

Perspectives of Auto-Correcting Lens Distortions in Mosaic-Based Underwater Navigation

Armagan Elibol, Birgit Möller and Rafael Garcia

Abstract—When Unmanned Underwater Vehicles (UUVs) perform missions near the ocean floor, optical sensors can be used to improve local navigation. Video mosaics allow to efficiently process the images acquired by the vehicle, and also to obtain position estimates. We discuss in this paper the role of lens distortions in this context, proving that degenerate mosaics have their origin not only in the selected motion model or in registration errors, but also in the cumulative effect of radial distortion residuals. Additionally, we present results on the accuracy of different feature-based approaches for self-correction of lens distortions that may guide the choice of appropriate techniques for correcting distortions.

I. INTRODUCTION

Exploration of unstructured unknown environments is a difficult task for a robot. If these environments are located underwater, close to the sea floor, the task becomes even more challenging. In the last few years several research groups have proposed vision-based systems to explore the ocean floor. Images are usually acquired by a down-looking camera carried by an underwater robot. Determining the motion between consecutive images of a sequence acquired this way can then be used to build image mosaics of the ocean floor, and to estimate the motion of the underwater vehicle [1].

Recovering the camera motion between two image frames is an indispensable prerequisite for mosaicing as well as vehicle motion estimation. It requires the choice of an appropriate motion model suitable to adequately describe changes in the images that result from the movements of the camera during acquisition. In the literature [2] several mathematical models for image transformations can be found whereas euclidean, similarity, affine and projective transformations are the most important ones. While euclidean transformations allow to model translations and rotations with 3 degrees of freedom (DOFs), similarity transformations with 4 DOFs additionally support image scaling. Affine transformations provide 6 DOFs subsuming translations, rotations, scaling and sheering. Finally, projective transformations (also called *homographies*) yield the largest flexibility given 8 DOFs which enable the modeling of projective mappings. Which model to choose mainly depends on the configuration of the camera and its degrees of freedom.

All the authors are with the Computer Vision and Robotics Group of the University of Girona, 17071, Girona, Spain, email: aelibol@eia.udg.edu.

Birgit Möller's permanent address: Institute of Computer Science Martin-Luther-University Halle-Wittenberg, Halle, Germany.

One of the first mosaicing systems for an underwater robot was proposed by Marks et al. [3]. This system achieved real-time performance due to the use of special purpose hardware for image filtering and correlation, allowing the creation of “single column” mosaics in real time. Since the robot could control its heading, a very restrictive (translation-only) motion model was assumed, without taking into account rotation, scaling, perspective distortion, etc. For the system to work, the robot was programmed to keep constant altitude. Provided that this system used a 2 DOFs motion model, no distortions appeared when mosaicing large sequences. However, when the robot performed either changes in altitude or motion in roll, pitch or yaw local misalignments in the final mosaic appeared.

In [4] a visual station-keeping system based on optical flow was presented. In this case motion was estimated according to a 2D similarity motion model with 4 DOFs. When underwater mosaics are built with 4 DOFs, scaling distortions may appear. If more DOFs are introduced in the motion model, however, the situation is even worse since perspective distortions may affect the final mosaic (especially when using homographies), e.g., resulting in an extreme shrinkage or expansion of images when they are mapped onto the mosaic.

Although very little discussion can be found in the literature about these scaling distortions, they are well known by researchers in the field of underwater mosaicing. Only the work presented in [5], based on the previous work of Sawhney and Kumar [6], deals with the problem of scaling distortions by adding a term to the function to optimize that bounds the size of the image diagonals once they are warped into the mosaic. Moreover, one term (k_1) of radial distortion is taken into account, since lens distortion correction is embedded in the mosaicing process as proposed in [6].

According to the literature, if a large area of the ocean is to be mosaiced, it is needed to limit the DOFs of the motion model, and lens distortion is assumed to have little impact on the final mosaic or estimated trajectories. However, in this paper we will show that radial distortions very well have an important effect on the final quality of a mosaic image. They should by no means be neglected since just accounting for related image scalings cannot compensate for registration errors induced by lens distortions. Moreover, we will show that lens distortions are one of the main error sources in image registration and have a serious impact on motion recovery of an underwater vehicle. Although a huge amount of calibration and correction techniques exist, we will also discuss perspectives for an auto-correction of

distortions, since standard approaches are difficult to apply in underwater domains.

