Robotica (2012) volume 30, pp. 865–878. © Cambridge University Press 2011
doi:10.1017/S026357471100110X

Combined visual odometry and visual compass for off-road mobile robots localization

Ramon Gonzalez†*, Francisco Rodriguez†, Jose Luis Guzman†, Cedric Pradalier‡ and Roland Siegwart‡

†*Department of Languages and Computation, University of Almería, Almería, Spain*
‡*Autonomous Systems Lab, ETH Zurich, Zurich, Switzerland*

(Accepted September 1, 2011. First published online: October 5, 2011)

SUMMARY
In this paper, we present the work related to the application of a visual odometry approach to estimate the location of mobile robots operating in off-road conditions. The visual odometry approach is based on template matching, which deals with estimating the robot displacement through a matching process between two consecutive images. Standard visual odometry has been improved using visual compass method for orientation estimation. For this purpose, two consumer-grade monocular cameras have been employed. One camera is pointing at the ground under the robot, and the other is looking at the surrounding environment. Comparisons with popular localization approaches, through physical experiments in off-road conditions, have shown the satisfactory behavior of the proposed strategy.

KEYWORDS: Mobile robots; Robot localization; Navigation; Computer vision; Service robots.

1. Introduction

Robot localization is defined as the process in which a mobile robot determines its current position and orientation relative to an inertial reference frame.[1] In the context of off-road mobile robots, localization techniques have to deal with particular features of off-road conditions, such as a noisy environment (vibrations when the robot moves, disturbance sources, etc.), changing lighting conditions, high degrees of slip, etc.

One of the most popular solutions for the mobile robotics community is wheel-based odometry (or odometry).[2,3] This technique is considered as relative or local localization, that is, robot location is incrementally calculated from an initial point. Odometry employs simple geometric equations (mobile robot kinematics) with wheel encoders that provide angular velocities of the wheels. Then the position and orientation are calculated by integrating these velocities. The main drawbacks of using wheel-based odometry are as follows: (i) Since encoder measurements are integrated, the noise is also integrated, and thus it causes an unbounded growth of errors along time and distance; and (ii) it is based on the assumption that wheel revolutions can be converted into linear displacement relative to the terrain, this assumption is limited in slip conditions.

An attractive alternative is the use of absolute or global techniques. These techniques determine the position of the robot with respect to a global reference frame, for instance, using beacons or landmarks.[4–6] The most popular technique is Global Positioning System (GPS), which is based on satellite signals to determine the absolute position of an object on the Earth (longitude, latitude, and altitude).[7] The main drawbacks of absolute techniques are as follows: (i) It requires a costly installation of beacons/markers on the area where the robot operates, and (ii) the mobile robot can only navigate over the area in which landmarks are located. Furthermore, the particular problems related to GPS are as follows: (i) The satellite signal is lost in partially covered areas (nearby trees, buildings, etc.); (ii) it cannot be used in covered areas (greenhouses, mines, etc.) or in space exploration;[7] and (iii) the consumer-grade GPS provides poor accuracy (several meters). Although more expensive solutions such as Differential GPS (DGPS) or Real-Time Kinematics GPS (RTK-GPS) can improve that accuracy significantly.

On the other hand, techniques that estimate robot location using visual information (images) are being successfully applied to off-road mobile robots, especially in space robotic exploration missions.[8–11] One of the most popular approaches is *visual odometry*, which is defined as the incremental online estimation of robot motion from image sequences using an on-board camera.[12,13]

In this paper, we present the work related to the application of a visual odometry approach based on template matching technique to estimate the location of a mobile robot operating in off-road conditions. Standard visual odometry has been improved using visual compass method for orientation estimation purposes. This issue constitutes the main contribution of this paper. Comparisons of this strategy with other localization techniques (e.g., wheel-based odometry) through physical experiments show a satisfactory behavior of the proposed scheme. Here, the results obtained using a tracked mobile robot available at the University of Almería (Spain), called *Fitorobot*,[14] are shown. Furthermore, successful results have also been obtained using the *CRAB rover* available at the ASL, ETH Zürich (Switzerland).[15] Videos related to the physical experiments are

* Corresponding author. E-mail: rgonzalez@ual.es. This work has been supported in part by the Spanish CICYT under grant DPI2010-21589-C05-04.

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

available at http://www.ual.es/personal/rgonzalez/videosVO.htm.

The paper is organized as follows. Section 2 describes the methodology to estimate the robot location combining the position obtained using visual odometry and the orientation using the visual compass method. Section 3 is devoted to implementation issues. Physical experiments are discussed in Section 4. Finally, conclusions and discussions about physical experiments are detailed in Section 5.

2. Methodology

Visual odometry constitutes a straightforward-cheap method to estimate the robot location.[9,16,17] A single consumer-grade camera can replace a typical expensive sensor suite (encoders, IMU, GPS, etc.). It is especially appropriate for off-road applications, since the visual information is used to estimate the actual velocity of the robot, thereby minimizing slip phenomena.[17] The main limitations of vision-based techniques are mainly related to the light and imaging conditions (i.e., terrain appearance, camera parameters, etc.) and the computational cost.

Generally, there are two ways to estimate the location of a mobile robot using the visual odometry paradigm. The most popular method is called *optical flow*.[13,18,19] It is based on tracking distinctive features between successively acquired images.[18] In this case, an image is matched with the previous one by individually comparing each feature on them and finding candidate matching features based on the Euclidean distance of their feature vectors. Afterwards, the velocity vector between these pairs of points is calculated and the displacement is obtained by using these vectors.[18] Optical flow is especially advisable for textured scenarios, such as urban and rough environments.[8,11,20] This approach has been tested using single,[16] stereo,[21] and omnidirectional cameras.[20]

A slightly different approach is the *template matching* method.[22–24] It avoids the problem of finding and tracking features, and instead it looks at the change in the appearance of the world (images). For that purpose, it takes a template or patch from an image and tries to match it in the previous image. The main difference with optical flow is that now no identification or tracking of features are involved, and there is no need to measure image velocities at different locations.[25] The appearance-based method has been successfully applied employing single[26,27] and omnidirectional cameras.[24]

The main difference between the optical flow and template matching approaches is that when the scene is low-textured, the number of detected and tracked features (single patterns) is low, which can lead to poor accuracy of motion estimate.[8] This fact means that optical flow can fail on almost featureless scenarios (such as sandy soils, urban floors, etc.) where images with few high gradients are grabbed. On the other hand, the template matching approach works properly in low texture scenarios, since a larger pattern (template) is employed, and, therefore, the probability of a successful matching is increased.[24]

Previous discussion motivates why the template matching method has been selected in this work. However, it is important to remark that if the matching process fails (false matches), the robot motion estimate can become degraded. In order to minimize this shortcoming, especially undesirable estimating robot orientation, a second camera is added. This solution is inspired by two recent works, in which a method called *visual compass* was proposed to estimate rotational information from omnidirectional cameras.[20,24] The visual compass technique is based on the use of a camera mounted vertical to the ground on a mobile robot. Then, a pure rotation on its vertical axis results in a single column-wise shift of the appearance in the opposite direction. In this way, the rotation angle is retrieved by matching a template between the current image (after rotation) and the previous one (before rotation).[20]

In this section, the steps carried out to estimate the robot location using visual odometry based on the template matching method are explained. Our strategy takes two image sequences as input. One image sequence comes from a single standard camera pointing at the ground under the robot, and the second one comes from a camera looking at the environment. The former is employed to estimate the robot longitudinal displacement (Section 2.2), and the latter is employed to estimate the robot orientation (Section 2.3). Firstly, the mathematical formulation of template matching is briefly described in the following subsection.

2.1. Template matching

The template matching method is defined as the process of locating the position of a sub-image inside a larger image. The sub-image is called the *template* and the larger image is called the *search area*.[22,23] This process involves shifting the template over the search area and computing the similarity between the template and a window in the search area. This is achieved by calculating the integral of their product. When the template matches, the value of the integral is maximized. There are several methods to address the template matching, see refs [28 and 29] for a review. Here, the cross-correlation solution has been implemented (a trade-off was realized comparing different methods and the best result, fewer false matches, was obtained using the cross-correlation approach). It is based on calculating an array of dimensionless coefficients for every image position (s, v) as[22,29]

$$R(s,v) = \sum_{i=0}^{h-1}\sum_{j=0}^{\varpi-1}(T(i,j) - \bar{T}(i,j))(I(i+s, j+v) - \bar{I}(i+s, j+v)), \quad (1)$$

where $h \in \mathbb{R}^+$ and $\varpi \in \mathbb{R}^+$ are the height and the width of the template, respectively, $T(i, j)$ and $I(i, j)$ are the pixel values at location (i, j) of the template and the current search area, respectively, and $\bar{T}(i, j)$ and $\bar{I}(i, j)$ are the mean values of the template and current search area, respectively. These mean values are calculated as

$$\bar{T}(i,j) = \frac{1}{(\varpi h)}\sum_{a=0}^{h-1}\sum_{c=0}^{\varpi-1} T(a,c), \quad (2)$$

