Visual navigation and servoing for object manipulation with mobile robots

DISSERTATION

submitted in partial fulfillment
of the requirements for the degree

Doktor Ingenieur
(Doctor of Engineering)

in the

Faculty of Electrical Engineering and Information Technology
at TU Dortmund University

by

Dipl.-Ing. Thomas Nierobisch
Schwäbisch Gmünd, Germany

Date of submission: 21th January 2014

First examiner: Univ.-Prof. Dr.-Ing. Prof. h.c. Dr. h.c. Torsten Bertram
Second examiner: Univ.-Prof. Dr.-Ing. Bernd Tibken

Date of approval: 22th June 2015

"Computer vision as a field is an intellectual frontier. Like any frontier, it is exciting and disorganised; there is often no reliable authority to appeal to - many useful ideas have no theoretical grounding, and some theories are useless in practice."

Forsyth and Ponce
Authors from Computer Vision: A Modern Approach

Abstract

In the future, autonomous service robots are supposed to remove the burden of monotonic and tedious tasks like pickup and delivery from people. Vision being the most important human sensor and feedback system is considered to play a prominent role in the future of robotics. Robust techniques for visual robot navigation, object recognition and vision assisted object manipulation are essential in service robotics tasks. Mobile manipulation in service robotics applications requires the alignment of the end-effector with recognized objects of unknown pose. Image based visual servoing provides a means of model-free manipulation of objects solely relying on 2D image information.

In this thesis contributions to the field of decoupled visual servoing for object manipulation as well as navigation are presented. A novel approach for large view visual servoing of mobile robots is presented by decoupling the gaze and navigation control via a virtual camera plane, which enables the visual controller to use the same natural landmarks efficiently over a large range of motion. In order to complete the repertoire of reactive visual behaviors an innovative door passing behavior and an obstacle avoidance behavior using omnivision are designed. The developed visual behaviors represent a significant step towards the model-free visual navigation paradigm relying solely on visual perception. A novel approach for visual servoing based on augmented image features is presented, which has only four off-diagonal couplings between the visual moments and the degrees of motion. As the visual servoing relies on unique image features, object recognition and pose alignment of the manipulator rely on the same representation of the object. In many scenarios the features extracted in the reference pose are only perceivable across a limited region of the work space. This necessitates the introduction of additional intermediate reference views of the object and requires path planning in view space. In this thesis a model-free approach for optimal large view visual servoing by switching between reference views in order to minimize the time to convergence is presented.

The efficiency and robustness of the proposed visual control schemes are evaluated in the virtual reality and on the real mobile platform as well as on two different manipulators. The experiments are performed successfully in different scenarios in realistic office environments without any prior structuring. Therefore this thesis presents a major contribution towards vision as the universal sensor for mobile manipulation.

Abstrakt

Autonome Serviceroboter sollen in Zukunft dem Menschen monotone und körperlich anstrengende Aufgaben abnehmen, indem sie beispielsweise Hol- und Bringedienste ausüben. Visuelle Wahrnehmung ist das wichtigste menschliche Sinnesorgan und Rückkopplungssystem und wird daher eine herausragende Rolle in zukünftigen Robotikanwendungen spielen. Robuste Verfahren für bildbasierte Navigation, Objekterkennung und Manipulation sind essentiell für Anwendungen in der Servicerobotik. Die mobile Manipulation in der Servicerobotik erfordert die Ausrichtung des Endeffektors zu erkannten Objekten in unbekannter Lage. Die bildbasierte Regelung ermöglicht eine modellfreie Objektmanipulation allein durch Berücksichtigung der zweidimensionalen Bildinformationen.

Im Rahmen dieser Arbeit werden Beiträge zur entkoppelten bildbasierten Regelung sowohl für die Objektmanipulation als auch für die Navigation präsentiert. Ein neuartiger Ansatz für die bildbasierte Weitbereichsregelung mobiler Roboter wird vorgestellt. Hierbei werden die Blickrichtungs- und Navigationsregelung durch eine virtuelle Kameraebene entkoppelt, was es der bildbasierten Regelung ermöglicht, dieselben natürlichen Landmarken effizient über einen weiten Bewegungsbereich zu verwenden. Um das Repertoire der visuellen Verhalten zu vervollständigen, werden ein innovatives Türdurchfahrtsverhalten sowie ein Hindernisvermeidungsverhalten basierend auf omnidirektionaler Wahrnehmung entwickelt. Die entworfenen visuellen Verhalten stellen einen wichtigen Schritt in Richtung des Paradigmas der reinen modellfreien visuellen Navigation dar. Ein neuartiger Ansatz basierend auf Bildmerkmalen mit einer erweiterten Anzahl von Attributen wird vorgestellt, der nach einer Entkopplung der Eingangsgrößen nur vier unerwünschte Kopplungen zwischen den Bildmomenten und den Bewegungsfreiheitsgraden aufweist. In vielen Anwendungsszenarien sind die extrahierten Referenzmerkmale nur in einem begrenzten Bereich des Arbeitsraums sichtbar. Dies erfordert die Einführung zusätzlicher Zwischenansichten des Objektes sowie eine Pfadplanung im zweidimensionalen Bildraum. In dieser Arbeit wird deswegen eine modellfreie Methodik für die zeitoptimale bildbasierte Weitbereichsregelung präsentiert, in der zwischen den einzelnen Referenzansichten umgeschaltet wird, um die Konvergenzzeit zu minimieren.

Die Effizienz und Robustheit der vorgeschlagenen bildbasierten Regler werden sowohl in der virtuellen Realität als auch auf der realen mobilen Plattform sowie zwei unterschiedlichen Manipulatoren verifiziert. Die Experimente werden in unterschiedlichen Szenarien in alltäglichen Büroumgebungen ohne vorherige Strukturierung durchgeführt. Diese Arbeit stellt einen wichtigen Schritt hin zu visueller Wahrnehmung als einziger und universeller Sensor für die mobile Manipulation dar.

Contents

List of abbreviations

The abbreviations used within the scope of this work are ordered alphabetically in the following.

ARIA	**A**dvanced **R**obot **I**nterface for **A**pplications
ARNL	**A**dvanced **R**obotics **N**avigation and **L**ocalization system
a.u.	**a**rbitrary **u**nits
AUTOSAR	**AUT**omotive **O**pen **S**ystem **AR**chitecture
BRIEF	**B**inary **R**obust **I**ndependent **E**lementary **F**eatures
CAD	**C**omputer-**A**ided **D**esign
CMAES	**C**ontrolled **M**odel-**A**ssisted **E**volution **S**trategy
CV	**C**urrent **V**iew
DBRVS	**D**istance-**B**ased **R**eference **V**iew **S**election
DOF	**D**egree **O**f **F**reedom
DoG	**D**ifference **o**f **G**aussian
EKF	**E**xtended **K**alman **F**ilter
FAST	**F**eatures from **A**ccelerated **S**egment **T**est
FCRVS	**F**ixed **C**onvergence **R**eference **V**iew **S**election
FSI	**F**ixed **S**cale **I**nterpolation
GFTT	**G**ood **F**eatures **T**o **T**rack
GF-HOG	**G**radient **F**ield-**H**istogram of **O**riented **G**radients
GLOH	**G**radient **L**ocation and **O**rientation **H**istogram
GV	**G**oal **V**iew
HIL	**H**ardware **I**n the **L**oop
HOG	**H**istogram of **O**riented **G**radients
IBVS	**I**mage-**B**ased **V**isual **S**ervoing
IR	**I**nfra**R**ed
LQR	**L**inear **Q**uadratic **R**egulator
MAES	**M**odel-**A**ssisted **E**volution **S**trategy
NN	**N**eural **N**etwork
ORB	**O**riented FAST and **R**otated BRIEF
ORVS	**O**ptimal **R**eference **V**iew **S**election
PBVS	**P**osition-**B**ased **V**isual **S**ervoing

PCA	**P**rinciple **C**omponent **A**nalysis
PD	**P**roportional **D**ifferential
PTZ	**P**an **T**ilt **Z**oom
RANSAC	**RAN**dom **SA**mple **C**onsensus algorithm
RMSE	**R**oot **M**ean **S**quare **E**rror
ROS	**R**obot **O**perating **S**ystem
RV	**R**eference **V**iew
SIFT	**S**cale **I**nvariant **F**eature **T**ransformation
SII	**S**cale **I**nvariant **I**nterpolation
SLAM	**S**imultaneous **L**ocalization **A**nd **M**apping
SNN	**S**ingle **N**earest **N**eighbor
SURF	**S**peeded **U**p **R**obust **F**eatures
ToF	**T**ime **o**f **F**light
ttc	**t**ime **t**o **c**ontact
VSLAM	**V**isual **S**imultaneous **L**ocalization **A**nd **M**apping
WANN	**W**eighted **A**verage among three **N**earest **N**eighbors

Nomenclature

In the present work vectors and matrices are printed in bold type. Vectors are hereby displayed by minuscule letters whereas matrices are represented by capital letters, and scalars are expressed in italic style. The nomenclature is sorted as following: the first classification criterion is latin before greek letters, afterwards lower-case before upper-case letters, and finally bold before italic type.

a	control action (for appearance based visual servoing)
a_h	scaling factor (for homography)
a_i, b_i	distance of an interest point to its appropriate epipolar line corresponding to the u- and v-direction, respectively
a_k	pixel displacement
a_m, b_m, c_m, d_m	model parameters for exponential function
$\mathbf{A}$	Hesse matrix
α	rotation around the x-axis (roll)
α_a	correction factor for the adaptive image Jacobian
$\alpha_c, \dot{\alpha}_c$	camera pan angle, respectively velocity
$\alpha_{\text{ia}}, \beta_{\text{ia}}, \gamma_{\text{ia}}$	interior angles
α_u, α_v	intrinsic camera parameter: scaling factor depending on λ and pixel dimensions
$\mathbf{b}_{C_{\text{ref}}}$	image features in the reference frame
β	rotation around the y-axis (pitch)
$\beta_c, \dot{\beta}_c$	camera tilt angle, respectively velocity
$\mathbf{c}$	performance criterion
conf_{avg}	mean of the confidence values
$\text{conf}_{\text{seg}(i,j)}$	confidence values in a window with the row and column position (i, j) of the cell
C, C_n, C_r	absolute, normalized and relative number of feature correspondences between the reference view and the current image
$C_{\text{ref}}, C_{\alpha,\beta}, C_R$	static and rotated camera coordinate systems, respectively, and camera coordinate system in the image plane
C_V	virtual camera coordinate system, respectively virtual camera plane
CVi	i-th reference view

