

Optimized Stereoscopic 3-D Object Reconstruction

E. Dumont, C. Constantin, A. Esse, A. Gréaux, and F. Techer

Abstract—Based on vision systems developed by living animals for their survival, this paper describes and analyzes the results of an active stereovision system comprising two identical sensor cameras mounted on a specific frame allowing different pan configurations and baseline variations. A 3D representation is then reconstructed using different algorithms run on a computer receiving images from the two cameras. SURF and SAD algorithms are evaluated. To improve their output quality, it is proposed here to actively vary parameters of stereovision system, ie distance between the two cameras and vision angles. As a consequence, these parameters can be optimized in proposed new prototype to obtain the best setup for 3D reconstruction.

Index Terms—3D reconstruction, parameters optimization, raspberry-pi, SAD, stereovision.

I. INTRODUCTION

The goal of present paper is 3D object reconstruction. Already living systems have been solving this problem, as animals tend to adopt physical attributes allowing them to best survive. For instance most of hunted (prey) animals have lateral eyes placement (e.g. rabbit, deer, chameleons) in order to increase danger perception. On the other hand, hunters have their two eyes located in front of their face (e.g. lion, cat, and bear) to allow them to track more accurately their prey. In technical domain, stereo-vision systems are more and more used in computer vision for transport and danger analysis [1], [2] (environment mapping where risks factors are extracted to warn the driver), humanoid robotic [3], [4] (gaze selection can be enhanced by using the variance between two eyes) and autonomous mobile platforms [5], [6] for instance. To increase their performance as required for higher quality service, a first possible solution [7] is to improve algorithm characteristics by choosing system architecture depending on signal processing, so that bigger algorithm can be implemented and so more accurate results can hopefully be obtained. But improvement of technology and of sensors performance implies correlatively to develop compatible processes. In parallel another additive way is also to increase data collection by grouping several sensors for a specific task and developing data fusion [8]. All these possibilities are paid by a much larger involvement in handling and treating collected information with delay problem in real time systems. Alternatively, it may be interesting to wander more elementarily whether these static systems could collect more efficient input by making them able, like living systems, to perform active data harvesting in adapting their configuration to environment. This is the aim of present study to propose an

active stereo electro mechanical platform mimicking natural eyes configuration, from which advantage and inconvenience of each biological configuration can be evaluated and optimization can be performed. Standards algorithms SAD [9] and SURF [10], [11] will be taken for application on data collected through basic sensors set up.



Fig. 1. Object and background used for 3D reconstruction algorithm.

To make analysis easier, a simple convex unicolor object with marked edges is used. As it does not constitute an environment, the background will be removed in order to detect the object. So a unicolor background is taken here to facilitate the process of background removing, see Figure 1. The purpose in taking an object with same color as the background is to see how the algorithm deals with it according to object marked edges.

Sensors accuracy is a main parameter when taking picture; actually a relation has been found between accuracy and picture details (background brightness). If original picture is more accurate, there will be more details to analyze but finally more filters will be applied to find the object and to reduce the noise.

II. STATE OF THE ART

Accurate 3D perception from video sequences is a central subject in computer vision and robotics, since it constitutes the basis of subsequent scene analysis. Classical operation performed in image processing is the correlation in which a set of image points of interest are compared in different positions to other images, and "best" position is kept according to searched data. The measure of similarity between two pixels (or blocks of pixels called "kernel") is called "correlation". There are many types of correlation, each with advantages and disadvantages (i.e. the difference in position between the matching points also called "disparity" [12]). But no satisfying algorithm combines the following four characteristics: 1) Timeliness, 2) Positional accuracy of discontinuities relief (main problem of correlation window), 3) Management of large disparities causing significant distortions between images, 3) Robustness gain/offset intensity between images, 4) Robustness to repetitive patterns.

The geometric distortion around the "epipoles" leads to problems for algorithms requiring grinding stage (even in polar coordinates). Researches in this area can roughly be

Manuscript received April 15, 2014; revised June 20, 2014. This work was supported in part by ECE Paris School of Engineering.