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

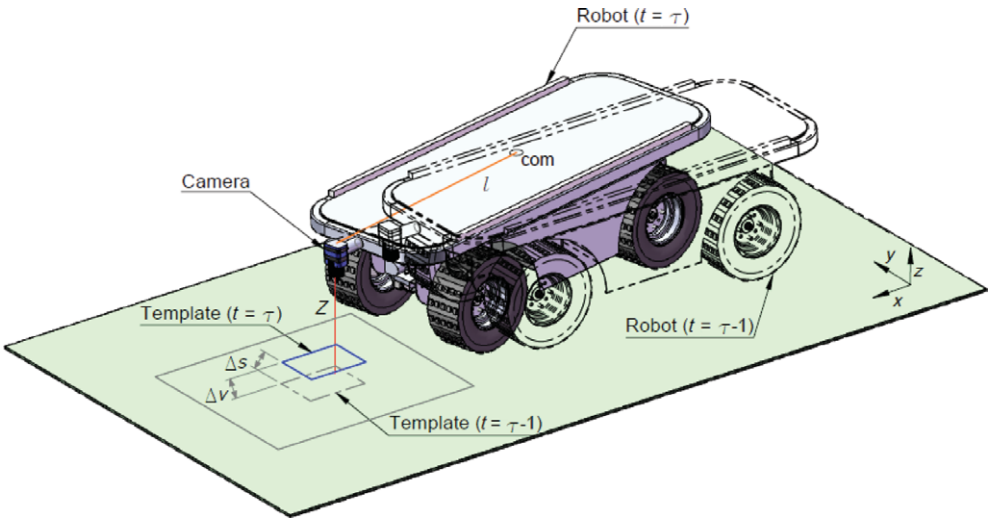


Fig. 1. (Colour online) Visual odometry based on template matching using a camera pointing at the terrain under the robot.

and

$$\bar{I}(i+s, j+v) = \frac{1}{(\varpi h)} \sum_{a=0}^{h-1} \sum_{c=0}^{\varpi-1} I(a+s, c+v). \quad (3)$$

Now, in order to avoid changes in the brightness between the template and the current image, every correlation coefficient is normalized.[28] For that purpose, it is divided by the standard deviation

$$N(s, v) = \sqrt{\sum_{i=0}^{h-1} \sum_{j=0}^{\varpi-1} \mathfrak{T}(i, j)^2 \mathfrak{I}(i+s, j+v)^2}, \quad (4)$$

where $\mathfrak{T}(i, j) = T(i, j) - \bar{T}(i, j)$ and $\mathfrak{I}(i+s, j+v) = I(i+s, j+v) - \bar{I}(i+s, j+v)$.

Finally, the normalized cross-correlation becomes

$$\tilde{R}(s, v) = \frac{R(s, v)}{N(s, v)}. \quad (5)$$

Notice that the value of $\tilde{R}$ changes between -1 and $+1$, and the closer the $\tilde{R}$ to $+1$, the more similar the template and the current image. For that purpose, the best match is defined as

$$\tilde{R}^M = \max(\tilde{R}(s, v)), \quad (6)$$

where $\tilde{R}^M$ is the maximum value of the array $\tilde{R}$ and (s^M, v^M) is the position of that point.

2.2. *Estimating robot displacement*

This subsection focuses on the estimation of the robot longitudinal displacement using the images taken by the camera pointing at the ground.

As shown in Fig. 1, at sampling instant, $t = \tau - 1$, the robot takes a picture of the ground under it. At the following sampling instant, $t = \tau$, the template matching approach is employed to find a defined template from the previous image in the current image. Finally, the pixel displacement $(\Delta s, \Delta v)$ is calculated as

$$\begin{aligned} \Delta s &= T_x - s^M, \\ \Delta v &= T_y - v^M, \end{aligned} \quad (7)$$

where $\Delta s \in \mathbb{R}$, $\Delta v \in \mathbb{R}$ are the longitudinal and lateral pixel displacements from the image sequence taken by the camera pointing at the ground, (T_x, T_y) is the position of the top left corner of the template (rectangle region centered at previous image), and (s^M, v^M) is the point of maximum correlation (see Eq. (6)). Notice that, for notational convenience, the time dependence on previous variables has been omitted.

Afterwards, camera units must be translated to physical world units using the camera calibration parameters,

$$\begin{aligned} \Delta x &= \Delta s \frac{Z}{f_x^g}, \\ \Delta y &= \Delta v \frac{Z}{f_y^g}, \end{aligned} \quad (8)$$

where $\Delta x \in \mathbb{R}$, $\Delta y \in \mathbb{R}$ are the camera longitudinal and lateral displacements in physical world units, respectively, $Z \in \mathbb{R}^+$ is the height of the camera above ground (see Assumption 1), and $f_x^g \in \mathbb{R}$, $f_y^g \in \mathbb{R}$ are the focal lengths of the camera pointing at the ground.

Assumption 1. *It is assumed that the distance between the camera and the ground is almost constant (see Remark 1).*

Remark 1. *We would like to remark that although on non-smooth surfaces the distance between the downward camera and the ground is not fixed due to vibrations, it is reasonable to assume this variation as zero or zero-mean, since such little oscillations are cancelled out during the experiment.*[30] *Notice that on a rougher surface an IMU sensor or a laser sensor should be used to estimate the height of the camera, leading to a 3D localization.*[27] *Recently, a novel approach consists in using telecentric cameras.*[31] *These cameras are electronically modified in such a way that the lens keeps the*

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

same field of view, regardless of the distance between the camera and the ground. Another possibility is to implement a 3D visual compass approach.[32]

Finally, the location of the robot along time is given by (see Remark 2)

$$x^{vo}(k) = x^{vo}(k-1) + \Delta x(k)\cos(\theta^{vo}(k)),$$
$$y^{vo}(k) = y^{vo}(k-1) + \Delta x(k)\sin(\theta^{vo}(k)), \quad (9)$$

where $[x^{vo} \quad y^{vo}]^T \in \mathbb{R}^2$ is the robot position. The estimation of the robot orientation ($\theta^{vo} \in \mathbb{R}$) is addressed in the following subsection.

Remark 2. *Notice that the robot orientation can be calculated using the information from the camera pointing at the ground. In this case, it is obtained as*[27]

$$\Delta\hat{\theta} = \arctan(\Delta y, l), \quad (10)$$

where $\Delta\hat{\theta} \in \mathbb{R}$ is the increment in robot orientation, and $l \in \mathbb{R}^+$ is the distance between the camera and the robot center (see Fig. 1). *Then the orientation at each sampling instant is given by*

$$\hat{\theta}^{vo}(k) = \hat{\theta}^{vo}(k-1) + \Delta\hat{\theta}(k). \quad (11)$$

However, the resulting orientation is extremely sensitive to systematic errors, such as inaccurate distance between the camera and the ground plane, inaccurate distance between the camera and the center of the robot, and false matches. These drawbacks cause that orientation becomes less and less accurate at each step.[20] *In order to minimize such effects, the visual compass technique is employed in this work.*

2.3. Estimating robot orientation: visual compass

The application of the visual compass technique to calculate the robot orientation is explained in this subsection. The visual compass approach was recently presented as a new way to estimate the robot orientation using vision. It was firstly presented in ref. [24], and it has been mainly applied to omnidirectional camera systems.[20,33]

The visual compass technique is also based on the template matching procedure to estimate the pixel displacement between two consecutive images. The difference is that now a camera looking at the environment ("panoramic view") is employed. In this way, a change in the robot orientation means an unidirectional pixel displacement between two consecutive images (see Fig. 2). The procedure consists of first obtaining the maximum correlation point between both images using Eq. (6), and secondly, calculating the pixel displacement (only in one direction) between the top left corner of the template and the maximum correlation point, that is,

$$\Delta u = T_y - v^M, \quad (12)$$

where $\Delta u \in \mathbb{R}$ is the pixel displacement from the image sequence taken by the camera looking at the environment. Finally, the rotation of the robot, $\Delta\theta \in \mathbb{R}$, supposing that the camera is mounted in the center of the robot, is given by

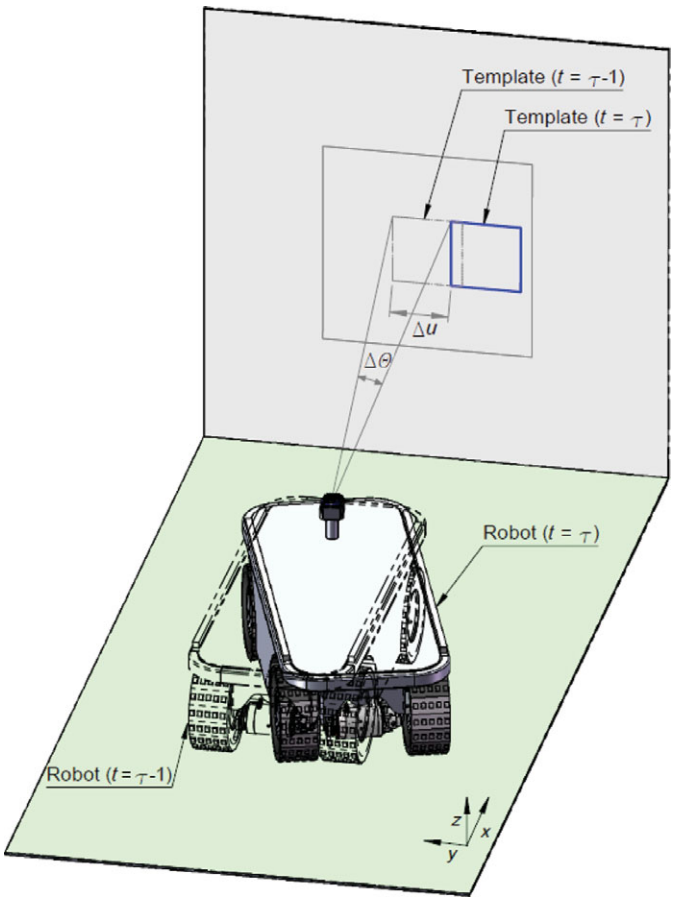


Fig. 2. (Colour online) Visual compass approach using a camera looking at the environment.