$\mathbf{d}_{\text{kp}}$	normalized keypoint descriptor of SIFT features
d	distance
$\mathbf{D}$	Difference-of-Gaussian
$\Delta\mathbf{f}$	error between desired and actual feature locations
$\Delta\hat{f}$	total normalized summed feature error
Δf_γ	correction along γ of the averaged keypoint rotation
$\Delta f_\omega, \Delta\mathbf{f}_\omega$	predicted motion of the image features caused by $\Delta\Theta_R$
$\Delta\varphi$	feature error between reference and current distortion (camera retreat problem)
$\Delta\Theta_R$	orientational task space error
Δx	lateral task space error
Δz	longitudinal task space error
$[\mathbf{e}_a^1, \mathbf{e}_a^2]^T$	epipoles from the actual image
$[\mathbf{e}_{\text{ref}}^1, \mathbf{e}_{\text{ref}}^2]^T$	epipoles from the desired view
$\mathbf{E}$	essential matrix describing the epipolar constraint
$\bar{E}(\theta), \bar{E}(\phi), \bar{E}(r)$	mean absolute error in azimuth, elevation and radius
E_u, E_v	entropy along the u- and v-axis, respectively
ε	residual error between model and data point (for error function of the M-estimator)
ε_{d}	dissimilarity (residual error)
ε_γ	estimation error for camera rotation
η_1, η_2	tuning variables
$\mathbf{f}$	current image features, stated depending on the context as $\mathbf{f}_i = [u_i, v_i]$ for the i-th image feature with coordinates u_i, v_i, in the context of SIFT features as $\mathbf{f}_i = [u_i, v_i, \phi_i, \sigma_i]$ with the additional attributes orientation ϕ_i and scale σ_i, also in the context of image moments as $\mathbf{f} = [f_\alpha, f_\beta, f_\gamma, f_x, f_y, f_z]$
$\mathbf{f}_{\text{ref}}$	reference image features, also used in the context of image moments
f_α	image moment for rotation around the x-axis
f_β	image moment for rotation around the y-axis
f_γ	image moment for rotation around the optical axis
f_x	image moment for translation along the x-axis
f_y	image moment for translation along the y-axis
f_z	image moment for translation along the camera axis
f_{zd}	image moment for translation along the camera axis, alternative expression via the distance between point features
F	cost function
$\mathbf{G}$	Gaussian filter
γ	rotation around the z-axis (yaw), respectively the optical camera axis
γ_t	angle between orientation of virtual camera plane and template plane
γ_V	angle between the virtual camera plane and the orientation of the robot
h	twice the distance between the parabola's vertex and the focus of an omnidirectional camera

$\mathbf{H}$, $\hat{\mathbf{H}}$	homography, estimated homography by feature correspondences
$H_u(i)$	relative frequency of features in i-th column
$H_v(i)$	relative frequency of features in i-th row
$\mathbf{I}$	current image, also denoted as $\mathbf{I}(u, v, t)$ in dependence of the pixel coordinates u, v and time t
$\mathbf{I_{ref}}$	reference image
$[I_u, I_v]^T$	spatial intensity gradient in u- and v-direction, respectively
$\mathbf{J}$	visual image Jacobian
$\mathbf{J}^+$	pseudoinverse of the image Jacobian
$\mathbf{J_a}$	Jacobian for appearance based visual servoing
$\mathbf{J_e}$	Jacobian for visual servoing on epipoles
$\mathbf{J}_{v\omega}$	separated Jacobian for rotational motion
$\mathbf{J}_{vt}$	separated Jacobian for translational motion
$\mathbf{J}_{v\xi u_\xi}$	separated Jacobian for angle and axis of rotation parametrization
$\mathbf{J}_{xz}$	separated Jacobian for translational motion, reduced to two degrees of freedom
$\mathbf{J}_{\mathrm{dk}}$	robot Jacobian for differential kinematics
$\mathbf{J}_{f_i}$	image Jacobian for the image moment in i, whereas i stands for x, y, z, α, β, γ
$J_{f_i,j}$	image Jacobian entry for the image moment in i with a movement in j, whereas both i and j stand for x, y, z, α, β, γ and $i = j$ (desired couplings)
$\tilde{J}_{f_i,j}$	image Jacobian entry for the image moment in i with a movement in j, whereas both i and j stand for x, y, z, α, β, γ and $i \neq j$ (undesired couplings)
J_ω	separated Jacobian for rotational motion, reduced to one degree of freedom
$\mathbf{k}$	constant proportional gain
$\mathbf{k}_a$	adaptive gain
k	proportional gain factor
$\mathbf{K}$	camera calibration matrix as a function of the intrinsic camera parameters
l_k	image displacement
$\mathbf{L}$	Gaussian-blurred image
λ	focal length
λ_e	evaluated individuals of λ-CMAES
λ_{eig}	eigenvalue
λ_i	Lagrange multiplier
λ_p	offspring of λ-CMAES
μ	control parameter for Levenberg-Marquardt optimization
$\mu_{(i,j)}$	mean of the time to contact values in a segment with the row and column position (i, j) of the cell
μ_p	parents of λ-CMAES

$\mathbf{n}$	normal vector of a plane
n, $n_{\min}$, $n_{\max}$	number of feature correspondences, respectively minimum/maximum
∇_{pw}	divergence for each pairing window
ω	rotational velocity
ω_R, $\omega_{R_{\max}}$	rotational velocity of non-holonomic robot, rotational velocity limit
$\mathbf{\Omega}$	spatial neighborhood around image feature, respectively point of interest
$\mathbf{p_{ci}}$	world point
$\mathbf{p_i}$	point in image plane
$\mathbf{p_v}$	point in virtual camera plane
$\boldsymbol{\pi}(s, a)$	optimal policy (for appearance based visual servoing)
ϕ	canonical orientation of the keypoint
$\varphi, \varphi_{\mathrm{ref}}$	current and reference angle between two points forming a line relative to the horizontal line
$\mathbf{q}$	robot joint angles
$\dot{\mathbf{q}}$	robot joint velocities
$\mathbf{Q}$	action value function (for appearance based visual servoing)
$\mathbf{r}$	camera position
$\dot{\mathbf{r}}$	camera velocity
r_{f}	horizontal distance from focus to parabola of an omnidirectional camera
r_{XY}	Pearson's correlation coefficient describing the linear dependency between two stochastic variables X and Y
$\mathbf{R}$	rotation matrix
ρ	error function of the M-estimator
ρ, α	polar coordinates
s	object appearance (in angular color cooccurrence histograms)
σ	image feature scale especially in the context of SIFT and SURF features, also referred to as the standard deviation of the Gaussian
σ_{e}	parameter to regulate outlier suppression (for error function of the M-estimator)
σ_u, σ_v	variance of the feature distribution
$\mathbf{t}$	translation vector
ttc, ttc_{avg}	time to contact, mean time to contact
ttc_{nv}	one of the m total time to contact estimates computed from the corresponding flow vectors
$\mathbf{T}_{C_R}^{C_{\alpha,\beta}}$	transformation from the camera coordinate system to the rotated camera coordinate system
$\mathbf{T}_{C_{\alpha,\beta}}^{C_{\mathrm{ref}}}$	transformation of the rotated camera coordinate system into the static camera coordinate system
$\mathbf{T}_{C_{\mathrm{ref}}}^{C_V}$	transformation from the fixed reference frame centered at the focal point to the virtual camera plane
$\mathbf{T}_{C_R}^{C_V}$	transformation from the camera plane to the horizontal virtual camera plane

$\mathbf{T}_{\text{ext}}$	extrinsic homogeneous transformation matrix
$\mathbf{T}_{\text{int}}$	intrinsic homogeneous transformation matrix
$\theta_{\text{az}}, \phi_{\text{el}}, r_{\text{sc}}$	reference azimuth, elevation and radius in spherical coordinates
$\hat{\theta}_{\text{az}}, \hat{\phi}_{\text{el}}, \hat{r}_{\text{sc}}$	estimated azimuth, elevation and radius in spherical coordinates
θ_{icp}	intrinsic camera parameter: angle between the axes of the retinal image
Θ_{m}	model parameters (for error function of the M-estimator)
Θ_R	orientation of the robot
u	pixel coordinate in x-direction of the camera coordinate system
$[u, v, 1]^T$	homogeneous 2D image coordinates
$[\hat{u}, \hat{v}, 1]^T$	normalized 2D image coordinates
$[\bar{u}, \bar{v}]^T$	deviation of the feature centroid from the origin
$[\dot{u}, \dot{v}]^T$	optical flow
$[u_0, v_0]^T$	intrinsic camera parameter: principle point describing intersection of optical axis with image plane
$[u_{\text{cog}}, v_{\text{cog}}, 1]^T$	feature centroid of current view
$[\hat{u}_{\text{cog}}, \hat{v}_{\text{cog}}, 1]^T$	feature centroid of goal view
$[u_V, v_V, 1]^T$	2D image coordinates in the virtual camera plane
$[u_{\text{vcog}}, v_{\text{vcog}}]$	centroid of the u-, respectively v-coordinate of the current view expressed in the horizontal virtual camera plane after the feature rotation about $\Delta\Theta_R$
$[\hat{u}_{\text{vcog}}, \hat{v}_{\text{vcog}}]$	centroid of the u-, respectively v-coordinate of the reference view expressed in the horizontal virtual camera plane after the feature rotation about $\Delta\Theta_R$
u_ξ	axis of rotation parametrization
$\mathbf{U}\mathbf{\Lambda}\mathbf{V}^T$	singular value decomposition (SVD) of a matrix
$\mathbf{v}$	velocity
v	pixel coordinate in y-direction of the camera coordinate system
v_R	translational velocity of non-holonomic robot, v_R is composed of v_{R_z} in longitudinal direction and v_{R_x} in lateral direction
$v_{R_{\text{Left}}}, v_{R_{\text{Right}}}$	commanded velocity for the left and right wheel of the robot, respectively
$v_{R_{\max}}$	translational velocity limit
w_i	dynamic weight for decoupling f_x and f_y
$w_{i,\text{norm}}$	normalized dynamic weight (to be independent of the distance z)
$w(u, v)$	weighting function, e.g. for optical flow or Hesse matrix
$\mathbf{x}$	position $[x, y, z]$ and orientation $[\alpha, \beta, \gamma]$ of the end-effector
$[x, y, z, 1]^T$	homogeneous point coordinates
$[x_R, z_R, \theta_R]^T$	state of non-holonomic robot
x_i	data point (for error function of the M-estimator)
$[X, Y]^T$, $[\bar{X}, \bar{Y}]^T$	stochastic variables, mean values of stochastic variables
ξ	angle of rotation parametrization
z_{f}	horizontal axis of parabolic mirror
ζ	constant for gain computation to avoid numerical instability

Chapter 1

Introduction

In the future service robots are supposed to liberate people from the burden of monotonic and tedious tasks. Robots perceive their environment by means of force, touch, proximity or visual feedback with the objective to perform complex manipulation tasks in dynamic, unstructured environments of a complexity that exceeds the capabilities of current robotic manipulators in industrial settings. Pickup and delivery tasks constitute a novel domain of application for intelligent service robots. This development is triggered by more powerful and affordable sensors, increased computational power and the advent of lightweight manipulators. This thesis is a contribution towards the goal of realizing mobile manipulation with autonomous service robots.

