Chapter 1
Unicycle–like Robots with Eye-In-Hand Monocular Cameras: from PBVS Towards IBVS

Daniele Fontanelli, Paolo Salaris, Felipe A. W. Belo and Antonio Bicchi

Abstract This chapter presents an introduction to current research devoted to the visual servoing problem of guiding differentially driven robots, more specifically, unicycle–like vehicles, taking into consideration limited field-of-view constraints. The goal is to carry out accurate servoing of the vehicle to a desired posture using only feedback from an on-board camera.

First, a position based scheme is proposed, adopting a hybrid control law to cope with limited camera aperture. This scheme relies on a localization method based on extended Kalman filter technique that takes into account the robot motion model and odometric data. To increase the potentiality of the visual servoing scheme with respect to existing solutions, which achieve similar goals locally (i.e., when the desired and actual camera views are sufficiently similar), the proposed method visually navigate the robot through an extended visual map before eventually reaching the desired goal. The map construction is part of the approach proposed here, which is then called Visual SLAM for Servoing.

Position based scheme accuracy are intrinsically related to the effectiveness of the localization process, which is related to the estimation of 3D information on both the robot and the environment. A shortcut overcoming the estimation process uses visual information directly in the image domain. In this spirit, an image based scheme is presented. The controller is devoted to constantly track desired image feature trajectories. Such trajectories represent optimal (shortest) paths for the vehicle from the 3D initial position towards the desired one. Optimal trajectories satisfies

Daniele Fontanelli
University of Trento, e-mail: fontanelli@disi.unitn.it

Felipe A. W. Belo
University of Pisa, e-mail: felipebelo@gmail.com

Paolo Salaris
University of Pisa, e-mail: salarispaolo@gmail.com

Antonio Bicchi
University of Pisa, e-mail: bicchi@ing.unipi.it

the additional constraint of keeping a feature in sight of the camera and induces a taxonomy of the robot plane of motion into regions. It follows that the robot uses only visual data to determine the region to which it belongs and, hence, the associated optimal path. Similarly to the previous case, the visual scheme effectiveness is improved adopting appearance based image maps.

1.1 Introduction

Vision–based control systems for vehicle control have attracted a lot of attention in the last decades. A plethora of different solutions have been proposed in literature, mainly tailored to the visual architecture of the system at hand. For example, stereo cameras provide accurate environmental measurements with an intrinsic robustness to occlusions. Furthermore, omnidirectional cameras do not suffer of field–of–view (FOV) limitations. Despite these advantages, both multiple and omnidirectional cameras need non trivial algorithms to extract useful data from collected images. Moreover, the system mechanical design becomes necessarily complicated and requires accuracy. Hence, it turns out that an economical approach to visual servoing for mobile vehicles is the use of conventional monocular cameras fixed onboard the robot. Nonetheless, the simplest mechanical choice to be used with off–the–shelf conventional cameras, e.g., web cams, reflects in a challenging control design. In fact, mobile robots are usually subject to non-holonomic kinematic constraints and, hence, they cannot rely on independent six-degrees of freedom operational space as robotic manipulators usually rely on. Moreover, when using a limited aperture camera fixed over the robot (a case known as Eye-In-Hand), the robot must deal with the problem of keeping observed image features within the FOV of the camera while the vehicle maneuvers. Also, the space entities estimate is derived using multiple measurements from different positions in space, taking advantage of multiple-view geometry. This chapter presents an overview of current research devoted to the visual servoing problem of guiding differentially driven robots, with standard Eye-In-Hand limited aperture monocular cameras, addressing the problems that such a simple and economical framework give birth to.

In the classic approach to visual servoing, a well known taxonomy is derived and classified in the seminal paper of Weiss in [34]: in *image based visual servoing* (IBVS) the control error is defined directly in the image space, based on *features* extracted from image data, e.g., visual cues like points, planes or lines; on the other hand, *position based visual servoing* (PBVS) computes the error in relation to a set of 3D parameters that are estimated from image measurements, e.g., robot position errors with respect to the desired position to reach. In the second case, position errors are usually computed in the robot Cartesian space and provided, as customary, to the control system. Robot position reconstruction is often referred to as robot *localization*. The two vision–based schemes thus described should be regarded as the end-points of a range of different possibilities, whereby the raw sensorial infor-

mation is gradually abstracted away to a more structured representation using some knowledge of the robot-environment model.

While PBVS methods are the most straightforward ones, they require metrical information about the feature positions, usually organized in a *map*. When the metrical information can't be computed directly from the camera view (which is the case for a monocular camera), pose estimation is usually achieved with the additional use of external measurements, such as odometric data. Therefore, the latest research focuses on *hybrid* methods or pure IBVS methods to overcome the estimation problem. Among the others, hybrid schemes have been proposed in [14], coming up to a scheme which is roughly half-way between IBVS and PBVS, and in [19], where a *switched* control approach that utilizes both schemes depending on the distance to the target has been implemented. Indeed, IBVS is more accurate than PBVS as soon as the reference and the target images are close enough ([7, 8]). For this reason, an image based task is often separated into a set of consecutive control problems using a sequence of target images, in other words using appearance based visual maps that not take into account any 3D spatial information ([31, 18]).

In the past few years, robot scientists have focused on the optimality of paths followed by visually servoed robots. For example, researchers have focused on the optimal control of visually guided robotic manipulators ([11]) or on optimal trajectory planning for robot manipulators controlled via a limited FOV camera ([16]). Minimal trajectories have been also presented in [28] in case of large displacements, again for a six degrees of freedom robot manipulator. Optimal paths for differentially driven robots with visibility restricted to limited FOV have recently been addressed by some researchers ([5, 33]). The solutions proposed in this field are more related to optimal path planning than robot reactive control, restricting the role of the visual control to path following. Of course, also in this case solutions can be divided in position based or image based, depending on the space in which the trajectories are derived.

In this chapter we aim to provide an overview to the visual servoing of differentially driven robot and the vision theoretic fundamentals needed in each case. Then, a PBVS control scheme that solves the problem in the 3D domain is proposed, together with a servoing–oriented *Simultaneous Localization And Mapping* (SLAM) algorithm to enhance the potentiality of the controller. To overcome the localization process, a combination of an IBVS and an optimal (shortest) path planner is then proposed. Again, the autonomous capability of the servoed robot are increased adopting an appearance map–based approach. A discussion on the advantages and drawbacks that pertain to each technique is also presented.

1.2 Problem Definition

The visual servoing problem as meant in this chapter is referred to mobile vehicles, with a rigidly fixed on–board camera. In particular, we consider vehicles that constantly move on a plane, as in typical indoor set–up, like factory or office floors.

We assume for the moving platform a driftless kinematic model, more precisely this chapter refers to a *unicycle–like* nonholonomic mobile robot. Without loss of generality, we assume that the robot coordinates are measured with respect to a dextrous reference frame $\langle W \rangle = \{O_w, X_w, Y_w, Z_w\}$, fixed with the static environment. It is also assumed that the Y_w coordinate $y(t) = 0$, $\forall t$, hence the state space of the mobile platform is $\xi(t) = (x(t), z(t), \theta(t))$, where the robot reference point $(x(t), z(t))$ is in the middle of the wheel axle and the robot direction $\theta(t)$ is zero when the vehicle heads to the X_w axis. Assuming that the control inputs are $u(t) = (v(t), \omega(t))$, with $v(t)$ and $\omega(t)$ are respectively the forward and angular velocities of the vehicle, the system kinematic model is

$$\dot{\xi} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega = f_v v + f_\omega \omega. \tag{1.1}$$

The mobile agent is equipped with a rigidly fixed camera with a reference frame $\langle C \rangle = \{O_c, X_c, Y_c, Z_c\}$ such that the optical center O_c corresponds to the robot's center and the optical axis Z_c is aligned with the robot's forward direction. If the robot orientation is null, the Z_c axis is parallel to the X_w axis, with the same direction, and the X_c axis is parallel to the Z_w axis, with opposite direction.