The remainder of the paper is organized as follows. Section II reviews the basics of camera geometry. Section III looks first at the theoretical background of lens distortion before introducing some common approaches for modeling distortion in computer vision. Section IV presents experimental observations and synthetic results regarding the influence of distortion in underwater navigation, while in section V self-correction techniques are discussed. A conclusion is given in section VI.

II. BASICS OF CAMERA GEOMETRY

In image-based localization and navigation an indispensable prerequisite for motion recovery is given by the assumption that 3D scene points are mapped onto different image planes according to a common imaging model that holds for all images. Usually this model is given by the well known *pinhole model* [2]. Following this model every 3D scene point P is projected onto a 2D image point \bar{p} according to the following equation:

$$\bar{p} = K [I | \vec{0}] \cdot P \quad \text{with} \quad K = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where matrix K encodes the *intrinsic camera parameters* and I denotes the 3×3 identity matrix. The calibration matrix K usually includes 4 parameters which are the *focal length* of the camera in x and y direction, α_x and α_y (which are both the same if square pixels are given), and its *principal point* $c_0 = (x_0, y_0)$. According to (1), which is written in homogeneous coordinates, 3D points and their images are related to each other by a *linear* mapping which significantly simplifies mathematical tractability of the image acquisition process and the modelling of camera/vehicle motion.

III. MODELING LENS DISTORTION

The pinhole camera model as introduced in the previous section assumes an ideal camera geometry which in practice rarely exists. Real cameras and lenses usually introduce some kind of distortion to the imaging process. As a consequence the mappings between 3D world points P and their projections on the various image planes are no longer linear, but include some non-linearities. Although these can be ignored in various applications –e.g., in object recognition or tracking– they are crucial in motion and navigation tasks (cf. [7]). The most important distortions that appear in practical image acquisition are given by *radial and tangential lens distortions* that are introduced in the following paragraphs.

A. Radial and Tangential Distortion

Lens distortions have at first been observed and treated by researchers from the photogrammetry community [8]. Within computer vision they had been ignored for a long period of time, however, with growing interest on geometric scene reconstruction and camera modeling they also gained more importance in this field. Hence, over the years several

approaches for modeling and correcting lens distortions have been investigated. The most common model used today is given by a radial-symmetric non-linear *polynomial mapping* between ideal, undistorted image points \bar{p} and their distorted correspondences $p_d = (x_d, y_d)$ as they are observable in acquired images:

$$p_d = c_0 + (1 + \kappa_1 \cdot r + \kappa_2 \cdot r^2 + \dots)(\bar{p} - c_0) \quad (2)$$

The κ_i are called distortion coefficients, c_0 denotes the center of radial distortion (which is often assumed to be identical to the principal point of the camera), and r gives the Euclidean distance of any point \bar{p} to the center of distortion defined as $r = \|\bar{p} - c_0\|$. In practice often only the terms with even exponentials are included. Besides, in several works (e.g., [6]) it has even been shown that sometimes already one single distortion term yields satisfying results for correction. In addition to the very common modeling of lens distortions by high-order polynomials in the last years a second kind of models has been investigated termed *division models*. The one introduced by Fitzgibbon [7] is given below:

$$\bar{p} = \frac{1}{1 + \lambda \cdot (\|p_d - c_0\|)} \cdot (p_d - c_0) \quad (3)$$

This model is not just an approximation to the common model in (2), but an alternative way to model lens distortion which has proven to have comparable power. For completeness it should be noted that, although *radial* distortions are the most important class of lens distortions, real lenses sometimes also exhibit portions of non-radial distortion. These are usually assumed to be *tangential* distortion.

In practice, however, tangential lens distortions are usually ignored since it is difficult and often numerically instable to estimate these portions from distorted images. Furthermore most of the time they only improve distortion correction slightly, compared to a pure radial distortion correction. Therefore, it does not seem to be worth spending much time on reconstruction of tangential coefficients.