Vision being the most important human sensor and feedback system is considered to play a prominent role in the future of robotics. Mobile manipulation in service robotic applications requires localization, navigation, object recognition as well as object manipulation. All these tasks are achieved with advanced sensors such as expensive laser scanners, affordable sonar as well as camera systems. Several tasks like obstacle avoidance and 3D world modeling are easily achieved by applying laser sensors. In order to disseminate service robots on a broad scale, their costs have to be reduced. Thus, new territory has to be entered in order to replace laser scanners in favor of cameras as a universal sensor. Camera systems offer the major advantage that they enable the recognition of objects as well as people including their gestures and mimics, in addition to their applicability for localization and navigation. They provide high dimensional and noisy data requiring information processing and reasoning in order to compensate for the information complexity compared to lasers. Therefore this thesis focuses on the challenging task to achieve mobile manipulation for autonomous service robots solely through computer vision.

1.1 Mobile manipulation

A general comprehensive outline of mobile manipulation is given by the Technical Committee on Mobile Manipulation:

> "*The ultimate goal of Autonomous Mobile Manipulation is the execution of complex manipulation tasks, in unstructured and dynamic environments, in which cooperation with humans may be required. To achieve this goal, several scientific and engineering challenges, currently beyond the state of the art in robotics, must be addressed.*" [146].

Mobile manipulation necessitates different skills such as planning, localization as well as deliberative navigation and object recognition in conjunction with object manipulation. The complexity of this mission arises from the high dimensional perceptual data afflicted with uncertainties as well as system complexity that emerges from the mobile platform itself but even more from the dynamics and ambiguities of the environment.

Given a scenario in which the human instructs the mobile platform with tasks such as table setting or pickup and delivery, the robot first of all has to localize itself in its dynamic environment as neither offices nor households are static. Localization is essential for planning as well as mission supervision. After the problem "Where am I?" is solved, navigation is required in order to address the problem of "How to get from A to B?". The navigation is supposed to guide the robot towards a goal destination for example passing a door, while simultaneously avoiding collisions. A large variety of different navigation schemes is provided in literature mostly using combinations of different sensors. This thesis follows the paradigm of purely vision-based navigation neglecting other kinds of sensor merely utilizing image data. Therefore all important skills for navigation of autonomous mobile robots such as obstacle avoidance, natural landmark orientation for goal-oriented navigation as well as door passing are designed solely based on visual perception. The skills for navigation using vision are supposed to be efficient to implement and robust to guarantee the safe operation of the mobile platform.

Once the designated goal location is reached the mobile platform needs to recognize and handle daily objects in household environments. The object recognition and manipulation relies on the same object representation, which is sparse in order to fulfill memory constraints of the underlying hardware. The task of object manipulation consists of the alignment of the end-effector with recognized objects of unknown pose. Image-based visual servoing provides a means of model-free manipulation of objects solely relying on 2D image information. Therefore this thesis provides a significant step towards manipulation of daily objects relying on natural texture even if the grasp pose of the object is outside the current view of the object.

Figure 1.1 shows the mobile robot equipped with camera and manipulator explicitly built for mobile manipulation tasks. It is based on a mobile platform from MobileRobots Inc.

equipped with sonar sensors. Two camera systems, a monocular pan-tilt camera and an omnidirectional camera are mounted on the platform for localization, navigation and object recognition. A manipulator with a two-finger gripper from Neuronics is installed on the platform. The eye-in-hand camera is designated for closed-loop object manipulation. The manipulator reduces the field of view of the omnidirectional camera. This imposes no constraint on the later-on described navigation with the omnidirectional camera because the remaining field of view of around 300° still contains all relevant environmental contents.

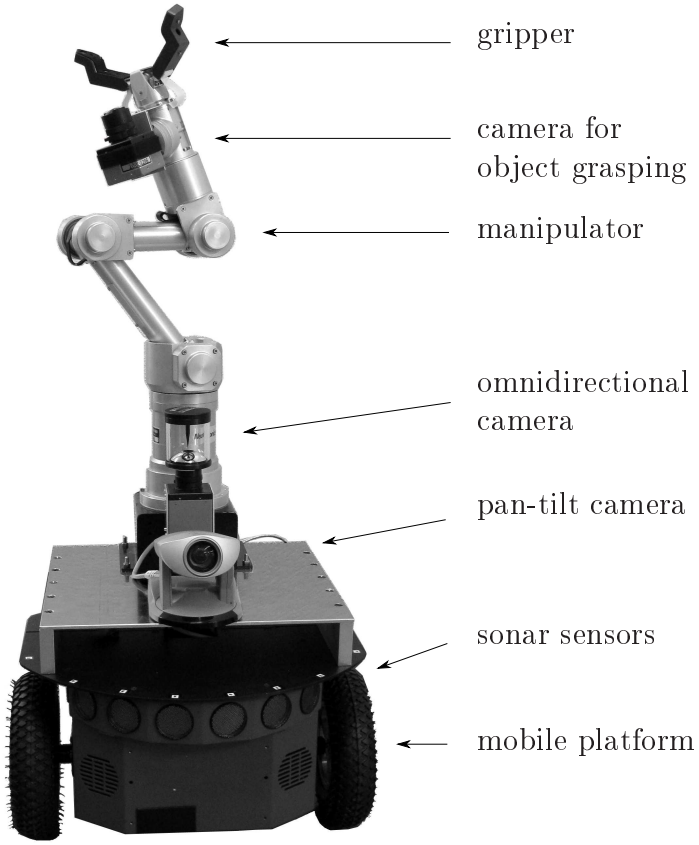


Figure 1.1: Mobile robot.

1.2 Related work

The mobile platform is provided with an Advanced Robot Interface for Applications (ARIA) [100]. ARIA already incorporates control of robot's velocities, odometric, sonar and laser measurements as well as collision-free navigation due to reactive behaviors based on its sonar or laser data. In order to achieve goal-oriented navigation additional packages for map building (laser mapping and navigation package), ARNL (Advanced Robotics Navigation and Localization System) for Markov based localization and MobileEyes for remote

control of the robot's actions, e.g. the progress of the task in the map, are at the disposal of the customer. The customer has a fully operational robot with these packages, which navigates after an initial mapping stage without collisions in a goal-oriented manner in dynamic environments. To achieve even more complex tasks in the context of service robotics such as human recognition, human-machine interaction as well as object recognition and manipulation additional sensors for visual perception are required. While a service robot inherits more tedious tasks from humans, it is indispensable to reduce the overall costs especially for the hardware in order to finally achieve the economic breakthrough in the consumer market. Therefore the motivation arises to design the crucial capabilities such as localization and navigation as well as advanced skills such as object manipulation with a single cost-efficient sensor system in conjunction with highly advanced control methodologies, rather than employing multiple kinds of expensive sensors in parallel. This trend from hardware to software intelligence occurs in many industry branches with severe pricing pressure e.g. automotive industry. Cameras represent an efficient solution to this dilemma because the range of possible applications and skills over price is much more advantageous compared to laser. Therefore in its first part this thesis aims at the objective to achieve similar performance for navigation with visual perception compared to the already existing commercial software with laser sensors. This provides the basis for additional applications such as object manipulation, which are treated in the second half of this thesis.

The robot control is based on a hybrid architecture [15] depicted in figure 1.2, composed of a planning layer, a coordination layer and a subordinate reactive layer. The role of the planning layer consists in generating the mission plan and its surveillance, including global localization of the robot, preloaded path planning for goal-oriented navigation as well as object manipulation. The coordination layer activates or deactivates those reactive behaviors that are necessary for successful realization of the plan and adequate in the current context. It is also responsible for the diagnosis of the robot's status, mission surveillance and emergency or fallback strategies. The operation of the reactive layer follows the behavior based paradigm [18], as it abandons any abstract representation of the environment but decides about the motion commands only based on the current perception provided by the sensors (behavior representation). A behavior is represented by a direct map from the stimulus, for example the distance measurement, to the response, in the case of mobile robots the motor commands. In case of navigation an obstacle avoidance behavior guarantees the safety of the robot with respect to collisions with surrounding objects. Other reactive behaviors e.g. constant velocity, corridor centering, homing are primarily useful for local navigation. The object manipulation requires a behavior which transfers the manipulator in a pre-grasping position. This thesis investigates the potential of camera systems to replace the sensor inputs for the planning and the reactive layer and completely dispense with distance sensors such as laser employed in commercially available robot systems.