$$\Delta\theta = \arctan(\Delta u, f_x^e), \quad (13)$$

$f_x^e \in \mathbb{R}$ being the focal length of the camera looking at the environment. Then the orientation along time is given by (see Assumptions 2 and 3)

$$\theta^{vo}(k) = \theta^{vo}(k-1) + \Delta\theta(k). \quad (14)$$

Assumption 2. *It is assumed that the robot is moving on a static environment. In this way, there are no moving objects appearing in the panoramic camera.*

Assumption 3. *It is assumed that the mobile robots considered in this work move at low velocities, what implies that parallax effects are negligible. For a further discussion on this issue, see ref. [20].*

Summing up, the localization scheme presented here based on visual odometry and visual compass operates as follows:

(1) Acquire a pair of consecutive frames from each camera.
(2) Select the template from images taken at time $t = \tau - 1$.
(3) Match the template with the current image ($t = \tau$) by using Eq. (1). Normalize the result by using Eq. (5).
(4) Estimate the pixel displacement between the template and the maximum correlation point with Eq. (7).

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

(5) Translate from camera plane to world plane using the camera calibration parameters by means of Eq. (8).
(6) Compute the rotation angle using the visual compass method using Eq. (13).
(7) Estimate the robot location using translation information given by the camera pointing at the ground with Eq. (9), and the rotation angle given by the camera looking at the environment with Eq. (14).
(8) Repeat from Step 1.

3. Implementation Issues

In this section, the computational aspects of template matching (Section 3.1) and the selection of the search area and template sizes (Section 3.2) are discussed.

3.1. Computational aspects of template matching

This subsection discusses some experiments carried out to select the most appropriate template/search area size for a satisfactory performance of the correlation algorithm and proper computation time.

The main drawback of template matching approach is its computation cost, since the template has to be slid over the whole search area. In the general case, the detection of a single template $T_{m\times m}$ within a image $I_{n\times n}$ by means of a matching process is $\mathcal{O} = m^2(n-m+1)^2$.[22] For that reason, two important issues to be investigated are the template size and the search area size.

Notice that here the possibility of speeding up the matching process from an algorithmic point of view is not considered. This subsection only deals with determining the template/search area size to reach a trade-off between performance and computation time.

Firstly, the proper template size is studied, and later on, a way to reduce the search area is analyzed. In this way, the template is obtained as a reduced squared window of the image taken at sampling instant $\tau - 1$. The template origin has been established in the image center, and the top left corner of the template is located at[27]

$$T^q(s) = \frac{W^q}{2} - \frac{T^q_{\text{size}}}{2},$$
$$T^q(v) = \frac{H^q}{2} - \frac{T^q_{\text{size}}}{2}, \tag{15}$$

where $T^q(s, v)$ is the top left corner of the template, q refers to the images taken by the camera pointing at the ground ($q = g$) and to the camera looking at the environment ($q = e$), $W^q \in \mathbb{R}^+$ and $H^q \in \mathbb{R}^+$ are the width and height of the original image, respectively, and

$$T^q_{\text{size}} = \frac{1}{\rho^q} H^q, \tag{16}$$

is the template size being $\rho^q \geq 1$, a reduction factor experimentally tuned (it is explained subsequently).

Notice that the larger the template, the smaller the probability that it is matched in the search area. This means that if a too large template is selected, it cannot be possible to find it in the following image. On the contrary, the smaller the template, the higher the probability to fail into false matches. That is, if a too small template is selected, several areas of the following image can match with that template.

As commented previously, the second way to speed up the correlation matching process consists in using a reduced window of the original image instead of the whole image. Such reduced search area is given by

$$Win^q_w = \frac{1}{\lambda^q} W^q,$$
$$Win^q_h = \frac{1}{\lambda^q} H^q, \tag{17}$$

where Win^q is the size of the new reduced image, and $\lambda^q \geq 1$ is a reduction factor tuned experimentally (it is explained subsequently). Then the reduced image will start at the point $Win^q(s, v)$ and it will have a size of $Win^q_w \times Win^q_h$. The top left corner of the new image is

$$Win^q(s) = Win^q_w - \frac{Win^q_w}{\lambda^q},$$
$$Win^q(v) = Win^q_h - \frac{Win^q_h}{\lambda^q}. \tag{18}$$

In this way the computation time is decreased, since correlation process is carried out over a smaller image as shown in the following subsection.

3.2. Selection of the search area and template sizes

Before carrying out physical experiments, the effect of the template and image sizes on the computation time has been analyzed. Image sequences taken during physical experiments are also employed here (see Fig. 7). Notice that experiments have been carried out on a computer Intel Core 2 Duo 2.5 GHz with 3.5 GB RAM using OpenCV (Version 1.1).[28]

Figure 3 shows the resulting computation time varying the template and image sizes ("Mean" is the mean computation time of the sequence of images and "Std" denotes the standard deviation). Here the template and image sizes of the image sequence employed by the visual compass method are fixed to $\rho^e = 4$ and $\lambda^e = 1.7$ for Eqs. (16) and (17), respectively. As observed, larger template size (smaller ρ^g) implies that the computation time is lower. When the matching process is applied over a smaller search area (larger λ^g), the computation time also decreases. The computation time when images do not have any reduction (black triangles) is also displayed. From this analysis, the following reduction factors $\rho^g = 3$ and $\lambda^g = 1.2$ have been selected, since they constitute a compromise between suitable computation time (< 0.2 s) and success in the matching process. It is important to remark that although smaller reduction factors can be considered, these reduced search areas lead to an unfeasible matching process. This means that for the experiments carried out in this work if smaller search areas were considered, the number of false matches increases to unsuitable values, and robot location cannot be reliably estimated.

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

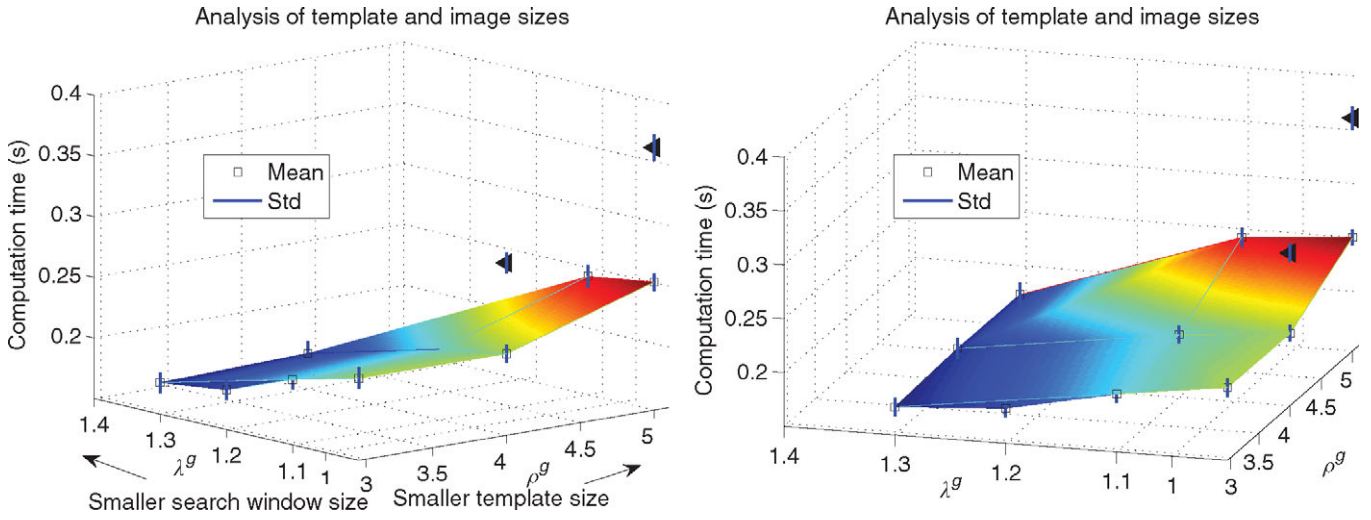


Fig. 3. (Colour online) Analysis of template and image sizes on computation time (images from camera pointing at the ground). The size of the images from the Pancam is fixed.

