Real-Time Geo-Referenced Video Mosaicking with the MATISSE System

A.G. Vincent, Ifremer
Anne.Gaelle.Vincent@ifremer.fr

N. Pessel, Ifremer
Nathalie.Pessel@ifremer.fr

M. Borgetto, Ifremer
Manon.Borgetto@ifremer.fr

J. Jouffroy, Ifremer
Jerome.Jouffroy@ifremer.fr

J. Opderbecke, Ifremer
Jan.Opderbecke@ifremer.fr

V. Rigaud, Ifremer
Vincent.Rigaud@ifremer.fr

Ifremer – Underwater Robotics, Navigation and Vision Department
Zone Portuaire de Brégaillon BP 330, 83507 La Seyne-sur-Mer Cedex, France

***Abstract*- This paper presents the *MATISSE* system: Mosaicking Advanced Technologies Integrated in a Single Software Environment. This system aims at producing in-line and off-line geo-referenced video mosaics of seabed given a video input and navigation data. It is based upon several techniques of image and signal processing which have been developed at Ifremer these last years in the fields of image mosaicking, camera self-calibration or correction and estimation of navigation data.**

I. INTRODUCTION

In this paper, we present a system dedicated to quantitative exploitation of video images in an underwater environment. This system aims at creating video maps in a geo-referenced environment. The videos come from a vertical camera fixed on an underwater vehicle, such as Ifremer's ROV *Victor 6000*. Geo-referencing is provided by measured or estimated data from a navigation system integrating dead-reckoning and acoustic USBL positioning [1].

In order to map an area of the seafloor, geo-referenced mosaics need to be computed from the video flow and navigation data. This is performed by four fundamental steps which are presented in Section 2. The first one concerns the process of the mosaic itself which requires image processing methods. Then, the camera self-calibration enables to compute the geographical size of the mosaics. A lighting correction is used to compensate for bad illumination conditions and to enhance mosaics. And finally, the image trajectory is corrected by blending with navigation data in order to obtain robust geo-referencing. Non-linear estimation of the vehicle trajectory from acoustic and dead-reckoning data can be used as a pre-processing step.

In section 3, the *MATISSE* (Mosaicking Advanced Technologies Integrated in a Single Software Environment) system including these functionalities is presented. The system is composed of a stand-alone software application which creates the geo-referenced mosaics. A second module is dedicated to the exploitation and management of the mosaics by the end-user in an appropriate geographic information system (GIS).

The last section presents some results obtained during an operational cruise. Future trends for the *MATISSE* system are also brought up at the end of the paper.

II. ALGORITHMS

A. 2D video mosaic processing

The process of video mosaicking consists in estimating the displacement between the successive images of an image stream in order to get a larger image which represents the global scene. Different methods enable to process mosaics and two of them have been developed within the *MATISSE* system. The estimation of the movement between two successive images is performed either by feature tracking methods or by robust optic flow algorithms.

The feature tracking method we have implemented is based on an algorithm proposed by Kanade, Lucas and Tomasi [2] [3]. The first stage of this algorithm aims at selecting good features, that is, features which will be easily tracked. So, the selected features consist of little windows, typically 7 pixel wide, which are well-textured. Given the selected features inside an image, the next step is to track them through an image stream. The displacement between two successive images is supposed to be quite small since the camera moves slowly. So, nearly all the points of an image I are also on the next image J and they are linked by a translation vector d:

$$I(x) = J(x-d) + \eta(x), \quad (1)$$

where η is the noise at a given position.

The method estimates the displacement for each window and thus, produces a list of matched points. Then, the global motion estimation between the two successive images is carried out by a least square method completed by an acceptance criterion. Several iterations enable to remove the false matched points and to improve the motion estimation.

A result of the feature tracking algorithm is presented Fig. 1. The image on top represents a video image of the seabed on the continental margin off Ireland acquired during the *Caracole* cruise in 2001. The partial mosaic at the bottom is built by the feature tracking algorithm, from an image sequence whose first image is presented below.

