Phase Correlation for Dense Visual Compass from Omnidirectional Camera-Robot Images

Fabio Morbidi, Guillaume Caron

***Abstract*—In this paper, we present a new omnidirectional visual compass for a camera-robot, based on the phase correlation method in the 2-D Fourier domain. The proposed visual compass is *accurate*, *robust* to image noise, and *frugal* in the use of computational resources. Moreover, unlike the majority of existing ego-motion estimators, it does not rely on any geometric image primitive, and it only requires a minimal knowledge of the internal camera parameters. Extensive real-world experiments conducted with a hypercatadioptric camera mounted on the end-effector of a Stäubli manipulator and on a Pioneer robot, show the effectiveness of our approach.**

***Index Terms*—Omnidirectional Vision, Visual Compass, Catadioptric Camera.**

I. Introduction

A. Motivation and related work

CAMERAS are being increasingly used in mobile robotics since they are small, lightweight, inexpensive, and they provide a richer information about the surrounding environment than other exteroceptive sensors. Differently from IMUs or gyroscopes, a single camera can meet all of robot's needs, since visual information can be used for high-level semantic analysis and low-level perception tasks alike. The problem of estimating the orientation of a camera between a reference and a query image is well-known in the robotics literature [1]. *Visual gyroscopes* estimate the attitude (roll, pitch, and yaw angles) of a camera-robot, while *visual compasses* limit themselves to a single axis (yaw angle). On the other hand, *visual odometers* exploit the entire sequence of images between the reference and query, for determining the camera pose (position and orientation).

Omnidirectional cameras have gained prominence for robot localization in unknown environments, since they permit to enlarge the field of view without losing the image features. A wealth of omnidirectional gyroscopes has been proposed in the recent literature [2]–[6]. With the exception of [3], all these methods assume a *calibrated* catadioptric camera with a single effective viewpoint, and rely on image features (e.g. the image projection of 3-D parallel lines, vanishing points, SIFT features) to estimate the attitude of a camera-robot. However, feature extraction and tracking require a nontrivial image-processing step, and these operations are sensitive to outliers and illumination changes. Another limitation of feature-based methods is that they generally rely on strong assumptions about the structure of the 3-D environment where the robot moves: for example, the approaches in [3]–[5] can only be used in man-made environments rich in straight lines.

To overcome these problems, several authors have proposed to exploit the pixel intensity of the *overall image*, interpreted as a 2-D signal, for estimating the pose of a camera-robot. This approach is alternatively referred to as *dense*, direct or global, depending on the context [7], and it led to the so-called appearance-based methods [8], [9] in the standard image domain, and to strategies based on *harmonic analysis* (a generalization of Fourier analysis), in the spatial-frequency domain. The latter approaches have appeared in the literature in various manifestations. In [10]–[12], by interpreting the rows of the panoramic cylinder as unidimensional signals, the Fourier components of the image (or "Fourier signature") are computed, and used for visual navigation: the Fourier signature is indeed crucial to reduce the computational and memory requirements of the algorithms developed in [10]–[12]. Other geometric works have exploited the Fourier transform defined on the two-sphere and on the special Euclidean/rotation groups, for 2-D [13] and 3-D rigid-motion estimation [14]–[16].

In the image processing and pattern recognition literatures, the 2-D Fourier transform is routinely used for registration, i.e. for the estimation of the translation, rotation and scaling between two given images [17]. A key tool here is the so-called *phase correlation method* (a.k.a. "Phase-Only Matched Filter" or POMF in short). By relying on the shift property of the 2-D Fourier transform, the phase correlation method computes the cross-power spectrum of two images and searches for the location of a peak in its Fourier inverse: the position of the peak corresponds to the translation vector between the two images [18]. By using a log-polar representation, the phase correlation method can be easily extended to estimate the rotation angle and scaling between a pair of images [19], [20]. Unfortunately, this method is not well known in the robotics community, and its potential for visual localization has not yet been fully exploited. The only prior research that we were able to locate in the robotics literature is [21], where the pose of a robotic platform is estimated via a calibrated ground-facing perspective camera, and [22], [23], where the Fourier-Mellin transform (akin to the polar Fourier transform) is used for the registration of a sequence of radar images, and for robot homing via visual-landmark matching, respectively.

B. Original contributions, organization and notation

In this paper, we design a new visual compass based on the phase correlation method, for a single-viewpoint

Manuscript received September, 9, 2016; Revised December, 9, 2016; Accepted December, 19, 2016.

This paper was recommended for publication by Editor F. Chaumette upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the ADAPT project. The ADAPT project was selected under the Interreg VA France (Channel) England - European cross-border cooperation programme, co-funded by the ERDF.

The authors are with the MIS laboratory, Université de Picardie Jules Verne, 33 Rue Saint-Leu, 80039 Amiens Cedex, France. {fabio.morbidi,guillaume.caron}@u-picardie.fr

Digital Object Identifier (DOI): see top of this page.

omnidirectional camera mounted on a robot undergoing a 2-D motion. The proposed visual compass has several attractive features. In fact, it is:

- *Correspondence-free*. The overall image is used: no need for segmentation and extraction of geometric features,
- *Accurate* and *robust* to image noise,
- *Frugal* in the use of computational resources,
- *Weakly calibrated*: it only needs the coordinates of the estimated principal point of the camera to center the omnidirectional (catadioptric or fisheye) images.

A new dataset of indoor and outdoor images, called OVMIS, has been generated with a hypercatadioptric camera mounted on the end-effector of a Stäubli manipulator and on a Pioneer robot. Thanks to the precise ground truth provided by these robotic platforms, we rigorously studied the performance of the proposed algorithm and compared it with that of competing approaches. We publicly released the OVMIS dataset on the Internet for the entire research community.

Note that the visual gyroscope studied in [14] is reminiscent of our approach. However, differently from [14], in this paper the shift property is not applied to the Fourier-transformed omnidirectional image mapped to the unit sphere of equivalence (knowing the full set of internal camera parameters), but *directly* to the image in the log-polar Fourier domain. Moreover, our algorithm is *non-iterative*, and it does not rely on nonlinear minimization techniques to estimate the yaw-angle of the camera-robot.

The rest of this paper is organized as follows. In Sect. II, we review some basics of 2-D Fourier transform, and introduce the phase correlation method. In Sect. III, we describe our omnidirectional visual compass, and in Sect. IV we present the results of a vast experimental campaign conducted with an industrial manipulator and a wheeled robot. Finally, in Sect. V, the main contributions of the paper are summarized, and some possible directions for future research are outlined.

