est celle qui sépare les programmes qui tournent en laboratoire sur quelques images et ceux qui travaillent de manière continue des heures durant sur des séquences de paires stéréoscopiques et produisent des résultats à des cadences et à des niveaux de qualité tels que l'on peut les utiliser pour guider un robot mobile sur sa trajectoire ou produire un modèle numérique de terrain. Nous croyons que cette frontière n'est franchie encore qu'assez rarement.

Real time correlation-based stereo: algorithm, implementations and applications

Olivier Faugeras Bernard Hotz Hervé Mathieu
Thierry Viéville Zhengyou Zhang
INRIA
2004 route des Lucioles, B.P. 93, 06902 Sophia-Antipolis cedex FRANCE
, **Pascal Fua**
SRI International
AI Center - 333 Ravenswool
Menlo Park, 94025 CA USA
fua@ai.sri.com
, **Eric Théron**
MS2i
Les Quadrants - 3 av du Centre
78182 Saint Quentin en Yvelines Cedex FRANCE
, **Laurent Moll**
Ecole Polytechnique
91128 Palaiseau Cedex FRANCE
moll@cma.cma.fr
, **Gérard Berry**
Ecole des Mines
Bld Albert Einstein
Sophia Antipolis FRANCE
berry@cma.cma.fr
, **Jean Vuillemin Patrice Bertin**
DEC-PRL
85 avenue Victor Hugo
92563 Rueil Malmaison cedex FRANCE
vuillemin@prl.dec.com
, **Catherine Proy**
CNES TOULOUSE
18, avenue Emile Belin
31055 Toulouse FRANCE
proy@hathor.cnes.fr

Contents

List of Figures

1 Introduction

Dense depth maps can be obtained from stereo at relatively high rates and resolutions using simple algorithms and hardware. They are useful for automatic land vehicles navigation in particular but in general for any task requiring a dense three-dimensional representation of three space. This article presents a summary of the work that has been going on at INRIA on the construction and use of dense three-dimensional maps from stereo. This work includes algorithmic development and coding on standard architectures, the porting of the algorithm on two very different parallel architectures, the use of one of these architectures in a robot navigation application indoors and on roads, and the construction of Digital Elevation Models (DEMs) for the navigation of a rover for future planetary exploration.

The literature on stereo is very large and we will not attempt to review it here. Let us just briefly recall the fact that the main difficulty in stereo vision is to establish correspondences between pairs (triplets, etc...) of images. Trying all possible correspondences is still out of the possibilities of current computers because of the combinatorial explosion problem and people have been using constraints to reduce it. These constraints are basically of three kinds:

1. Geometric constraints imposed by the imaging system: probably the most important such constraint is the epipolar constraint thanks to which we can transform a two-dimensional search for correspondence into a one-dimensional one.

2. Geometric constraints arising from the objects being looked at: we can assume, for example that their distance to the imaging system varies slowly almost everywhere. This is, for example, the origin of the disparity gradient constraint.

3. Physical constraints such as those arising from models of the way objects interact with the illumination. The simplest and most widely used such model is the Lambertian model [15].

Stereo algorithms also differ according to the type of tokens they attempt to match and to the type of features they use to represent them. Almost all possibilities of tokens have been proposed, from the single pixel, the edge pixel, curves of various sorts up to image regions. For each of these tokens numerous features have been used, the limit being set by the researchers' imagination.

The algorithms for stereo fall broadly into four main categories

Correlation-based algorithms : intensity based area-correlation techniques have been investigated extensively for commercial applications in stereo-photogrammetry [17, 9] but are also one of the oldest methods used in computer vision [12, 25]. A recent application of this class of techniques to Planetary rovers can be found in [23].

Relaxation-based algorithms : The basic idea of this class of techniques is to allow the pixels that are to be put into correspondence make "educated guesses" as to what their match should be and then let the matches reorganize themselves by propagating some of the above constraints. Three famous examples of this class of algorithms are the Marr-Poggio algorithm [20, 21], the Grimson algorithm [13, 14], and the Pollard-Mayhew-Frisby algorithm [27].

Dynamic programming : The problem of matching primitives between images can also be cast as a problem of minimizing a cost function. Dynamic programming is a way of efficiently minimizing (or maximizing) functions of a large number of discrete variables. Successful attempts at using dynamic programming for solving the stereo matching problem are those of Baker and Binford [4], and Ohta and Kanade [26]. In both cases, they were using edges as the basic primitives.

Prediction and verification : This is a category of stereo algorithms where the tokens put into correspondence are of a higher symbolic level than pixels. This approach has been followed in particular by Medioni and Nevatia [24], Ayache and Faverjon [2], Ayache and Lustman [3], Yachida [18], and Robert [28], among others.

For a recent review, the reader is referred to [5].

Since we were interested for our applications in obtaining dense three-dimensional maps at a reasonable computational cost, this oriented us from the beginning toward the first category. The algorithm described in this paper is a variant of this class of techniques.