Notice that, as remarked in ref. [27], there is another important parameter to be considered in the selection of the template size, that is, the robot velocity. It is experienced that a smaller template size permits a high robot velocity, and a large template size limits the robot velocity. Regarding this issue, the images used for the experiment displayed in Fig. 3 were collected for a robot velocity that ranges between 0.4 m/s and 0.5 m/s. Nevertheless, the selected template and image sizes still work properly for small variations of those velocities.

4. Results

In this section, the physical experiments carried out to localize a tracked mobile robot using the suggested visual odometry approach are discussed. In this case, the robot was teleoperated on a sunlit illuminated off-road terrain. For comparison purposes, we collected vision data (cameras), global position (DGPS), odometry data (encoders), and absolute orientation (magnetic compass). The frames were grabbed at 5 Hz and the robot velocity ranged between 0.4 m/s and 0.5 m/s. Notice that for the kind of applications in which our mobile robot will be applied (greenhouse tasks),[14] these are considered an appropriate sampling period and velocity range. Here the DGPS and the magnetic compass data are considered as ground-truth for position and orientation, respectively. Notice that the position obtained using the DGPS is translated to relative position. For this purpose, the global position (latitude/longitude) was converted to Universal Transverse Mercator (UTM) grid system.[7]

We have tried several experiments. In this case, we firstly present a physical experiment in which the robot was driven along a squared trajectory where the total travelled distance was close to 160 m. After that, we discuss a S-shaped trajectory with a total travelled distance close to 290 m. Finally, we show a circular trajectory in which the total travelled distance was 65 m.

4.1. Testbed

The robot available at the University of Almería (Spain) is a tracked mobile robot called *Fitorobot* (see Fig. 4).[14] The mobile robot has a mass of 500 kg and its dimensions are 1.5-m long × 0.7-m wide. It is driven by a 20-HP gasoline engine.



Fig. 4. (Colour online) Tracked mobile robot *Fitorobot* at the experiment site. Observe the position of the two cameras on the robot.

We have employed two consumer-grade cameras, Logitech 2 Mpixel QuickCam Sphere AF webcam with maximum frame rate of 30 fps. In this case, a resolution of 640 × 480 has been employed. For calibration purposes, the Matlab's camera calibration toolbox has been used.[34]

The rest of sensors were one magnetic compass (C100, KVH Industries Inc.), two incremental encoders (DRS61, SICK AG), and one DGPS (R100, Hemisphere). The performance of the DGPS under motion is about 0.20 m. The resolution of the magnetic compass is 0.1°.

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press



Fig. 5. (Colour online) Images taken by the camera pointing at the ground (velocities > 1 m/s): (a) Blur effect (flat terrain); (b) vibrations effect (bumpy terrain).

Notice the position of the cameras for visual odometry (circles) in Fig. 4. The camera looking at the environment was mounted on the top center of the robot. The camera pointing to the ground in front of the robot is in the middle of both tracks at a height of 0.49 m and the distance between the camera and the robot center is 0.9 m.

4.2. Preliminary experiments: shadows and blur phenomenon

From physical experiments, it was noticed that when the robot moves at velocities greater than 1 m/s on flat terrains, blur phenomenon corrupts the images taken by the camera pointing at the ground (see Fig. 5(a)). Blur phenomenon occurs when an image is captured while the camera is moving during the exposure time or shutter time.[35] This phenomenon constitutes a difficult issue to be removed and elaborate solutions have to be considered to minimize its influence. For instance, in ref. [36], authors formulate a learning policy as a trade-off between the localization accuracy and the robot velocity. In ref. [35], authors propose to carry out a preprocessing step before detecting features in the image. In this work, a preprocessing of the images, such as an enhancing filter, is not appropriate, since it would mean to raise the computation time assigned to the vision algorithm. Bounding the robot velocity can be a successful solution; however, it would entail a certain degree of conservativeness for the motion controllers.

In relation to the image shown in Fig. 5(b), it is also interesting to remark that blur phenomenon is stressed by the vibrations affecting the mobile robot. It is a difficult issue to be removed, since the tracked mobile robot employed for physical experiments has a limited suspension mechanism that produces unavoidable vibrations on the robot structure. In conclusion, as a first approach in this work, visual odometry was employed when the robot was moving at velocities lower than 1 m/s.

Another important issue observed from outdoor physical experiments is the problem found in environments with changing lighting conditions, which can lead to shadows in the images taken by the camera pointing at the ground (see Fig. 6). After analyzing many experiments, it was concluded that when there are shadows in the images, the risk for false matches increases highly. For this reason, this phenomenon has been deeply studied and two approaches to minimize its effect have been proposed.



Fig. 6. (Colour online) Position and image acquired by groundcam with shadows. (a) Height of the groundcam. (b) Shadows in the groundcam.

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

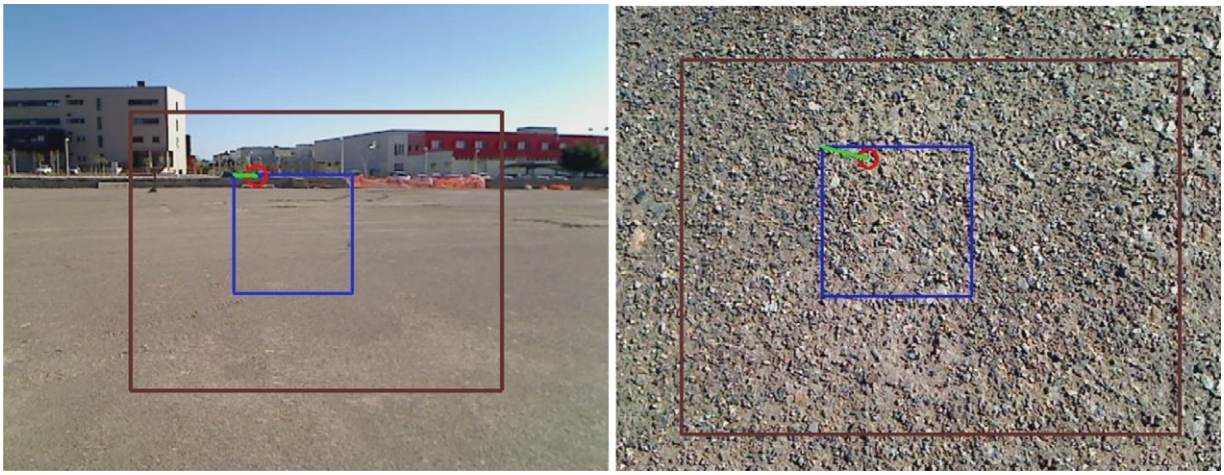


Fig. 7. (Colour online) Result of template matching in the experiment site (gravel soil). (a) Panoramic view. (b) Ground view.

First, the position and the height of the camera pointing at the ground were studied carefully. In this case, the camera was mounted in front of the robot between both tracks at a height of 0.49 m, see Fig. 6(a). This distance was obtained as a trade-off between shadow-reduction and template matching performance, that is, higher distance leads to more features but shadows can appear. On the contrary, shorter distance corresponds to smaller field of view, where the probability of shadows in the images is reduced. However, it can lead to featureless images.

Secondly, a threshold filter has been tuned. It compares the current pixel displacement with the previous ones; if the difference is greater than the threshold (experimentally selected), then the current value is considered as an outlier. In this way, these peaks or outliers, due to false matches, are partially compensated. As shown in the following subsection, this filter works properly and requires a small computation time.

4.3. *Physical experiments in off-road conditions*

Several trajectories were tested to check the performance of the suggested localization approach. In this case, three experiments were selected. In the first one, the robot was driven along a rectangular trajectory of approximately 55-m long and 20-m wide. The total travelled distance was close to 160 m. In the second experiment, the robot was driven along an S-shaped trajectory with three parallel paths to the x-axis of 80 m and two perpendicular paths of 20 m. The total travelled distance was close to 290 m. Finally, a circular trajectory is selected, the total travelled distance was close to 65 m. Notice that trajectories similar to those selected here are usually employed in off-road mobile robotics, see, for instance, refs. [12, 20, 27]. The sampling period was T_s = 0.2 s and the robot velocity ranged between 0.4 m/s and 0.5 m/s.

The compared localization techniques are visual odometry with visual compass trajectory (denoted as "VO + VC" in the figures), visual odometry using only the downward camera (see Remark 2; referred to as "VO"), and the wheel-based odometry (denoted as "Odo"). The position ground-truth comes from a DGPS (labelled as "DGPS"), and the orientation ground-truth comes from a magnetic compass (denoted as "Compass").

4.3.1. Experiment 1. Rectangular trajectory. In this experiment, the robot was manually driven on a sunlit illuminated gravel terrain following a rectangular trajectory. In this case, the lighting conditions did not produce any significant shadows during the experiment.

Figure 7 shows two frames employed by the vision-based localization technique during this experiment. The pixel displacement is marked by the green line and the red circle, the template is labelled by the blue rectangle, and the black rectangle means the reduced area in which the matching process is carried out.