E. Dumont, C. Constantin, A. Gréaux, and F. Techer are with ECE Paris School of Engineering, France (e-mail: dumont@ece.fr, techer@ece.fr).

A. Esse is with ECE Paris School of Engineering and Pierre and Marie Curie University, Paris 6, France (e-mail: esse@ece.fr).

divided in two classes: improving accuracy regardless of computing time, and scene reconstruction in real-time. In [13] it is tried to fill the gap, by providing instructions on how to implement correlation based disparity calculation, with high speed and reasonable quality. This implementation can be used for different applications like 3D reconstruction. A variant of the sum of absolute (SAD) is chosen by computing the sum of absolute RGB differences and by trying to improve the results. Of other algorithms, the main types are:

- Algorithms used in segments of the picture (edges). They are fast and discontinuities relief has a precise location. But their results are not dense, only edges are detected [12].
- Algorithms working on the regions of the image (Blocks of pixel). The regions of an image are very rich in information (outline, texture, geometry, connectivity). But they are not really used in 3D reconstruction because of their complexity to obtain a good disparity. Moreover the design depends on the setup [14].
- Global Optimization Algorithms (layered stereo, graph cut, SAD ...). They are accurate for discontinuities relief. But their computation time is prohibitive for real images or high disparity and both robustness to noise and gain/offset intensity is weak [12].

The goal in this article is to observe the result of standard algorithms on sensors improvement. To do so, the used algorithm is SAD described below, where in the same calculation, the background can be removed and disparity map is computed.

III. ALGORITHMS

Three main steps are followed for analysis of collected data and transforming them in understandable and interpretable information on surrounding environment.

A. Transform from RGB to Grayscale Image

When shot, the pictures are immediately changed from RGB to grayscale using a simple mean value:

$$I_{\text{grayscale}} = \frac{I_{\text{Red}} + I_{\text{Green}} + I_{\text{Blue}}}{3} \quad (1)$$

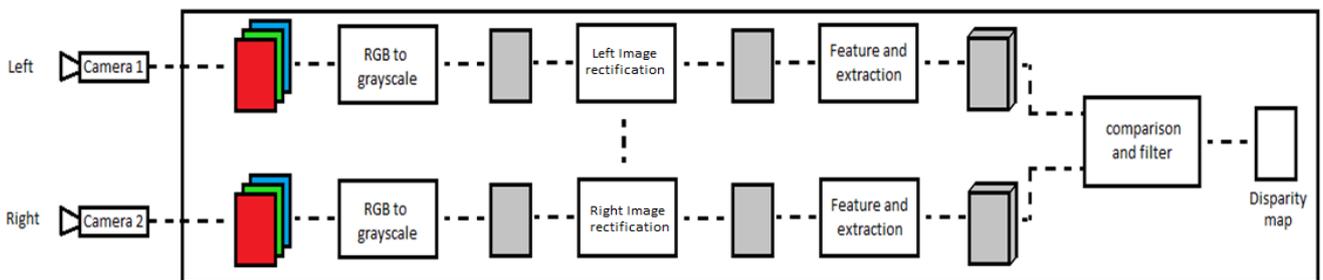


Fig. 2. Stereoscopic recognition global overview.

The number of pixels accepted with same minimum SAD value filter is used for background removing. Generally, main used algorithms consist in taking uncolored background and trying to adapt a threshold in order to remove it. The main disadvantage of this technique is coming from color change (due to brightness for instance), and threshold must be adapted every time (if setup environment is changed, the threshold has to be readapted). Thanks to this filter, the background is removed during the same process of disparity

map calculation.

Then, in order to correlate points from left picture to the same points on right image, two algorithms are usually implemented: Census [15] and the Sum of Absolute Differences (SAD) [9].