We started working on the algorithm in 1990 [10]. This original version was improved later and ported on a DSP board in collaboration with Matra-MSII and on a XLinks board by Molle, Berry, and Vuillemin. These hardware implementations have been used in several navigation and exploration tasks of which we describe two. The first application is the navigation of the INRIA mobile cart indoors and on the road using a combination of three sensory processes: odometry, inertia, and correlation based stereo. In this application the stereo is used in a very simple manner to localize the edges of the hallway for the indoors part, and of the road for the outdoors part. The second application is the construction of Digital Elevation Models (DEMs) for the navigation of a rover in a rocky environment. The DEMs are intended to help planning trajectories and are built incrementally by fusing local 3-D representations obtained by the stereo rig mounted on the rover from several viewpoints.

The paper is organized as follows. In section 2 we recall briefly the original algorithm described in [10, 11] and the latest improvements. In section 3 we describe the first hardware parallel implementation on a board composed of four DSP's from Motorolla and present its performances. In section 4 we describe the second hardware implementation on the DECPeRLe-1 board and present its performances. In section 5 we present the indoors and outdoors navigation experiments, in section 6 we present the DEM's construction experiment.

2 Description of the algorithm

The algorithm that we are about to describe falls in the category of the correlation based stereo algorithms. The tokens which are used in the correspondence process are the image pixels themselves with one feature, the intensity at the pixel. The algorithm uses two geometric constraints, the epipolar constraint to reduce the search for correspondences, and the constraint that the disparity (or depth) is locally constant in the vicinity of a pixel. The second constraint is clearly only an approximation. The algorithms also uses the constraint that the intensities at two corresponding pixels are approximately the same. This is in fact a physical constraint related to the hypothesis that the observed objects are approximately lambertian.

2.1 Calibration and rectification

Calibration of a stereo rig is necessary for using the epipolar constraint and for three-dimensional reconstruction.

2.1.1 Camera sytem calibration

Calibration of a stereo camera system requires a calculation of intrinsic and extrinsic parameters for both cameras. This amounts to finding a relationship (perspective projection) between the 3-D points in the scene and their different camera images [8, 29, 30]. This is a crucial stage in the vision process as it allows to simplify the correspondence problem and to obtain a 3-D representation of the results. Calibration has been performed in two ways. Initially it was performed using images of a predefined calibration grid. Specific points were extracted to obtain a sufficient number of combinations (pixel, 3-D point) to compute the perspective projection matrices. Using a calibration grid might be problematic for some applications. Therefore we also used the so called “weak” calibration technique [7, 19] for the Mars rover system. Weak calibration means that only the epipolar geometry of the stereo rig is known, but not the intrinsic parameters of the cameras. This only requires the knowledge of a small number of pixel correspondences between images. It does not require either the knowledge of any three-dimensional information or the use of a special calibration grid. One drawback of this calibration scheme is that 3D reconstruction is done in an unknown affine or projective frame [6]. Hence, adjustment against a metric frame is compulsory before exploiting results, at least at the present stage of our knowledge.

2.1.2 Epipolar geometry simplification: Image rectification

The pinhole camera model implies that the correspondent of a given point lies on its epipolar line in the other image. Corresponding points can then be found by scanning every point's epipolar lines. Unfortunately the epipolar lines form a pencil of lines, i.e. they all go through a point called the epipole. This makes the scanning task fairly complex and thus inefficient. This is why the images are reprojected onto a plane parallel to the line between the optical centers, in the case of a binocular stereo rig, or to the optical centers plane for a trinocular system. In the case of a binocular rig, we can align the epipolar lines with the image rows which greatly simplifies the correspondence process. In the case of a trinocular rig, an L camera setup allows to align the epipolar lines with the image rows or columns and to reduce the image deformations due to rectification with no perturbing effects on the following pairing stage.

2.2 Matching algorithm

A number of correlation-based algorithms attempt to find points of interest on which to perform the correlation. This approach is justified when only limited computing resources are available, but with modern hardware architectures it becomes practical to perform the correlation over all image points and retain only matches that appear to be "valid". The hard problem is then to provide an effective definition of what we call validity and we will propose one below.

Correlation scores are computed by comparing a fixed window in the first image to a shifting window in the second. The second window is moved in the second image by integer increments along the corresponding epipolar line and a curve of correlation scores is generated for integer disparity values. The mesured disparity can then be taken to be the one that provides the largest peak. To compute the disparity with subpixel accuracy, we fit a second degree curve to the correlation scores in the neighborhood of the extremum and compute the optimal disparity by interpolation.