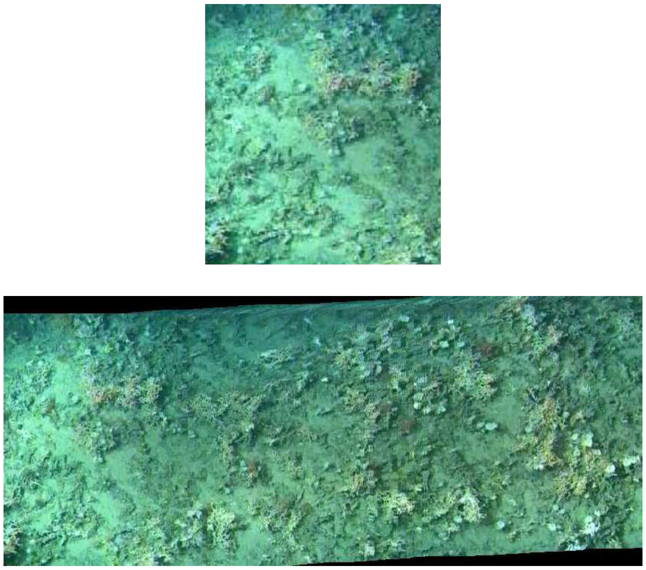


Fig. 1. First image of a video sequence and mosaic obtained with feature tracking.

The second method we have included in the *MATISSE* system is the Robust Multi-Resolution (RMR) method which is based upon robust optic flow [4]. The advantage of this method is that motion is estimated from the whole image. The method is based on a "coarse-to-fine" strategy: the Gaussian pyramids of the two successive images are built. At each level, the parameters of the flow vector are estimated using the flow constraint equation linking spatial and temporal intensity gradients:

$$\nabla I(x,y)\cdot v(x,y)+I_t(x,y)=0\,, \quad (2)$$

where $\nabla I(x,y)$ is the spatial gradient vector of the intensity, $I_t(x,y)$ is the partial temporal derivative of the intensity relative to time and $v(x,y)$ is the vector field. The motion estimation is achieved by a maximum likelihood estimator considering an affine 2D model of the motion field.

B. Camera self-calibration

In order to quantitatively exploit images in the varying optical conditions of the sub sea environment, tools for rapid automated camera calibration without external means are required. We need to identify the intrinsic parameters of the camera, given an image sequence corresponding to a specific vehicle trajectory for a non-structured natural scene. The four intrinsic parameters $(\alpha_u, \alpha_v, v_0, u_0)$ to be estimated correspond respectively to products of the scale factors according to the axis u and v by the focal distance f, and to the coordinates of the intersection of the optical axis with the image plane.

As the images used for calibration are provided by one vertical camera in motion, the point matching is achieved by the KLT (Kanade-Lucas-Tomasi) algorithm which allows to track features in an image sequence. The tracked features through the image sequence enable to study the epipolar geometry between two successive images. The knowledge of several matches in two successive images allows to define algebraically the geometric relation between the two images. This relation is represented by the fundamental matrix F which links the image coordinates q_1 and q_2 of a same point in two successive images:

$$q_2^T F q_1 = 0 \quad (3)$$

The estimation of the fundamental matrix is carried out by using Hartley's normalized 8-point algorithm [5]. This method allows to compute the fundamental matrix from a set of at least 8 matched points $q_1 \leftrightarrow q_2$ between two views of the same scene.

A difficulty arises from the possible presence of bad matches. We propose to integrate a robust estimator that aims at eliminating bad point matches: a RANSAC (RANdom Sample Consensus) algorithm scheme [6] is constructed around the 8-point algorithm computing the fundamental matrix. The selection is based on the accuracy of the equation (3). The fundamental matrix is estimated from this equation and the reprojection error is determined. A point match is considered to be valid if the reprojection error is inferior to a predefined threshold. A study of the RANSAC algorithm for this application is presented in [7].