Motionless features will be defined in reference to the camera frame as ${}^cP = [{}^cx, {}^cy, {}^cz]^T$ and to the fixed frame $\langle W \rangle$ as ${}^wP$. The coordinate transformation between $\langle W \rangle$ and $\langle C \rangle$ is given by $[{}^cP^T, 1]^T = [{}^cx, {}^cy, {}^cz, 1]^T = {}^cH_w[{}^wP^T, 1]^T$, where ${}^cH_w$ is the transformation matrix between frames. Assume a *pinhole camera model* where α_x and α_y are the focal lengths of the intrinsic camera calibration matrix ([21])

$$K_c = \begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.2}$$

The origin O_I of the image plane reference frame $\langle I \rangle = \{O_I, X_I, Y_I\}$ is assumed to be coincident with the principal point — i.e. the intersection of the camera axis Z_c with the image plane. The feature coordinates in the image frame (measured in pixels) is ${}^Ip = [{}^Ix, {}^Iy]^T$. Hence, the projective transformation (or *homography*) that describes a general mapping from ${}^wP$ to ${}^Ip$ can be summarized as

$$\lambda {}^I\bar{p} = K_c[I_3|\mathbf{0}^T]{}^cH_w {}^w\bar{P}, \tag{1.3}$$

where λ is an homogeneous scale factor, and ${}^I\bar{p}$ and ${}^w\bar{P}$ are the homogeneous representation of vectors ${}^Ip$ and ${}^wP$.

With this assumption, the image feature velocities are derived using (1.1) and (1.2), usually referred to as *image Jacobian*

$$
{}^I\dot{p} = \begin{bmatrix} {}^I\dot{x} \\ {}^I\dot{y} \end{bmatrix} = \begin{bmatrix} \frac{{}^Ix\,{}^Iy}{\alpha_y\,{}^wy} & \frac{{}^Ix^2 + \alpha_x^2}{\alpha_x} \\ \frac{{}^Iy^2}{\alpha_y\,{}^wy} & \frac{{}^Ix\,{}^Iy}{\alpha_x} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \tag{1.4}
$$

In the visual servoing literature, whenever an eye–in–hand configuration is considered (as is a camera rigidly fixed on a moving platform), the objective of the control task is to stabilize the robot towards the desired position controlling the camera position ([7, 8, 26]). More precisely:

Definition 1. Given the desired and the current robot positions, which correspond the desired $\langle C \rangle_d = \{O_{cd}, X_{cd}, Y_{cd}, Z_{cd}\}$ and the current $\langle C \rangle_c = \{O_{cc}, X_{cc}, Y_{cc}, Z_{cc}\}$ reference frames respectively, the stabilization in the desired position is accomplished if $\langle C \rangle_c \equiv \langle C \rangle_d$ at the end of the control task.

Remark 1. Since the mobile robot is moving constantly on a plane, the visual servoing control approach is used only to stabilize at most a 3D subspace of the state space $\xi \in \mathbb{R}^n$.

As it is customary in the visual servoing literature, by a suitable robot state variables change of coordinates, $\langle W \rangle \equiv \langle C \rangle_d$. Hence, the visual servoing control problem turns into a *point–to–point stabilization* problem, i.e., we require that $\xi(t) \to 0$ as $t \to +\infty$. In the particular framework proposed in this chapter, $\langle W \rangle \equiv \langle C \rangle_d$ corresponds to choosing $X_w = Z_{cd}$, $Y_w = Y_{cd}$ and $Z_w = -X_{cd}$.

A visual servoing scheme for robot control relays on the straightforward reformulation of Definition 1 from the use of image feature positions (albeit the equivalence between the two problems holds only if singularity configurations are avoided, see [7, 29]).

Definition 2. Given n current $F_c = [{}^I x_{c_1}, {}^I y_{c_1}, {}^I x_{c_2}, \ldots, {}^I y_{c_n}]^T$ *image feature* positions, the servoing task is fulfilled if at the end of the controlled trajectory, F_c matches the desired image feature positions $F_d = [{}^I x_{d_1}, {}^I y_{d_1}, {}^I x_{d_2}, \ldots, {}^I y_{d_n}]^T$, i.e., ${}^I x_{d_i} = {}^I x_{c_i}$ and ${}^I y_{d_i} = {}^I y_{c_i}$, $\forall i = 1, \ldots, n$.

In the presented chapter, we consider the visual servoing with an explicit feasibility constraint: the image features must be always within the FOV of the camera along the robot stabilizing trajectories (henceforth referred to as the *FOV constraint*), which ensures that a visual feedback can be always performed. Probably, the problem of keeping the features in view during the robot manoeuvres is one the most relevant problem to address for effective robot control. Multiple solutions have been proposed in literature, ranging from omnidirectional cameras ([4]), image path planning ([12]), or switching visual servoing schemes ([10]). In the presented dissertation, the FOV constraint will be addressed in the controller design, which simplifies the mechanical set–up and lowers the overall cost. Moreover, we will not focus on image processing backgrounds, giving for granted the feature extraction, tracking and association among the image features. In particular, we adopt the well–established *Scale Invariant Feature Transform* (SIFT) proposed in [24].

1.2.1 Position Based Visual Servoing

The main attractive feature of the PBVS approach is probably the relative simplicity of the control design. Indeed, the control law can be synthesized in the usual working coordinates for the robot ([9]). Unfortunately, a position estimation algorithm has to be necessarily provided. More precisely, since the estimated robot posture $\hat{\bar{\xi}}(t)$ can be derived by the knowledge of $^c\hat{H}_w(t)$ (at least for the subspace of interest), the objective of the estimation algorithm, often dubbed *localization* or *pose estimation* algorithm, is to reconstruct the relative position between $\langle C \rangle_c$ and $\langle W \rangle$.

Notice that if $^c\hat{H}_w(t)$ is composed of a rotation R and a translation T, (1.3) is rewritten as $\lambda^I\bar{p} = K_c[R|T]^w\bar{P}$. The fundamental matrix F associated with this projective map is defined as $F = K_c^T S(T) R K_c^{-1}$, where $S(T)$ is the skew-symmetric matrix associated with the translation T. Trivially, if ^Ip_d is the feature position in the desired camera position, we have $\lambda^I\bar{p}_d = K_c[\mathbf{I}|0]^w\bar{P}$, with the fundamental matrix F satisfying the condition $^I\bar{p}_d F^I\bar{p} = 0$. Therefore, assuming a calibrated camera (K_c is known), the fundamental matrix can be robustly estimated knowing image features correspondences $^Ip \leftrightarrow {}^Ip_d$ ([21]). Therefore, two main design requirements are derived:

1. full camera calibration (as implicitly assumed in the rest of the chapter);
2. a priori knowledge of ^wP for each feature.

The latter condition is not strictly needed if the algorithm that performs the servoing works in agreement with a mapping algorithm. In this case, the visual servoing scheme can be applied to previously unknown portion of the environment, thus increasing the autonomicity of the mobile vehicle. The *Visual Simultaneous Localization And Mapping for Servoing* (VSLAM for Servoing) is an example of such an architecture. In fact, since the PBVS approach abstracts sensor information to a higher level of representation, it allows the integration of different sensorial sources, thus make it suitable to cooperate with SLAM based architectures. In our example of a camera mounted on a mobile robot, for instance, the synergistic use of odometry and visual feedback is viable if these information are described in the same coordinate frame, where they can be fused coherently.