2.2.1 Correlation criteria

To quantify the similarity between two correlation windows, we must choose among many different criteria the one that produces reliable results in a minimum computation time. We denote by $I_1(x,y)$ and $I_2(x,y)$ the intensity values at pixel (x,y). The correlation window has dimensions $(2n+1)\times(2m+1)$. Therefore, the indexes which appear in the formula below vary between $-n$ and $+n$ for the i-index and between $-m$ and $+m$ for the j-index. We have extensively tested the four criteria defined below (we take the case of horizontal epipolar lines with the same image row index, so that we have no y- or vertical disparity):

$$C_1(x,y,d) = \frac{\sum_{i,j}[I_1(x+i,y+j) - I_2(x+d+i,y+j)]^2}{\sqrt{\sum_{i,j} I_1(x+i,y+j)^2} \times \sqrt{\sum_{i,j} I_2(x+d+i,y+j)^2}}$$

$$C_2(x,y,d) = \frac{\sum_{i,j} I_1(x+i,y+j) \times I_2(x+d+i,y+j)}{\sqrt{\sum_{i,j} I_1(x+i,y+j)^2} \times \sqrt{\sum_{i,j} I_2(x+d+i,y+j)^2}}$$

$$C_3(x,y,d) = \frac{\sum_{i,j}[(I_1(x+i,y+j) - \overline{I_1(x,y)}) - (I_2(x+d+i,y+j) - \overline{I_2(x+d,y)})]^2}{\sqrt{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x,y)}]^2} \times \sqrt{\sum_{i,j}[I_2(x+d+i,y+j) - \overline{I_2(x+d,y)}]^2}}$$

$$C_4(x,y,d) = \frac{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x,y)}] \times [I_2(x+d+i,y+j) - \overline{I_2(x+d,y)}]}{\sqrt{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x,y)}]^2} \times \sqrt{\sum_{i,j}[I_2(x+d+i,y+j) - \overline{I_2(x+d,y)}]^2}}$$

where $\overline{I_k(x,y)}$, $k=1,2$ is the average of image k within the window of size $(2n+1)\times(2m+1)$. These four criteria can be expressed compactly in vector form if we consider the vectors of size $(2n+1)(2m+1)\times 1$ obtained by stacking the columns of the image windows. With obvious notations, we can rewrite them as follows:

$$C_1(x,y,d) = \frac{\|\mathbf{I}_1(x,y)-\mathbf{I}_2(x+d,y)\|^2}{\|\mathbf{I}_1(x,y)\|\cdot\|\mathbf{I}_2(x+d,y)\|} \qquad C_2(x,y,d) = \frac{\mathbf{I}_1(x,y)\cdot\mathbf{I}_2(x+d,y)}{\|\mathbf{I}_1(x,y)\|\cdot\|\mathbf{I}_2(x+d,y)\|}$$

$$C_3(x,y,d) = \frac{\|\mathbf{J}_1(x,y)-\mathbf{J}_2(x+d,y)\|^2}{\|\mathbf{J}_1(x,y)\|\cdot\|\mathbf{J}_2(x+d,y)\|} \qquad C_4(x,y,d) = \frac{\mathbf{J}_1(x,y)\cdot\mathbf{J}_2(x+d,y)}{\|\mathbf{J}_1(x,y)\|\cdot\|\mathbf{J}_2(x+d,y)\|}$$

The C_1 and C_3 criteria use the difference between the gray levels of the images and must be minimized with respect to the disparity d. The C_2 and C_4 criteria multiply the gray level values together and must be maximized with respect to d. C_3 and C_4 are similar to C_1 and C_2 respectively, except for the fact that the mean gray level value over the correlation window is substracted from the intensity values. From the expressions of C_i, $i=1,\cdots,4$, it is apparent that the value of C_1 does not change if we replace the two images I_1 and I_2 with aI_1+b and aI_2+b with the *same* values of a and b, that the value of C_2 does not change if we replace I_1 by aI_1 and I_2 by bI_2, that the value of C_3 does not change if we replace I_1 by aI_1+b_1 and I_2 by aI_2+b_2, and that the value of C_4 does not change if we replace I_1 by $a_1I_1+b_1$ and I_2 by $a_2I_2+b_2$. We have found C_3 and C_4 performed best in practice because they are the most invariant to affine transformations of the images which may result from slightly different settings of the cameras. C_2 has similar performances to C_3 and C_4 except when the difference in the distribution of gray levels between the images is important. C_1 clearly produces worse results than the others criteria.

2.2.2 Validating matches

As shown by Nishihara [25], the probability of a mismatch goes down as the size of the correlation window and the amount of texture increase. However, using large windows leads to a loss of accuracy and to the possible missing of important image features. For smaller windows, the simplest definition of validity would call for a threshold on the correlation score; unfortunately such a threshold would be rather arbitrary and, in practice, hard to choose. Another approach is to build a correlation surface by computing disparity scores for points in the neighborhood of a prospective match and checking that the surface is peaked enough [1]. It is more robust but also involves a set of relatively arbitrary thresholds.