The intrinsic parameter estimation is realized with the Mendonça and Cipolla algorithm [8] applied to a sequence of five successive images. The authors propose to minimize a cost function, which takes the intrinsic parameters as arguments and the fundamental matrix as parameters. The cost function is :

$$C(K)=\sum_{i=1}^{n}\sum_{j\rangle i}^{n} w_{ij}\,\frac{{}^1\sigma_{ij}-{}^2\sigma_{ij}}{{}^2\sigma_{ij}} \quad (4)$$

where K is a function of the intrinsic parameters, w_{ij} is the degree of confidence of the fundamental matrix F_{ij} estimation and ${}^1\sigma_{ij}$ and ${}^2\sigma_{ij}$ are the non-zero singular values of $K^T F_{ij} K$, with ${}^1\sigma_{ij} > {}^2\sigma_{ij}$.

Several experiments have been carried out with simulated and real data. Some results are presented in [7]. We have obtained identification of the intrinsic parameters up to a precision of 2% in natural sea-floor experiments.

C. Lighting correction for mosaicking enhancement

On the deep sea ROV *Victor 6000*, photos or videos are acquired using artificial lighting that produces non-uniform luminance within the image. In the example below, the top center of the image is very bright while the surrounding area is darker (Fig. 2).

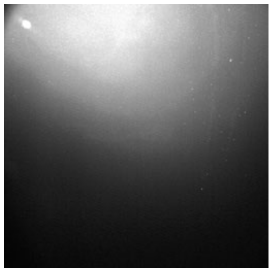


Fig. 2. VICTOR 6000 descent.

This non-uniformity of lighting has an obvious impact on the mosaic creation process. Correcting these adverse lighting effects on each image of the mosaic should improve the quality of the mosaic.

Two methods of correction were investigated in [9]: reflectance-illumination modeling of a digital image [10] and radiometric correction [11].

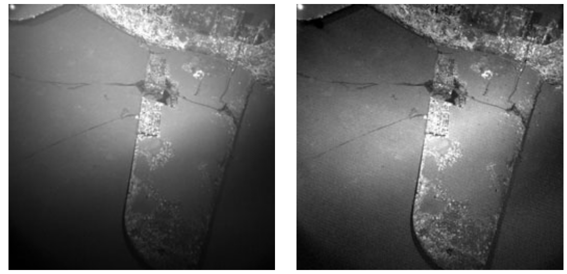
The first method assumes that an image can be separated into two components: reflectance and illumination. Actually, illumination varies continuously and slowly through the image while reflectance can vary abruptly on object edges. The low frequencies in the image cover illumination effects while the high frequencies cover reflectance effects. The idea is thus to bring both components apart by filtering in the frequency domain.

Currently, an operator is still needed to adjust the parameters of the frequency filter. Despite this drawback, the method yields good results.

The second method relies on the CCD camera radiometric correction, mostly used in remote sensing and astronomical imaging. Basically, the image is divided by a so-called Uniform Lit Reference Image (image of a uniformly lit scene without any object – see Fig. 2 for example). Several ways of obtaining this reference image were proposed: by convolving each image by a Gaussian filter, by acquiring this image while diving, by tuning up the focal to infinity or by modeling the lighting by a Gaussian based on an elliptic basis.

Some results of the second method are presented on Fig. 3. These photos represent an airplane wreck acquired during a 1200 meter dive. The first one is the original one while the second one is corrected using the reference image presented Fig. 2.

A bright halo and a darker area are well noticeable on Fig. 3(a). Since the reference image has not be taken under the same conditions, it remains a slight halo in the middle of the corrected image (Fig. 3(b)). But, as we can see, the image is enhanced and more details of the airplane are distinguishable than in the original image, particularly at the bottom of the image.



(a) Original image (b) Resulting image

Fig. 3. Result obtained on an airplane wreck by using the reference image taken while diving (Fig. 2).

Using “natural” reference images (image acquired while diving or by tuning up the focal to infinity), this method is the most promising because only one image has to be acquired before the image sequence acquisition. No further processing is needed to use this reference image, resulting in a simple operational procedure on the ROV.