Summarizing, the PBVS approach estimates the error between the current and desired robot position through an approximation of $^c\hat{H}_w(t)$. For this reason, we can conclude that the PBVS approach verifies the Definition 1, which implies the satisfaction of Definition 2 in accordance with the accuracy of the estimation algorithm.

1.2.2 Image Based Visual Servoing

IBVS and other sensor-level control schemes have several advantages in relation to PBVS, such as robustness (or even insensitivity) to modelling errors, and hence suitability to unstructured scenes and environments ([27]). Although IBVS is demonstrated to be quite effective for manipulators ([31]), its control design turns out to

be more challenging for nonholonomic mobile vehicles. Indeed, the *image Jacobian* (1.4), which relates image features and robot's motion, cannot be used to solve the general stabilization problem directly, a fact reported in famous results on nonholonomic systems stabilization ([6]). However, as presented here, the classical visual servoing scheme can be used together with path planners on image maps ([7]). Such maps can be either given a priori or constructed on–line, according to a VS-LAM approach.

For IBVS the image error between two different postures is computed directly on the image measurements. Hence, an IBVS controller relates to Definition 2, which implies the satisfaction of Definition 1 if pathological image feature postures (singularities) are avoided.

1.2.3 Optimal trajectories

Path planning solutions to vision–based parking have been proposed in literature ([26]). Among all the possible choices for robot trajectories, a particular mention deserves those one that are optimal, e.g., minimum length. Moreover, optimal path planning turns to be particularly challenging when the nonholonomicity of the platform combines with the FOV constraint. A solution to this problem has been provided very recently by [5], where shortest paths are shown to be comprised of three maneuvers: a rotation on the spot, a straight line and a logarithmic spiral. The results there proposed are additionally refined in [33], where a global partition of the motion plane in terms of optimal trajectories has been derived.

As the optimal trajectories are retrieved, a visual servoing scheme can be used to control the robot along the optimal trajectories. For example, [23] proposes a PBVS homography–based controller. Alternatively, IBVS approaches can be used, as reported in what follows.

1.3 PBVS in the Large

As described previously, the PBVS comprises two major components that are described briefly in what follows: a localization algorithm and the controller design. The quantities referred in this section are reported in Fig. 1.1(a), where the fixed frame $\langle W \rangle$ and the camera frame $\langle C \rangle$ are reported.

1.3.1 Robot Localization

Let $^w\xi = [\xi_1, \xi_2, \xi_3]^T$ be the set of Cartesian coordinates of the robot (see Fig. 1.1(A)). The current position of the feature ^cP_i in the camera frame is related to ^wP_i in the
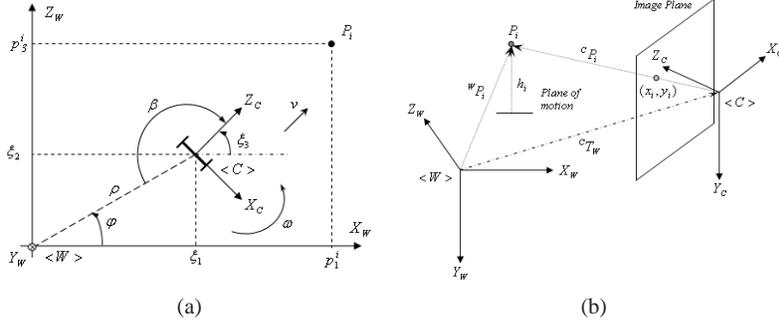
Fig. 1.1 (a) Fixed frame $\langle W \rangle$, camera frame $\langle C \rangle$, and relative coordinates (ξ_1, ξ_2, ξ_3) and (ρ, ϕ, β); and (b) fixed frame $\langle W \rangle$, camera frame $\langle C \rangle$ and relative feature coordinates.

fixed frame by a rigid-body motion ^cH_w, which can be computed assuming the knowledge of the height of the features ^cy_i, $\forall i$. Indeed, with respect to Fig. 1.1(B), we have

$$\begin{bmatrix} ^cx_i \\ ^cz_i \end{bmatrix} = \begin{bmatrix} ^wz_i & ^wx_i & 1 & 0 \\ -^wx_i & ^wz_i & 0 & 1 \end{bmatrix} b, \tag{1.5}$$

with $b = [-\cos\xi_3, \sin\xi_3, \xi_2\cos\xi_3 - \xi_1\sin\xi_3, -\xi_1\cos\xi_3 - \xi_2\sin\xi_3]^T$. Equation 1.5 can be regarded as providing two nonlinear scalar equations in the 3 unknowns (ξ_1, ξ_2, ξ_3), for each feature observed in the current and desired images. Assuming a number $n \geq 4$ of features, the actual unknown position and orientation $\xi(t)$ of the unicycle can be evaluated by solving for b in a least-squares sense (see [29] for further details).

The localization method thus presented is based on punctual estimation and does not take into account the known robot motion model. To better exploit the available information, an extended Kalman filter (EKF) is adopted, given for granted the association between current and desired point features. However, should feature outliers occur in the process, more robust filtering should be used in place of simple EKF, such as e.g. those described in [35] or [25].

The EKF localization state is $S = [\xi_1, \xi_2, \xi_3, ^cx_1, \dots, ^cx_n, ^cz_n]^T$. Estimated state initial guess is computed using the previously presented least mean squares static localization. The initial model covariance matrix P_0 is block diagonal. $P_0^r \in \mathbb{R}^{3\times3}$, i.e., initial covariance matrix for the vehicle, has its values set to half a meter for ξ_1 and ξ_2, while it is one radian for ξ_3. $P_0^i \in \mathbb{R}^{2\times2}$, i.e., the block diagonal matrices for the features, have their entries set to ≈ 20 centimeters in our experimental setting. Unicycle kinematic model, with odometric data, is assumed for state prediction.

Two different noise sources are taken into account. The prediction errors are modelled as additive and zero mean Gaussian noises with covariance matrix $Q = P_0$. Systematic errors are assumed to be removed by suitable calibration, hence nonzero mean errors are not modelled. The odometry errors γ_r, γ_l, for the right and left wheel respectively, are assumed to be zero mean and Gaussian distributed. They are also

assumed to be equal for both wheels. This is a simple but easily verified assumption in respect to generic unicycle like vehicles, and is computed taking into account lack of accuracy in odometry (typically due to wheels slipping and skidding). The covariance matrix of the prior estimate (model prediction) is then calculated by the formula

$$P_k^- = A_k P_{k-1} A_k^T + \sigma_\gamma^2 B_k B_k^T + Q,$$

where σ_γ^2 is the input variance ($\gamma = \gamma_r = \gamma_l$ by assumption), A_k and B_k are the model Jacobians and P_{k-1} is the model covariance matrix at the previous step.

During experiments, the EKF–based localization thus derived shows to outperform the least mean squares approach, as it may be expected.