Here we propose a definition of a valid disparity measure [11] in which the two images play a symmetric role and that allows us to greatly reduce the probability of error even when using very small windows. We perform the correlation twice by reversing the roles of the two images and consider as valid only those matches for which the reverse correlation has fallen on the initial point in the left image.

This validity test is likely to work in the presence of an occlusion. Indeed, let us assume that a portion of a scene is visible in the left image I_1 but not in the right image I_2. The pixels in I_1 corresponding to the occluded area in I_2 will be matched, more or less at random, to points of I_2.. The reverse correlation will usually find better matches in I_1 for those points. The matches for the occluded points will therefore be declared invalid and rejected.

In fact, the density of such consistent matches in a given area of the image appears to be an excellent indicator of the quality of the stereo matching. An occasional "false positive" (a pixel for which the same erroneous disparity is measured when matching both from left to right and right to left) may occur. But, except in the presence of repetitive patterns, we have never encountered a situation that gave rise to a large clump of such errors.

When the correlation between the two images of a stereo pair is degraded our algorithm tends, instead of making mistakes, to yield sparse maps. In other words, a relatively dense disparity map is a *guarantee* that the matches are correct, at least up to the precision allowed by the resolution being used. If we reject not only invalid matches but also isolated valid matches (using a simple method based on successive erosions and dilatations) we can increase even more the ratio correct/incorrect matches without losing a large number of the correct answers.

2.2.3 More information about matches

It is very interesting to really know if a validated match is reliable or not. If the match is situated in a dense area in the disparity map, the probability of a correct correlation is very high, except on repetitive patterns. But the form of the correlation curve (criterion value for all integer disparity values) can be used to decide if the probability of the match to be an error is high or not. Indeed, errors occur when a wrong peak slightly higher than the right one is chosen. So, if in the correlation curve we can notice the presence of several peaks with approximately the same height, the risk of choosing the wrong one increases, especially if the images are noisy. We have therefore defined a confidence coefficient proportionnal to the difference of height between the two most important peaks (which must be sufficiently distant when using small windows for which the correlation curve has a lot of small noisy peaks). On repetitive patterns the correlation curve has a periodic-like shape and the confidence will receive a very low value.

We can extract another type of information from the correlation curve. Indeed, the shape of the optimal peak shows us if the matched points are situated in bland areas or not. The narrower the peak is, the more precise the localisation of the matched point is. So, a good way to quantify the accuracy of the sub-pixel disparity computed by the parabolic approximation is to measure the spread of the optimal peak. In order to do this, we assume that the peak can be locally represented as a gaussian with standard deviation σ and take the sub-pixel precision proportionnal to σ.

2.2.4 Hierarchical algorithm

To increase the density of our potentially sparse disparity map, we use windows of a fixed size to perform the matching at several levels of resolution (computed by subsampling gaussian smoothed images), which is almost equivalent to, but computationally more efficient, than matching at one level of resolution with windows of different sizes More precisely, it amounts to performing the correlation using several frequency bands of the image signal.

We then merge the disparity maps by selecting, for every pixel, the highest level of resolution for which a valid disparity has been found. The reliability of our validity test allows us to deal very simply with several resolutions without having to introduce, a correction factor accounting for the fact that correlation scores for large windows tend to be inferior to those for small windows.

The computation proceeds independently at all levels of resolution and this is a departure from traditional hierarchical implementations that make use of the results generated at low resolution to guide the search at higher resolutions. While this is a good method to reduce computation time, it assumes that the results generated at low resolution are more reliable, if less precise, than those generated at high resolution; this is a questionable assumption especially in the presence of occlusions. For example in the case of tree images, it could lead to a computed distance for the area between some neighbouring trunks that would be approximately the same as that of the trunks themselves, which would be wrong. Furthermore, in the absence of repetitive patterns, the output of our algorithm is not appreciably degraded by using the large disparity ranges that our approach requires.

2.2.5 Trinocular algorithm

As suggested by several researchers [3, 18, 28], more than two images can and should be used whenever practical. When dealing with three images or more, we take the first one to be our reference frame, compute separately disparity maps for all pairs formed by this image and one of the others and merge these maps by selecting for example the match that has produced the best correlation score. In this way we can reduce the size of occluded areas and generate a denser depth map.

In particular, we use at INRIA a trinocular stereo system in which the cameras are situated on the vertices of a right-angle triangle. This particular disposition is needed to cause small deformations of the images in the rectification process. Indeed, to simplify the research of the best candidate along the epipolar line, the images are first reprojected onto the same image plane (parallel to the plane defined by the three optical centers) so that all epipolar lines become parallel. It is easy to show that the axes of the rectified images referentials can be chosen in such a way as to make the epipolar lines horizontal on the same line or vertical on the same column, without introducing severe deformations. The image from the camera located at the square corner of the right-angle triangle is taken to be the reference one. This reference image is then correlated horizontally with the second image and vertically with the third. The two disparity maps obtained are then merged together to produce a dense fused depth map.