D. Trajectory

At this stage, the mosaics can be built, enhanced by lighting correction. The next step introduces dead-reckoning navigation data into the mosaic computation, which leads to mosaic geo-referencing. But the trajectory needs to be improved to be more reliable. Indeed, dead-reckoning data are provided by integrating data from sensors such as Doppler velocity log, gyrocompass or inertial systems, which imply a long-term drift. We propose two methods to improve dead-reckoning data. The first one is a global approach which estimates the vehicle trajectory for instance from USBL and dead-reckoning data. The second one computes the image trajectory on the sea floor by blending vehicle navigation and image motion, both of which are linked by the camera model as well as attitude and altitude (echosounder) measurements.

1) Trajectory estimation

Obtaining a good estimation of the position and orientation of an underwater vehicle at any moment on a mission is of great importance as its usefulness often goes beyond mere navigation issues. Indeed, good trajectory estimation can also be helpful for many off-line data analysis such as cartography, mosaic creation, or planning of spots for future missions.

To be in accordance with these different needs, the objective is then to get an optimal trajectory with respect to a criterion computed over a complete segment. This “most suitable trajectory” must reproduce the measured vehicle motion while respecting global position references.

Such an estimation is clearly done through the appropriate combination of several different sensors [12] (Gyrocompass, Gyro-Doppler, USB, LBL), each one having its own features and specificities. Such a task has also to deal with “real world” constraints such as non availability of sensor data, outliers, sensor misalignment.

In order to take into account these different aspects for off-line application purposes, our approach consists in a global processing of a whole trajectory segment. More specifically, a correct trajectory estimation $\hat{X}$ should minimize the following integral:

$$\int_{t_b}^{t_e} \left(\nabla\hat{X} - V_{dop}\right)^2 + k(t)\cdot\left(\hat{X} - X_{ac}\right)^2 dt\,, \tag{5}$$

where ∇ stands for time differentiation, V_{dop} is the vehicle velocity vector given by the Gyro/Doppler system, and X_{ac} the position vector from a hydro-acoustic measurement system. $k(t)$ is a tuning function.

Such an integral means that the estimated trajectory should be a balance between a close match of $\hat{X}$ with the measured position, and another one of the derivative of $\hat{X}$ with the measured velocities. This minimization criterion is then easily computed using PDE-based techniques [13] where the above integral is the solution obtained through the following gradient descent method:

$$\frac{\partial \hat{X}}{\partial s} = \nabla(\nabla\hat{X} - V_{dop}) - k(t)\cdot(\hat{X} - X_{ac}) \tag{6}$$

This kind of approach, which can be vastly and simply augmented through the appropriate use of nonlinear functions to incorporate additional constraints, gives good results on real data obtained during missions of the ROV *Victor 6000*.

Fig. 4 shows a result of combining Gyro-Doppler and noisy USB data. We can notice the outlier in the upper right corner. The PDE filter exhibits both good absolute positioning and incremental precision (typical from a Gyro/Doppler system) at the same time.

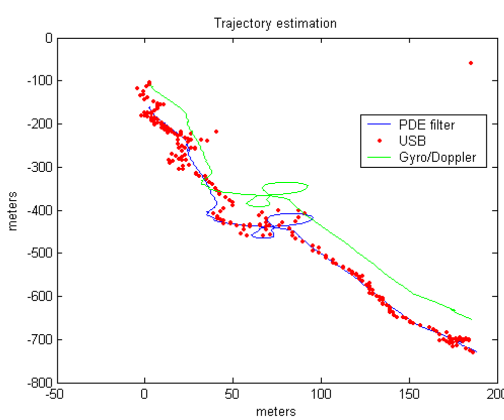


Fig. 4. Trajectory estimation using PDE filtering.

2) Fusion of the dead reckoning and the mosaic

The trajectories followed by a ROV during an exploration of interested zones have generally the same shape as the one presented in Fig. 5:

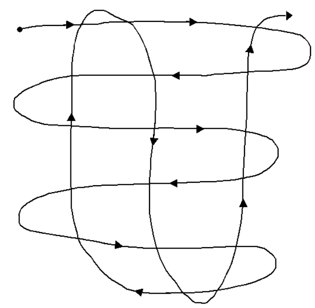


Fig. 5. Typical trajectory of a ROV exploring an interest site.