1.3.2 Visual-Servoing with Omnidirectional Sight

Consider that only one feature has to be tracked, coincident with the origin O_w. Consider for this problem a new set of coordinates, which is better suited to describe the angle by which the feature is observed from the vehicle, described by $\Phi : \mathbb{R}^2 \times S \to \mathbb{R}^+ \times S^2$ (see Fig. 1.1(a)) and the new dynamics

$$\begin{bmatrix} \rho \\ \phi \\ \beta \end{bmatrix} = \begin{bmatrix} \sqrt{\xi_1^2 + \xi_2^2} \\ \arctan(\frac{\xi_2}{\xi_1}) \\ \pi + \arctan(\frac{\xi_2}{\xi_1}) - \xi_3 \end{bmatrix}, \text{ and } \begin{bmatrix} \dot{\rho} \\ \dot{\phi} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\rho\cos\beta & 0 \\ \sin\beta & 0 \\ \sin\beta & -1 \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix}, \quad (1.6)$$

where we let $u = \frac{v}{\rho}$. Observe that this change of coordinates is a diffeomorphism everywhere except at the origin of the plane (exactly where the feature point is). A continuous, time-invariant control law can in principle stabilize the system (1.6) – indeed, the two control vector fields are now linearly dependent at the origin, thus making Brockett's negative result [6] unapplicable.

Consider the candidate Lyapunov function $V = \frac{1}{2}(\rho^2 + \phi^2 + \lambda\beta^2)$, with $\lambda > 0$ a free parameter to be used in the following controller design. Substituting

$$u = \cos\beta, \text{ and } \omega = \frac{\phi\sin\beta\cos\beta + \lambda\beta\sin\beta\cos\beta}{\lambda\beta} + \beta \quad (1.7)$$

in the Lypunov derivative one gets $\dot{V} = -\rho^2\cos^2\beta - \lambda\beta^2 \leq 0$. By using LaSalle's invariant set theorem, the controlled dynamics obtained plugging (1.7) in (1.6) turns to be asymptotically stable.

1.3.3 Visual Servoing with FOV Constraint

The basic idea to be applied in this section is rather simple, and is based on the fact that the Lyapunov-based control described in the previous section is not uniquely defined. Rather, a whole family of controllers can be defined by simply redefining the control Lyapunov function candidate. It can be expected that for such different candidates, the resulting stabilizing control laws and ensuing trajectories are different, and that switching among these control laws should be enabled when the FOV constraint for the i–th feature

$$\gamma(\rho,\phi,\beta) = \phi - \pi - \beta - \arctan\frac{{}^w z_i - \rho\sin\phi}{{}^w x_i - \rho\cos\phi} = \arctan\frac{{}^I x_i}{\alpha_x} \in [-\Delta,\Delta] \qquad (1.8)$$

is about to be violated (see [29] for further details). Notice that the limited FOV is described by a symmetric cone centered in the optical axis Z_c with semi-aperture Δ.

The switching controller is expressed in a set of different polar coordinates, which are conveniently denoted by introducing the two vectors $\tilde{\beta} = [\beta, \beta - \pi, \beta - \pi, \beta + \pi, \beta + \pi]$ and $\tilde{\phi} = [\phi - \pi, \phi - 2\pi, \phi, \phi - 2\pi, \phi]$. Correspondingly, a set of five distinct candidate Lyapunov functions can be written as $V_i(\rho,\alpha,\beta) = \frac{1}{2}(\rho^2 + \tilde{\phi}_i^2 + \tilde{\beta}_i^2)$, with $i = 1,\ldots,5$. The control law choice, i.e.

$$u = \cos\beta, \text{ and } \omega = \lambda\tilde{\beta}_i + \frac{\sin\beta\cos\beta}{\tilde{\beta}_i}(\tilde{\phi}_i + \tilde{\beta}_i),$$

is such that all the Lyapunov candidates have negative semi-definite time derivatives and (by LaSalle's invariant set principle) asymptotically stable. These five different control laws (parameterized by λ) define in turn five different controlled dynamics (analogous to (1.6)) that are globally asymptotically stable in the state manifold $\mathbb{R}^+ \times S^2$. Although none of these control laws alone can guarantee that the FOV constraint is satisfied throughout the parking maneuver, it is shown that a suitable switching logic among the control laws achieves this goal. The switching law is triggered when, during the stabilization with one of the five control laws, a feature approaches the border of the field of view by a threshold $\Delta_j < \Delta$, i.e. when $|\gamma| \geq \Delta_j$. It should be noticed that a dead zone is introduced in the controller for $\rho \leq \rho_D$, within which the forward velocity control u is set to zero and the so-called Zeno phenomenon are avoided.

1.3.4 Visual Servoing in the Large

The classic approach for visual servoing, however, so far have focussed on local stabilization, in the sense that the initial and desired conditions of the system are assumed to be close enough so that a significant number of features remain in view all along the maneuver. The purpose of this section is to define the tools that enables
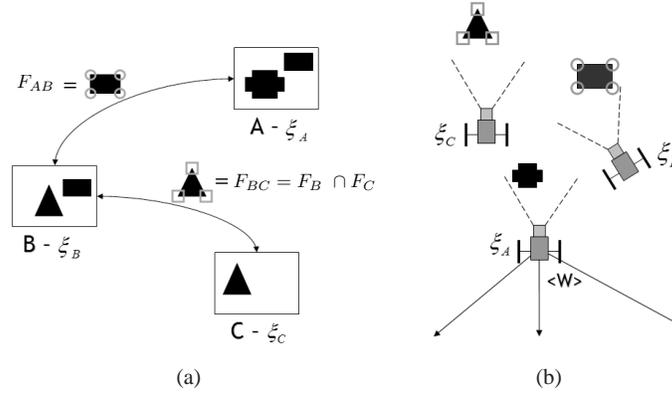
Fig. 1.2 (a) Topological map: grabbed images are indicated with a capital letter, say I_A, I_B, I_C; each grabbed image corresponds to both a certain robot configuration in the metric map (b) and a node in the topological map; the nodes A and B are connected if and only if the set of features $F_{AB} = F_A \cap F_B$ is not empty; and (b) metric map: the positions A, B, C, representing each image node in the topological map (a), ξ_A, ξ_B and ξ_C are a set of 3D robot postures.

its use to servo the vehicle *in the large*, i.e. across paths connecting totally different initial and final views. The necessary information are stored in images, called *way–points*, which can be used to topologically connect the initial and desired images, and in a *metric* map, which stores sufficient data to implement the PBVS. A representation of the environment that conveys these metric and topological information will be referred to as a *hybrid visual map*.

The literature on the problem of simultaneous visual–based localization and map building (v-SLAM) is rather extensive (see e.g. [30, 13, 15, 32]), and COTS software is already available ([22]). These results are clearly fundamental for the approach to servoing here presented, which instead relates to nonholonomic vehicles. In the hybrid map representation, the metric information is represented by a set of robot postures, along with the corresponding 3D position estimates for the features observed from such postures. The topological information is represented by an undirected reachability graph (we assume indeed that possible environment changes do not affect the traversability of the space by the robot, [17]). The hybrid map construction method is described in what follows:

1. From the initial unknown position of the vehicle (i.e. $^W \xi_A = {}^W[0,0,0]^T$) an image I_A of a portion of the scene in view is grabbed and stored in the first node A of the hybrid map (see Fig. 1.2(a))).
2. From the image in view, a subset F_A of n_A features is selected.
3. The vehicle moves, avoiding obstacles with proximity sensors, in an arbitrary direction using a simple control law that keeps the image point features in view.
4. An EKF is implemented using odometry and camera measurements to estimate the relative spatial position of the feature in camera frame $\langle C \rangle$.

5. Once 3D feature position estimates have converged to a value under given level of uncertainty determined by the covariance matrix, the robot stops moving, updates the metric map (Fig. 1.2(b)) and then a new node corresponding to the current pose is added to the hybrid map.
6. To add new nodes from the already created ones, the procedure starts again from step 2.