Correlating both horizontally and vertically has the advantage to introduce redundancies which allow us to obtain more reliable results. Indeed we can verify that two matches are compatible by testing if the matched points in the second and third images lie on the same diagonal epipolar line. If this condition is true, then the two matches measure the same depth and we can be almost absolutely sure they are correct because the search of the two corresponding points is made on completely different areas horizontally and vertically. If not, at least one match is false, and we have to select the best one. We have tested this trinocular algorithm on outdoor rocks scenes and have noticed that such a simple philosophy can build error-free depth maps.

2.2.6 Algorithmic implementation

In this paragraph we restrict our attention to a binocular correlation aligned horizontal epipolar lines, so that we have no verical disparities. Let us consider criterion C_2 for the moment. A first simplification is obtained by noticing that the first term in the denominator is constant when x and y do not vary. We can therefore consider the simpler criterion

$$C'_2(x,y,d) = \frac{\sum_{i,j} I_1(x+i,y+j) \times I_2(x+d+i,y+j)}{\sqrt{\sum_{i,j} I_2(x+d+i,y+j)^2}}$$

For simplification purposes, we split the computation into different parts:

$$\begin{aligned}
O(x,y) &= \max_d\{N(x,y,d) \times R(x+d,y)\} \\
N(x,y,d) &= \sum_{i,j} I_1(x+i,y+j) \times I_2(x+d+i,y+j) \\
R(x,y) &= 1\sqrt{M(x,y)} \\
M(x,y) &= \sum_{i,j} I_2(x+i,y+j) \times I_2(x+i,y+j)
\end{aligned}$$

We can see that the numerator uses $(2n+1)(2m+1)$ redundant multiplications. Using recursions over the indices i and j, we can avoid redoing the same computation. We compute the numerator N as follows:

$$\begin{aligned}
P(x,y,d) &= I_1(x,y) \times I_2(x+d,y) \\
Q(x,0,d) &= \sum_j P(x,j,d) \\
Q(x,y+1,d) &= Q(x,y,d) + P(x,y+2m+1,d) - P(x,y,d) \\
N(0,y,d) &= \sum_i Q(0,y,d) \\
N(x+1,y,d) &= N(x,y,d) + Q(x+2n+1,y,d) - Q(x,y,d)
\end{aligned}$$

Similarly, for the denominator M:

$$\begin{aligned}
P_2(x,y) &= I_2(x,y) \times I_2(x,y) \\
Q_2(x,0) &= \sum_j P_2(x,j) \\
Q_2(x,y+1) &= Q_2(x,y) + P_2(x,y+2m+1) - P_2(x,y) \\
M(0,y) &= \sum_i Q_2(0,y) \\
M(x+1,y) &= M(x,y) + Q_2(x+2n+1,y) - Q_2(x,y)
\end{aligned}$$

The equations for R and O are as above. Those simplifications are represented graphically in figure 1. The computation of the correlation criterion for a given value of x and y is performed by first calculating and storing the first square-root term of the denominator. Then the cross-term of the numerator and the second square-root term of the denominator are computed for each disparity value and divided by the previous square root term to obtain the criterion values.

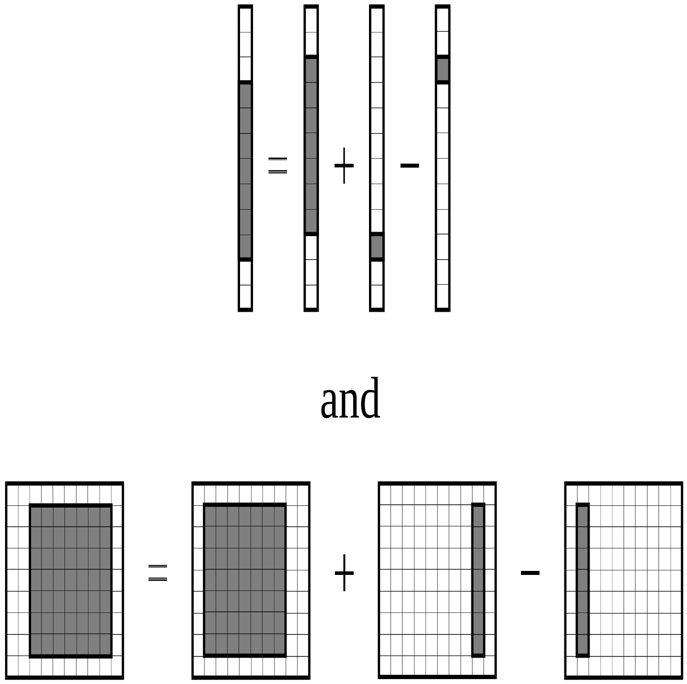

Figure 1: Simplifications to the computation of criterion C_2.