Different approaches for robot navigation are known from literature [15, 18, 5] focussing mostly on methodologies for distance sensors. In their general survey about vision for

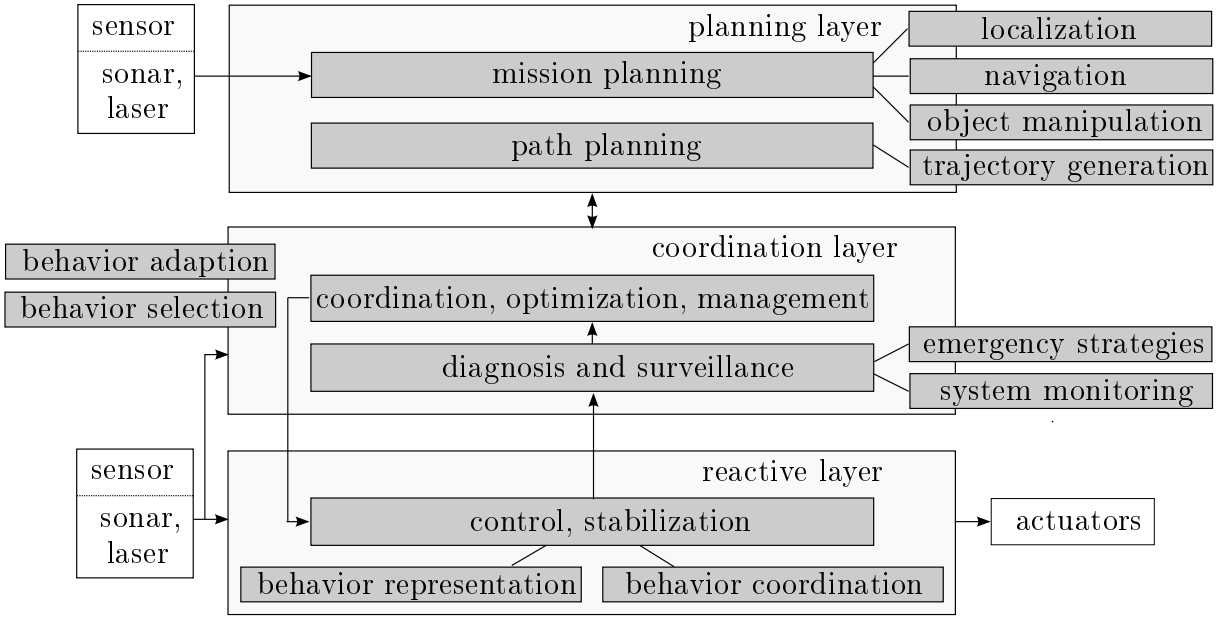


Figure 1.2: Hybrid three-layer model for robot control with planning, coordination and reactive layer, with laser and sonar as input for localization and navigation in the planning layer as well as for the behaviors in the reactive layer.

mobile robot navigation [39] distinguish between indoor and outdoor navigation. A comprehensive overview for **visual navigation** is provided by [16], which categorizes visual navigation as map-based navigation and mapless navigation, whereas map-based navigation is subdivided into metric and topological map-based navigation. Metric maps represent the environment in relative coordinates with respect to an absolute world coordinate system, whereas topological maps possess a graph-like structure with nodes and edges, representing abstract locations and the repertoire of behaviors to transit between them without any geometric information [86]. Localization techniques using laser sensors are well-established [51]. A framework for **S**imultaneous **L**ocalisation **A**nd **M**apping (SLAM) is provided by [147] by building a map from scratch while continuously localizing itself in the online generated map. Efficient approaches such as FAST-SLAM [102] achieve nowadays real-time mapping of the environment. Despite the substantial progress regarding VSLAM (**V**isual **S**imultaneous **L**ocalisation **A**nd **M**apping) [142, 36, 138], maps provided by VSLAM using local feature extraction are sparse and therefore not dense enough for metric navigation required by standard laser based navigation schemes. However, these maps are suited for robot localization [76]. Recent approaches [148] generate off-line dense 3D maps due to stereo vision with additionally integrated landmarks, nonetheless the overall localization is inferior to simple topological localization approaches using omnivision such as [55]. In [44] a VSLAM scheme provides a 3D-voxel map by FAST-SLAM in conjunction with the Kinect sensor, which solves inherently the 3D reconstruction problem of visual senors by actively emitting structured light [98]. This thesis follows the topologi-

cal map-based navigation paradigm using passive visual senors, representing environments by a directed graph. Topological maps require less memory and are suitable for the representation of large indoor environments. Topological SLAM using local feature extraction is presented in the works of [155, 3], which seems to outperform appearance-based visual SLAM by global feature extraction [65]. The choice of the localization methodology has a direct impact on the required collection of behaviors (referred to as mapless navigation in [16]). Topological map-based navigation requires visual perception representing the visual nodes, also referred to as waypoints, as well as the visual behaviors associated with the edges in order to navigate between them. Depending on the degree of integration of the image processing systems into the hybrid control architecture the approaches are classified throughout this work into vision-guided and visual navigation schemes. Visual navigation solely uses visual information as input for the planning as well as for the reactive layer, whereas vision-guided approaches are supplemented by active distance sensors such as sonar or laser sensors providing further input for the reactive layer.

Visual reactive behaviors omit metric maps for representing the environment, instead they perceive and track objects by coupling the immediate decision about the robot movement directly with the visually observed appearance of the local environment. Such approaches are either based on locating specific landmarks in the environment, or follow an appearance based approach [154] or measure the optical flow [4]. The corridor centering described in [4] operates by balancing the optical flow in the right and left hemisphere of an omnidirectional camera system, however, it fails if texture is missing or non-uniformly distributed in the corridor environment. Vision-based navigation in unstructured environments solely uses natural features and structures without adding supplementary landmarks or texture elements to facilitate the navigation task. [105] describes a vision-based homing behavior with gaze control for decoupling the camera and the robot movement via a virtual camera plane. However, in this context the environment is structured systematically by placing landmarks at selected waypoints to support vision-based navigation.

Robotic manipulation of daily-life objects in unstructured environments is an essential requirement in service robotic applications. **I**mage-**B**ased (IBVS) and **P**osition-**B**ased **Visual Servoing** (PBVS) grow in visibility due to their importance for robotic manipulation and grasping. Visual servoing is defined in the standard tutorial [70] as:

"*the use of one or more cameras and a computer vision system to control the position of the robot's end-effector relative to the work piece as required by the task*".

Position-based visual servoing estimates the object's pose relative to camera, as the error between the actual and the goal pose is defined in the Cartesian space. The main drawbacks of position-based visual servoing are 3D model generation of objects, on-line estimation of 3D pose, system instabilities because of coarse pose estimations as well as objects leaving the field of view [23]. Image-based visual servoing solely relies on 2D image information for the alignment of the end-effector with an object of unknown pose. The desired pose

for grasping is demonstrated to the robot during a learning stage and a set of reference features is extracted from the image. A geometric object model or an explicit reconstruction of the object scene becomes obsolete for image-based visual servoing. Due to these two major advantages this approach is particulary promising for mobile manipulation, namely model-free and easy to demonstrate for the instructor.

The categorization of [24] and [25] for different image-based visual servoing concepts is pursued and different approaches in literature are ranked regarding their applicability to mobile manipulation. Jacobian based visual servoing inverts the analytical relation between differential changes in task space to differential changes of pixel coordinates to reduce the error in the image space between the actual and desired feature coordinates [151]. Hybrid visual servoing defines the error between actual and desired pose partially in image and Cartesian space [26]. Partitioned visual servo, respectively visual servoing with decoupled image moments, defines image moments which are related approximately in a one-to-one relationship to their degrees of motion, resulting in a simple linear control problem in the image space [143]. Appearance based visual servoing [37] captures the overall appearance of an object rather than single features and relates this appearance by an offline learned interaction matrix to control values to steer the end-effector in the reference pose. Other approaches for visual servoing such as visual servoing on epipoles [120] or by structured light are neglected because of their minor importance for service robotics.

Figure 1.3 depicts a radar chart in order to compare different visual servoing concepts with respect to various aspects. Visual servoing by image Jacobian, hybrid visual servoing, visual servoing by decoupled moments as well as appearance based visual servoing are compared regarding stability, calibration issues, convergence, compliance with service robotic specifications and biology inspiration. Stability is divided into global asymptotic and local asymptotic stability as well as heuristic approaches for stability analysis e.g. convex polygons. Hybrid visual servoing has the highest ranking due to its global asymptotic stability. Appearance based visual servoing has the lowest ranking as the stability analysis of the optimal policy (feed-forward) is not analytically feasible. On the contrary appearance based approaches require in principle no intrinsic or extrinsic camera calibration and therefore achieve the highest ranking in this category. Nonetheless even if the three other approaches require intrinsic camera calibration, this is nowadays no severe limitation because of the standard tools for camera calibration [136]. The aspect of convergence contains computational complexity as well as the convergence (behavior) of the image error, the task space error in addition to the required actuating variables. Hybrid and visual servoing with decoupled moments exhibit fast convergence in conjunction with low computational complexity. The computational complexity of course highly depends on the feature extraction methodology and its application parameters. On the contrary appearance based visual servoing has high computational demands for extracting appearance, whereas Jacobian based approaches partially show slow convergence depending on the relative pose between actual pose and goal pose because of their couplings between rotational and translational degrees of freedom. The term service robotic applications compromises e.g. the robustness

regarding occlusion, unstructured cluttered environments with highly structured objects as well as changing light conditions. Additionally object recognition as well as visual servoing should rely on the same object representation in order to reduce memory requirements. Appearance based visual servoing requires accurate object segmentation to discriminate different object poses, which is difficult to achieve in textured environments. Nonetheless this methodology directly fulfills the requirement for the same object representation for recognition and positioning. Feature based approaches in literature are presented most frequently using simple feature primitives such as [135]. These features are very efficient to implement but not realistic for service robotic applications because of their low perceptibility across large regions of the workspace as well as their minor ability to discriminate among different objects. The potential of feature based approaches is much more promising than appearance based visual servoing concerning robustness due to feature redundancy and under the assumption of solved correspondence problem. Even if appearance based approaches are ranked highest in the category biology inspiration, these approaches are suboptimal regarding the other categories and are therefore not pursued in the context of this thesis. It is an interesting point that approaches adopted from nature are less robust than purely technical motivated methodologies regarding mobile manipulation.

Conclusively it can be stated that visual servoing with decoupled moments and hybrid visual servoing are best suited for service robotic applications and are further investigated to achieve full applicability for mobile object manipulation. Furthermore this thesis postulates visual servoing with decoupled moments, as no partial pose estimation requiring intrinsic camera calibration as well as geometric assumptions of the scene are required. Exploitation of the potential of visual servoing with decoupled image moments regarding decoupling the translational and rotational degrees of freedom as well as fulfilling service robotic specifications is a challenging task. The authors in [117], however, state that:

"*Finding a set of visual features which produces a decoupled interaction matrix for any camera pose seems an unreachable issue*".

Nonetheless a diagonal interaction matrix is much desired and therefore investigated in the context of this thesis with the success of finding a resulting interaction matrix with only four remaining couplings independent of the camera pose.