Notation: Throughout this paper, we use the symbol $\mathtt{i} = \sqrt{-1}$ to indicate the imaginary unit, $\mathbb{C}$ the set of all complex numbers, $\mathbb{R}^n$ the n-dimensional Euclidean space, $\mathbf{I}_{n\times n}$ the $n \times n$ identity matrix, and $\delta(x)$ and $\delta(x,\, y)$ the unidimensional and bidimensional Dirac deltas, respectively. $\mathrm{SO}(2)$ is the special orthogonal group of the plane, and $\mathrm{SIM}(2)$ the similitude group of the plane, i.e. the set of transformations consisting of rotations, scalings and translations:

$$(\psi_j,\, \sigma_j,\, \mathbf{x}_j) \;:\; \mathbf{x} \mapsto \sigma_j\, \mathbf{R}^T(\psi_j)\, \mathbf{x} + \mathbf{x}_j,$$

$$\mathbf{R}(\psi_j) = \begin{bmatrix} \cos\psi_j & -\sin\psi_j \\ \sin\psi_j & \cos\psi_j \end{bmatrix} \in \mathrm{SO}(2),\; \sigma_j > 0,\; \mathbf{x}_j \in \mathbb{R}^2.$$

Given a complex-valued function $f(x,\, y) : \mathbb{R}^2 \to \mathbb{C}$, $f^*(x,\, y)$ indicates its complex conjugate and $\big|f(x,\, y)\big|$ its magnitude. $\diamond$

II. Preliminaries

In this section, we provide relevant background on the 2-D Fourier transform, and recall how the phase correlation method can be used to compute the rigid motion and scale factor between two bivariate functions. The reader is referred to [24, Ch. 4] for more details on classical Fourier analysis.

Given an integrable function $f(x,\, y) : \mathbb{R}^2 \to \mathbb{C}$, we denote its Fourier transform in rectangular coordinates by:

$$\begin{aligned} \mathfrak{F}[f(x,\, y)] \,&=\, \widehat{f}(u,\, v) \\ &\triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,\, y)\, e^{-\mathtt{i}(u\,x\,+\,v\,y)}\, dx\, dy, \end{aligned} \tag{1}$$

where $(u,\, v)$ are the Fourier variables, and $\mathfrak{F}[\,\cdot\,]$ is the Fourier operator. If $f(x{+}x_0,\, y{+}y_0)$ is a translated replica of $f(x,\, y)$, the *shift property* of the Fourier transform is:

$$\mathfrak{F}[f(x + x_0,\, y + y_0)] \,=\, \widehat{f}(u,\, v)\, e^{\mathtt{i}(u\,x_0\,+\,v\,y_0)}. \tag{2}$$

Given two functions f_1 and f_2, satisfying $f_2(x,\, y) = f_1(x + x_0,\, y + y_0)$, from (2) we have that:

$$\frac{\widehat{f}_2(u,\, v)\, \widehat{f}_1^{\,*}(u,\, v)}{\big|\widehat{f}_2(u,\, v)\big|\, \big|\widehat{f}_1^{\,*}(u,\, v)\big|} \,=\, e^{\mathtt{i}(u\,x_0\,+\,v\,y_0)}.$$

Thus, the translation vector $[x_0,\, y_0]^T$ can be determined by computing the inverse Fourier transform,

$$\mathrm{Corr}(x,\, y) \,\triangleq\, \mathfrak{F}^{-1}[e^{\mathtt{i}(u\,x_0\,+\,v\,y_0)}] \,=\, \delta(x + x_0,\, y + y_0),$$

and finding the single peak of the *normalized phase correlation function* $\mathrm{Corr}(x,\, y)$. By changing to polar coordinates in both the $(x,\, y)$ and $(u,\, v)$ domains, i.e. by defining $x = r\cos\theta$, $y = r\sin\theta$ and $u = \rho\cos\psi$, $v = \rho\sin\psi$, the original function is $f(x,\, y) = f(r,\, \theta)$, and the Fourier transform in (1) can be rewritten as [25, Sect. 4.7.1]:

$$\widehat{f}(\rho,\, \psi) = \int_0^{\infty}\int_0^{2\pi} f(r,\, \theta)\, e^{-\mathtt{i}\,\rho\, r\cos(\theta\,-\,\psi)}\, r\, dr\, d\theta, \tag{3}$$

which is referred to as the *polar Fourier transform*.

Let us now focus on the most general case, and let f_2 be a *translated*, *rotated*, and *scaled* replica of function f_1, i.e.

$$f_2(\mathbf{x}) \,=\, f_1(\sigma\, \mathbf{R}^T(\psi_0)\, \mathbf{x} \,+\, \mathbf{x}_0), \tag{4}$$

where $\mathbf{x} = [x,\, y]^T$, $\mathbf{x}_0 = [x_0,\, y_0]^T$ is the translation vector, $\mathbf{R}(\psi_0) \in \mathrm{SO}(2)$, and $\sigma > 0$ is the scale factor. By taking the polar Fourier transform of both sides of (4), and using again (2) and the scaling property of the Fourier transform [24, Sect. 4.6.1], we find that:

$$\widehat{f}_2(\rho,\, \psi) \,=\, \frac{1}{\sigma^2}\, \widehat{f}_1\Big(\frac{\rho}{\sigma},\, \psi \,+\, \psi_0\Big)\, e^{\mathtt{i}(u\,x_0\,+\,v\,y_0)}.$$

If we now define $M_1(\rho,\, \psi) \,=\, \big|\widehat{f}_1(\rho,\, \psi)\big|$ and $M_2(\rho,\, \psi) \,=\, \big|\widehat{f}_2(\rho,\, \psi)\big|$, we have that:

$$M_2(\rho,\, \psi) \,=\, \frac{1}{\sigma^2}\, M_1\Big(\frac{\rho}{\sigma},\, \psi \,+\, \psi_0\Big). \tag{5}$$

Finally, using log-polar coordinates, we can rewrite (5) as:

$$M_2(\ln\rho,\, \psi) \,=\, \frac{1}{\sigma^2}\, M_1(\ln\rho - \ln\sigma,\, \psi \,+\, \psi_0), \tag{6}$$

where $\ln(\cdot)$ denotes the natural logarithm[1]. By leveraging the log-polar Fourier transform, rotations and scalings are thus reduced to translations, and ψ_0 and σ can be recovered via

[1]Note that in terms of *energy distribution* as a function of frequency, from (6) we have that $E_{f_1}(\ln\rho \,-\, \ln\sigma,\, \psi \,+\, \psi_0) \,=\, \sigma^4\, E_{f_2}(\ln\rho,\, \psi)$ where $E_{f_2}(\ln\rho,\, \psi) \,=\, M_2^2(\ln\rho,\, \psi)$ is the energy spectral density of f_2 [26, Sect. 1.2].

phase correlation applied to M_1 and M_2 (irrespective of the translation vector $\mathbf{x}_0$). Using (6) to estimate the rotation angle ψ_0 results in an ambiguity of π [20]. This ambiguity can be resolved by rotating f_1 by the two possible angles, ψ_0 and $\psi_0 + \pi$, and then recovering the relative translation $\mathbf{x}_0$ and correlation peak for each angle. The parameters corresponding to the highest correlation peak are selected as the result.

III. Omnidirectional visual compass

In this section, we leverage the phase correlation method described in Sect. II to design an omnidirectional visual compass for a camera-robot undergoing a 2-D motion. For simplicity of presentation, N × N images are henceforth considered. The discrete counterpart of (3) will be utilized to compute the polar Fourier transform of these square images.