The use of these relationships allows us to avoid any redundancies in the criterion computation and makes the processing time independent of the window size. We can also take large windows to produce more reliable results without being penalized in time. There is no problem of roundness because the values I_1 and I_2 are integer. Similar ideas can be used for computing criterion C_1.

The criteria C_3 and C_4 cannot be computed exactly in the same way because the mean values $\overline{I_k}, k = 1, 2$ which are substracted are the same for all points of the correlation windows. If we first substract at every pixel the mean value (computed on a rectangular neighbourhood of the same size as the correlation window and centered on points) and then perform the correlation using the non-normalized criteria C_1 or C_2, it amounts to using the C_5 or C_6 criteria defined as below:

$$C_5(x,y,d) = \frac{\sum_{i,j}[(I_1(x+i,y+j) - \overline{I_1(x+i,y+j)}) - (I_2(x+d+i,y+j) - \overline{I_2(x+d+i,y+j)})]^2}{\sqrt{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x+i,y+j)}]^2} \times \sqrt{\sum_{i,j}[I_2(x+d+i,y+j) - \overline{I_2(x+d+i,y+j)}]^2}}$$

$$C_6(x,y,d) = \frac{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x+i,y+j)}] \times [I_2(x+d+i,y+j) - \overline{I_2(x+d+i,y+j)}]}{\sqrt{\sum_{i,j}[I_1(x+i,y+j) - \overline{I_1(x+i,y+j)}]^2} \times \sqrt{\sum_{i,j}[I_2(x+d+i,y+j) - \overline{I_2(x+d+i,y+j)}]^2}}$$

C_5 and C_6 produces almost the same results as C_3 and C_4, respectively; there is sometimes a little difference between scores curves but most of the time the criteria vary in the same way. The use of C_5 or C_6 allows us to have a robust normalized cross-correlation in a minimum computation time.

2.3 3D Reconstruction

By intersecting the optical rays of two matched pixels we reconstruct the corresponding 3-D point. This is done by inverting a 3×3 matrix. Prior knowledge of the epipolar position uncertainty and disparity precision is used to compute a 3-D covariance matrix for every reconstructed point.

When weak calibration is used, 3-D reconstruction is performed in a projective frame. As of today, the reconstruction results can only be used for applications in a metric frame of reference which, in the case of the Mars rover, must be obtained without the help of a calibration grid. This can be done if the robot features a calibrated laser vision system providing metric information. Switching from the projective frame to a metric one can be done after computing the projective transformation betweeen the two frames which can be achieved from five correspondences between laser points and matched image pixels. Figure 2 shows an original stereo pair of a rock scene used to test vision algorithms developed for the VAP project in Toulouse. Two corresponding epipolar lines are also shown. Figure 3 shows the rectified stereo pair after estimation of the epipolar geometry. Figure 4 shows on the left the depths (distances to the cameras) and on the right the elevations (distance from the ground). Both are represented in shades of gray, dark meaning close, white meaning far, and black meaning that the algorithm has returned "don't know". Finally, figure 5 shows two perspective representations of the 3-D reconstruction of the scene of figure 2: the left part shows the reconstruction obtained by calibrating the stereo rig traditionnally with a known three-dimensional calibration pattern, the right part shows the same reconstruction obtained by "weakly" calibrating the stereo rig, using only the images of a few points in the scene. The two reconstructions have then been put in the same metric frame for display purposes by using the 3-D metric coordinates of five points in the two views. The two reconstructions are clearly very similar, thus validating the weak calibration approach.