In many scenarios the features extracted in the reference pose are only perceivable across a limited region of the work space. Different terminologies are reported in literature for visual servoing across several intermediate reference views of the object in order to navigate towards the final reference pose. Path planning in image space [97], visual servoing due to visual memory [123] as well large view visual servoing [105] are conceptualized for global visual servoing. Notice that local visual servoing is defined by the visual servoing towards a single reference image, whereas global visual servoing is concerned with the navigation and control in a set of connected, partially overlapping reference images, respectively in the overall image space. Achieving a model-free and time-optimal convergence towards

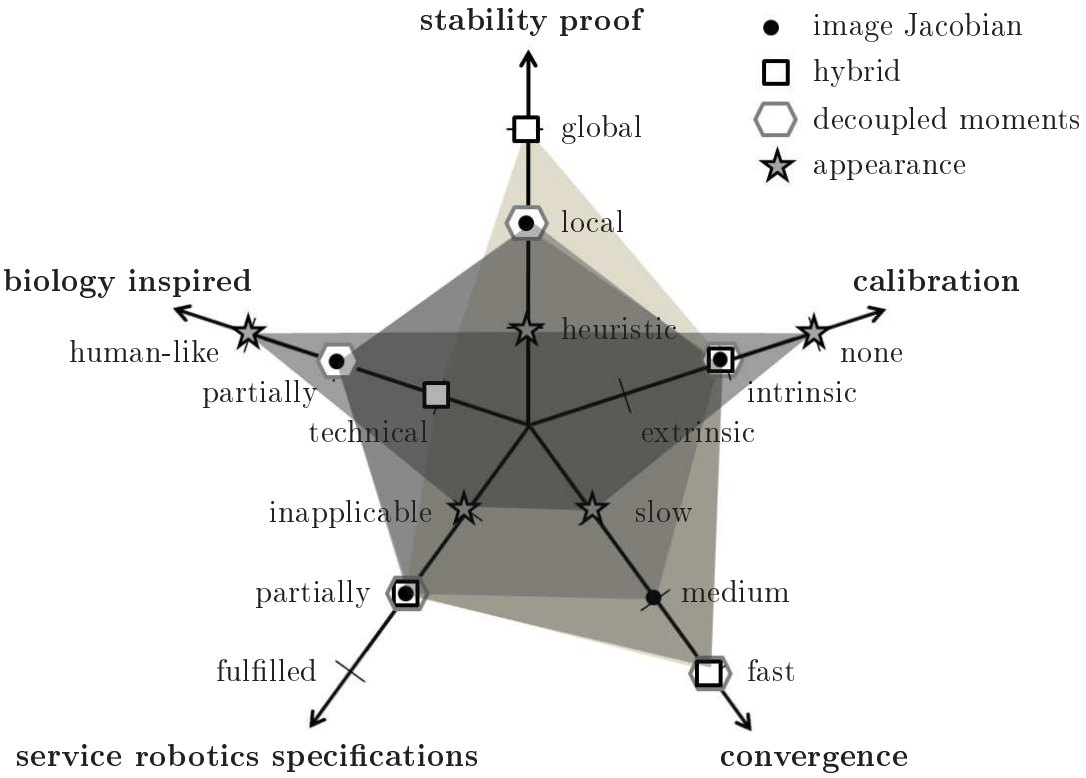


Figure 1.3: Characteristics of different visual servoing concepts regarding stability, convergence, service robotic specifications and biology inspiration.

the desired pose by switching between reference views is the ultimate goal of the cited approaches. Global visual servoing is a challenging task, which is imperative to achieve mobile manipulation independent of the object's initial view in the camera image.

1.3 Objective of this thesis

This thesis provides a contribution towards mobile manipulation in unstructured environments with the ambitious goal to accomplish all skills and tasks exclusively by means of visual perception. In order to achieve mobile manipulation solely relying on visual perception this work yields new insights in two major domains namely visual navigation in the first part and visual servoing for object manipulation in the second part.

For visual navigation the following questions are addressed:

- How to achieve time-optimal visual homing for mobile robots dealing with natural texture in dynamic environments with camera systems with limited field of view requiring gaze and position control in parallel?
- How to design collision-free navigation using omnivision considering noisy image measurements and sparsely textured office environments?
- How to accomplish door detection, door tracking and door passing in a coherent purely vision-based framework with closed-loop door traversing?
- How to design visual navigation in unstructured office environments with matchable performance in comparison to state-of-the-art approaches using sonar and laser sensors?

Visual servoing for object manipulation is mainly concerned with the following challenges:

- How to achieve markerless and decoupled visual servoing for optimal convergence in task space in the context of object manipulation of daily objects?
- How to realize time-optimal visual positioning of the gripper relative to an object even if the desired grasping position is outside the current field of view of the camera?
- Which strategy is better? A look-then-move strategy in conjunction with local visual servoing close to the reference pose or visual servoing over several reference images in the context of service robot applications?

This thesis is organized as follows: Chapter 2 provides the state of the art of computer vision as well as the visual servoing in order to keep this thesis self-contained. The chapter 3 is dedicated to the progress from vision-guided navigation with laser based stimuli to purely vision-based navigation by relying solely on visual stimuli. Global visual homing based on visual servoing with an omnidirectional in conjunction with a pan-tilt camera is introduced in chapter 4. A comparison of vision-guided and visual navigation is additionally provided at the end of chapter 4. In order to accomplish mobile manipulation chapter 5 demonstrates a novel approach for markerless and decoupled visual servoing to align the robot end-effector with recognized objects of unknown pose. Conventional point features are augmented by additional attributes like scale and orientation, which establish a one-to-one correspondence between the individual image moment and its corresponding degrees of freedom. The limited visibility of features necessitates the introduction of additional intermediate reference views of the object and requires path planning in view space. Therefore a new methodology for global (large view) visual servoing is introduced in chapter 6. The path planning in the image space is flexible as the decoupled visual servoing relies on a dynamic set of feature correspondences rather than a static set of individual features. This property allows the online selection of optimal reference views during servoing to the goal view resulting in time-optimal control. Finally this thesis concludes with a summary and outlook on future work in chapter 7, in which the major developments concerning the challenges and open questions raised here and the major results and insights are summarized.

Chapter 2

State of the art of computer vision and visual servoing

This chapter provides the basis for computer vision and visual servoing, the required terminology for the comprehension of this thesis as well as the classification of this thesis into the scientific context. This chapter is organized as follows: Image formation is described in section 2.1 for perspective and multiple cameras as well as for omnivision. Image understanding by robust feature detection for object recognition is treated in section 2.2. The two major topics visual navigation and image based visual servoing are described in detail in sections 2.3 and 2.4, respectively, as well as the experimental systems in section 2.5.

2.1 Perspective camera, multiple-view geometry and omnivision

The general perspective projection model describes the relation between a homogeneous point $\mathbf{p}_c(x, y, z, 1)$ in the 3D camera space coordinate system and its projection onto the 2D image coordinate system in homogeneous coordinates $\mathbf{p}(u, v, 1)$, whereas λ denotes the focal length:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} \begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}. \tag{2.1}$$

The image point $\mathbf{p}(u, v, 1)$ on the retinal image is transformed to the normalized image plane according to equation 2.2. This transformation yields the normalized pixel coordinates $[\hat{u}, \hat{v}, 1]^T$ independent of the intrinsic camera parameters, i.e. enabling the direct

comparison of images originating from different camera systems:

$$\begin{bmatrix} \hat{u} \\ \hat{v} \\ 1 \end{bmatrix} = \mathbf{K^{-1}} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad \text{with} \quad \mathbf{K} = \begin{bmatrix} \alpha_u & -\alpha_u \cot(\theta_{\text{icp}}) & u_0 \\ 0 & \frac{\alpha_v}{\sin(\theta_{\text{icp}})} & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.2}$$

The intrinsic camera parameters α_u and α_v describe the scaling factors depending on λ and the pixel dimensions. The intersection of the optical axis with the image plane is described by the principle point $[u_0, v_0]^T$. Due to manufacturing imperfections of an actual camera, the angle θ_{icp} between the axes of the retinal image may not be equal to 90°. The extrinsic camera parameters consider the position and orientation of the camera coordinate system relative to the world coordinate system. To express this relation, the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$ are combined in a homogeneous transformation matrix $\mathbf{T}_{\text{ext}}$:

$$[u, v, 1]^T = \frac{1}{z}\mathbf{T}_{\text{int}}\mathbf{T}_{\text{ext}}\,[x, y, z, 1]^T \quad \text{with} \quad \mathbf{T}_{\text{int}} = (\mathbf{K}\,\mathbf{0}). \tag{2.3}$$

Intrinsic camera parameters as well as radial distortions of the pixel coordinates u and v caused by lens imperfections are determined by a camera calibration process [136]. The radial distortion is corrected by a polynomial function of the squared distance between the optical center of the image and the given pixel coordinates (cf. chapter 3.3 in [50]). Detailed information about the complete camera system layout and the image formation process can be found in [67], whereas standard references [50], [78] mainly focus on the image analysis from low level to high level vision.

Multiple view geometry is concerned with partial or full 3D reconstruction, respectively, of the environment based on multiple views of a scene. The essential and fundamental matrices describe the epipolar constraint for calibrated and uncalibrated camera systems which relates a point in one image to a line in the other independent of the scene's geometry [90]. The essential matrix is stated as:

$$\mathbf{E} = [\mathbf{T_x}]\mathbf{R}, \tag{2.4}$$

where the vector $\mathbf{t}$ is expressed as a skew-symmetric matrix $\mathbf{T}_x$ so that $\mathbf{t} \times \mathbf{x} = [\mathbf{T}_x]\mathbf{x}$. The essential matrix degenerates for small translations, rendering it unsuitable for automatic control engineering topics such as visual servoing or image-based oscillation measurements. The homography $\mathbf{H}$, however, describes a point-to-point transformation between two perspective views of a plane:

$$a_h\,[\hat{u}_2, \hat{v}_2, 1]^T = \mathbf{H}\,[\hat{u}_1, \hat{v}_1, 1]^T \quad \text{with} \quad \mathbf{H} = \mathbf{R} + \frac{\mathbf{n}^T}{d}\mathbf{t}, \tag{2.5}$$

whereas $\mathbf{R}$ and $\mathbf{t}$ are defined by the rotation and translation between the optical camera centers. $\mathbf{n}$ is the normal vector of the plane and d the distance between the optical center of the first camera and the plane. Contrary to the essential matrix the homography matrix does not degenerate because $\mathbf{t}$ is additive.

The homography is estimated from at least four corresponding features located on a common plane, assuming that the scaling factor $\hat{h}_{33} = 1$, via:

$$\mathbf{p}_2 = \hat{\mathbf{H}}\mathbf{p}_1 \quad \Leftrightarrow \quad \begin{bmatrix} \hat{u}_2 \\ \hat{v}_2 \\ 1 \end{bmatrix} = \begin{bmatrix} \hat{h}_{11} & \hat{h}_{12} & \hat{h}_{13} \\ \hat{h}_{21} & \hat{h}_{22} & \hat{h}_{23} \\ \hat{h}_{31} & \hat{h}_{32} & \hat{h}_{33} \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{v}_1 \\ 1 \end{bmatrix}, \tag{2.6}$$

where $\hat{\mathbf{H}}$ is, apart from a scaling factor a_h, identical to the actual homography matrix $\mathbf{H}$. The estimated homography $\hat{\mathbf{H}}$ is decomposed via singular-value decomposition into the unknowns rotation matrix, scaled direction vector as well as the normal vector [47]:

$$\hat{\mathbf{H}} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T \quad \Leftrightarrow \quad \mathbf{\Lambda} = \mathbf{U}^T\hat{\mathbf{H}}\mathbf{V} \quad \Leftrightarrow \quad \mathbf{\Lambda} = \mathbf{U}^T(d\mathbf{R} + \mathbf{t}\mathbf{n}^T)\mathbf{V}. \tag{2.7}$$

As the decomposition of the homography yields ambiguous solutions, the correct solution is obtained by taking into account only the physically plausible solutions and a subsequent comparison of the estimated with the assumed normal vector. Multiple view geometry for partial or complete real world reconstruction e.g. homography is treated extensively in the works of [60] and [134].