Let $I_1(\mathbf{x}) = I_1(x, y)$ be the omnidirectional grayscale image taken by the camera-robot at the initial (or reference) pose, and $I_j(\mathbf{x})$, $j \in \{2, \ldots, m\}$, a grayscale query image taken along an unknown 2-D trajectory at a successive time instant. Our goal is to estimate the angle ψ_j between $I_1(\mathbf{x})$ and $I_j(\mathbf{x})$, $j \in \{2, \ldots, m\}$, denoted by $\overline{\psi}_j$, with the phase correlation method. Note that owing to condition (4), that in the notation of this section reads:

$$I_j(\mathbf{x}) = I_1(\sigma_j \mathbf{R}^T(\psi_j)\,\mathbf{x} + \mathbf{x}_j),\ j \in \{2, \ldots, m\}, \quad (7)$$

meaning that the only transformations between $I_1(\mathbf{x})$ and $I_j(\mathbf{x})$ which are admitted belong to SIM(2), the phase correlation method will not provide correct angular estimates unless the camera-robot undergoes a *pure rotational motion*. To get around this limitation and deal with general 2-D camera displacements non-conforming to the similarity-transformation constraint, an *incremental approach* can be adopted. In fact, instead of directly computing the rotation angle between $I_1(\mathbf{x})$ and $I_j(\mathbf{x})$, we can progressively cumulate the angular increments between subsequent frames $k-1$ and k, for $k \in \{2, \ldots, j\}$. Being the difference between $I_{k-1}(\mathbf{x})$ and $I_k(\mathbf{x})$ "small", the following infinitesimal version of condition (7) is now guaranteed:

$$I_k(\mathbf{x}) = I_{k-1}\big((1 + d\sigma)\,\mathbf{R}^T(d\psi)\,\mathbf{x} + d\mathbf{x}\big),\ k \in \{2, \ldots, j\}, \quad (8)$$

where $d\psi$, $d\sigma$, and $d\mathbf{x}$ respectively denote infinitesimal increments of the rotation angle, scale factor, and image translation vector, and

$$\mathbf{R}(d\psi) = d\psi \begin{bmatrix} 0 & -\omega_\psi \\ \omega_\psi & 0 \end{bmatrix} + \mathbf{I}_{2\times 2},$$

is a differential rotation, being ω_ψ the angular velocity of the camera about the optical axis. The argument of I_{k-1} on the right-hand side of (8) is the (small-angle) four-parameter *Helmert* (or *Bursa-Wolf*) *transformation* [27].

For the reader's convenience, the pseudo-code that allows to implement our visual compass is reported in **Algorithm 1**, where the `PhaseCorr` function applies the phase correlation method to the transformed images M_{j-1} and M_j, and returns the estimated *incremental* angle $\overline{\psi}^j_{j-1}$, scale factor $\overline{\sigma}^j_{j-1}$, and image translation vector $\overline{\mathbf{x}}^j_{j-1}$. The angular estimate $\overline{\psi}_j$ is stored in the temporary variable ψ_{tmp}, whereas $\overline{\sigma}^j_{j-1}$ (ideally equal to zero, being the images of equal size) and $\overline{\mathbf{x}}^j_{j-1}$ are computed at each iteration to ensure enhanced flexibility to the algorithm, but they are not returned as output. Further details on the computation of the discrete log-polar Fourier transform of an omnidirectional image are provided in Sect. IV-B.

Algorithm 1: Pseudo-code of the omnidirectional visual compass ($I_1(\mathbf{x}), \ldots, I_m(\mathbf{x})$ are N × N grayscale images)

Input : The omnidirectional image $I_1(\mathbf{x})$ at the reference pose of the camera-robot.

Output: The estimated rotation angles $\overline{\psi}_2, \ldots, \overline{\psi}_m$ between the omnidirectional images $I_1(\mathbf{x})$ and $I_j(\mathbf{x})$, $j \in \{2, \ldots, m\}$.

Initialization;
Compute $M_1(\rho, \psi) = \big|\widehat{I}_1(\rho, \psi)\big|$;
$j := 2$;
$\psi_{\text{tmp}} := 0$;

Begin;
while *The camera-robot moves* **do**
  Take $I_j(\mathbf{x})$ at the current pose;
  Compute $M_j(\rho, \psi) = \big|\widehat{I}_j(\rho, \psi)\big|$;
  $[\overline{\psi}^j_{j-1}, \overline{\sigma}^j_{j-1}, \overline{\mathbf{x}}^j_{j-1}] =$ `PhaseCorr`(M_{j-1}, M_j);
  $\psi_{\text{tmp}} := \psi_{\text{tmp}} + \overline{\psi}^j_{j-1}$;
  $\overline{\psi}_j := \psi_{\text{tmp}}$;
  return $\overline{\psi}_j$;
  $j := j + 1$;
end

IV. Experimental validation

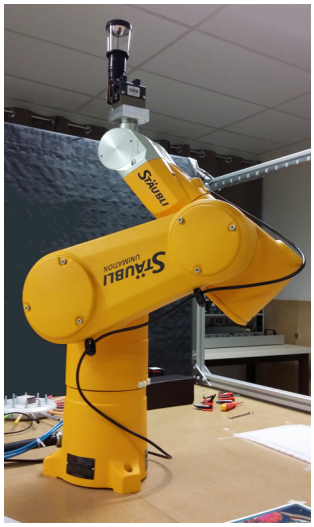
A. Description of the OVMIS dataset

A publicly-available dataset of omnidirectional images, called OVMIS ("*Omnidirectional Vision, MIS laboratory*"), has been created to evaluate the accuracy and robustness of our visual compass[2]. The catadioptric system used to generate the dataset consists of an IDS uEye UI-3370CP-C-HQ color camera screwed on a VStone VS-C450MR-TK objective. The resolution of the uEye camera is 620 × 620 pixels (fitted to mirror border), and the VStone objective (diameter 59 mm, height 136 mm) includes a hyperboloidal mirror (vertical field of view, around 70°; horizontal field of view, 360°) and perspective lenses. Two robotic platforms were considered in the OVMIS dataset: *1)* an industrial manipulator, and *2)* a wheeled robot.

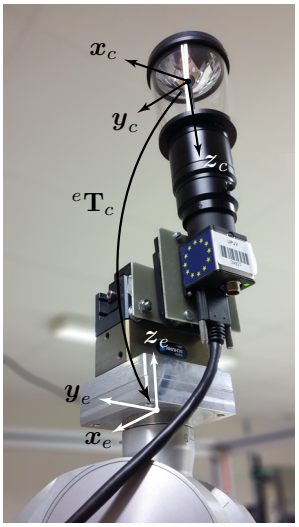
1) Stäubli manipulator: Since industrial manipulators guarantee high positioning accuracy and repeatability, they are well-suited for simulating the motion of a camera-robot and obtaining a precise ground truth. We mounted our catadioptric system on the end-effector of a 6 DOF Stäubli TX60 robot located in a 10.05 m × 7.03 m × 2.70 m room with neon lighting (see Figs. 1(a) and 1(b)). To account for the geometric constraints (maximum arm extension, 670 mm) and kinematic singularities of the robot, we limited ourselves to camera poses within a horizontal disk of radius 450 mm around the first (vertical) axis of the manipulator. To sample this disk,

[2] OVMIS dataset (2.72 GB): http://mis.u-picardie.fr/~g-caron/pub/data/OVMIS_dataset.zip