B. Sum of Absolute Differences (SAD)

SAD method calculates the correlation between two parts of a picture. For each pixel, a surrounding block of pixels is compared to the other block of pixels, with following metrics:

$$SAD = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |Left_{i,j} - Right_{i,j}| \quad (2)$$

where $Left_{i,j}$ corresponds to a pixel in the block of left picture and $Right_{i,j}$ to a pixel in the block of right picture, M is the size of the block length and N the block size in width. Smaller value of SAD result provides higher correspondence between each part. SAD process is used with satisfying results when invariant rotation points are searched.

C. Applied Filters

Before and after SAD calculation, the following filters are applied, see Table I:

Filter	Description
Difference between two analyzed pixels	If the two analyzed pixels (one from left image, the other from right image) have a difference between their value higher than filter value, the SAD is not computed and the process continues for next pixel
Size of analyzed kernel	Size of the matrix used around the pixel to compute the SAD
Number of pixels accepted with same minimum SAD value	When SAD values are computed, if the number of pixels with same minimum SAD found is higher than filter value, it is considered as unknown (disparity cannot be found for this pixel).
Closest pixel	Otherwise, if the number of pixels with same minimum SAD value is lower than previous filter value, the closest (Euclidian distance) pixel is kept and disparity is computed from this pixel

IV. RESULTS

Results tend to confirm that the algorithm is currently not efficient. With minimum possible threshold the background is well removed and the object is only slightly locally eroded due to bound size. As expectable, pyramid white part is also removed (a simple solution could be to reconstruct it at

calculation end with a simple gradient between two points). Object disparity is not correct. When checked by hand, some parts are correctly found but the main part of correlated object is completely out. An error rate cannot be precisely determined due to implied calculation and to the fact that standard data can just be computed manually. When thresholds are increased in order to improve the result, observation can be done that obviously, with increase of kernel size (or of difference threshold between two pixels), calculation time is also increased and the algorithm is slower. Table II below summarizes results for different elements and with different schemes.

TABLE II: RESULTS OBTAINED AFTER MODIFICATION OF IMPLEMENTED FILTERS

Threshold modified	Number of correlated point increased	Worse results on disparity map	Background removing less efficient	Not usable results
1-Increased difference between the two analyzed pixels	Yes	No change	No	No
2- Number of accepted pixels with same SAD value increased	Yes	Yes	Yes	No
3- Increased size of analyzed kernel	No, decreased	Yes	No	No
1 + 2	Yes	Yes	Yes, object not well detected	Yes
1 + 3	Yes	Yes	No	No
2 + 3	No change	Yes	Yes, object not detected and truncated	Yes
1 + 2 + 3	Yes	Yes	No background removing	Yes

In summary, available data treated with SAD algorithm do not allow fair pyramid reconstruction. As expected the background is removed easily but the same algorithm removes also points of the same color on the object and provides wrong information concerning the correlated point

of analyzed pixel. So an all-in-one algorithm is not sufficient enough for performing 3D reconstruction of an object from camera collected data set, and another algorithm should be used.

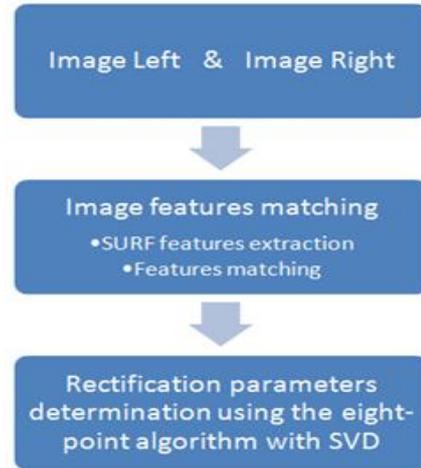


Fig. 3. Stereoscopic rectification algorithm.

V. FURTHER IMPLEMENTATIONS

Recognition system, see Fig. 2, is not calibrated for every configuration. Besides mechanical construction is not perfect, and focal axes are not coplanar. These problems need to be corrected and an extra-step is added to basic algorithm.