Let the robot be in a generic mapped position, say $^{W}\xi_{A} = {}^{W}[\xi_{1}, \xi_{2}, \xi_{3}]^{T}$ (or a node A with image I_{A}). Suppose that the robot has to reach a new position, say ξ_{K}, expressed in the metric map. If $^{W}\xi_{K}$ corresponds to a topologically mapped location K, which has an associated image I_{K} (found using SIFT technique), a standard graph visiting algorithm is used for the path selection from A to K in the image map, therefore, permitting the vehicle to steer through the map nodes using the servoing presented previously. For space limits, we refer the interested readers to [17] for further details on the mapping/navigation processes here briefly discussed.

1.3.5 Experimental Results

Two experiments, with different experimental set–up are described. The first experimental setup comprises of a TRC LabMate vehicle, equipped with an analogical monochromatic camera Jai CVM-50 [2] placed on the robot so that a vertical axis through the camera pinhole intersects the wheel axis in the midpoint. The camera allows for a $\Delta = \pi/6$ semi-aperture of the optical cone, while a threshold of $\Delta_{j} = \pi/8$ was used in the experiment to switch among different controllers. The controller is implemented under Linux on a 300MHz PentiumII PC equipped with a Matrox MeteorI frame grabber. The XVision library is used to compute optical flow and to track features. The hardware communication between the robot and the PC is performed by a RS-232 serial cable. In this experiment, the PBVS controller alone is shown. Four features from the scene are used as the desired feature set, belonging to the desired image, recorded in a preliminary phase of the experiment (see Fig. 1.3). Images grabbed from the robot camera in the initial, offset configuration and at the end of the visually-servoed parking maneuver are shown in Fig. 1.4, along with ground-reference views showing the experimental environment.

In the second experimental set–up, a low-cost apparatus was employed to highlight the potential of the proposed technique. The experimental setup is comprised of a K-Team Koala vehicle [3], equipped with a commercial web–cam placed on the front part of the robot platform. The controller is implemented under Windows XP on a 1130MHz Pentium III laptop mounted on-board. SIFT elaboration is performed using ERSP vision library (see [20, 22]). The Intel OpenCV [1] library was used to compute optical flow and to track features. The hardware communication between the robot and the laptop is again performed by a RS-232 serial cable.

In the experiments both the mapping phase (topological and metric) and the navigation phase (visual servoing) are reported. In the mapping experiment, the robot collects a set of images and, for each pair of images, say I_{i} and I_{j}, it estimates
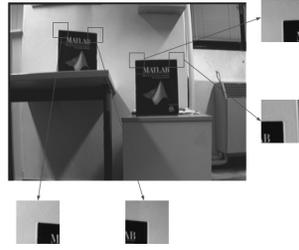
Fig. 1.3 Image grabbed from the target position, with the four selected control features shown in detail.

Fig. 1.4 External views (left column) and subjective images (right column) as taken from the vehicle, in the initial configuration (top row) and in the final configuration (bottom row), after reaching convergence under the proposed visual feedback control scheme. The bottom right image should be compared with the target image in fig. 1.3.

the three–dimensional coordinates of the image feature points F_{ij} and the three–dimensional robot positions ξ_i and ξ_j.

The experimental mapping process runs for 50 seconds and the sampling period (i.e. the inverse of the frequency of the EKF steps) is $T = 0.1$ seconds. The sampling period T is determined by the worst-case frame rate available for commercial webcams. Although even low cost cameras ensure about 20 to 30 frames per second, the rate changes depending on ambient illumination variations.

Fig. 1.5 reports an image graph created during the topological mapping phase.

The visual servoing controller has been used to travel the distance between the mapped images, parking the vehicle in the position x_T^3 (position I_3 in the topological map, see fig. 1.6(A). The initial robot position is unknown, but the architecture solves the kidnapped robot problem identifying the topological position x_T^5 (see
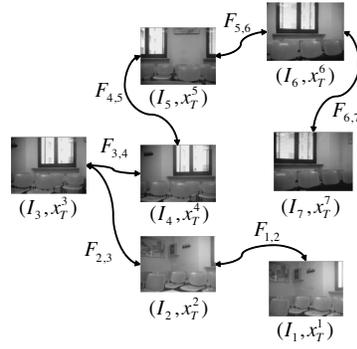
Fig. 1.5 The topological image–based map.

(a)   (b)

Fig. 1.6 Map navigation: (a) desired image; and, initial image (b).

fig. 1.6(B). Hence, the visual servoing path corresponds to a travel between image node I_5 to I_3.

In fig. 1.7, the nodes crossed by the robot during the parking are represented. In the top row, the images stored in the topological map (i.e. desired images for the visual servoing) are represented, while in the bottom row, the images grabbed from the camera after each path are depicted. Approaching an intermediate node ends once it is possible to localize and to track features of the next node to be reached.

A wide movement in the mapped environment comprises several limited movements between each pair of images (fig. 1.7). Nevertheless, the visual servoed motion between successive images is still quite small. Indeed, it is well known in the literature (e.g. [7, 8]) that large image errors (hence, large robot movements) decrease accuracy and robustness of the visual servo controller. In the proposed architecture, the granularity of the topological map is then related to the visual servoing accuracy.

Fig. 1.7 Desired images from each topological map node on the top row. Images grabbed from the camera after each visually servoed path, on the bottom row.

1.4 Optimal trajectories

This section presents a solution to the path planning problem of a unicycle–like vehicle subject to limited FOV constraint. In particular, the paths derived are the minimum length (optimal) paths between initial and desired positions.

Assume that the tracked feature position is coincident with the origin of the reference frame O_w and that the total horizontal aperture of the camera's optical cone is given by $\delta = 2\Delta$. The desired position P of the robot on the plane of motion w.r.t. $\langle W \rangle$ is assumed to lie on the X_w axis, with polar coordinates $(\rho_P, 0)$. Hence, the desired state of the robot is $(\rho_P, 0, \pi)$ with respect to the model (1.6). The objective of the optimal trajectory problem with FOV constraint is the partition of the motion plane into regions, depending on the desired position P. From all initial positions Q inside a region, the structure of the shortest paths turns to be invariant.

Denote with "$*$" a zero–length rotation on the spot, with SL a straight line and with T a logarithmic spiral. The three different kinds of maneuvers that compose the optimal paths, according to [5, 33], are then symbolically described. While the mathematic description of $*$ and SL is straightforward, logarithmic spirals deserve some additional explanations. In general, the equation of the logarithmic spiral that passes through a point Q (whose polar coordinates are (ρ_Q, ψ_Q) w.r.t. $\langle W \rangle$) is expressed as $T_Q : \left(\rho_Q e^{(\psi_Q - \psi)g}, \psi \right)$, where $g = \frac{\cos \alpha}{\sin \alpha}$ and α is the spiral's characteristic angle. The spiral rotates clockwise if $\alpha < 0$ (denoted with $T1_Q$) and counterclockwise if $\alpha > 0$ (denoted with $T2_Q$). Notice that if $\alpha = 0$, the logarithmic spiral is a straight line passing through the origin O_W, and, if $\alpha = \pm\pi/2$, it is a circumference.

Using standard tools from optimal theory, it can be shown that optimal words that minimizes the total length of the path
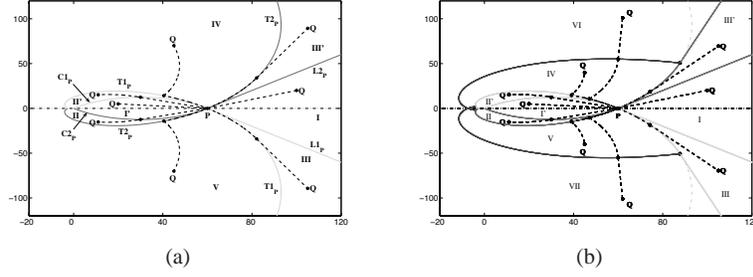
Fig. 1.8 Shortest paths: (a) according to [5]; and (b) according to [33].

$$\mathscr{L} = \int_0^T |v|\, dt \, ,$$

are words of extremal arcs that can be covered forward or backward w.r.t. the direction of motion. Summarizing, extremal paths will be characterized by sequences of symbols $\{*, SL, T1, T2\}$.