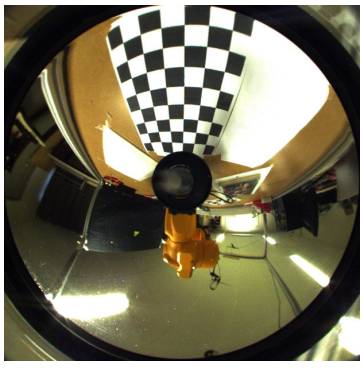
located 603.3 mm above the platform where the robot is fixed (cf. Fig. 1(a)), we chose 25 Collection Points (CPs) on 8 rays with an angular spacing of 45°: on each ray, three locations, at a distance of 100, 250 and 450 mm, were considered (see the black dots in Fig. 2). At each CP, the camera was rotated by 360° about its optical axis $\boldsymbol{z}_c$ with a step size of 2.5°, yielding 144 images per CP. For each of the $25 \times 144 = 3600$ PNG images, we recorded the 3-D coordinates of the origin of the camera frame $\{\boldsymbol{x}_c, \boldsymbol{y}_c, \boldsymbol{z}_c\}$ and its orientation with respect to the base frame of the robot. We positioned the Stäubli manipulator on the 25 CPs by running a VAL3 program (that behaves as a server) on the CS8C controller. Neither master gain nor gamma correction was considered for the camera to reduce image noise. However, to get images with enough luminance to be usable by our algorithm, we set the exposure time to 200 ms, leading to a frame rate of 5 fps. Such a low frame rate was not an issue in practice, since the camera was stopped before every acquisition to avoid image blur.



(a) Stäubli TX60 manipulator inside its cell.



(b) Hypercatadioptric camera mounted on the end-effector of the manipulator.



(c) One of the images of the checkerboard pattern used for the calibration of the catadioptric system on the Stäubli robot.

(d) Pioneer 3-AT robot equipped with the hypercatadioptric camera.

Fig. 1. *Robotic platforms used for the generation of the OVMIS dataset.* In (b), ${}^e\mathbf{T}_c$ denotes the 3-D rigid transformation between the camera's frame $\{\boldsymbol{x}_c, \boldsymbol{y}_c, \boldsymbol{z}_c\}$ and the end-effector's frame $\{\boldsymbol{x}_e, \boldsymbol{y}_e, \boldsymbol{z}_e\}$.

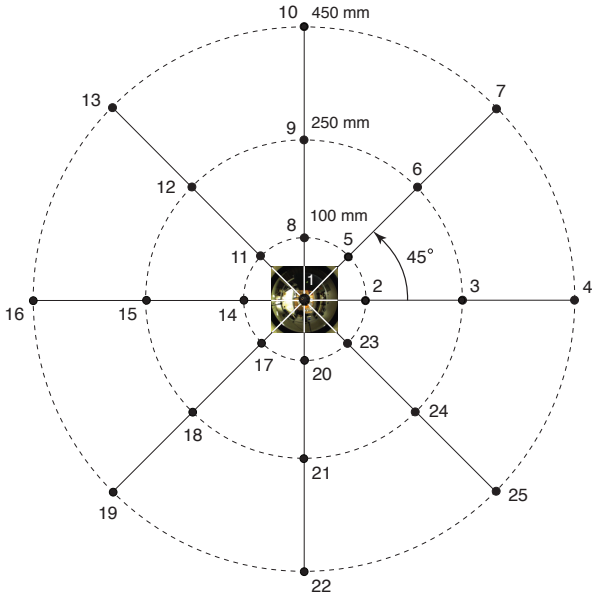
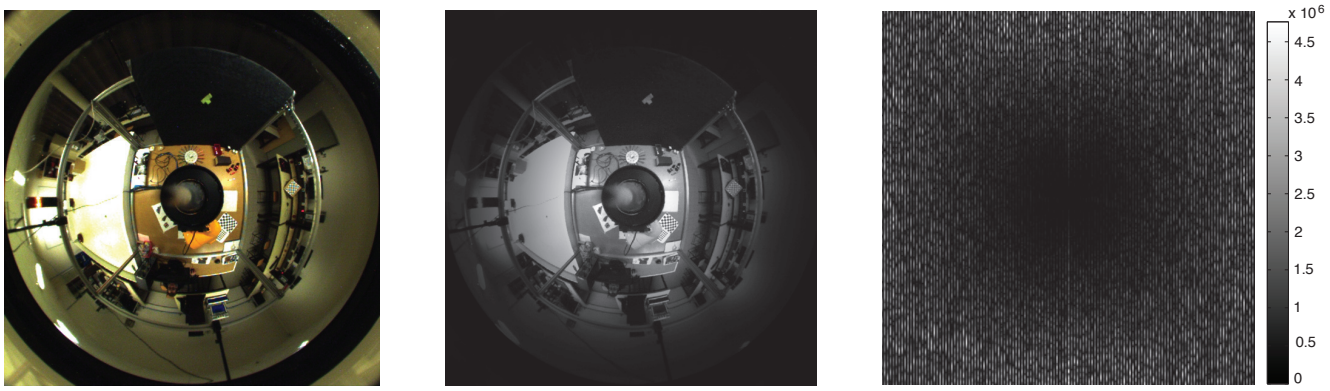


Fig. 2. Graphical illustration of the structure of the dataset generated with the Stäubli manipulator. It consists of 3600 images taken at 25 CPs (black dots) lying on 8 rays with a regular spacing of 45°.

For the intrinsic calibration of our catadioptric system, we leveraged Barreto's unified central projection model [28] and used HYSCAS [29]. Note that *only* the coordinates of the principal point, $[\mathrm{u}_0, \mathrm{v}_0]^T = [304.71, 308.76]^T$ pixels, were actually exploited for the validation of our visual compass (see Sect. IV-C). We fed the calibration algorithm with a set of image points of a checkerboard pattern taken at 6 different camera poses, the first of which is shown in Fig. 1(c). Let ${}^{c_i}\mathbf{T}_p$ be the 3-D rigid transformation between the frame attached to the checkerboard pattern, "p", and the camera's frame at pose "c_i", $i \in \{1, \ldots, 6\}$. Since the camera is rigidly attached to the robot arm, the end-effector's pose corresponding to each of these camera poses should also be considered. Let then ${}^{e_i}\mathbf{T}_b$ be the rigid transformation between the robot's base frame, "b", and the end-effector's frame at pose "e_i". ${}^{c_i}\mathbf{T}_p$ and ${}^{e_i}\mathbf{T}_b$, $i \in \{1, \ldots, 6\}$, were given as input to the Tsai and Lenz's algorithm (in the ViSP-library's implementation [30]) for the estimation of the extrinsic camera parameters. We thus obtained ${}^e\mathbf{T}_c$, the transformation between the camera's frame $\{\boldsymbol{x}_c, \boldsymbol{y}_c, \boldsymbol{z}_c\}$, and the end-effector's frame $\{\boldsymbol{x}_e, \boldsymbol{y}_e, \boldsymbol{z}_e\}$ (see Fig. 1(b)). This transformation was essential for precisely positioning the camera in the 25 CPs with the CS8C controller of the Stäubli robot.