B. Correcting Lens Distortion

Lens distortion is an important problem in various computer vision applications and leaving distorted images uncorrected may lead to completely inaccurate results (cf. section IV). To avoid this a huge amount of algorithms for correcting lens distortions emerged over time. The various approaches can roughly be divided into three different classes.

- a) *Pattern-based approaches* where images of calibration patterns that show a rectangular grid of squares or at least some straight lines, e.g., chessboards, are used for calibration [9].
- b) *Plumb line methods*, i.e., algorithms that ground on assumptions about specific structures in a scene and their appearance in acquired images e.g., ellipsoids that should map to circles; for distortion calibration images are acquired and distortion parameters can then be estimated by transforming the

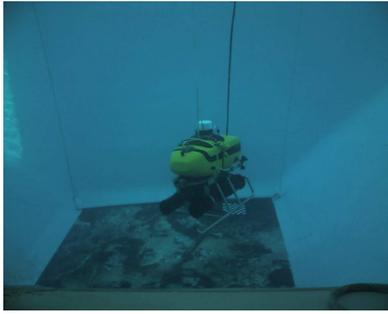


Fig. 1. Snapshot of UUV operating in the test pool. The floor of the pool is covered by a poster of real sea floor as to simulate realistic sea floor images for assessing algorithms.

images as to make the elements appear as they are supposed to [10].

- c) *Self-correction techniques* that basically rely on two or more images taken of an arbitrary scene; distortion parameters are reconstructed based on fundamental matrix relations or homographies that should hold between the images. Since these constraints are usually not fulfilled between distorted images the errors in geometric reconstruction yield valuable hints for distortion estimation[11], [12].

IV. EXPERIMENTAL OBSERVATIONS

In underwater navigation often image sequences are taken from the seafloor which is monitored by the vehicle during its operation. The motion of the vehicle can then be reconstructed from these sequences by estimating parameters of an appropriate motion model between subsequent images of the sequence. In our framework we assume that the seafloor is more or less planar and the camera motion can be described by an *8-parameter projective homography* which has proven to be a reasonable model in our case.

A. Real Image Registration

We performed tests on real imagery acquired with our robot (Fig. 1). On the ground of this pool a huge poster simulating the sea floor was placed yielding a controlled environment for test data acquisition. In particular, since the floor of the pool is perfectly planar and the robot performs movements in 3D space, camera motion in acquired sequences should be described perfectly by 8-DOF homographies.

A number of test datasets was acquired by the robot monitoring the poster. We started our experiments on mosaic-based underwater navigation by applying various mosaicing algorithms to the data. On the one hand feature-based approaches were applied where the homography parameters are robustly reconstructed based on correspondences in both images formerly detected by a Harris corner detector [13]. On the other hand also a featureless technique that basically estimates the optical flow between two images constraint by the projective motion model [14]. Featureless and feature-based algorithms were both combined with local estimation strategies while the latter ones were additionally embedded

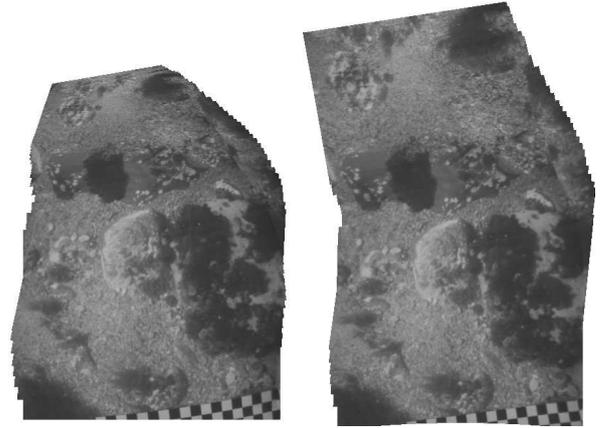


Fig. 2. Left: exemplary mosaic resulting from registering approximately 40 images of the test dataset. Right: mosaic calculated from the same images, however, this time after radial lens distortion correction.

in a global framework. By applying a global strategy all images are processed simultaneously while in local processing the data is only analyzed incrementally.