Let us denote the two logarithmic spirals that pass through the desired position P as $T1_P$ and $T2_P$, with characteristic angles $\alpha = -\Delta$ and $\alpha = \Delta$, respectively. The two spirals divide the plane into four disjoint regions. Using the geometric properties of the problem, these regions are further subdivided considering the set of points Q for which the optimal path is given by a straight-line from Q to P without violating the FOV constraints ([5]). With this intuitive subdivision, eight regions are derived, depicted in Fig. 1.8(a). Defining a smooth transition between segments with the symbol "–", the optimal path from each region is defined as

- *Region I*: region between the half–lines $L1_P$ and $L2_P$. Shortest path: SL;
- *Region I'*: region between the two arc circles $C1_P$ and $C2_P$. Shortest path: SL;
- *Region II*: region between the circle arc $C2_P$ (bound of *Region I'*) and the spiral $T2_P$ (bound of *Region V*), symmetric to *Region II'*. Shortest path: $T2 - SL$;
- *Region III*: region between the half–line $L1_P$ (bound of *Region I*) and the spiral $T1_P$ (bound of *Region V*), symmetric to *Region III'*. Shortest path: $SL - T1_P$;
- *Region IV*: region between the spiral $T2_P$ (bound of *Region III'*) and the horizontal line passing through the feature and the desired position P (bound of *Region V*). The *Region V* is symmetric to *Region IV*. Shortest path: $T2 * T1_P$;
- *T Region*: region defined by $T1_P$ and $T2_P$.

However, in [33] it has been shown that the previous taxonomy is only locally valid, i.e. near the desired configuration. As a consequence, Regions II, II', IV, V have to be further subdivided. Indeed, there exists a subspace of Regions IV and V, for which the shortest paths are $SL - T2 * T1_P$ or $SL - T1 * T2_P$ (see Regions VI and VII in Fig. 1.8(b)), and a subspace of of Regions II, II' where the shortest paths are of kind $T1 * T2 - SL$ or $T2 * T1 - SL$.

1.4.1 IBVS and Optimal Paths

As the optimal paths are described and a taxonomy of the motion plane is derived, a visual–based controller able to track the optimal trajectories is needed in order to close the loop. Even though PBVS solutions can be adopted ([23]), the proposed IBVS solution avoids localization algorithms and the inherit intrinsic higher robustness with respect to PBVS approaches ([8]).

The IBVS scheme here proposed is feasible if: 1) optimal paths can be computed using only visual information and 2) once the vehicle optimal 3D words are "translated" to the image space. For the former point, it can be shown that only the orientation Ω between the initial position Q and the desired one P is needed ([33]). Since Ω is defined as $\int_0^\tau \omega(t)dt = \Omega$ along the optimal path, it can be estimated using epipolar geometry and the fundamental matrix F if at least eight feature points (in a non singular configuration) are given ([21]). Estimation robustness can be achieved if even more features are used.

1.4.1.1 Trajectories on the Image Plane

From the definition of the optimal paths in Section 1.4, it follows that only three specific kind of robot maneuvers needs to be translated: pure rotations, i.e., $v = 0$, pure translations, i.e., $\omega = 0$, and logarithm spirals. In what follows we assume that $^Ip_i = [^Ix_i, {}^Iy_i]^T$, $^Ip_c = [^Ix_c, {}^Iy_c]^T$ and $^Ip_d = [^Ix_d, {}^Iy_d]^T$ are the feature initial, current and desired positions respectively.

Pure Rotation: by plugging into the image Jacobian (1.4), the condition $v = 0$ and $\omega = \bar{\omega} = $ constant and solving for the ^Iy_c feature coordinate, yields to

$$^Iy_c = \frac{{}^Iy_i \cos\left(\arctan\left(\frac{{}^Ix_i}{\alpha_x}\right)\right)}{\cos\left(\arctan\left(\frac{{}^Ix_c}{\alpha_x}\right)\right)}, \tag{1.9}$$

that is the equation of a conic, i.e., the intersection between the image plane and the cone with vertex in the camera center (optical center) and base circumference passing through the 3D feature position. On the other hand, solving for ^Ix_c, one gets

$$\theta - \theta_0 = \arctan\left(\frac{{}^Ix_d}{\alpha_x}\right) - \arctan\left(\frac{{}^Ix_i}{\alpha_x}\right), \tag{1.10}$$

where θ_0 is the initial value of θ, the robot orientation. Therefore, the angle variation can be computed for pure rotations by means of (1.10).

Pure Translation: by plugging into the image Jacobian (1.4), the condition $\omega = 0$ and $v = \bar{v} = $ constant and solving for the ^Iy_c feature coordinate, yields to

$$^Iy_c = \frac{{}^Iy_i}{{}^Ix_i}{}^Ix_c, \tag{1.11}$$

(a)   (b)   (c)
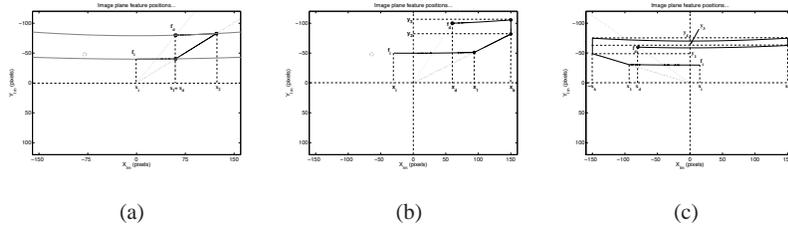
Fig. 1.9 Optimal paths on image plane: (a) unconstrained optimal path in Region I; (b) constrained optimal path in Region III; and (c) path for one feature in Region VII.

which describes a straight line passing through the initial position of the feature and the principal point.

Logarithmic Spiral: consider the condition ${}^I\dot{x}_c = 0$, i.e., ${}^I x_c = {}^I\bar{x}_c =$ constant. Using (1.4), the condition reflects in

$$\omega = -\frac{\alpha_x}{\alpha_y \, {}^w y} \frac{{}^I\bar{x}^2 + \alpha_x^2}{{}^I\bar{x} \, {}^I y} v. \tag{1.12}$$

Since ${}^I x_c$ is constant, if follows that the angle γ in (1.8) between the forward direction of the vehicle and the feature direction is constant. Since this is the condition verified by a logarithmic spiral path, the condition (1.12) identifies such a path.

1.4.1.2 From Image paths to Servoing

Once the alphabet of feature sub-paths has been defined, the rules to choose and construct the right sequence of maneuvers (the word) that correspond to the shortest path must be defined directly on the image plane. To this end, path of increasing complexity, i.e., increasing number of symbols, are computed. If a path is feasible, i.e., the feature path remains constrained inside the image view, than it is chosen. Otherwise, it is assumed that the robot state pertains to a different region in space and the algorithm is reiterated. The path solutions derived in Section 1.4.1.1 are computed numerically using the Levenberg-Marquardt algorithm ([33]).