Conventional monocular cameras have a limited field of view. In order to overcome this constraint, omnidirectional cameras, also referred to as catadioptric cameras, consist of a combination of lenses (refractive, i.e. dioptric) and mirrors (reflective, i.e. catoptric) to enlarge the field of view. The most important design objective for catadioptric sensors is to achieve a single effective viewpoint, which allows the reconstruction of perspective views and panoramic images with arbitrary orientations. A detailed overview of single viewpoint catadioptric sensors and the image formation process is provided by [8, 53].

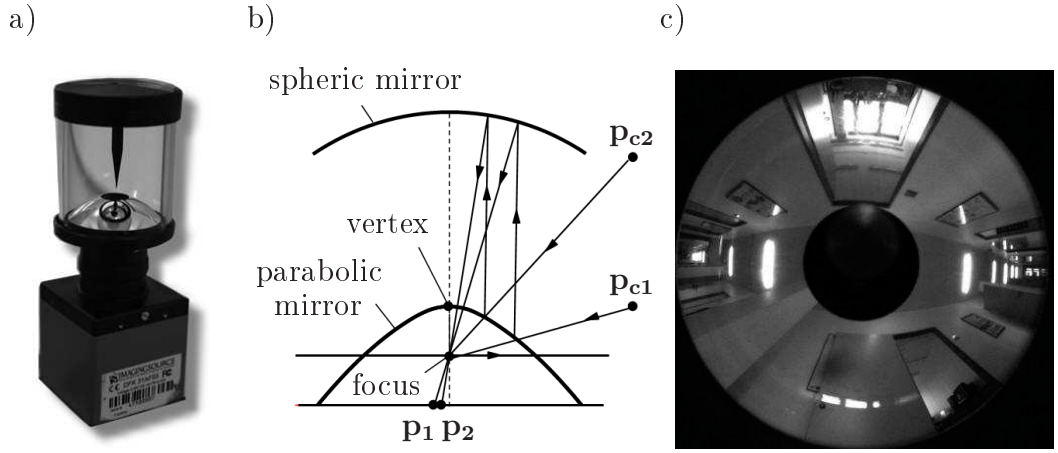


Figure 2.1: a) Omnidirectional camera; b) Geometry of a parabolic omnidirectional camera; c) Omnidirectional image.

The omnidirectional sensor used in this thesis consists of a camera DFK-31AF03 from Imaging Source and a D40 optic from RemoteReality. It has a field of view of 360° in

azimuth and approximately 60° in elevation. Figure 2.1 depicts the omnidirectional camera (a), a schematic view of the projection geometry (b) as well as an omniview (c) referred to in the following as omnivision. The catadioptric sensor consists of a parabolic mirror in conjunction with a spheric mirror and a perspective lens system. Parabolic mirrors have an orthographic projection, which guarantees that the light rays from the environment are reflected parallel towards the spheric mirror. The spheric mirror also satisfies the single viewpoint constraint, whereas the center of projection lies in the center of the sphere. A sharp single viewpoint image is obtained as the center of the sphere coincides with the focal point of the perspective lens system. Figure 2.1 b) shows the geometry of such a parabolic omnidirectional camera. The world points $\mathbf{p_{c1}}$ and $\mathbf{p_{c2}}$ are orthographically reflected to the points $\mathbf{p_1}$ and $\mathbf{p_2}$ in the image plane. The vertex of the parabola has the distance $h/2$ to the focus which is the single viewpoint of the parabola. The parameter h is also the radius r_f at $z_\mathrm{p} = 0$. Thus, the expression for the reflecting surface follows as:

$$z_\mathrm{p} = \frac{h^2 - r_\mathrm{f}^2}{2h}. \tag{2.8}$$

In figure 2.1 c) the omniview is presented which shows the blind spot in the center, an analogy to the human eye, originating from a pin in the center of the spheric mirror to prevent multiple reflections.

Omnivision is well suited for mobile robot applications as it captures the entire surrounding, which facilitates robot localization as well as robot navigation. Furthermore, due to their large field of view, omnidirectional camera systems are optimal for work space surveillance of product assistants [141].

2.2 Robust point feature detection for recognition

For developing vision-based control concepts for mobile manipulation in unstructured environments unambiguous and recognizable features have to be extracted from the camera images. Contrary to the industrial context where markers or labels are imprinted on objects and in the surrounding environments, for service robotic tasks this approach is not feasible. Thus, the algorithms employed in this thesis have to recognize the features in the camera image if the camera-object distance changes (scaling invariance), the lighting conditions vary, the camera rotates around its optical axis or is subject to affine transformations. Associating the same feature in different perspectives is referred to as correspondence problem.

In the following, two prominent and useful algorithms from literature for local feature extraction and for solving the correspondence problem are presented in detail. Primarily **G**ood **F**eatures **T**o **T**rack (GFTT) [135], which is implemented e.g. in the OpenCV library [71], is described as it already contains all significant steps required for robust feature

extraction and matching. Based on this efficient implementation, a sophisticated method for feature extraction, **S**cale **I**nvariant **F**eature **T**ransformation (SIFT), is described which is utilized within the scope of this work.

GFTT consists of an edge detection in order to localize interest points and subsequently track the same feature over consecutive images. Strong corners in the image are detected with the Hesse matrix according to the ideas of the Harris edge detector [59]:

$$\mathbf{A} = \sum_u \sum_v w(u,v) \begin{bmatrix} I_u^2 & I_u I_v \\ I_u I_v & I_v^2 \end{bmatrix}, \tag{2.9}$$

with the image derivatives I_u and I_v in u- and v-direction, respectively, and the isotropic weighting $w(u,v)$ such as a Gaussian kernel. The two eigenvalues λ_{eig_1} and λ_{eig_2} are extracted from $\mathbf{A}$. If λ_{eig_1},λ_{eig_2} are close to zero then the image region is homogeneous. If one of the two eigenvalues is much greater than the other the image region contains an edge. A corner is detected only if both eigenvalues have large positive values and satisfy the constraint $\min(\lambda_{\text{eig}_1},\lambda_{\text{eig}_2})$ larger than a threshold. The corner represents an interest point which is tracked in consecutive images by a small window $\mathbf{\Omega_s}$ assuming purely translational motion. In order to avoid false tracking of features the dissimilarity is measured for a large window $\mathbf{\Omega_l}$ as follows:

$$\varepsilon_{\text{d}} = \iint_{\mathbf{\Omega_l}} [\mathbf{I}_2(\mathbf{R}\mathbf{p}_r + \mathbf{t}) - \mathbf{I}_1]^2 d\mathbf{p}_r. \tag{2.10}$$

If the residual error ε_{d} exceeds a certain threshold the feature is classified as lost and is therefore rejected. GFTT are well suited for local feature tracking and are therefore not suited for advanced service robotic applications. Of course scale-invariance can also be achieved by a scale-independent Harris edge detection using a Gaussian pyramid, nonetheless the feature extraction described in the following has a better representation of the features suited for recognition even under large displacement and rotations as well as changes in lighting conditions.

Scale Invariant Feature Transformation introduced by Lowe [93] is an approach to detect and extract local features from an image with similar methodology as GFTT but with superior performance in terms of recognition, because of combinations of the progress in image processing since the first presentation of GFTT. They demonstrate invariance with respect to scale, orientation and illumination. SIFT features are conveniently matched across similar views of the same scene. The utilization of specific markers in vision-based applications becomes obsolete as the environment and textured objects naturally contain suitable SIFT features. SIFT features are distinguishable as their associated keypoint descriptor includes a compact, albeit specific representation of the surrounding image region. These properties make them particularly suitable for vision-based localization, visual servoing, object recognition and pose estimation. As their properties are essential for the later on introduced visual controllers, the four major computation stages are briefly described.

(1) Scale-space extrema detection: Interest points in the image for SIFT features are the ones which correspond to local extrema of Difference-of-Gaussian (DoG) filters at

different scales. The scale of the SIFT feature is defined by σ. The difference of Gaussians is calculated from the difference of convoluted images at neighboring scales σ, respectively $k\sigma$. Given a Gaussian-blurred image $\mathbf{L}$

$$\mathbf{L}(u, v, \sigma) = \mathbf{G}(u, v, \sigma) * \mathbf{I}(u, v) \quad \text{where} \quad G(u_i, v_i, \sigma_i) = \frac{1}{2\pi\sigma_i^2} \exp\left(\frac{-(u_i^2 + v_i^2)}{\sigma_i^2}\right) \tag{2.11}$$

is a variable scale Gaussian, $\mathbf{I}$ denotes the image to be processed and $*$ is the convolution operator. The convolution of an image with a DoG filter is defined by

$$\mathbf{D}(u, v, \sigma) = (\mathbf{G}(u, v, k\sigma) - \mathbf{G}(u, v, \sigma)) * \mathbf{I}(u, v) = \mathbf{L}(u, v, k\sigma) - \mathbf{L}(u, v, \sigma). \tag{2.12}$$

The converted images are grouped by octaves which correspond to doubling the value of σ, resulting in a pyramid of DoG images with different scale.

(2) Keypoint localization: The interest points in the image are referred to as keypoints. They are identified either by their local maxima or minima of the DoG images across the scales. Every pixel in the DoG image is checked for its candidate validity by comparing it with its eight neighbors at the same scale and also with its nine corresponding neighbors at neighboring scales. If the pixel exhibits either a local maximum or local minimum it is selected as a candidate keypoint. Every candidate keypoint needs interpolation to accurately determine its position. Keypoints with low contrast values are removed and responses along the edges are also eliminated. Once the positions of the keypoints are assigned their orientation can be determined.