The rectification process, see Fig. 3, analyses automatically both stereo pictures, extracts feature couple using SURF method [10], and corrects images thanks to utilized eight-point algorithm technique [16]. However, this algorithm needs more characteristic points from the two left and right images for finding a correlation. It is applied on an environment rather than on an object.

Fig. 4 shows the result obtained with SURF algorithm after rectification process. In red is represented the object considered in the foreground, in blue, the background. As observed, image output (colored right) is very noisy (though it can be cleaned with basic de-noising filter but with the risk of information loss). Also, some domains are still not well determined and disparity map can be improved. This technique gives better 3D estimation than SAD method. Improving it (noise removing, object detection...) will increase calculations number, computation time, and resources utilization.

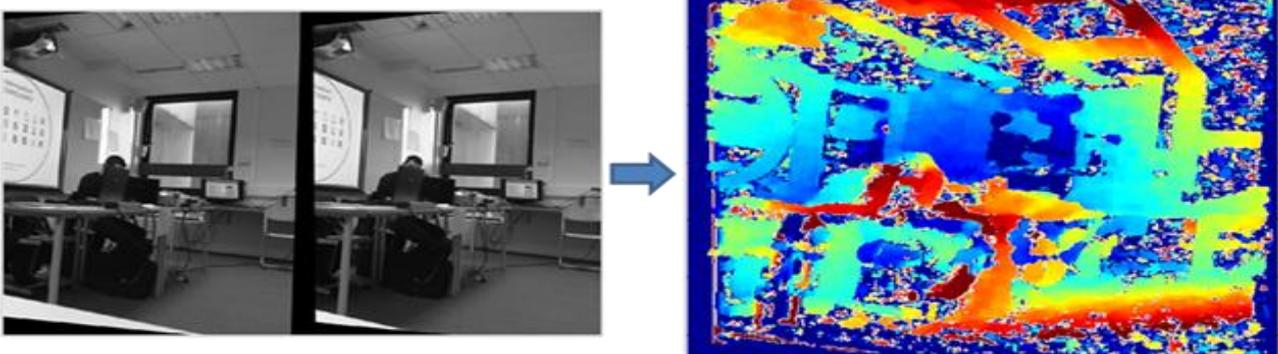


Fig. 4. 3D Reconstruction with rectification.

At this point, two steps have been explored. In first one, standard and easy-to-use algorithm has been implemented

and tested using a basic stereo vision system. The main results are mainly not usable due to setup parameters (vertical

offset of the two cameras). In a second step, SURF algorithm has been tested. Similar results have been obtained but after using rectification, disparity map is getting close to reality. In next step, a new setup almost automatically optimizing 3D reconstruction is introduced.

VI. SETUP DESCRIPTION

Up to now sensor system parameters are fixed, and collected data in this static situation are treated to extract meaningful information for further analysis. To get rid of this constraint, a prototype has been built up in order to vary precisely the selected parameters. With this setup, it is possible to change the distance between cameras and their inclination as they are fixed on a track, see Fig. 5. Each module is composed of an actuator, several holding parts built up with a 3D printer and an electronic system based on a Raspberry-Pi. The following three elements are in order.

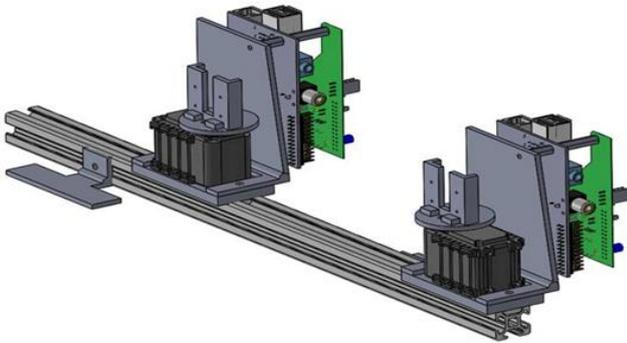


Fig. 5. 3D model of setup used for parameter analysis.