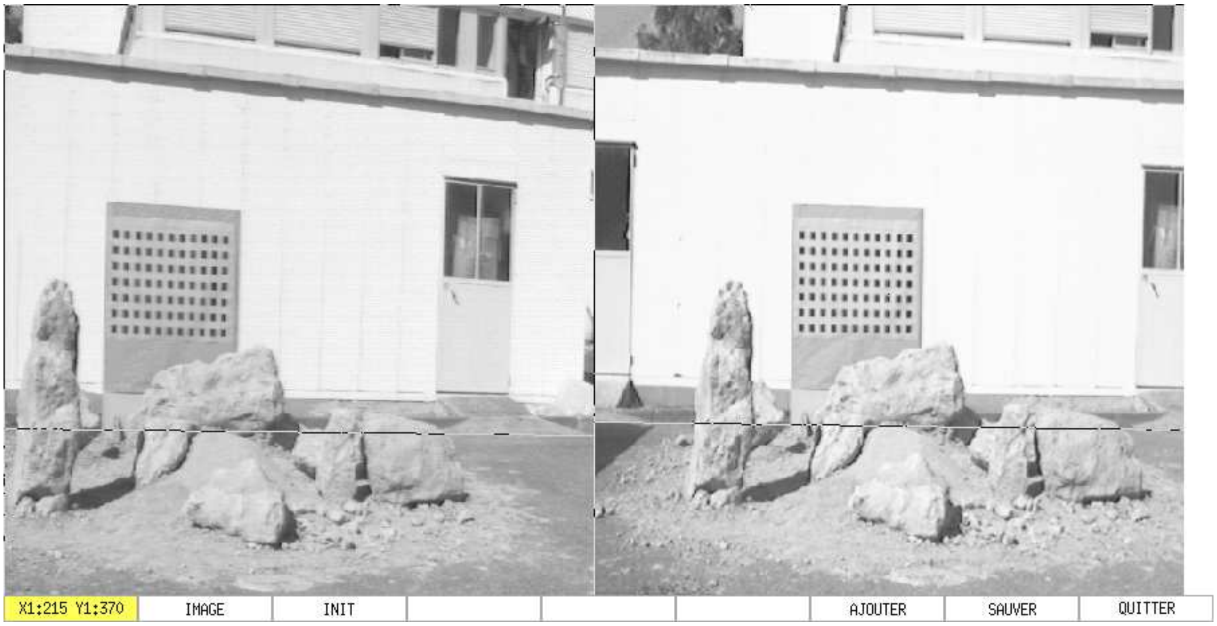

Figure 2: Original stereo pair

3 DSP board implementation

We now describe the first of the two parallel implementations of the stereo algorithm described in section 2. The idea of this implementation is to use in parallel several powerful processors. Each of this processors can be programmed in a fairly high-language such as C and the existence of cross-compilers available on most workstations has made the parallelization of the workstation version of the code relatively easy.

3.1 Architecture

DSPs (Digital Signal Processors) are well adapted for many computer vision algorithms because of their hardware architecture. This is why we developed the MD96 board, a Multi-