Figure 8 shows the resulting trajectories. It is observed that the visual odometry with visual compass trajectory closely follows the ground-truth, while the wheel-based odometry estimate diverges largely from the ground-truth, particularly odometry fails at turns. The trajectory obtained using the image sequence from the camera pointing at the ground to estimate orientation is also plotted, and it has a similar result to that obtained using the approach combining information from both cameras (visual odometry with visual compass).

A deeper analysis is obtained looking at Fig. 9. Here, the error between each localization method and the ground-truth is shown quantitatively. In this case, the Euclidean distance between the initial and the final positions of the robot in the four parts of the trajectory is calculated, that is, two parallel paths to the x-axis (Parts 1 and 3) and two perpendicular ones (Parts 2 and 4). From these data, it is observed that the visual odometry with the visual compass approach achieves the smallest error. Another vision-based technique also achieves an admissible error. The relative mean errors with respect to the total travelled distance are 1.45% for visual odometry with visual compass, 2.33% for visual odometry alone (using only the downward camera), and 16% for wheel-based odometry.

Figure 10 displays the orientations. Here, it is checked that the orientations obtained through the visual odometry-based approaches follow properly the ground-truth. The mean orientation errors are 8.2^o for visual odometry with visual compass, 14.1^o for visual odometry, and 39.37^o for wheel-based

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

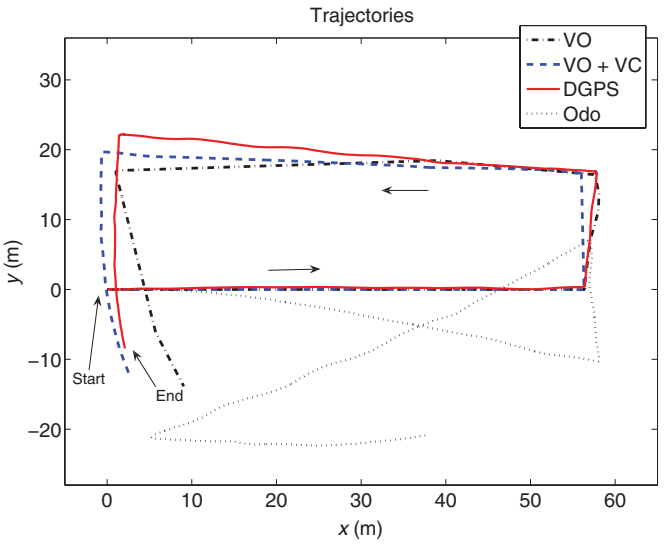


Fig. 8. (Colour online) Experiment 1. Rectangular trajectory.

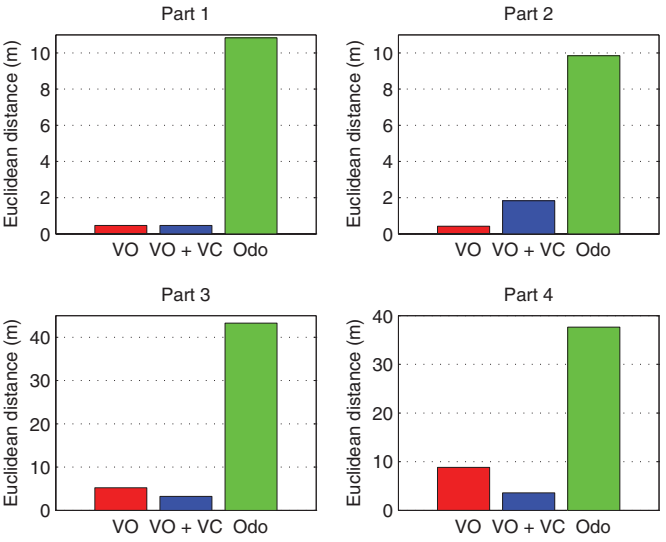


Fig. 9. (Colour online) Experiment 1. Comparison of the Euclidean distance with respect to the ground-truth.

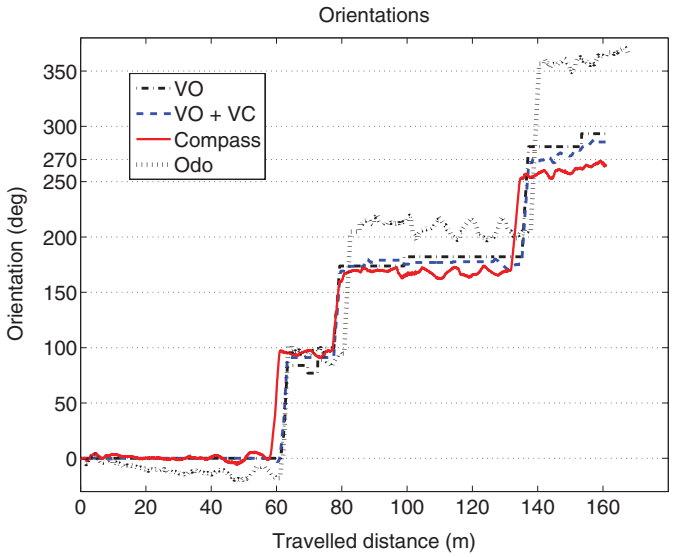


Fig. 10. (Colour online) Experiment 1. Orientations.

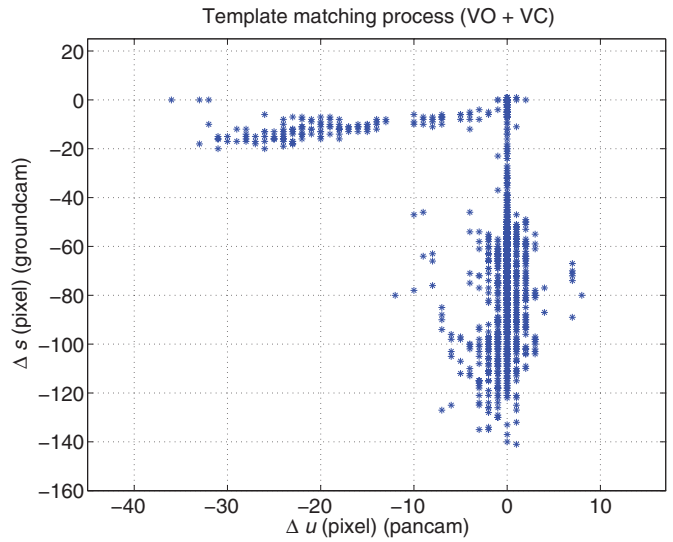


Fig. 11. (Colour online) Experiment 1. Lateral and longitudinal pixel displacements (VO + VC).

odometry. In this figure, it is possible to observe the unavoidable error growth phenomenon for odometry-based solutions, that is, the deviation between the ground-truth and the rest of techniques increases along the travelled distance (integration of the noises and the error over time).

In Fig. 11, the longitudinal (Δs) and lateral (Δu) pixel displacement values related to the visual odometry with the visual compass approach are shown. Notice that values close to zero mean small displacements (low velocity), and high values mean large displacements (high velocity). In this plot, it is observed that the points are aligned in two directions, being this effect due to the pixel displacements during straight motions, Δs component, and during turns, Δu component. It is checked that template matching is highly robust with few outliers (false matching or unsuccessful matching). It is important to point out three interesting conclusions from this plot. Firstly, since the robot always turns in the same sense (to the left side), lateral pixel displacement (Δu) is also aligned in one direction. Secondly, when the robot is turning, it does not move forward, since, as observed, Δs is close to zero at turns. Finally, note that in the range $\Delta s = (-20, -40)$ pixel, Δu is zero, which means the moment in which the robot is stopping before turning.

4.3.2. Experiment 2. S-shaped trajectory. In this experiment, a longer trajectory, in which the robot changed direction several times, was tested. Furthermore, in some parts of the experiment site, the lighting conditions produced shadows that affected the performance of the vision-based localization strategies.

Figure 12 shows the resulting trajectories. It is observed that the visual odometry with the visual compass trajectory does not follow accurately the ground-truth mainly for one reason. As checked during the first perpendicular path to the x-axis and the second parallel path, the trajectory is shorter than the ground-truth. This fact is due to false matches obtained from the camera pointing at the ground caused by shadows (see Fig. 15(b)). This erroneous behavior is worst

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

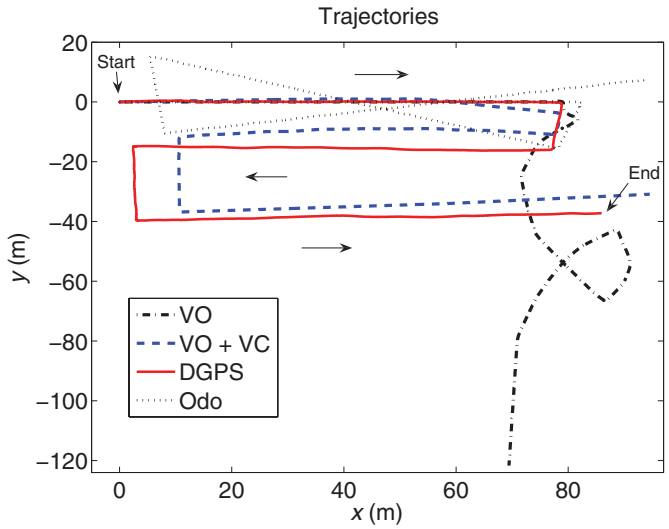


Fig. 12. (Colour online) Experiment 2. S-shaped trajectory.