Independent of the applied algorithms and strategies we always got severely distorted images. From the example mosaic shown on the left of Fig. 2 it can be seen that already after short periods of time, images were severely scaled and distorted which is very uncommon in our scenario. Since the robot is moving approximately parallel to the poster, the resulting mosaic is expected to be almost undistorted and no image scaling should appear at all. Such severe distortions as observed can usually result only if the orientation of the camera's image plane is no longer parallel to the seafloor. Additionally, error accumulation over time which cannot be avoided in local registration could neither explain these effects since they also appeared in global registration modes. Thus, further tests were carried out to discover the reasons for the distortions.

B. Intensity Residual Analysis

On the one hand a detailed analysis of intensity residuals in registration was performed to extract error characteristics. One typical result of this analysis is shown in Fig. 3 where two registered images are shown with significant residuals (exceeding a threshold of 10) marked in green. As is apparent from the picture the residuals are unequally distributed over the image and, concentrate in image corners. Such error patterns may not appear in case of pure registration errors due to misalignment, since projective homographies cannot distort images non-linearly and errors along image borders should always enforce errors in the image center as well. The distribution moreover implied some kind of radial distortion according to the radial-symmetric distribution.

C. Lens Distortion Correction

To verify this hypothesis we have tried to extract lens distortion coefficients from the image sequences applying (2) with only κ_2 and assuming the center of radial distortion equal to the principal point and located in the center of

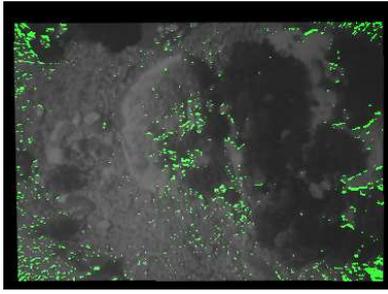


Fig. 3. Typical registration result on lens distorted images. The picture shows significant intensity residuals remaining after registering two images. They are concentrated along the image borders while in the center only few errors appear which is very unlikely for projective transformations.

the image. Given a result of $\kappa_2 \approx -1.5e^{-6}$ the images were corrected and mosaiced again. The result, shown on the right of Fig. 2, was much less distorted and by this supported our hypotheses. Even on long image sequences with more than 150 images it became possible to close loops which was absolutely impossible by just using the original dataset. Fitzgibbon [7] presented similar results with regard to path reconstruction of a mobile robot. Nevertheless, to completely eliminate the possibility that the distortion might solely be due to perspective effects we further on simulated effects of radial distortion on synthetic data as outlined in detail in the next section.

D. Simulations on Synthetic Data

To verify the effects of lens distortion in image registration we performed synthetic tests. Given the distortion-free source image of the pool poster we cropped 60 images from this poster whereas a vertical camera translation was simulated between each pair of images. This sequence was then registered applying featureless estimation. As a result the poster strip could be recovered very well as it can be seen from Fig. 4, left. Afterwards the images were distorted by applying the formerly estimated κ_2 to each single image before they were registered again. This time the resulting poster was severely scaled and distorted, as it can be seen from Fig. 4, right. The total strip has only half size compared to the ground truth one, and in horizontal direction also significant scaling appears. Comparing the estimated homographies in every registration step an error in reconstructed translation of approximately 1.5 pixels appears which in the end results in the severe scaling that we observed. It should be noted, however, that by analyzing residual errors in the mosaicing process nearly no residuals could be detected and patterns like the one shown in Fig. 3 could not be discovered. This led to the conclusion that lens distortions may sometimes be compensated to a large degree by the projective motion model which obviously happened in the synthetic tests (and was later on also verified by experiments on self-calibration, see next paragraph).