(3) Orientation assignment: Orientation of the keypoint is determined using a gradient orientation histogram in the neighborhood of the keypoint. The contribution of each neighboring pixel is weighted by the gradient magnitude and a Gaussian window with a width σ that is 1.5 times the scale of the keypoint. Peaks in the histogram correspond to dominant orientations. A separate keypoint is created for the direction corresponding to the histogram maximum and any other direction within 80% of the maximum value. The properties of the keypoints are all described relative to the keypoint orientation to accomplish orientation invariance.

(4) Keypoint descriptor: With the information about the keypoint orientation, a keypoint descriptor is constructed which is a set of orientation histograms on the neighboring 4 by 4 pixels. The histograms are expressed with respect to the keypoint orientation. The histogram has eight bins and each descriptor has an array of four histograms around its keypoint. Each SIFT feature consists of a normalized keypoint descriptor $\mathbf{d}_{\text{kp}}$ with 4 by 4 by $8 = 128$ elements.

Matching of SIFT features: Matching of SIFT features involves the determination of corresponding features in two views of the same scene. Therefore the SIFT features are extracted in both views and the similarity of their keypoint descriptor is calculated. The similarity is defined by the Euclidian distance between the two keypoint descriptors of

length 128. In order to make the matching even more robust the relative rather than the absolute similarity is evaluated using the relationship between the highest and the second highest value of similarity which is required to exceed a specified threshold.

The presented control concepts can be realized identically with SURF (**S**peeded **U**p **R**obust **F**eatures) [13, 12] because they also contribute additional attributes as scale and orientation of the features. Other methods for local feature extraction such as GLOH (**G**radient **L**ocation and **O**rientation **H**istogram) [99], HOG (**H**istogram of **O**riented **G**radients) [34] or its significant extension GF-HOG (**G**radient **F**ield-**H**istogram of **O**riented **G**radients) [68] only differ in the methodology to capture the local appearance of the feature descriptor. [127] recently introduced ORB (**O**riented FAST and **R**otated **B**RIEF), which combines in an efficient way the keypoint detector FAST [125] with the efficient feature descriptor BRIEF [21]. FAST extracts keypoints even faster than GFTT or SIFT. However, as these methods do not offer any major improvement apart from faster computational time e.g. based on discretization by integral images like SURF, they are not considered further.

Literature reports two distinct approaches to solve the pose estimation problem. Model based methods rely on the extraction of specific geometric features in the image such as corners and edges. Robust features like SIFT, GFTT or SURF are mandatory for model-based object recognition and pose estimation. Clusters of robust image features are utilized in the first step to recognize the object. Afterwards the extracted features are compared and related to a known geometric model of the object. Efficient and reliable approaches for model based pose estimation with known correspondences have been proposed by [38, 115]. The drawback of these methods like any other model based approaches is the requirement of an a-priori geometric model of the object, an exact camera calibration as well as the solution of the correspondence problem, which becomes inherently more difficult in case of occlusion and ambiguous features. Following the model based paradigm, [56] therefore describes an approach for the construction of 3D metric models from multiple images taken with an uncalibrated handheld camera for augmented reality applications.

In contrast, global appearance based methods capture the overall visual appearance of an object, e.g. the multidimensional receptive fields introduced by [132]. Neither do they depend on the extraction of individual features nor do they face the correspondence problem. The basic idea is to capture the appearance by statistical representations such as histograms in order to calculate a probability of the object's presence in the current image view, an idea which is inherent to almost every appearance based approach. The methodology consists roughly of three steps, primarily low-dimensional local feature descriptors are calculated on a regular grid on the image, these descriptors are then quantized and aggregated in multi-dimensional histograms and finally compared to stored histograms of known objects exploiting the Bayes rule. The major difference between object recognition by clusters of SIFT features and by means of multidimensional receptive fields can be summarized as follows: SIFT features extract solely keypoints representing corner points and thereby assured textured image regions from which a highly distinguishable high-dimensional feature

descriptor can be determined, thereby exploiting all image information available. Multidimensional receptive fields on the contrary calculate a low-dimensional feature descriptor on a regular grid, thereby giving away information in textured highly distinguishable image regions and additionally sampling homogeneous regions with less information for the histograms as well. [22] propose distance color cooccurrence histograms for object recognition of multi-colored, textured objects, emphasizing the conservation of geometric information as the major advantage of color cooccurrence histograms compared to regular color histograms. Based on this fundamental idea, [43] propose color cooccurrence histograms for object recognition as well as 1 DOF pose estimation. The angular extension of color cooccurrence histograms is suggested by [106] in the context of pose estimation of robot players (AIBOs) as well as for 2 DOF pose estimation of multi-colored, textured objects [107]. [104] introduce a method that combines appearance and geometric object models in order to achieve robust and fast object detection as well as 2 DOF pose estimation. Their major contribution is the integration of the known 3D geometry of the object during matching and pose estimation by a statistical analysis of the distribution of feature appearances in the view space. Nonetheless their approach requires a 3D model of the object, which is difficult to generate for objects of complex shape and therefore the inherent problem of all model based approaches.

Image-based visual servoing presented in section 2.4 provides the means for model-free object manipulation for service robot applications without prior pose estimation requiring only an object recognition with e.g. clusters of GFTT or SIFT features and a subsequent control in the image space towards the desired locations of the features in the image plane. This approach leads to a high position accuracy, but nonetheless achieves only local convergence due to viewpoint limitations. Therefore an initial pose estimation is again mostly mandatory as the current object view does not necessarily contain the features close to the manipulation position. Global visual servoing introduced in chapter 6 overcomes the above stated limitations, thereby constituting a promising and more efficient approach compared to model and appearance based object recognition and pose estimation, neglecting any model knowledge but still incorporating the high position accuracy.

2.3 Visual navigation

The characterization of the different visual navigation concepts leads to the appraisal of topological map-based navigation with reactive visual behaviors as stated in section 1.3. Visual navigation draws its inspiration from biology which provides numerous examples of visual behaviors of insects and birds. It is challenging to design behaviors that are not based on distance sensors but on visual stimuli considering the burden of high computational complexity and noisy data. The authors in [1] extract the elements of early vision by defining the so-called plenoptic function which describes the visual information available to an observer at any point in space and time. Analyzing the plenoptic function yields the

definition of only four fundamental visual primitives, namely color, texture, disparity and optical flow to be utilized for designing visual behaviors.

Color corresponds to the different wavelengths in the visible range of the light spectrum. It requires model knowledge about the surrounding world e.g. the color information of objects like doors and side walls. Additionally the problem of color constancy is not solved yet, assigning always the same color to a homogeneous monochromatic area in spite of different illuminating conditions as described by the Dichromatic Reflection Model [82]. Therefore color is not suited for the navigation in unstructured environments. "*Texture is a phenomenon that is widespread, easy to recognise and hard to define*" [50]. Texture is understood by two similar but distinct meanings.

(1) Texture is defined as repeated patterns like carpet, hair or grass which have a specific response in the frequency domain, thereby extractable and distinguishable by filter banks as Garbor filters.
(2) Texture is defined as any difference to homogeneous regions exhibiting the same wavelength. Thus texture includes simple white spots in a black environment as well as paintings with a lot of unique structures and shapes maybe expressed by a set of widespread colors. The texture from definition 1 is a subspace of definition 2. Neither kind of texture is caused by shadowing, surface shape or other lighting effects.

Texture accordingly to definition (1) requires also model knowledge like carpet patterns about the surrounding world and is therefore not suited for mobile manipulation, e.g. navigation in unstructured environments. Texture from definition (2) is required for visual navigation to extract primitives like disparity and optical flow or advanced information such as visual landmarks. Notice that texture from definition (1) hinders the robust optical flow extraction thereby requiring more sophisticated algorithms to deal with repeating ambiguous patterns. Therefore texture is understood according to definition (2) throughout this thesis. Disparity is a brilliant clue as it directly leads to distance measurements, which allows mimicking of distance-based behavior. The determination of disparity requires a second extrinsically calibrated monocular camera and the solution of the correspondence problem, but is subject to the same shortcomings as optical flow as it necessitates the presence of texture in the environment. Optical flow is defined by the pixel motion between two images in the image space caused by the egomotion of the observer or moving objects in the field of view. The methods for calculation of optical flow can be classified into three groups: differential, frequency-based and matching [10]. A classification in terms of accuracy and density of the flow field is given by [11]. In this work differential methods are used for the sake of computational efficiency which compute velocities from spatio-temporal derivatives of image intensities. This is equivalent to the integration of velocities normal to the local intensity structure into full velocities either locally by least-square calculation [94] or globally via regularization [66]. Differential methods are based on the 2D motion

constraint equation:

$$\nabla \mathbf{I}(u,v,t)\mathbf{v} = -\frac{\partial \mathbf{I}(u,v,t)}{\partial t} \quad \Leftrightarrow \quad [I_u, I_v][\dot{u}, \dot{v}]^T = -I_t(u,v,t), \tag{2.13}$$

where $[I_u, I_v]$ is the spatial intensity gradient of the image $\mathbf{I}(u,v,t)$ and the image velocity $\mathbf{v}$, respectively the optical flow $[\dot{u}, \dot{v}]$ at the pixel $[u,v]$ at time t. The Lucas and Kanade algorithm [94] uses a weighted least-squares fit of local first-order constraints to a constant model for the image velocity in a small spatial neighborhood $\mathbf{\Omega}$ by minimizing:

$$\sum_{u,v \in \mathbf{\Omega}} w^2(u,v)[[I_u, I_v][\dot{u}, \dot{v}]^T + I_t(u,v,t)]^2 \tag{2.14}$$

with the weighting function $w(u,v)$ giving more influence to constraints at the neighborhood's center than at its periphery. These methods provide a dense optical flow field, nonetheless the motion direction is not estimated accurately enough, because homogeneous image regions cause ambiguous solutions of the correspondence problem and large flow is not observable due to phase restriction. The limitations of this algorithm are overcome by the Lucas Kanade pyramid algorithm [17] by creating a Gaussian image pyramid of the two images and calculating the optical flow iteratively at every level, thus providing the input for the next level. As the optical flow is determined only for qualified strong corners indicated by the Harris corner detector [59], the optical flow field is in comparison with other differential, intensity based methods sparse but therefore more accurate, even for large optical flow. Because of these advantages this method is used in the scope of this work for the estimation of the optical flow.

Therefore only texture and optical flow are required for the design of visual behaviors in unstructured indoor environments. Visual behaviors for a perspective camera as visual homing, collision avoidance and obstacle avoidance are introduced by [27]. The approach relies on fast image segmentation by template matching of carpet patches to detect free space in front of the robot and is therefore not suited for unstructured environments as it requires a huge amount of model knowledge. Their method is nonetheless at the same time point of departure as well as inspiration for sophisticated and more general visual behaviors.