2) Pioneer robot: The catadioptric system described in Sect. IV-A1 was also mounted on an Adept MobileRobots Pioneer 3-AT robot. In order to have an elevated viewpoint, it was placed on top of a vertical support, 134 cm above the ground (110 cm above the robot, see Fig. 1(d)). We used a two-axis bubble level to orient the optical axis of the camera perpendicularly to the ground plane. We did not estimate the rigid transformation between the camera's and robot's frames



(a) (b) (c)

Fig. 3. *Computation of the log-polar Fourier transform of a hypercatadioptric image*. (a) Recentered input image (from the Stäubli dataset); (b) Application of the 2-D Hamming window (9) to image (a) converted to grayscale; (c) High-pass filtered[3] version of the magnitude of the log-polar Fourier transform of the image in (b): note that for easy visualization, the zero-frequency component is shown in the middle of the spectrum.

as with the Stäubli manipulator, but we aligned their vertical axes by rotating the platform 360°, and by ensuring that the omnidirectional images taken at 0° and at 360° matched precisely. Although simple, this procedure was adequate for our experimental validation (see Sect. IV-D). The Pioneer has an integrated single-axis (yaw) gyroscopic sensor for increased rotational accuracy, and the robot controller automatically incorporates the gyroscopic corrections into the odometry-based estimates (from wheel encoders). In order to obtain reproducible results, we precisely synchronized the image acquisition system, and the module that logs the pose estimates of the robot via the *ARIA* library (our ground truth). Three scenarios were considered with the Pioneer robot:

- *Scenario 1*: Indoor, pure rotation ($m = 160$, total time 49.69 s),
- *Scenario 2*: Outdoor, pure rotation ($m = 156$, total time 50.22 s),
- *Scenario 3*: Outdoor, rotation and translation ($m = 318$, total time 160.99 s, length of the trajectory 25.27 m, average robot speed 0.187 m/s).

In Scenarios 1 and 2, we applied a constant angular velocity to the skid-steering robot in order to have a regular rotational step (for a pose estimate to be logged, the step must be of at least 2°). The images in Scenario 1 were recorded in a laboratory setting, while those in Scenarios 2 and 3 in Bois Bonvallet, a 4.5 ha public park in Amiens, France. Except for a lamppost and a paved footpath, few recognizable landmarks are present in Scenarios 2 and 3. In Scenario 3, the robot was manually driven along a figure-of-eight trajectory (see Fig. 5(b)): its forward and angular velocities were limited to 0.2 m/s and 12 deg/s, respectively. The camera was re-calibrated with HYSCAS [29], yielding an estimated principal point $[\mathrm{u}_0,\, \mathrm{v}_0]^T = [319.70,\, 310.94]^T$ pixels. The exposure time of the camera was left free but limited to 50 ms, yielding a frame rate of 20 fps.

B. Computation of the discrete log-polar Fourier transform

In order to implement our visual compass algorithm, we need to compute the discrete log-polar Fourier transform of a grayscale omnidirectional image. To this end, we first crop the images so that their geometric center coincides with the estimated principal point $[\mathrm{u}_0,\, \mathrm{v}_0]^T$ (see Fig. 3(a)). We then apply a 2-D Hamming window to the images (see Fig. 3(b)):

$$\mathrm{w}_{\mathrm{hm}}(x,\, y) \,=\, \mathrm{w}_{\mathrm{hm}}(x)\, \mathrm{w}_{\mathrm{hm}}(y), \tag{9}$$

where $\mathrm{w}_{\mathrm{hm}}(x) = \alpha - (1-\alpha)\cos(2\pi x/(\mathrm{N}-1))$, $x \in \mathcal{S} = \{0, \ldots, \mathrm{N}-1\}$, is the univariate Hamming window with length N and $\alpha = 0.54$. Other windows (Kaiser's, Hann's, Blackman's [31, Sect. 6.3]) were tested: however, we experimentally observed that the type of window selected had a minimal impact on the performance of the phase correlation method. Finally, for the conversion from rectangular to log-polar coordinates, we used a bicubic interpolation for the angles and distances, and a nearest-neighbor interpolation for the scale factor (see Fig. 3(c)). As we will see in the sequel, the size N_{lp} of the log-polar image is a tuning parameter, trading off accuracy and computational complexity.

C. Experimental results: Stäubli manipulator

The omnidirectional visual compass was used to estimate the yaw angle between the first image, $I_1(\mathbf{x})$, and each of the subsequent images $I_j(\mathbf{x})$, $j \in \{2, \ldots, m\}$, $m = 144$, at the 25 CPs of the Stäubli dataset. Note that since *pure rotations* are considered in this case, condition (7) is fulfilled and our visual compass algorithm can be implemented in an "absolute fashion" (i.e. without cumulating the incremental estimates of the yaw angles). Our Matlab implementation of the phase correlation method produces a list of candidate rotation angles sorted in decreasing order according to the height of the rotation-scaling correlation peaks and translation correlation peak. We limited ourselves to 3 candidates trading off speed for accuracy, and retained the yaw angle $\overline{\psi}_j$ corresponding to the top-ranked value in the list. By considering 256-by-256 log-polar images, it took an average of 1.53 s (over ten trials) to compute the 3 candidate angles on a MacBook Pro with 2.4 GHz Intel Core i7 processor and 8 GB RAM (note that all the computation times in this paper are wall-clock times). To gain more insight into the performance of the

[3]The high-pass filter is $H(u,\, v) = (1 - \eta_u \eta_v)(2 - \eta_u \eta_v)$ where $\eta_u = \cos(\pi(-\frac{1}{2} + \frac{u}{\mathrm{N}_{lp}-1}))$ with $u \in \{0, \ldots, \mathrm{N}_{lp}-1\}$.

proposed algorithm, we compared $|\psi_j - \overline{\psi}_j|$ with $|\psi_j - \overline{\psi}_j^{\text{PHO}}|$, $j \in \{2, \ldots, m\}$, where the estimate $\overline{\psi}_j^{\text{PHO}}$ is provided by a *photometric algorithm* exploiting the pixel intensity of the images. $\overline{\psi}_j^{\text{PHO}}$ is computed by brute-force search, i.e.

$$\overline{\psi}_j^{\text{PHO}} = \underset{\psi_j \in \{0^\circ, \tau, \ldots, 360^\circ - \tau\}}{\operatorname{argmin}} \sum_{\mathbf{x} \in \mathcal{S}^2} \left[I_1(\mathbf{R}^T(\psi_j)\mathbf{x}) - I_j(\mathbf{x}) \right]^2, \tag{10}$$