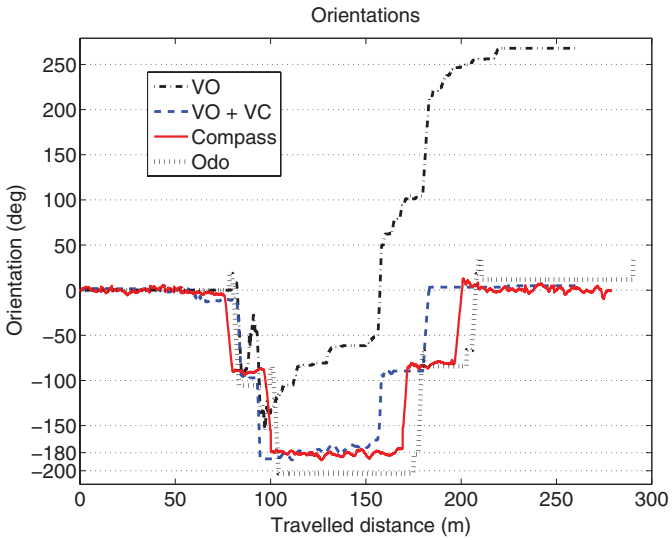


Fig. 14. (Colour online) Experiment 2. Orientations.

in the case of visual odometry alone, since, now, longitudinal displacement and orientation are obtained from the camera pointing at the ground. The largest deviation is obtained during the first perpendicular path to the x-axis. Again, the wheel-based odometry diverges largely from the ground-truth, particularly, odometry fails at turns.

In Fig. 13, the error between each localization technique and the ground-truth is displayed. In this case, the Euclidean distance between the initial and the final position of the robot is calculated in five parts, that is, three parallel paths to the x-axis (Parts 1, 3, and 5) and two perpendicular ones (Parts 2 and 4). As expected, the visual odometry with visual compass approach obtains an admissible error except during the second and the third paths. The relative mean errors with respect to the total travelled distance are 2.46% for visual odometry with visual compass, 7.60% for wheel-based odometry, and 19.50% for visual odometry.

In Fig. 14, the orientations are plotted with respect to the travelled distance. Here the erroneous behavior of the visual odometry approach during the first parallel path to the x-axis is noticed. The visual odometry with the visual compass approach estimates the orientation properly and follows the ground-truth. The mean orientation errors are 4.8° for visual odometry with visual compass, 10.2° for wheel-based odometry, and 148.2° for visual odometry. The mean orientation error for the case of visual odometry cannot be considered as a comparable value, since it has a large standard deviation.

In Fig. 15(a), the longitudinal (Δs) and lateral (Δu) pixel displacement values are shown. In contrast to the previous

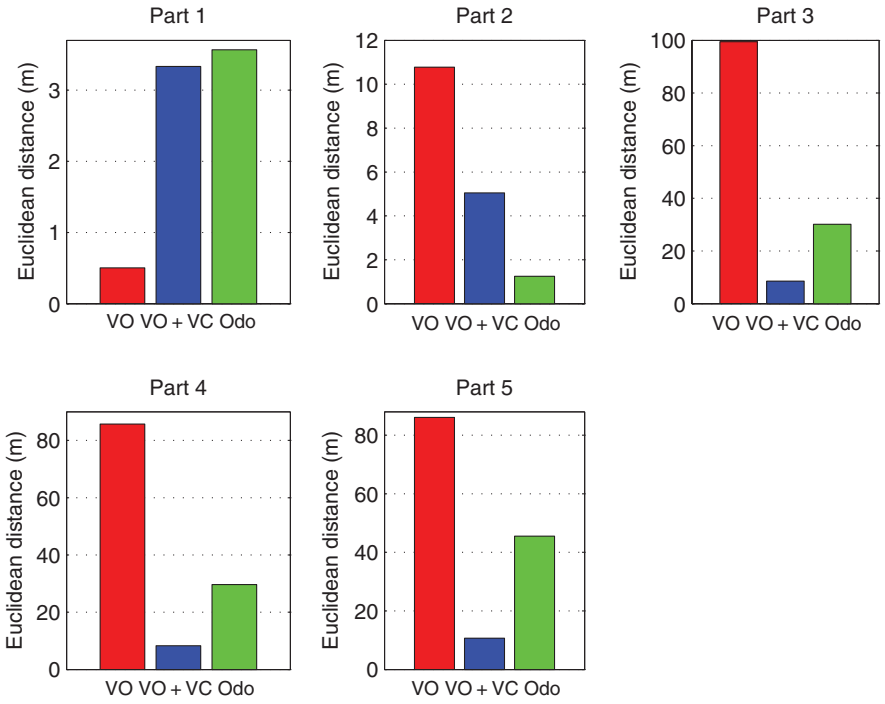


Fig. 13. (Colour online) Experiment 2. Comparison of the Euclidean distance with respect to the ground-truth.

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

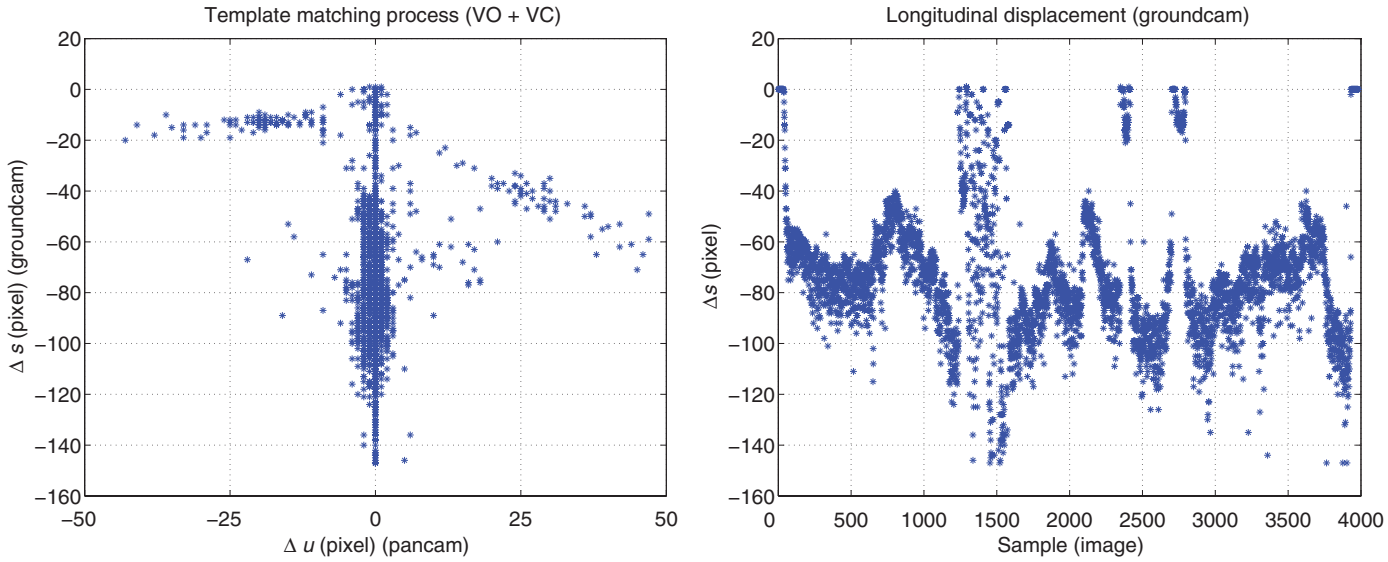


Fig. 15. (Colour online) Experiment 2. Template matching using both cameras. (a) Lateral and longitudinal displacement. (b) Longitudinal displacement (groundcam).

experiment, now the pixels related to the lateral displacement (Δu) are aligned in two directions (right and left turns). Notice that the turns to the right side were carried out at higher linear velocity than the turns to the left side. In this case, there are some outliers, when the robot moved in straight line ($\Delta s < -40$ pixel), which can explain the small deviation obtained at the end of the first parallel path to the x-axis (see Fig. 12). On the other hand, in Fig. 15(b), the longitudinal pixel displacements (Δs) with respect to the acquired images are displayed. As noticed, during the samples (1200, 1600), there is an erroneous behavior (false matches). This behavior explains why the trajectories obtained with the vision-based approaches are shorter than the ground-truth during the first perpendicular path to the x-axis. The false matches found in the interval (2300, 2800) explain why the trajectories are shorter than the ground-truth during the second parallel path to the x-axis.

A deeper understanding of the erroneous behavior of the visual odometry approach is obtained by analyzing Fig. 16. Recall that, for the case of visual odometry alone, the robot orientation comes from the lateral pixel displacement obtained from the camera pointing at the ground (see Remark 2). As checked in Fig. 16(a), many outliers appear in the Δv component; compare it with the visual compass pixel displacement (Δu) obtained from the camera looking at the environment in Fig. 15(a). In Fig. 16(b), notice that these outliers occur within two intervals in which false matches appeared due to shadows (see Fig. 15(b)).