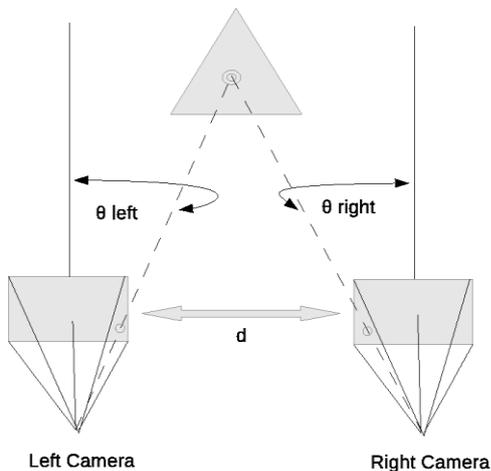


Fig. 6. System parameters.

Distance between Camera setup: The track is used to allow both cameras to move on the same axis. To make the prototype easier to manipulate, only one camera is fixed to the track. The other one can be displaced along it by pushing/pulling to set the distance. The minimal distance for the cameras is 5 centimeters.

Camera Inclination: The inclination for each camera is actuated and can move from 0 degree (parallel vision) to 90 degrees (cameras are face to face). Precision on this parameter is 0.58 degree and step is 1 degree (due to servomotors choice).

System Automation: To use this setup in the easiest way, a client-server system has been built up, based on a UDP unicast and SFTP protocols. The idea is to make the system as much automatic as possible to speed up test benches. This network allows centralizing all the commands like rotating the system, taking pictures and getting them directly to a specified folder. The PC works as a client which sends a request to one of the Raspberry-Pi.

The system allows cameras to gaze right (Dextro-version) or left (Levo-version) which can be used to optimize scene reconstruction. Consider the optical center merged with the rotation axis. In order to optimize this technique, pictures are first taken with both angles set to 0 ($\theta_{left} = \theta_{right} = 0$), see Fig. 6. Then angle α is computed to move to gaze into target object knowing D_y , the distance between camera center and the pixel to gaze to, with:

$$\alpha = \arctan(D_y/f) \quad (3)$$

D_y is the distance in mm determined from the number S of mm/pixel and D_{yPixel} the distance in pixels with $D_y = D_{yPixel} \times S$. Finally, the setup with smallest disparity is kept and reconstructed.

VII. CONCLUSION AND FUTURE WORKS

To deal with inevitable “curse to complexity” when improving 3D image reconstruction by always more sophisticated algorithms and even using data fusion, an alternative and simpler way based on natural living systems choice for their survival is explored here. It is mainly organized on observation that these systems are using their “hardware” sensors parameters as optimization variables they adjust in real time for higher efficiency. Along this line, a prototype is proposed which allows parameter modification such as distance between the two cameras, distance to the object and pan angle. System improvement could aim at total automation of remaining movements (currently done by hand) such as tilt and baseline variations. So the prototype could trend to adopt a traditional humanoid ocular vision. Right now however, dynamical environment cannot be studied because actual system cannot take two pictures at the same time due to server/client protocol. Hopefully, chosen cameras can adapt their precision (number of pixels in one image), depending on the result and the computing time, the size was increased or decreased in order to reach a compromise between accuracy and calculation time.

Concerning algorithm side, a real-time interaction between image processing and servomotor can be implemented and parameters can be analyzed thanks to setup and algorithm. A further improvement is to fully automate the system for making it a real-time parameter optimizer, ie a system which can automatically take best geometrical configuration based on environment for 3D reconstruction.

ACKNOWLEDGMENT

The authors are very much indebted to ECE for material support, Dr. A. Houelle for guidance and Pr. M. Cotsaftis for help in preparation of the manuscript.