The approach we propose is made up of two parts and consists in fusing dead-reckoning data (speed and angles) with the image mosaic in order to increase the precision of the mosaic localization. First, a better trajectory is estimated using speed and angles measured and knowing the extreme positions and “rendezvous,” that is to say times where the vehicle passes on the same positions, given by images [14]. Then, the new trajectory is fused with the mosaic already built.

Integrating the angles and speeds from an unknown absolute position from dead reckoning data gives the ROV position. As the angles and speeds are acquired, some errors affect data. These errors are then integrated and the reconstituted trajectory is biased. So the first step of our approach is to re-estimate the trajectory.

The idea is to integrate the angles and speeds in two directions: forward, knowing the first position, and backward, knowing the final position, and to fuse them by least squares estimation - forward and backward at time t_k: the one knowing the first position and the $k-1$ first measured angles and speeds, with the one knowing the last position and the $N-k$ following measures (N being the number of measures).

This method called “auto-estimation of a trajectory,” is explained on simple trajectories, that is, without knowing any retiming positions. To process a complex trajectory (as seen on Fig. 5), additional data (retiming times) are introduced to simplify and refine the treatment. This is done thanks to the graph theory, more particularly thanks to the shortest path algorithm developed by Dijkstra [15].

We proceed on the following way:

1. Find the shortest path between the two known extreme positions,
2. Auto-estimate this sub-trajectory,
3. Consider that the auto-estimated retiming positions are known,

4. Find the shortest path between all the couples of auto-estimated retiming times,
5. Auto-estimate each sub-trajectory,
6. Return to 4.

This method was tested on simulated data and yields good results. It is being to be tested in real conditions.

Concerning the step of fusing the mosaic with the re-estimated trajectory, a method of warping is currently being developed. Our idea is to warp the image center trajectory (initial position and translations and rotations estimated by the KLT are available) into the re-estimated trajectory.

The next stage of our research is a preprocessing phase applied to each image before the mosaic building.

III. GIS INTEGRATION AND SOFTWARE ERGONOMICS

A The MATISSE system

The *MATISSE* system is made up of two parts: on one hand, the *MATISSE* software including the functions of geo-referenced mosaic processing and, on the other hand, a module *MATISSE-GIS* embedded in the Geographic Information System ArcView^{TM}.

The *MATISSE* software and the GIS module can run on the same PC but the performances are increased if they run each one in a dedicated computer.

We present on Fig. 6 the diagram of the *MATISSE* system. This system needs two computers, one for mosaic creation with *MATISSE*, the other for embedding the mosaics into a GIS. Both the computers are linked with a network communication. Moreover, the *MATISSE* software operates with two inputs necessary for geo-referenced mosaic processing: digital or analogue video data and trajectory data.

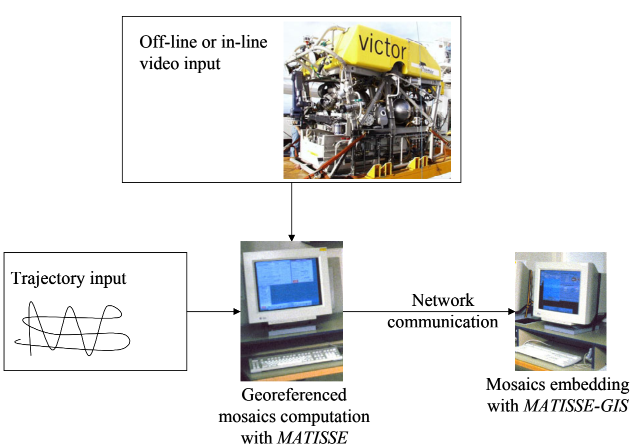


Fig. 6. The *MATISSE* system.

Here below both parts of the *MATISSE* system are detailed.

B. The MATISSE software

The *MATISSE* software has been developed to embed all the algorithms of geo-referenced mosaicking. In order to be more adaptable and to let the mosaicking algorithms evolve, the *MATISSE* software integrates them as DLL. So, any change in an algorithm concerns only the said algorithm. This architecture allows us also to add other algorithms in the future.