4.3.3. Experiment 3. Circular trajectory. Finally, a circular trajectory was tested in order to check the performance of the proposed localization strategies estimating the robot orientation. The most challenging issue about this experiment is that the robot is always turning and, hence there are always

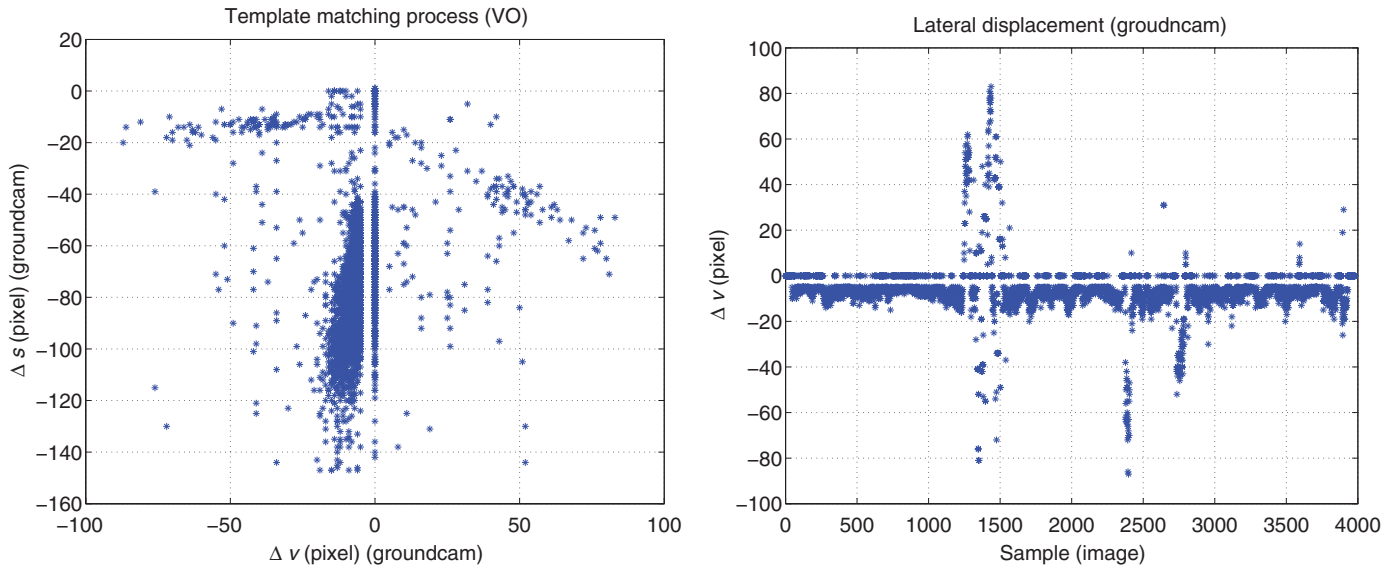


Fig. 16. (Colour online) Experiment 2. Template matching process using groundcam. (a) Lateral and longitudinal displacement. (b) Lateral displacement (groundcam).

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

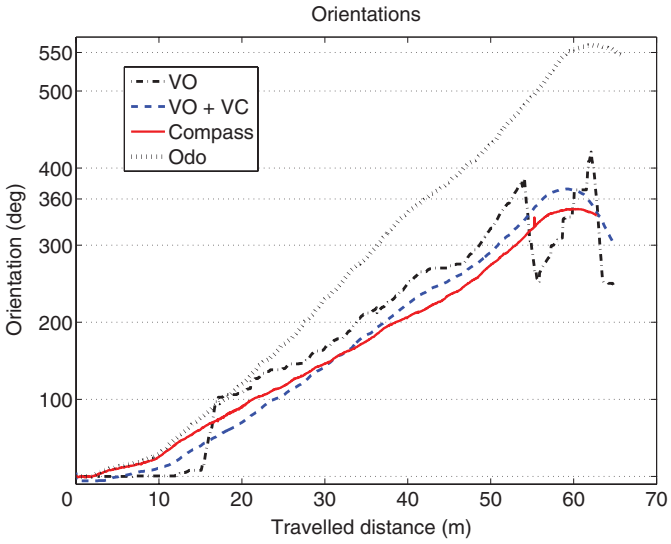


Fig. 17. (Colour online) Experiment 3. Orientations.

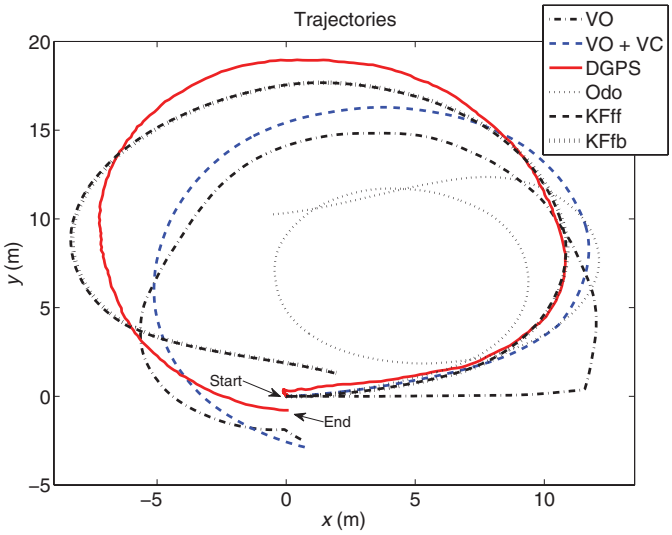


Fig. 18. (Colour online) Experiment 3. Circular trajectory.

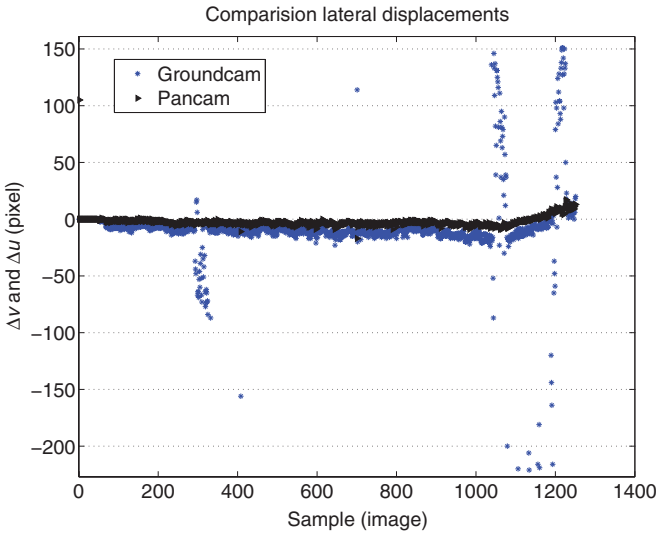


Fig. 19. (Colour online) Experiment 3. Lateral pixel displacements (groundcam and pancam).

shadows in the images obtained from the camera pointing at the ground. Furthermore, this circular trajectory highlights the main inconvenience of the odometry-based localization techniques, that is, the integration of the orientation from the starting point that leads to inaccurate robot localization for long-range circular trajectories.

In Fig. 17, the orientations obtained during the test are shown with respect to the travelled distance. As advised, the typical effect of the odometry-based solutions is observed. Notice that the orientation obtained using the odometry-based techniques diverges from the ground-truth (recall that the experiments were carried out in open-loop). However, it is interesting to remark that acceptable behavior of the orientation is obtained using the visual compass technique. For instance, note that outliers and mismatches highly affect the estimated orientation using only the camera pointing at the ground. The mean errors between the ground-truth and orientation are 15.02º for visual odometry with visual compass, 40.43º for visual odometry, and 107.84º for wheel-based odometry.

Figure 18 shows the resulting trajectories. From this figure, it is possible to understand the effect of the orientation integration in the odometry-based localization techniques. In this way, small orientation deviations lead to a high error in the position along time and distance. However, it is important to remark the acceptable behavior of the visual odometry with the visual compass technique for the first 20 m. As observed in previous experiments, the Euclidean distance with respect to the ground-truth is calculated. In particular, the mean errors with respect to the total distance travelled are 2.77% for the visual odometry with visual compass, 4.61% for the visual odometry, and 10.92% for wheel-based odometry.

As observed in Fig. 19, the template matching result for the camera pointing at the ground (denoted as "Groundcam") suffers from outliers and false matches, especially at the end of the experiment. This explains the behavior observed in Fig. 17. The images employed by the visual compass approach (labelled as "Pancam") are not affected by shadows, and hence there are no significant outliers during the matching process.

In conclusion, Table I summarizes the most important data from the physical experiments. In this case, the mean error with respect to the DGPS and the total travelled distance, and the mean error with respect to the magnetic compass are shown. In particular, it is interesting to point out the satisfactory results obtained using the visual odometry with the visual compass approach. Recall that in this particular case, two consumer-grade cameras can replace more expensive sensors, such as encoders, Doppler radar, IMU, gyroscopes, etc.

5. Conclusions and Discussion

In this paper, we have shown the application of a visual odometry technique based on template matching to off-road mobile robots localization. Standard visual odometry (using a single camera) has been improved using the visual compass method. This strategy has been implemented using two consumer-grade monocular cameras. Physical experiments

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

Table I. Summary of localization methods.