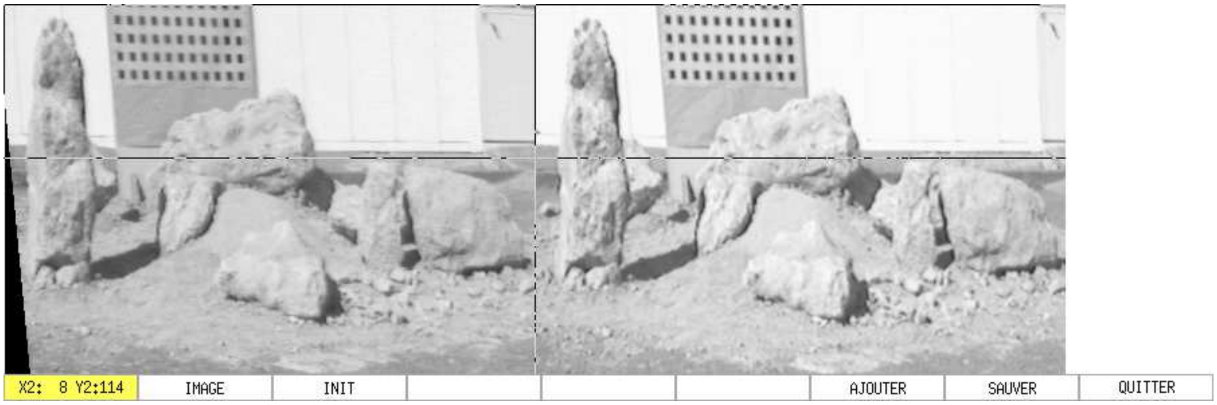

Figure 3: Rectified stereo pair

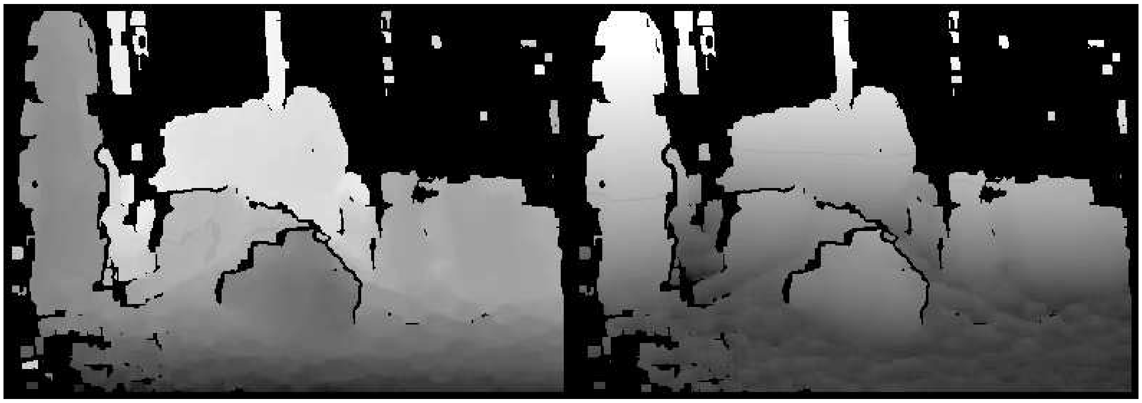

Figure 4: Depth and elevation

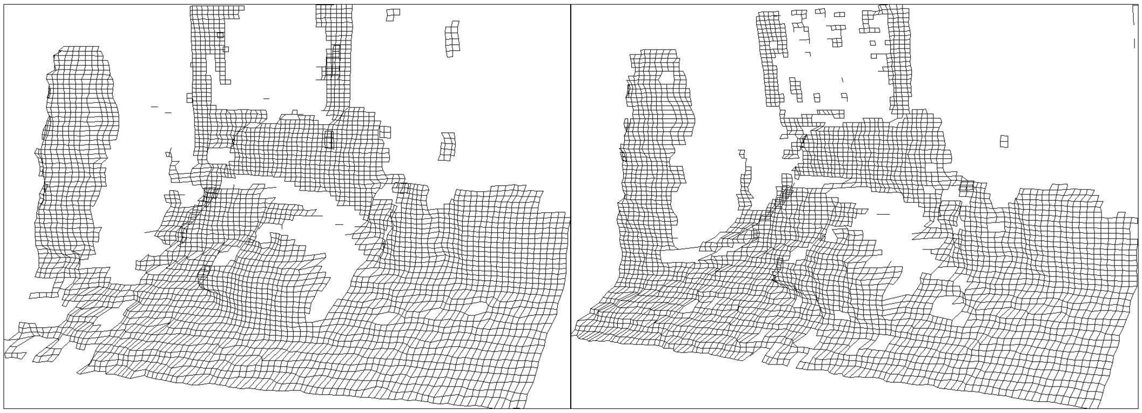

Figure 5: 3D models (grid calibration and self-calibration)

DSP board, with four Motorola 96002 Digital Signal Processors and interfaced with the VMEbus. This design was done in 1990 within an ESPRIT European project as a collaboration between Hervé Mathieu from INRIA and Eric Théron from Matra MSII. Detailed technical information about this board can be found in [22]. The MD96 board has a peak processing power of 240 MFLOPS (Mega FLoating-point Operation per Second) and has the following features:

- Four Processing Elements working in parallel. Each one of them is a 96002 Digital Signal Processor running at 40 MHz with two 256Kx32 (256 Kilo-words of 32 bits) memory modules packaged in JEDEC shape.
- An on-board shared bus between the four Processing Elements, the VMEbus and a 256Kx32 memory (Communication Memory). This shared bus allows Processing Elements to use fast communication channels between them and to share data through a communication memory.
- Each DSP can be master or slave on the VMEbus allowing the board to work without any master board. (except for booting). The VMEbus interfaced module is fully compliant with the VMEbus specification (Revision C.1).
- The MD96 is made of standard CMOS/TTL components, and is implemented on an extended Euro-Card (220 mm x 233 mm).

3.1.1 DSP96002 Overview

The DSP96002 is a dual-port IEEE floating-point programmable CMOS processor. The device is available with a 33 MHz (resp. 40 MHz) clock, for 16,5 (resp. 20) Million Instructions per second (MIPS) and 49.5 (resp. 60) MFLOPS.

The main features of the DSP96002 are :

- A 32x32 bits floating-point and integer multiplier unit.
- A 32/64 bits floating-point and integer ALU.
- A full 32 bits Address Generation Unit.
- 2Kx32 bits internal memory (in three banks).
- Two A32/D32 channels DMA controller.
- Full compatibility with IEEE 32/64 floating point and integer data format. This means that format conversions are not required. In fact, the internal device architecture has been designed for efficient C implementation.

The architecture of the MD96 board is shown in figure 6.

3.1.2 Software

The programming of the board can be done in a fairly high-level language such as C and compiled with the C Compiler Intertools delivered by Intermetrics Inc. A C library allows to drive the MD96 board via a host computer, and some C libraries have been written for the MD96, allowing communication between DSPs or between a DSP and other boards connected on the same VMEbus.

3.1.3 Conclusion

As a conclusion, the MD96 board accepts up to 9 mega-bytes of fast access memory, and all the programming can be done in C. Each DSP works on its memory with zero Wait State and no bus arbitration, and the shared bus is used as a communication channel by each DSP with one Wait State, or by the VMEbus with a minimum of arbitration.

3.2 Implementation

We now describe how the binocular correlation described in section 2 has been implemented on the MD96 board.

Our current configuration uses two MD96 boards which are embedded in a VMEbus box containing a Motorola MVME167 with a 68040 processor running the real-time operating system VxWorks from Wind River System Inc.. This box also includes a video frame grabber allowing to scan synchronously two 512×512 interlaced images.

Our implementation has four parts :

- Rectification in order to obtain horizontal epipolar lines.
- Binocular correlation programmed in floating-point to achieve sub-pixel accuracy. We have implemented criterion C_5.