This piece of software is available for in-line or off-line working and operates with digital or analogue video and navigation data. We present on Fig. 7 the interface during the mosaic processing from an image sequence. In case of the ROV *Victor 6000* exploring a zone, the camera is fixed on the ROV. All video data acquired by the camera go up in real-time to the surface, enter the *MATISSE* software through a video acquisition board and are processed with navigation data to produce a sequence of geo-referenced mosaics in-line.

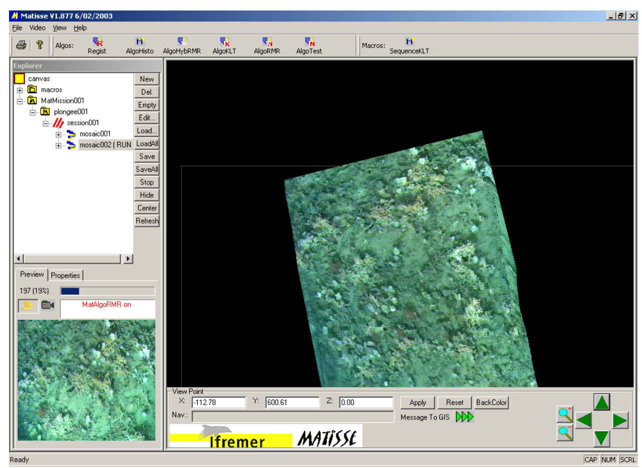


Fig. 7. Real-time mosaicking with the *MATISSE* software.

C. The GIS module

The *MATISSE* system functionalities are increased by a module integrated into a GIS. It enables to integrate data in an appropriate environment, that is in a geographic environment (ArcView^{TM}). All the mosaics processed in the *MATISSE* software can be imported into the GIS in-line thanks to a real-time communication between *MATISSE* and the GIS or off-line. On Fig. 8, we can see a few mosaics imported in-line and their rectangular marks.

This tool provides us with some more functionalities. When mosaics are embedded in the GIS, some of their attributes are recorded in a dedicated database and allow a spatio-temporal study of underwater areas. With this tool, one can search for video mosaics from one area, at one time, and can display these mosaics with other geo-referenced observations.

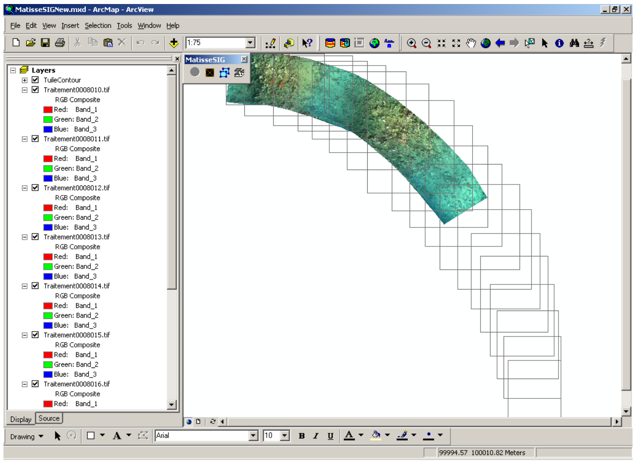


Fig. 8. Embedding of mosaics in the *MATISSE-GIS*.

IV. RESULTS AND FUTURE TRENDS

All the functions of the *MATISSE* system have been validated either with laboratory trials or during operational cruises, such as Caracole in 2001 (RV L'Atalante and *Victor 6000*, high resolution mapping and sampling of carbonate mounds on the continental margin off Ireland) or Marmarascarps in 2002 (RV L'Atalante and *Victor 6000*, high resolution mapping and sampling of geo-tectonic faults in the Marmara Sea). Fig. 9 presents one of the results obtained during Caracole cruise. *Victor 6000* has criss-crossed an area and videos were transmitted in-line to *MATISSE* inside which geo-referenced mosaics have been created. All the mosaics have then been embedded in the GIS where some data resulting from microbathymetry have been superposed.