V. ROBUST FEATURE-BASED SELF-CORRECTION

For correcting lens distortions there are basically the three classes of possible approaches that were outlined in

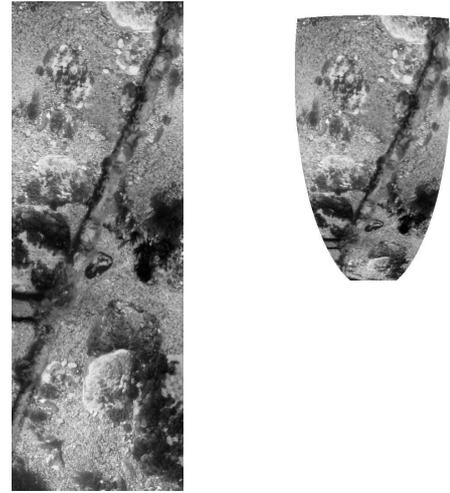


Fig. 4. Comparison of registration results on synthetic images affected by lens distortion: on the right the sequence images are distorted with $\kappa_2 = -1.5e^{-6}$ (cf. Eq. (2)), on the left they are undistorted. It can clearly be seen that radial distortion may severely falsify motion estimation results.

section III-B. It has early been stated and proven by various authors (e.g., [15]) that pattern-based calibration is usually the best thing one can do. Although also plumb line and self-calibration techniques yield satisfying results in many applications, the accuracy of these results is usually not as good as of pattern-based approaches. Nevertheless, in many applications it is simply not possible to do pattern-based calibration. E.g., in our case underwater images are usually acquired in real world environments where neither calibration patterns nor specific scene structures may be assumed. Additionally, zoom and with this also focal length and distortion coefficients sometimes change during image acquisition, e.g., due to detailed exploration or even auto-focus. In these cases the only chance for distortion correction is self-calibration from motion or geometric constraints.

We have tried to recover radial distortion parameters by using correspondences between images and related homographies. Let $\bar{p} = (x, y, 1)$ and $\bar{p}' = (x', y', 1)$ be projections of the same 3D point in two different images. The homography between both images can be calculated by minimizing the following error function $\epsilon_{\mathbf{H}}(\mathbf{H})$:

$$\epsilon_{\mathbf{H}}(\mathbf{H}) = \sum_i^n |\bar{p}' - \mathbf{H} \cdot \bar{p}| \quad (4)$$

where n is total number of correspondences. This error function is minimized by employing non-linear iterative optimization methods [16]. Since in real images only distorted points p_d and p'_d are given, \bar{p} and \bar{p}' have to be derived from these by distortion correction:

$$\bar{p} = \begin{pmatrix} x_c = x_0 + (1 + k_1 \cdot r^2 + k_2 \cdot r^4) \cdot (x_d - x_0) \\ y_c = y_0 + (1 + k_1 \cdot r^2 + k_2 \cdot r^4) \cdot (y_d - y_0) \end{pmatrix} \quad (5)$$

where (x_0, y_0) is center of radial distortion, k_1 and k_2 are radial distortion parameters and $r^2 = (x_d - x_0)^2 + (y_d - y_0)^2$. Embedding corrected points into Eq. 4, allows for

simultaneous radial distortion parameters and homography estimation [5].

Different synthetic data sets were used in our experiments. We report here the results obtained by taking a grid of points and applying a known affine homography to obtain pairs of correspondences, which will serve as ground-truth (\mathbf{H}_{real}). The assumed size of the images was 300×300 . A radial distortion equivalent to that of the camera of the underwater robot that was used for experiments was applied to the original grid and the corresponding distorted one, inducing a maximum pixel offset of up to 20 pixels in each grid. Both point sets were corrupted with various levels of zero-mean Gaussian noise to carry out the different experiments. We first examined two approaches to recover the camera motion in terms of an affine transformation \mathbf{H} , ignoring lens distortions completely:

- A1: The homography was estimated directly from the distorted points, ignoring lens distortions completely (\mathbf{H}_{A1})
- A2: Since radial distortion is a function of the distance from every pixel to the center of distortion in the image, points that are close to the image corners are more distorted than points located close to the distortion center. Therefore, the second approach only used those correspondences that were inside the area of a circle with radius r located at center of distortion (in our case image center).