where the step size $\tau = 0.5^\circ$ (obviously, the smaller τ the smaller the estimation error, but the higher the computational burden). An annular mask centered at the black hole was used to select those parts of an omnidirectional image whose intensity is not constant, and which are thus informative for estimating ψ_j. Table I(a) reports the mean, standard deviation, minimum and maximum values of the magnitude of the estimation errors for the *phase correlation* (COR) and *photometric* (PHO) approach in the 25 CPs of the Stäubli dataset. The former method guarantees an average error of $0.46^\circ \pm 0.32^\circ$ over the 3600 images of the dataset, with an average maximum error of 1.44° (see the last row of Table I(a)). The average error of the photometric algorithm is slightly larger, but the average maximum error is smaller (this is, however, an artifact imputable to our half-degree angular quantization τ). Note that the overall complexity of the phase correlation approach is dominated by the computation of the discrete log-polar Fourier transform. Thus, for a fixed j, the complexity is at least $O(N^2 \log N)$ (cf. Cooley-Tukey FFT algorithm [31, Ch. 5]). On the other hand, for a fixed j, the complexity of the photometric approach is $O(360\,\tau^{-1}N^2)$. Practically speaking, to solve problem (10) for a fixed j, it took on average (ten trials) 62.68 s in Matlab, under the previous conditions. In fact, by searching over 720 images, the photometric visual compass is about 41 times slower than the phase correlation algorithm. As a final comparison, Table I(b) reports the statistics of the magnitude of the estimation errors of the method in [5] in the first 4 CPs. This method detects 3-D parallel lines on the unit sphere of equivalence (it thus needs a *fully-calibrated* camera), and it computes the vanishing points from which the roll and pitch angles are deduced. We adapted [5] to estimate the yaw-angle of the camera, and we worked with the same Matlab implementation of the authors to identify an optimal set of threshold parameters. It came as no surprise that [5] is less accurate than the dense visual compasses, since only few parallel lines are present in the images (two bundles of 6 to 8 straight lines). In addition, the average computation time for a fixed j is 4.33 s, i.e. 2.83 times longer than that of the phase correlation method.

(a)	COR				PHO			
CP	Mean	Std	Min	Max	Mean	Std	Min	Max
1	0.41	0.29	0.00	1.41	0.46	0.33	0.00	1.00
2	0.37	0.24	0.00	1.09	0.59	0.50	0.00	1.50
3	0.39	0.27	0.00	1.09	0.47	0.43	0.00	1.00
4	0.44	0.31	0.00	1.41	0.49	0.44	0.00	1.00
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
25	0.68	0.46	0.00	1.88	0.58	0.34	0.00	1.00
	0.46	**0.32**	**0.00**	**1.44**	**0.48**	**0.38**	**0.00**	**1.06**

(b)	Bazin *et al.* [5]			
CP	Mean	Std	Min	Max
1	1.72	1.23	0.09	11.69
2	1.79	4.54	0.00	37.40
3	2.06	6.02	0.00	48.22
4	5.83	4.25	0.11	37.24

TABLE I
(a) Magnitude of the estimation errors (in degrees) of the *phase correlation* (COR) and *photometric* (PHO) approach in the 25 CPs of the Stäubli dataset. The last row of the table (boldface) reports the average values of the eight columns; (b) Magnitude of the estimation errors (in degrees) of the approach in [5] in the first 4 CPs of the Stäubli dataset.

Remark 1: Note that in order to determine the *maximum* translational displacement of the camera allowed by the phase correlation algorithm for a given level of accuracy, we can relate the statistics of the magnitude of the estimation error at CP 2 (see the second row of Table I(a)), with those obtained by estimating the yaw-angle between the first image at CP 1 and all the images at CP 2. Under a linear growth model, we can impose, for example, that the mean value in the latter case (8.04°), is within 3 sigma of the mean value in the former case (i.e. $0.37 + 3 \times 0.24 = 1.09^\circ$), thus obtaining a maximum translational displacement of $(10 \times 1.09)/8.04 = 1.35$ cm. As a comparison, with 1024-by-1024 log-polar images and with the photometric approach, the maximum translational displacement becomes $(10 \times 1.02)/4.19 = 2.43$ cm and $(10 \times 2.09)/1.79 = 11.68$ cm, respectively. ◇

D. Experimental results: Pioneer robot

To further evaluate the robustness of our approach, we ran **Algorithm 1** in the three scenarios described in Sect. IV-A2. Because of the incremental nature of the algorithm, we increased the sensibility of the phase correlation method by considering 1024-by-1024 log-polar images. The price to pay was a slight increase of the average computation time (less than 9% over ten trials, for Scenario 3). Fig. 4 reports the results obtained in Scenarios 1 and 2: in particular, Figs. 4(a), 4(d), Figs. 4(b), 4(e), and Figs. 4(c), 4(f), respectively show the robot setups, the first image of each scenario, and the incremental estimation errors $\psi_{j-1}^j - \overline{\psi}_{j-1}^j$ (blue) and $\psi_{j-1}^j - \overline{\psi}_{j-1}^{j,\text{PHO}}$ (red) over the 360° angular range, for $j \in \{2, \ldots, m\}$. We computed $\overline{\psi}_{j-1}^{j,\text{PHO}}$ as (cf. equation (10)):

$$\overline{\psi}_{j-1}^{j,\text{PHO}} = \underset{\psi_{j-1}^j \in \Omega}{\operatorname{argmin}} \sum_{\mathbf{x} \in \mathcal{S}^2} \left[I_{j-1}(\mathbf{R}^T(\psi_{j-1}^j)\mathbf{x}) - I_j(\mathbf{x}) \right]^2,$$

where $\Omega = \{-6^\circ + \tau, -6^\circ + 2\tau, \ldots, 6^\circ - \tau\}$ and $\tau = 0.1^\circ$. The first two rows of Table II report the statistics of the magnitude of the incremental estimation errors in Scenarios 1 and 2. The minimum, maximum and end-value (i.e. for $j = m$), of $|\psi_j - \overline{\psi}_j|$ and $|\psi_j - \overline{\psi}_j^{\text{PHO}}|$ are, instead, 0.06°, 8.64°, 8.64° and 0.03°, 2.26°, 1.96° in Scenario 1, and 0.25°, 6.09°, 2.97° and 0.12°, 2.91°, 2.72° in Scenario 2. Finally, Fig. 5 refers to Scenario 3: Fig. 5(a) reports the first image of the dataset, and Fig. 5(b) the trajectory of the Pioneer. Fig. 5(c) shows the yaw angles of the robot estimated by the phase correlation ($\overline{\psi}_j$, blue) and photometric approach ($\overline{\psi}_j^{\text{PHO}} = \sum_{k=2}^{j} \overline{\psi}_{k-1}^{k,\text{PHO}}$, red), against the ground truth (ψ_j, black). Note that despite