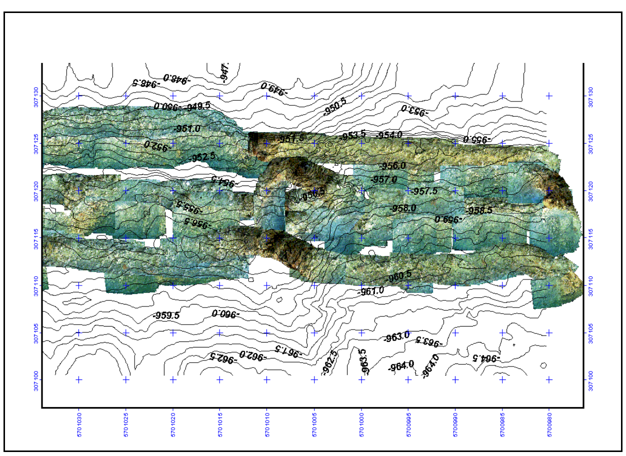


Fig. 9. Example of a map obtained with *MATISSE*.

To conclude, we have presented the *MATISSE* system [16] and detailed several functions which have already been integrated in or those which are part of Ifremer's ongoing researches and will be soon embedded. This system is now operational and is used during technical and operational cruises.

The perspective for *MATISSE* concerns the automatic common representation of optic and acoustic maps in the GIS environment or in Ifremer's *CARAIBES* bathymetry mapping software.

REFERENCES

[1] C. Viellard, V. Rigaud, S. Valestra, "The use of GPS buoys for geo-referencing mosaic charts and environmental data collection," *Oceanology International 2000*, Brighton UK, March 2000.

[2] J. Shi and C. Tomasi, "Good features to track," *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, June 1994.

[3] C. Tomasi and T. Kanade, "Detection and tracking of point features," *Technical report*, CMU-CS-91-132, Carnegie Mellon University, April 1991.

[4] J.M. Odobez and P. Bouthémy, "Robust multi-resolution estimation of parametric motion models applied to complex scenes," *IRISA Intern Report*, n° 788, January 1994.

[5] R. Hartley, "In defence of the 8-point algorithm," *Proceedings of the Fifth International Conference on Computer Vision*, Cambridge, Mass., June 1995.

[6] M.A. Fischler and R.C. Bolles, "Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography," *Communication Association and Computing Machine*, 24(6), pp. 381-395, 1981.

[7] N. Pessel, J. Opderbecke and M.J. Aldon, "An experimental study of a robust self-calibration method for a single camera," *ISPA 2003*, Rome, Italy, September 2003, in press.

[8] P. Mendonça and R. Cipolla, "A simple technique for self-calibration," *Conference on Computer Vision and Pattern Recognition*, 1999.

[9] M. Borgetto, V. Rigaud and J.F. Lots, "Lighting correction for underwater mosaicking enhancement," *Proceedings of the 16th International Conference on Vision Interface*, Halifax, Canada, June 2003.

[10] H.G. Adelmann, "Butterworth equations for homomorphic filtering of images", *Computer in Biology and Medicine*, vol. 28, pp. 169-181, 1998.

[11] J.P. Tarel, "Calibration radiométrique de caméra," *INRIA technical report*, March 1995.

[12] J.C. Kinsey and L.L. Whitcomb, "Towards in-situ calibration of gyro and Doppler navigation sensors for precision underwater vehicle navigation," *Proceedings of the 2002 Int. Conf. On Robotics and Automation*, May 2002, Washington, DC.

[13] J. Weickert, "Efficient image segmentation using partial differential equations and morphology," *Pattern Recognition*, 34, pp 1813-1824, 2001.

[14] M. Borgetto, C. Jauffret and V. Rigaud, "Auto-localisation d'un engin sous-marin quadrillant une zone," *Proceedings of the 2003 GRETSI*, Paris, September 2003, in press.

[15] E.W. Dijkstra, "A note on two problems in connexion with graphs", *Numerische Mathematik*, 1959, vol.1, pp 269-271.

[16] http://www.ifremer.fr/flotte/r&d/mosaique.htm