The optimal feature path for *Region I* or *Region I'* is shown in Fig. 1.9(a) and is composed of: a piece of a conic passing through the initial position of the feature (rotation); a piece of straight line passing through the principal point (translation); and a piece of conic passing through the final position of the feature (rotation). Given that Ω is known (or estimated), then the path can be computed solving the following equation

$$\theta_i + \theta_d = \Omega = \left[\arctan\left(\frac{{}^I x_1}{\alpha_x}\right) - \arctan\left(\frac{{}^I x_i}{\alpha_x}\right)\right] + \left[\arctan\left(\frac{{}^I x_d}{\alpha_x}\right) - \arctan\left(\frac{{}^I x_2}{\alpha_x}\right)\right],$$

where θ_i and θ_d are the orientation of the vehicle in the initial and final positions, the variables $^{I}x_1$ or $^{I}x_2$ correspond to the intermediate (unknown) positions of the feature along the path and are the initial and final feature position for the intermediate straight path.

The optimal feature path for the *Region III* is depicted in Fig. 1.9(b) and it is determined solving the following equation

$$\Omega = \left[\arctan\left(\frac{^{I}x_1}{\alpha_x}\right) - \arctan\left(\frac{^{I}x_i}{\alpha_x}\right)\right] +$$
$$+ \left[\frac{x_b}{\alpha_x}\ln\left(\frac{^{I}y_2}{^{I}y_3}\right)\right] + \left[\arctan\left(\frac{^{I}x_d}{\alpha_x}\right) - \arctan\left(\frac{x_b}{\alpha_x}\right)\right],$$

where the second addendum in the second member corresponds to the angle variation of the robot's orientation θ during the spiral path. x_b is the horizontal image bound.

Finally, the optimal feature paths that correspond to the regions *IV, V, VI* or *VII* are plotted in Fig. 1.9(c). The solution in this case is given by:

$$\Omega = \left[\arctan\left(\frac{^{I}x_1}{\alpha_x}\right) - \arctan\left(\frac{^{I}x_i}{\alpha_x}\right)\right] + \left[\frac{x_b}{\alpha_x}\ln\left(\frac{^{I}y_2}{^{I}y_3}\right)\right] +$$
$$+ 2\left[\frac{x_b}{\alpha_x}\ln\left(\frac{^{I}y_3}{^{I}y_4}\right)\right] + \left[\arctan\left(\frac{^{I}x_d}{\alpha_x}\right) - \arctan\left(\frac{x_b}{\alpha_x}\right)\right].$$

Notice that, for each $^{I}y_4$, there exists a valid image path. To disambiguate, the choice of the optimal path in this last case can be done only by the use of its 3–D reconstruction (since both the path $T1 - T2_P$ or $SL - T1 - T2_P$ are feasible). It is worthwhile to note that such a reconstruction does not need to be exact, since a scaled one is sufficient ([33]).

Finally, the robot's proportional control laws $u = (v, \omega)$ are obtained using Lyapunov theory. For each image trajectory component a different Lyapunov function is chosen in order to minimize the feature tracking error.

1.4.2 Experimental Results

Experimental results for the IBVS and optimal path framework are now presented. The experimental setup comprises of a Quickcam Ultravision camera, whose resolution is 320x240 pixels, mounted over the front-part of a K-team Koala vehicle. The ERSP vision library ([20]) is used to perform SIFT recognition. The controller bandwidth is almost 7 Hz.

In the first experiment, the optimal path SL reported in Fig. 1.9 (a) is accomplished. The path is composed of the following maneuvers: a counter–clockwise rotation on the spot (the image conic $^{I}x_i \rightarrow {}^{I}x_1$), a forward motion (the straight line $^{I}x_1 \rightarrow {}^{I}x_2$), and a clockwise rotation on the spot (the image conic $^{I}x_2 \rightarrow {}^{I}x_d$). Fig. 1.10 reports the related experimental result.

In the second experiment, the optimal path $SL - T1_P$ reported in Fig. 1.9 (b) is accomplished. The path is composed of the following sequence of maneuvers:
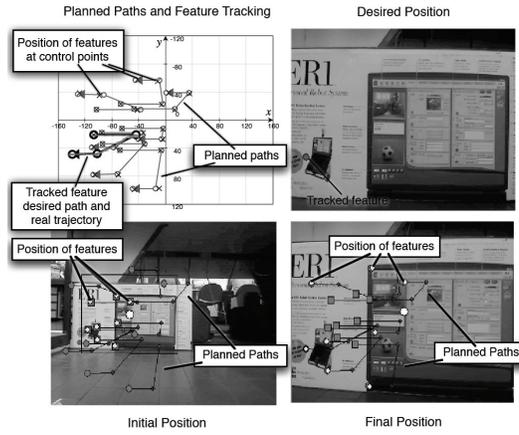
Fig. 1.10 Experiment 1. Planned paths for all features and the trajectory of the tracked feature (up left). Initial (bottom left), final (bottom right) and desired (up right) images taken from the vehicle. The planned paths and also the actual position of the features are plotted over the initial and final images.
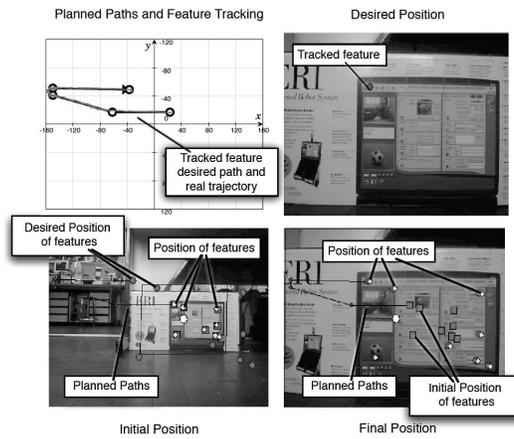
Fig. 1.11 Experiment 2. Planned paths and trajectory of the tracked feature (up left). Initial (bottom left), final (bottom right) and desired (up right) images taken from the vehicle. The initial and desired positions of the features are plotted over the initial and final images taken from the vehicle. The actual positions of the features are also shown in these images.

a counter–clockwise rotation on the spot (the image conic $^{I}x_i \rightarrow {}^{I}x_1$), a forward motion (the straight line $^{I}x_1 \rightarrow {}^{I}x_b$), a logarithmic spiral (the straight line $^{I}y_2 \rightarrow {}^{I}y_3$) and a clockwise rotation on the spot (the image conic $^{I}x_b \rightarrow {}^{I}x_d$). Fig. 1.11 reports the related experimental result.

The method accomplished the task of driving the tracked feature through the planned path. It can be observed from the figures that the final positions of the features are almost coincident with the desired positions. The final feature position error is due to the tracking noise and to the estimation of the angle Δ (that determines the feature trajectories). Hence, the final position of the robot is close to the desired one and the error is mostly due to a translation along the Y_w axis. The observed errors can be explained by image quantization deviations, presence of noisy data, camera specifications, low number of features used to compute the planned trajectory, estimation errors related to the SIFT recognition system, erroneous camera calibration parameters and, finally, the fact that the robot control was based on the tracking of just one feature in the image.

It is worthwhile to note that the previously presented PBVS VSLAM for Servoing can be compared to the IBVS controller with a VSLAM architecture. Indeed, due to the very nature of the proposed solution, pure appearance based visual maps can be adopted, without considering any 3D spatial information (i.e., metric maps). Due to space limits, the interested reader is referred to [18].

1.5 Conclusions

The problem of visually guide unicycle–like vehicles towards desired position in space has been analyzed in this chapter. Solutions here discussed ranges from PBVS to IBVS approaches applied "in the large", which differs in which image data are treated in the control procedure.