Given a camera motion between two images, it is not only possible to recover the motion of the camera, but it is also possible to recover lens distortions in the images. The basic idea is to use registration residuals that cannot be compensated by the camera motion model to estimate lens distortions. We examined three different approaches to recover the coefficients k_1 and k_2 for correcting the distortions:

- B: In this case at first radial distortion parameters (k_1, k_2) are estimated by minimizing Eq.4 for these two parameters. \mathbf{H} is taken from applying method A2 before, and (p, p') are the noisy distorted points located outside the circle. In this case, the homography \mathbf{H} is fixed during the minimization process and only the radial distortion terms are optimized. Once (k_1, k_2) have been found, the noisy distorted correspondences are undistorted applying the obtained parameters. Finally, \mathbf{H}_B is computed from the resulting undistorted correspondences.
- C1: Here Eq.4 is minimized given \mathbf{H}_{A2} . This time \mathbf{H} is not fixed and it is optimized *simultaneously* with (k_1, k_2) , as it is proposed in [5].
- C2: Finally, the last approach consists of undistorting the noisy distorted correspondences with radial distortion parameters (k_1, k_2) obtained from C1, and then computing (\mathbf{H}_{C2}) from these undistorted points.

Fig. 5 illustrates the results. The vertical axis represents the remaining residuals when applying the estimated homogra-

phies to the set of undistorted ground-truth correspondences (Eq.4). If the motion is completely and correctly recovered, $\epsilon_{\mathbf{H}}(\mathbf{H})$ equals to zero. The bigger the residual is, the worse is the motion reconstruction. In the graph on the left in Fig. 5, zero mean Gaussian noise with $\sigma = 1$ was applied to both point sets, while on the right only $\sigma = 0.25$ was chosen. From the results, it can be observed that for very small radii (up to 25 pixels) \mathbf{H}_{A2} performs worse than \mathbf{H}_{A1} . On the contrary, for bigger radii \mathbf{H}_{A2} outperforms the blind estimation. If the chosen circle includes all the correspondences, performance of both methods is same as they are using the same data to compute the homography. It should be noted that residuals are always computed for all the correspondences to allow comparison among the different approaches. On the other hand, given a high noise ratio, optimizing together for the homography and the distortion parameters (B) yields a worse estimation of the real homography than optimizing only (k_1, k_2) , after fixing the homography as computed from the central points of the image (C1). Finally, the good performance of estimating the homography with just the central, but nevertheless distorted points of the images (\mathbf{H}_{A2}) and (\mathbf{H}_B), fixing the homography, and then estimating radial distortion parameters and re-estimating the homography, should be outlined. Although Fig. 5 just shows the results for a given configuration of test points, and two noise levels, extensive simulations have been run, showing the same tendency in the results.

VI. CONCLUSIONS AND FUTURE WORK

Growing capabilities of autonomous robots require robust motion recovery and navigation algorithms. As analyzed in this paper such techniques significantly rely on good-quality image data and are very sensitive to lens distortions. Hence, distortion correction is indispensable.

The idea behind radial distortion estimation from correspondences is to compensate errors that cannot be compensated by homographies and thus, getting a better estimation of radial distortion parameters. From our experiments, it can be seen that the results are highly dependent on a good homography estimation. The better the homography estimation is, the better radial distortion estimation is and vice-versa. Our experiments show that instead of simultaneously estimating both homography and radial distortion parameters, estimating the homography by using the correspondences that are closer to the distortion center (\mathbf{H}_{A2}) then estimating radial distortion parameters and re-estimating the homography, (\mathbf{H}_B), results in a better estimation.

VII. ACKNOWLEDGMENTS

This work has been partially funded through the MARMARNET EU project MRTN-CT-2004-505026, in part by the FREESUBNET EU project MRTN-CT-2006-036186 and in part by the Spanish Ministry of Education and Science under grant CTM2007-64751. Armagan Elibol has been funded by Generalitat de Catalunya under grant 2004FI-IQUC1/00130. Birgit Möller was supported by a fellowship within the Postdoc-Programme of the German Academic Exchange Service (DAAD).