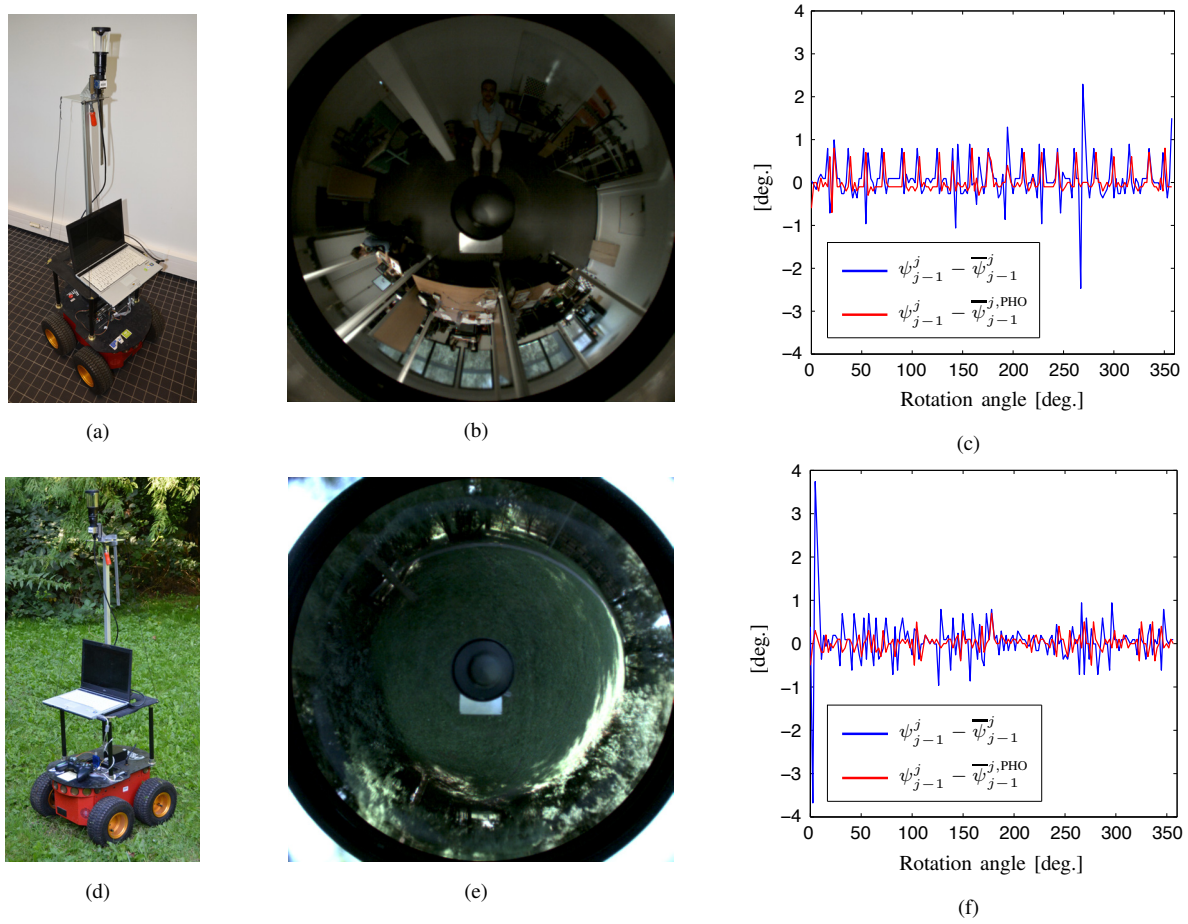


Fig. 4. *Scenario 1 (top row) and Scenario 2 (bottom row).* (a), (d) Robot setups; (b), (e) Initial images; (c), (f) Estimation errors $\psi^j_{j-1} - \overline{\psi}^j_{j-1}$ (blue) and $\psi^j_{j-1} - \overline{\psi}^{j,\mathrm{PHO}}_{j-1}$ (red), in degrees, over the 360° angular range.

	COR				PHO			
Scen.	Mean	Std	Min	Max	Mean	Std	Min	Max
1	0.31	0.39	0.00	2.47	0.17	0.21	0.00	0.78
2	0.32	0.46	0.00	3.74	0.14	0.14	0.00	0.70
3	0.60	0.67	0.00	3.47	0.30	0.26	0.00	1.40

TABLE II
Statistics of $|\psi^j_{j-1} - \overline{\psi}^j_{j-1}|$ and $|\psi^j_{j-1} - \overline{\psi}^{j,\mathrm{PHO}}_{j-1}|$ (degrees) of the *phase correlation* (COR) and *photometric* (PHO) approach in Scenarios 1-3.

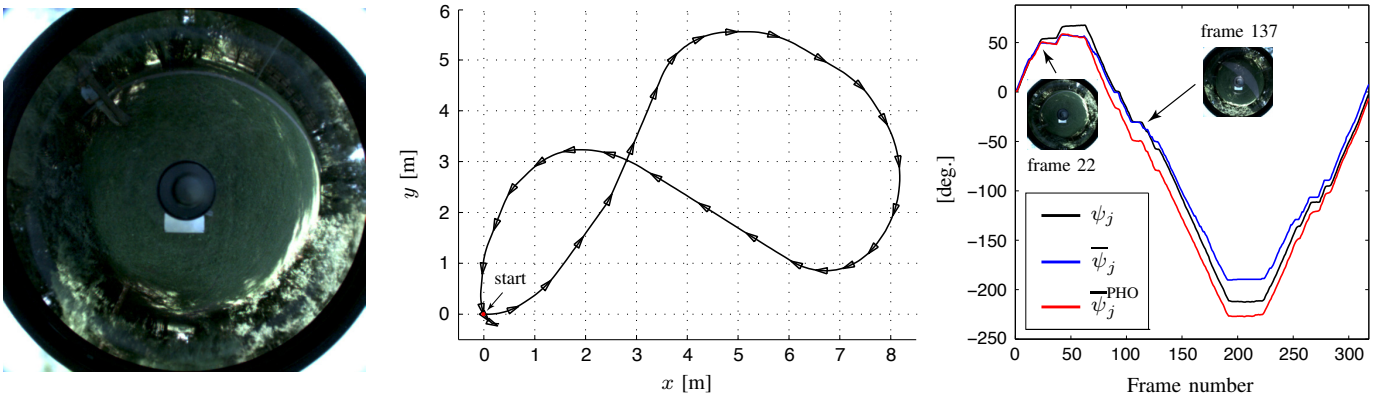
the drift at frames 22 and 137, caused by the oscillations of the catadioptric system when the Pioneer negotiated the lawn from the footpath (see Fig. 5(c)), small incremental estimation errors are obtained, as reported in the third row of Table II. The minimum and end-value of $|\psi_j - \overline{\psi}_j|$ and $|\psi_j - \overline{\psi}^{\mathrm{PHO}}_j|$ are relatively small as well: 0.01°, 9.63° and 0.18°, 4.47°, respectively.

Discussion: Note that in Scenarios 2 and 3, challenging experimental conditions were considered[4]. The robot traversed *uneven terrain* (resulting in a non-planar camera displacement) under *variable illumination conditions* (moving cast shadows of the trees). Moreover, in the Stäubli dataset, we experienced *partial image occlusions* due to the spatial reconfiguration of the robot arm (see frames 14, 15 and 70, 71 in CP 1). In spite of these important sources of error and of being *partially agnostic* of the internal camera parameters, our visual compass turned out to be *accurate* and *robust* against unexpected image changes. Moreover, it was between one and two orders of magnitude *faster* than the photometric algorithm, and almost three times *faster* than [5]. It is finally worth pointing out here that the performance of [5] was unsatisfactory in the *non-structured environments* of Scenarios 2 and 3, because of the almost total absence of straight lines in the images. The methods in [3], [4] rely on line images too, and they are also expected to perform poorly in these scenarios.

[4]The videos of the experiments are available in the multimedia attachment accompanying this submission and at the address: http://mis.u-picardie.fr/~g-caron/pub/data/OVMIS_videos.zip

V. Conclusion and Future Work

In this paper, we have designed and experimentally validated on different robotic platforms, a dense omnidirectional visual compass based on the phase correlation method in the spectral domain. We have also described a new publicly-available dataset of hypercatadioptric images, called OVMIS, which was instrumental in evaluating our approach and in comparing it with competing methods.