REFERENCES

- [1] K. Mühlmann, D. Maier *et al.*, “Calculating dense disparity maps from color stereo images, an efficient implementation,” *Intern. J. Computer Vision*, vol. 47, no. 1-3, pp. 79-88, 2002.
- [2] C. D. Pantilie, S. Bota, I. Haller, and S. Nedeveschi, “Real-time obstacle detection using dense stereo vision and dense optical flow,” in *Proc. 2010 IEEE Intern. Conf. on Intelligent Computer Communication and Processing (ICCP)*, 2010, pp. 191–196.
- [3] K. Welke, D. Schiebener, T. Asfour, and R. Dillmann, “Gaze selection during manipulation tasks,” in *Proc. 2013 IEEE Intern. Conf. on Robotics and Automation (ICRA)*, May 6-10 2013, pp. 652-659.
- [4] Y. Kuniyoshi, N. Kita, T. Suehiro, and S. Rougeaux, “Active stereo vision system with foveated wide angle lenses, in recent developments in computer vision,” S. Z. Li, D. P. Mital, E. K. Teoh, and H. Wang, Éd.s. Springer Berlin Heidelberg, 1996, pp. 191-200.
- [5] A. Achtelek *et al.*, “Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments,” in *Proc. SPIE 7332, Unmanned Systems Technology XI*, 733219, April 30, 2009.
- [6] C. E. Bichot, “Analyse Vidéo-Stéréoscopique Embarquée sur un Mini-drone à Ailes Fixes Permettant l'Évitement d'Obstacles,” Report of Internship, Master Computer, LIRIS, Lyon, 2011.
- [7] P. Courbin, A. Pedron *et al.*, “Parallélisation d'Opérateurs de TI: Multi-coeurs, Cell ou GPU?” *Traitement du Signal*, vol. 27, no. 2, pp. 161- 187, 2010.
- [8] M. E. Liggins, D. L. Hall, and J. Llinas, *Multisensor Data Fusion*, 2nd Ed., *Theory and Practice (Multisensor Data Fusion)*, CRC Press, Boca Raton, 2008.
- [9] C. Watman, D. Austin *et al.*, “Fast sum of absolute differences visual landmark detection,” in *Proc. ICRA '04 (2004 IEEE Intern. Conf. on Robotics and Automation)*, 2004, vol. 5, pp. 4827-4832.
- [10] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, “Speeded-up robust features (SURF),” *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346-359, 2008.
- [11] Distinctive Image Features from Scale-Invariant Keypoints. [Online]. Available: <http://link.springer.com/article/10.1023/B:VISI.0000029664.99615.94>
- [12] E. Trucco and A. Verri, *Introductory Techniques for 3-D Computer Vision*, Prentice-Hall, 1998.
- [13] S. Lefevbre, “La stereovision passive,” Ph.D Thesis, Univ. Sciences et Technologies, Lille, 2008.
- [14] M. Grigorescu, Tiberiu T. Cocias *et al.*, “Stereo vision-based 3D camera pose and object structure estimation,” in *Proc. VISAPP2012*, 2012, pp. 355-358.
- [15] K. B. Young., H. Jo. Jung, and M. L. Kyoung, “Fast census transform-based stereo algorithm using SSE2,” in *Proc. 12th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, Feb. 2-3, 2006, Tokushima, Japan.
- [16] R. I. Hartley, “Defense of the eight-point algorithm,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580-593, 1997.

E. Dumont has always been living in Paris. He received his bachelor of science degree from Aalborg University. He is now studying embedded systems in ECE Paris School of Engineering. He is highly interested in all topics related to robotics.

C. Constantin started his studies in clermont-ferrand. Then he moved to Paris for studying information systems in ECE Paris School of Engineering specialized in functional design. He is now working in Avanade.

A. Esse was born and raised in the Paris area. He is currently studying embedded systems in ECE Paris School of Engineering and following a double degree program, he is also studying advanced systems and robotics in Pierre and Marie Curie University – Paris 6.

His research interests include computer vision and robotics, he is currently working in collaboration with Parrot and the French aerospace lab, ONERA on micro-drones vision.

A. Gréaux was living in St Barts until he started studying energy and environment in Paris as a ECE Paris School of Engineering.

F. Techer grew up in Réunion island and he is currently studying embedded systems in ECE Paris School of Engineering.

His research interests include electronic and sciences. He is now working on the electronic system conception of drones developed by Parrot.