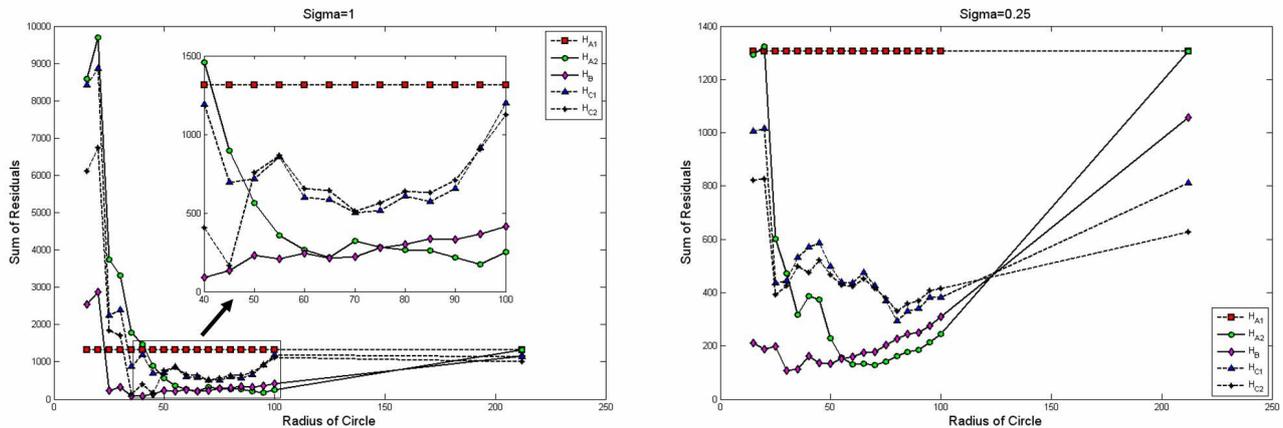


Fig. 5. Residuals obtained by the different approaches to recover motion between two images affected by lens distortion. (Left) zero-mean Gaussian noise with $\sigma = 1$ pixel. (Right) $\sigma = 0.25$ pixel

REFERENCES

- [1] N. Gracias, S. Zwaan, A. Bernardino, and J. Santos-Victor, "Mosaic based navigation for autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 3, pp. 609–624, Oct. 2003.
- [2] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2004.
- [3] R. L. Marks, S. M. Rock, and M. J. Lee, "Real-time video mosaicking of the ocean floor," *IEEE Journal of Oceanic Engineering*, vol. 20, pp. 229–241, July 1995.
- [4] S. Negahdaripour and Xun Xu, "Mosaic-based positioning and improved motion-estimation methods for automatic navigation of submersible vehicles," *IEEE Journal of Oceanic Engineering*, vol. 27, no. 1, pp. 79–99, Jan 2002.
- [5] O. Pizarro and H. Singh, "Toward large-area mosaicing for underwater scientific applications," *IEEE Journal of Oceanic Engineering*, vol. 28, no. 4, October 2003.
- [6] H. Sawhney and R. Kumar, "True multi-image alignment and its application to mosaicing and lens distortion correction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 3, pp. 235–243, 1999.
- [7] A.W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. 125–132.
- [8] K. Kraus, J. Jansa, and H. Hager, *Photogrammetry: Advanced Methods and Applications*, Dümmler Verlag Bonn, 1997.
- [9] R.Y. Tsai, "A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 4, pp. 323–344, 1987.
- [10] T. Thormaehlen, H. Broszio, and I. Wassermann, "Robust line-based calibration of lens distortion from a single view," in *Proceedings of International Conference Mirage*, March 2003, pp. 105–112.
- [11] Gideon P. Stein, "Lens distortion calibration using point correspondences," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1997.
- [12] R.M. Steele and C.O. Jaynes, "Overconstrained linear estimation of radial distortion and multi-view geometry," in *European Conference on Computer Vision*, 2006, pp. 253–264.
- [13] R. Garcia, J. Battle, X. Cufí, and J. Amat, "Positioning an underwater vehicle through image mosaicking," in *IEEE International Conference on Robotics and Automation*, Seoul, Rep. of Korea, May 2001, vol. 3, pp. 2779–2784.
- [14] B. Möller, D. Williams, and S. Posch, "Robust image sequence mosaicing," in *Pattern Recognition, Proc. of 25th DAGM Symposium*, September 2003.
- [15] Z. Zhang, "On the epipolar geometry between two images with lens distortion," 25-29 Aug 1996, vol. 1, pp. 407–411.
- [16] J. Nocedal and S.J. Wright, *Numerical optimization*, Springer, New York, 1999.