(a) (b) (c)

Fig. 5. *Scenario 3.* (a) Initial image; (b) Figure-of-eight trajectory of the Pioneer robot (in order to have a time reference, we marked the robot with a triangle every 10 frames or about every 5 s); (c) Yaw angles estimated by the phase correlation (blue) and photometric approach (red), against the ground truth (black).

This work opens several directions for further research. Although our results are encouraging, an effort is still required to achieve real-time performance via machine-specific optimized implementations of **Algorithm 1**. Working towards a more sophisticated robot-localization unit, we also aim at incorporating a vision-based loop-detection module into our algorithm. Finally, we plan to extend our approach to 3-D camera motions for estimating the attitude of an aerial robot (e.g. a quadrotor). To this end, we are currently considering the possibility to adapt the volume registration method based on the 3-D pseudo-polar Fourier transform presented in [32], to a visual gyroscope.

REFERENCES

[1] D. Scaramuzza and F. Fraundorfer. Visual Odometry - Part I : The First 30 Years and Fundamentals. *IEEE Rob. Autom. Mag.*, 18(4):80–92, 2011.

[2] G. Caron, E. Mouaddib, and E. Marchand. 3D model based tracking for omnidirectional vision: A new spherical approach. *Robot. Autonom. Syst.*, 60(8):1056–1068, 2012.

[3] G.L. Mariottini, S. Scheggi, F. Morbidi, and D. Prattichizzo. An accurate and robust visual-compass algorithm for robot-mounted omnidirectional cameras. *Robot. Autonom. Syst.*, 60(9):1179–1190, 2012.

[4] J. Bermudez-Cameo, L. Puig, and J.J. Guerrero. Hypercatadioptric line images for 3D orientation and image rectification. *Robot. Autonom. Syst.*, 60(6):755–768, 2012.

[5] J.-C. Bazin, C. Demonceaux, P. Vasseur, and I. Kweon. Rotation estimation and vanishing point extraction by omnidirectional vision in urban environment. *Int. J. Robot. Res.*, 31(1):63–81, 2012.

[6] D. Churchill and A. Vardy. An Orientation Invariant Visual Homing Algorithm. *J. Intell. Robot. Syst.*, 71(1):3–29, 2013.

[7] M. Irani and P. Anandan. About Direct Methods. In *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science*, pages 267–277. Springer, 2000.

[8] F. Labrosse. The Visual Compass: Performance and Limitations of an Appearance-Based Method. *J. Field Robot.*, 23(10):913–941, 2006.

[9] D. Scaramuzza and R. Siegwart. Appearance-Guided Monocular Omnidirectional Visual Odometry for Outdoor Ground Vehicles. *IEEE Trans. Robot.*, 24(5):1015–1026, 2008.

[10] T. Pajdla and V. Hlaváč. Zero Phase Representation of Panoramic Images for Image Based Localization. In *Computer Analysis of Images and Patterns*, volume 1689 of *Lecture Notes in Computer Science*, pages 550–557. Springer, 1999.

[11] E. Menegatti, T. Maeda, and H. Ishiguro. Image-based memory for robot navigation using properties of omnidirectional images. *Robot. Autonom. Syst.*, 47(4):251–267, 2004.

[12] W. Stürzl and H. Mallot. Efficient visual homing based on Fourier transformed panoramic images. *Robot. Autonom. Syst.*, 54(4):300–313, 2006.

[13] A. Kyatkin and G. Chirikjian. Pattern Matching as a Correlation on the Discrete Motion Group. *Comput. Vis. Image Und.*, 74(1):22–35, 1999.

[14] A. Makadia and K. Daniilidis. Direct 3D-Rotation Estimation from Spherical Images via a Generalized Shift Theorem. In *Proc. IEEE Conf. Comp. Vis. Pattern Rec.*, volume 2, pages 217–224, 2003.

[15] A. Makadia, C. Geyer, and K. Daniilidis. Correspondence-free structure from motion. *Int. J. Comput. Vision*, 75(3):311–327, 2007.

[16] H.E. Benseddik, H. Hadj-Abdelkader, B. Cherki, and S. Bouchafa. Camera Rotation Estimation using 3D Mesh Surfaces Representation of Spherical Images. In *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst.*, pages 2514–2520, 2016.

[17] B. Zitová and J. Flusser. Image registration methods: a survey. *Image vision comput.*, 21(11):977–1000, 2003.

[18] A. Alba, J.F. Vigueras-Gomez, E.R. Arce-Santana, and R.M. Aguilar-Ponce. Phase correlation with sub-pixel accuracy: A comparative study in 1D and 2D. *Comput. Vis. Image Und.*, 137:76–87, 2015.

[19] E. De Castro and C. Morandi. Registration of Translated and Rotated Images Using Finite Fourier Transforms. *IEEE Trans. Pattern Anal.*, 9(5):700–703, 1987.

[20] Y. Keller, Y. Shkolnisky, and A. Averbuch. The Angular Difference Function and its Application to Image Registration. *IEEE Trans. Pattern Anal.*, 27(6):969–976, 2005.

[21] T. Kazik and A.H. Göktoğan. Visual Odometry Based on the Fourier-Mellin Transform for a Rover Using a Monocular Ground-Facing Camera. In *Proc. IEEE Int. Conf. Mechatronics*, pages 469–474, 2011.

[22] R. Cassinis, D. Duina, S. Inelli, and A. Rizzi. Unsupervised matching of visual landmarks for robotic homing using Fourier-Mellin transform. *Robot. Autonom. Syst.*, 40(2):131–138, 2002.

[23] P. Checchin, F. Gérossier, C. Blanc, R. Chapuis, and L. Trassoudaine. Radar Scan Matching SLAM Using the Fourier-Mellin Transform. In *Field and Service Robotics*, volume 62 of *Springer Tracts in Advanced Robotics*, pages 151–161. Springer, 2010.

[24] R.C. Gonzalez and R.E. Woods. *Digital Image Processing*. Prentice Hall, 2nd edition, 2002.

[25] G.S. Chirikjian and A.B. Kyatkin. *Engineering Applications of Noncommutative Harmonic Analysis: with Emphasis on Rotation and Motion Groups*. CRC press, 2001.

[26] P. Stoica and R.L. Moses. *Introduction to Spectral Analysis*. Prentice Hall, 1997.

[27] G.A. Watson. Computing Helmert transformations. *J. Comput. Appl. Math.*, 197(2):387–394, 2006.

[28] J. Barreto. A unifying geometric representation for central projection systems. *Comput. Vis. Image Und.*, 103(3):208–217, 2006.

[29] G. Caron and D. Eynard. Multiple Camera Types Simultaneous Stereo Calibration. In *Proc. IEEE Int. Conf. Robot. Automat.*, pages 2933–2938, 2011.

[30] E. Marchand, F. Spindler, and F. Chaumette. ViSP for Visual Servoing: A Generic Software Platform with a Wide Class of Robot Control Skills. *IEEE Rob. Autom. Mag.*, 12(4):40–52, 2005.

[31] B. Porat. *A Course in Digital Signal Processing*. Wiley, 1997.

[32] Y. Keller, Y. Shkolnisky, and A. Averbuch. Volume registration using the 3-D pseudopolar Fourier transform. *IEEE Trans. Signal Proces.*, 54(11):4323–4331, 2006.