Feature	Trajectory	Rectangular	*S*-shaped	Circular
Total travelled distance (m)		160	290	65
Mean error (DGPS) (%)	Odo	16.00	7.60	10.92
	VO	2.33	19.50	4.61
	VO + VC	1.45	2.46	2.77
Mean error (compass) (deg)	Odo	39.37	10.2	107.84
	VO	14.1	148.2	40.43
	VO + VC	8.2	4.8	15.02

have confirmed the appropriate behavior of the proposed scheme with a mean error lesser than 3% with respect to the total travelled distance. Furthermore, an acceptable computation time (<0.17 s) has been achieved taking into account the purposes of our testbed. However, it is important to remark that the current code can be highly optimized.

The point related to reduce the size of the search area during the matching process can be considered as an incipient approach to decrease the computation time. The best improvement in which we are currently working consists of using the robot motion to reduce accordingly the image size and to estimate the position of the template during the matching process. We are also considering it in terms of a multi-objective problem (success of matching process, reduction of template size, and reduction of image size).

The problem with shadows and blur phenomenon, and hence with false matches, constitutes the most important shortcoming of the vision-based localization techniques. In this work, a deep analysis has been carried out to minimize these issues. The height of the downward camera was carefully selected and a threshold filter was tuned. Currently, we are investigating two ways to minimize such undesirable effects. Firstly, mounting the downward camera just under the vehicle and using an artificial uniform source to light the ground (shadows issue), and secondly, acquisition of a new camera with a shorter exposure time (blur phenomenon issue).

On the other hand, probabilistic techniques, such as the Kalman filter or particle filter, will be employed fusing the orientations obtained using visual compass and an absolute orientation sensor. In this way, we will reduce the unavoidable error growth of relative localization techniques.

Finally, we will study to improve the planar motion assumption (fixed height of the camera pointing at the ground) employing an IMU sensor or a telecentric camera.

References

1. S. Thrun, S. Thayer, W. Whittaker, C. Baker, W. Burgard, D. Ferguson, D. Haehnel, M. Montemerlo, A. Morris, Z. Omohundro, C. Reverte and W. Whittaker, "Autonomous exploration and mapping of abandoned mines," *IEEE Robot. Autom. Mag.* **11**(1), 79–91 (2004).
2. J. Borenstein, "The CLAPPER: A Dual-Drive Mobile Robot with Internal Correction of Dead-Reckoning Errors," **In**: *Proceedings of IEEE Conference on Robotics and Automation, IEEE*, San Diego, CA, USA (May 8–13, 1994) pp. 3085–3090.
3. O. Horn and M. Kreutner, "Smart wheelchair perception using odometry, ultrasound sensors, and camera," *Robotica* **27**(2), 303–310 (2009).
4. H. Beom and H. Cho, "Mobile robot localization using a single rotating sonar and two passive cylindrical beacons," *Robotica* **13**(3), 243–252 (1995).
5. S. Cho and J. Lee, "Localization of a high-speed mobile robot using global features," *Robotica* **29**(5), 757–765 (2010) Available on CJO.
6. R. Siegwart and I. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, 1st ed. A Bradford book. (The MIT Press, Cambridge, MA, USA, 2004).
7. B. Hofmann-Wellenhof, H. Lichtenegger and J. Collins, *Global Positioning System: Theory and Practice*, 5th ed. (Springer, Germany, 2001).
8. A. Johnson, S. Goldberg, C. Yang and L. Matthies, "Robust and Efficient Stereo Feature Tracking for Visual Odometry," **In**: *Proceedings of IEEE International Conference on Robotics and Automation, IEEE*, Pasadena, USA (May 19–23, 2008) pp. 39–46.
9. L. Matthies, M. Maimone, A. Johnson, Y. Cheng, R. Willson, C. Villalpando, S. Goldberg and A. Huertas, "Computer vision on mars," *Int. J. Comput. Vis.* **75**(1), 67–92 (2007).
10. C. Olson, L. Matthies, M. Schoppers and M. Maimone, "Rover navigation using stereo ego-motion," *Robot. Auton. Syst.* **43**(4), 215–229 (2003).
11. I. Parra, M. Sotelo, D. Llorce and M. O. Na, "Robust visual odometry for vehicle localization in urban environments," *Robotica* **28**(3), 441–452 (2010).
12. J. Campbell, R. Sukthankar, I. Nourbakhsh and A. Pahwa, "A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision," **In**: *Proceedings of IEEE International Conference on Robotics and Automation, IEEE*, Barcelona, Spain (Apr. 18–22, 2005) pp. 3421–3427.
13. L. Matthies, "Dynamic Stereo Vision," *Ph.D Thesis* (Pittsburgh, USA: Carnegie Mellon University, 1989).
14. R. González, F. Rodríguez, J. Sánchez-Hermosilla and J. Donaire, "Navigation techniques for mobile robots in greenhouses," *Appl. Eng. Agr.* **25**(2), 153–165 (2009).
15. R. González, "Localization of the CRAB rover using Visual Odometry," *Technical Report* (Autonomous Systems Lab, ETH Zürich, Switzerland), available at: http://www.ual.es/personal/rgonzalez/english/publications.htm (2009) online. (Accessed September 2011).
16. D. Nistér, O. Naroditsky and J. Bergen, "Visual odometry for ground vehicle applications," *J. Field Robot.* **23**(1), 3–20 (2006).
17. A. Angelova, L. Matthies, D. Helmick and P. Perona, "Learning and prediction of slip from visual information," *J. Field Robot.* **24**(3), 205–231 (2007) .
18. D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.* **60**(2), 91–110 (2004).
19. B. Lucas and T. Kanade, "An Iterative Image Registration Technique with An Application to Stereo Vision," **In**: *Proceedings of DARPA Imaging Understanding Workshop, DARPA*, Monterey, USA (Aug. 24–28, 1981) pp. 121–130.
20. D. Scaramuzza, "Omnidirectional Vision: From Calibration to Robot Motion Estimation," *Ph.D Thesis* (Zürich, Switzerland: Swiss Federal Institute of Technology, 2008).

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press

21. V. Matiukhin, "Trajectory Stabilization of Wheeled System," **In**: *Proceedings of IFAC World Congress, IFAC*, Seoul, Korea (2008) pp. 1177–1182.
22. R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*, (John Wiley, New Jersey, USA, 2009).
23. A. Goshtasby, S. Gage and J. Bartholic, "A two-stage correlation approach to template matching," *IEEE Trans. Pattern Anal. Mach. Intell.* **6**(3), 374–378 (1984).
24. F. Labrosse, "The visual compass: Performance and limitations of an appearance-based method," *J. Field Robot.* **23**(10), 913–941 (2006).
25. M. Srinivasan, "An image-interpolation technique for the computation of optic flow and egomotion," *J. Biol. Cybern.* **71**(5), 401–415 (1994).
26. S. Kim and S. Lee, "Robust mobile robot velocity estimation using a polygonal array of optical mice," *Int. J. Inf. Acquis.* **5**(4), 321–330 (2008).
27. N. Nourani-Vatani, J. Roberts and M. Srinivasan, "Practical Visual Odometry for Car–like Vehicles," **In**: *Proceedings of IEEE International Conference on Robotics and Automation, IEEE*, Kobe, Japan (May 12–17, 2009) pp. 3551–3557.
28. G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*, (O'Reilly, Sebastopol, CA, USA, 2008).
29. J. Rodgers and W. Nicewander, "Thirteen ways to look at the correlation coefficient," *Am. Stat.* **42**(1), 59–66 (1988).
30. M. Elmadany and Z. Abduljabbar, "On the statistical performance of active and semi-active car suspension systems," *Comput. Struct.* **33**(3), 785–790 (1989).
31. K. Nagatani, A. Ikeda, G. Ishigami, K. Yoshida and I. Nagai, "Development of a visual odometry system for a wheeled robot on loose soil using a telecentric camera," *Adv. Robot.* **24**(8–9), 1149–1167 (2010).
32. J. Montiel and A. Davison, "A Visual Compass based on SLAM," **In**: *Proceedings of IEEE International Conference on Robotics and Automation, IEEE*, Orlando, USA (May 15–19, 2006) pp. 1917–1922.
33. J. Sturm and A. Visser, "An appearance-based visual compass for mobile robots," *Robot. Auton. Syst.* **57**(5), 536–545 (2009).
34. J. Bouguet, "Camera calibration toolbox for Matlab," available at: http://www.vision.caltech.edu/bouguetj/calib_doc/ (2008). (Accessed September 2011).
35. A. Pretto, E. Menegatti, M. Bennewitz, W. Burgard and E. Pagello, "A Visual Odometry Framework Robust to Motion Blur," **In**: *Proceedings of IEEE International Conference on Robotics and Automation, IEEE*, Kobe, Japan (May 12–17, 2009) pp. 1685–1692.
36. A. Hornung, M. Bennewitz and H. Strasdat, "Efficient vision-based navigation. Learning about the influence of motion blur," *Auton. Robots* **29**(2), 137–149 (2010).

https://doi.org/10.1017/S026357471100110X Published online by Cambridge University Press