For position based, we have proposed a visual servoing scheme for a non–holonomic vehicle in unknown indoor environments. The proposed approach is divided into two phases: a servoing process and a map building for servoing process. The servoing hybrid control is based on a PBVS scheme that is also presented. Map construction complies with the need of overcoming the limits of the servoing scheme in a large environment. The work could be regarded as an attempt to connect control techniques (action) and sensorial data interpretation(perception).

Furthermore, a method that associates space regions to optimal vehicle trajectories, combined with a limited FOV camera, is presented. Then, optimal trajectory words are translated from 3D space to image space, in order to adopt an IBVS controller to track image feature trajectories. Notice that, the mobile robot autonomicity, i.e., the ability to move in a large environment, is increased using appearance based visual maps, in which metric data are not of interest.

Experiments on real nonholonomic robot platforms are reported for both approaches. The robot successfully reached the desired position while keeping the tracked feature inside the FOV. Moreover, even if the initial and the desired images do not have a common set of features, the servoing is still feasible using image data stored in the maps.

References

1. Intel Open Source Computer Vision Library.
   http://www.intel.com/research/mrl/research/opencv/
2. Jai CVM-50 Monochrome Camera.
   http://www.compumodules.com/imaging/cameras_list.shtml
3. K-Team Koala Robot.
   http://www.k-team.com/robots/koala/index.html
4. Benhimane, S., Malis, E.: A new approach to vision–based robot control with omni–directional cameras. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 526–531. Orlando, Florida, USA (2006)
5. Bhattacharya, S., Murrieta-Cid, R., Hutchinson, S.: Optimal paths for landmark–based navigation by differential–drive vehicles with field–of–view constraints. IEEE Transactions on Robotics **23**(1), 47–59 (2007)
6. Brockett, R.: Asymptotic stability and feedback stabilization. In: M. Brockett, Sussmann (eds.) Differential Geometric Control Theory, pp. 181–191. Birkhauser, Boston, U.S. (1983)
7. Chaumette, F., Hutchinson, S.: Visual servo control, Part I: Basic approaches. IEEE Robotics and Automation Magazine **13**(4), 82–90 (2006)
8. Chaumette, F., Hutchinson, S.: Visual servo control, Part II: Advanced approaches. IEEE Robotics and Automation Magazine **14**(1), 109–118 (2007)
9. Chen, J., Dawson, D., Dixon, W., Chitrakaran, V.: Navigation function-based visual servo control. Automatica **43**(7), 1165–1177 (2007)
10. Chesi, G., Hashimoto, K., Prattichizzo, D., Vicino, A.: A swiching control law for keeping features in the field of view in eye-in-hand visual servoing. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3929–3934. Taipei, Taiwan (2003)
11. Chesi, G., Hung, Y.: Global path-planning for constrained and optimal visual servoing. IEEE Transactions on Robotics **23**(5), 1050–1060 (2007)
12. Chesi, G., Hung, Y.: Visual servoing: a global path-planning approach. In: Proc. IEEE Intl. Conf. on Robotics and Automation, pp. 2086–2091. Roma, Italy (2007)
13. Chiuso, A., Favaro, P., Jin, H., Soatto, S.: Structure from motion casually integrated over time. IEEE Trans. on Pattern Analysis and Machine Intelligence **24**(4), 523–535 (2002)
14. Collewet, C., Chaumette, F.: Positioning a camera with respect to planar objects of unknown shape by coupling 2-d visual servoing and 3-d estimations. IEEE Trans. on Robotics and Automation **18**(3) (2002)
15. Davison, A.J.: Real–time simultaneous localization and mapping with a single camera. In: Proc. IEEE Int. Conf. on Computer Vision, vol. 2, pp. 1403–1410 (2003)
16. Deguchi, K.: Optimal motion control for image-based visual servoing by decoupling translation and rotation. In: IEEE/RJS Intl. Conf. on Intelligent Robots and Systems, pp. 705–711. Victoria, B.C., Canada (1998)
17. Fontanelli, D., Danesi, A., Belo, F., Salaris, P., Bicchi, A.: Visual Servoing in the Large. The International Journal of Robotics Research **28**(6), 802–814 (2009). DOI 10.1177/0278364908097660
18. Fontanelli, D., Salaris, P., Belo, F., Bicchi, A.: Visual appearance mapping for optimal vision based servoing. In: Experimental Robotics, *Springer Tracts in Advanced Robotics (STAR)*, vol. 54, pp. 353–362. Springer Berlin / Heidelberg (2009)
19. Gans, N., Hutchinson, S.: Stable visual servoing through hybrid switched system control. IEEE Transactions on Robotics **23**(3), 530–540 (2007)
20. Goncalves, L., DiBernardo, E., Benson, D., Svedman, M., Ostrowski, J., Karlsson, N., Pirjanian, P.: A visual front–end for simultaneous localization and mapping. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 44–49. Barcelona, Spain (2005)
21. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2003)
22. Karlsson, N., DiBernardo, E., Ostrowski, J., Goncalves, L., Pirjanian, P., Munich, M.E.: The vSLAM algorithm for robust localization and mapping. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 24–29. Barcelona, Spain (2005)

23. López-Nicolás, G., Bhattacharya, S., Guerrero, J., Sagüés, C., Hutchinson, S.: Switched homography-based visual control of differential drive vehicles with field-of-view constraints. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 4238–4244. Rome (2007)
24. Lowe, D.: Distinctive image features from scale-invariant keypoints. In: Int. Jour. of Computer Vision, vol. 20, pp. 91–110 (2003)
25. Lu, L., Dai, X., Hager, G.D.: Efficient particle filtering using RANSAC with application to 3D face tracking. Image and Vision Computing **24**(6), 581–592 (2006)
26. Mariottini, G.L., Oriolo, G., Prattichizzo, D.: Image–based visual servoing for nonholonomic mobile robots using epipolar geometry. IEEE Transactions on Robotics **23**(1), 87–100 (2007)
27. Mezouar, Y., Chaumette, F.: Path planning for robust image–based control. IEEE Transactions on Robotics and Automation **18**(4), 534–549 (2002)
28. Mezouar, Y., Remazeilles, A., Gros, P., Chaumette, F.: Images interpolation for image–based control under large displacement. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3787–3794. Washington, DC (2002)
29. Murrieri, P., Fontanelli, D., Bicchi, A.: A hybrid-control approach to the parking problem of a wheeled vehicle using limited view-angle visual feedback. Int. Jour. of Robotics Research **23**(4–5), 437–448 (2004)
30. Rekleitis, I., Sim, R., Dudek, G., Milios, E.: Collaborative exploration for the construction of visual maps. In: Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1269–1274 (2001)
31. Remazeilles, A., Chaumette, F., Gros, P.: Robot motion control from a visual memory. In: Proc. IEEE Int. Conf. on Robotics and Automation, vol. 5, pp. 4695–4700 (2004)
32. Royer, E., Bom, J., Dhome, M., Thuilot, B., Lhuillier, M., Marmoiton, F.: Outdoor autonomous navigation using monocular vision. In: Proc. of the IEEE/RSJ Conference on Intelligent Robots and Systems, pp. 1253–1258 (2005)
33. Salaris, P., Belo, F., Fontanelli, D., Greco, L., Bicchi, A.: Optimal paths in a constrained image plane for purely image–based parking. In: Proc. IEEE Int. Symp. Intelligent Robots and Systems, pp. 1673–1680. Nice, France (2008)
34. Sanderson, A.C., Weiss, L.E.: Image–based visual servo control using relational graph error signals. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1074–1077 (1980)
35. Vedaldi, A., Jin, H., Favaro, P., Soatto, S.: KALMANSAC: Robust filtering by consensus. In: Proc. of the Int. Conf. on Computer Vision (ICCV), vol. 1, pp. 633–640 (2005)