Visual door passing: Robust and reliable door passing is feasible with laser range scanners as demonstrated by the reactive door passing behavior in [114]. However 2D laser range scanners are not suitable for the detection of closed or partially opened doors [75]. The robust visual detection and localization of doors still remains a challenging task despite a number of successful implementations in the past [42, 153, 140].The authors in [140, 101] detect doors by means of a monocular camera in conjunction with sonar, with the main disadvantage that the final door detection at close range relies on sonar information only. The second approach [101] relies on the assumption that the robot already faces the door, which excludes more realistic scenarios in which the robot travels along a corridor parallel to the doors. The door traversal approach by [42] is robust with respect to individual

pose errors, scene complexity and lighting conditions as door hypotheses are filtered and verified for consistency across multiple views. The door detection relies on a binocular pan-tilt camera system whereas the proposed approach uses an omnidirectional camera.

Visual obstacle avoidance and corridor centering: In order to achieve obstacle avoidance in indoor environments the authors in [31] determine the time to contact (*ttc*) in driving direction based on the divergence of the optical flow. The optical flow is a powerful image clue used for egomotion estimation [19], structure from motion [139] and for visual behaviors like corridor centering, wandering and target point following [32, 41]. [29] employ a monocular camera in conjunction with a lidar system in order to estimate obstacle velocities by a Kalman filter to avoid moving obstacles, whereas [28] integrate a laser based obstacle avoidance into the visual navigation.

Visual homing: Literature reports several distinct approaches for a visual homing behavior. Appearance-based homing of a non-holonomic robot is presented in [35]. Other approaches prefer feature-based navigation, e.g. in [46] a Euclidean reconstruction is performed based on a homography matrix relating the visual feedback to the position and orientation of the mobile robot in a local coordinate system. [57] presents a promising approach for merging the desired movements with the feasible motor commands of the non-holonomic robot. The robot uses a monocular vision system in conjunction with a Jacobian and geometry-based controller. In [20] a spherical image projection is applied in order to overcome the numerically ill-conditioned system equations for large pan angles. Their system uses natural landmarks which are either selected manually or automatically [83] and are detected by region-based image correlation. [14] introduce visual servoing on epipoles with visual memory. The stored trajectory of the epipoles in image space is learnt during a demonstration stage, which represents the desired robot trajectory.

2.4 Image-based visual servoing

A classification of visual servoing concepts is introduced by [129]. Therein two questions are addressed, primarily whether visual servo control directly drives the joints (direct visual servoing) or provides the input for an underlying joint controller (look-and-move). Most visual servoing implementations employ the look-and-move structure with underlying joint controllers, with the intention of decoupling kinematic and visual singularities, suppressing kinematic singularities by standard joint controllers, using different bandwidths for image processing and joint control as well as standard robotic interfaces with setpoints for Cartesian velocity and incremental movements. In this thesis look-and-move structures are used exclusively. Secondly, visual servoing is classified into position or image-based visual servoing, depending on whether the control input consists of a pose estimation of the end-effector with respect to the work piece or a direct calculation of the error signal in the image space. Position-based visual servoing (PBVS) issues model generation for every

object to be manipulated, solving the correspondence problem and pose estimation as well as an intrinsic and extrinsic camera calibration. In addition to errors in the model generation caused e.g. by imperfect intrinsic and extrinsic camera calibration, deviations in the inverse kinematics also contribute to control deviation. PBVS is therefore well suited for robotic manipulators in industrial settings with predefined and predictable systems but not for service robot applications, which incorporate high uncertainty about the environment. This thesis advocates in the following image-based visual servoing (IBVS). Image-based visual servoing offers the advantages that camera calibration and robot's kinematics errors do not result in a control deviation and that it does not require any object model.

The visual information is provided either by camera systems fixed in the workspace observing the robot's motion or by a so-called eye-in-hand configuration, in which the camera is attached directly to the robot and thereby exhibiting the robot's motion in the task space. Eye-in-hand configurations are convenient for mobile robots in unstructured environments due to their complete awareness of the surrounding. Additionally they provide high position accuracy close to the goal pose because of their projection scale, therefore visual servoing in this thesis postulates eye-in-hand configurations. Figure 2.2 illustrates the image-based visual servoing employing a look-and-move structure for eye-in-hand object manipulation. The visual reference features $\mathbf{f}_{\text{ref}}$ are defined directly in the 2D image plane, making a geometric model or reconstruction of the environment obsolete. The task space velocities and the corresponding joint velocities of the manipulator are calculated according to the error $\Delta\mathbf{f}$ between the desired $\mathbf{f}_{\text{ref}}$ and the actual feature locations $\mathbf{f}$. The robot and the camera motion regulate the feature error, which vanishes as the current and reference pose coincide. A known shortcoming of image-based visual controllers is the

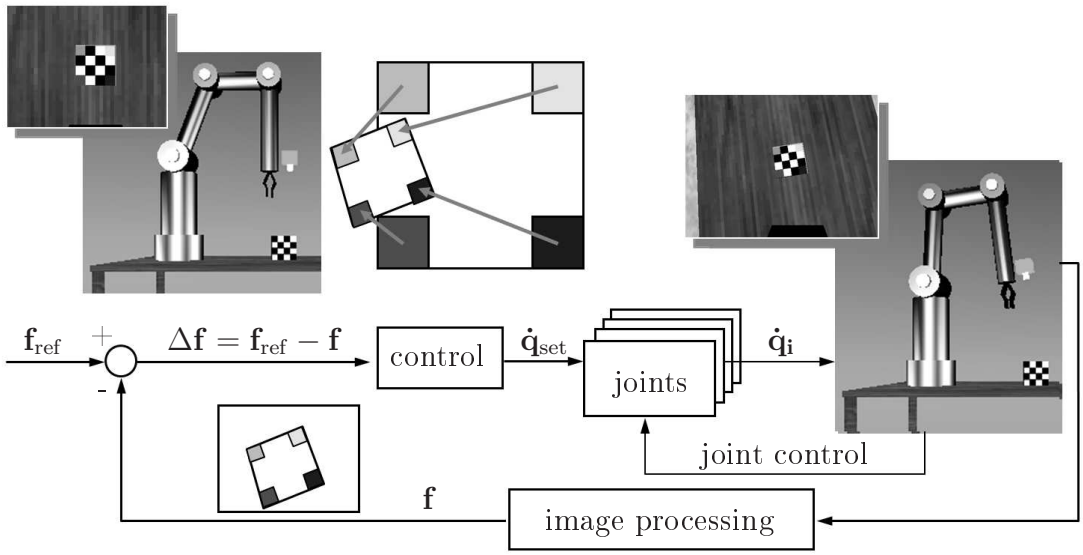


Figure 2.2: Image-based visual servoing (IBVS) in a look-and-move structure for eye-in-hand visual object manipulation.

camera retreat problem. The problem is constituted by the fact that optimal trajectories in the image space might result in singularities or infeasible trajectories in task space. The image-based controller minimizes the image error linearly in the image space. If the camera is only rotated by 180° compared to the goal pose, instead of the appropriate motion in task space, namely a counter-rotation of the camera around the optical axis of about 180°, the camera retreats from the scene in order to minimize the error linearly. As the camera retreats from the scene the feature points travel to the image center and end up in a singularity. A possible solution for manipulators to decouple the translational and rotational velocities is proposed in [33]. Based on perspective projection an angular criterion is developed, which takes into account the trapezoidal distortion of a square based on a rotation around one of the axes spanning the image plane. The presented prerequisite for image-based visual servoing follows the classification of [24, 25] for the different approaches in literature and summarizes the pros and cons for service robotic applications.

Visual servoing with (adaptive) image Jacobian: Visual servoing based on the image Jacobian $\mathbf{J}$ inverts the analytical relation between differential changes in task space to differential changes of pixel coordinates to reduce $\Delta\mathbf{f}$ [151]. The simple proportional control law is given by:

$$\dot{\mathbf{r}} = -\mathbf{k}\mathbf{J}^{+}(\mathbf{r})(\mathbf{f}_{\text{ref}} - \mathbf{f}), \tag{2.15}$$

where $\mathbf{J}^{+}$ is the pseudoinverse of the image Jacobian $\mathbf{J}$ and $\mathbf{k}$ a constant gain factor. $\mathbf{k}$ ensures an exponential decrease of the error as $\Delta\dot{\mathbf{f}} = -\mathbf{k}\Delta\mathbf{f}$. The image Jacobian $\mathbf{J}$ also referred to as interaction or sensitivity matrix is derived in [70]:

$$\mathbf{J} = \begin{bmatrix} -\frac{\lambda}{z} & 0 & \frac{u}{z} & \frac{uv}{\lambda} & \frac{-\lambda^2-u^2}{\lambda} & v \\ 0 & -\frac{\lambda}{z} & \frac{v}{z} & \frac{\lambda^2+v^2}{\lambda} & \frac{-uv}{\lambda} & -u \end{bmatrix}. \tag{2.16}$$

As the analytical determination of $\mathbf{J}$ requires the knowledge of z for each point feature, different approaches for the determination are stated in literature. $\mathbf{J}$ is determined with the distance z^* at the goal pose and remains static during visual control. A better performance in terms of convergence is achieved by determining $\mathbf{J}$ via the algebraic mean of z in the current and z^* in the goal configuration [24]. As an alternative an adaptive approach is introduced by [77] in which the image Jacobian $\mathbf{J}$ is estimated by the predicted feature motion due to the motion of the camera $\Delta\mathbf{r}$ and the observed motion $\Delta\mathbf{f}$ (optical flow) according to:

$$\mathbf{J}_{k+1} = \mathbf{J}_k + \alpha_a \frac{(\Delta\mathbf{f} - \mathbf{J}_k\Delta\mathbf{r})\Delta\mathbf{r}^T}{\Delta\mathbf{r}^T\Delta\mathbf{r}}. \tag{2.17}$$

α_a denotes the correction factor for the adaptive image Jacobian. As no knowledge of the distance z is required this approach seems particulary interesting for service robot applications. As the control by the image Jacobian assumes a linear model between the pixel and camera motion, a trust region control [137] is especially suitable to guarantee strictly linear control:

$$\dot{\mathbf{r}} = -\mathbf{k}_a\mathbf{J}^{+}(\mathbf{r})\Delta\mathbf{f} \quad \text{with} \quad \mathbf{k}_a = \min\left(1, \frac{a_k}{l_k}\right). \tag{2.18}$$

The constant proportional gain $\mathbf{k}$ is replaced by an adaptive gain $\mathbf{k}_a$, which is determined by the boundary of the pixel displacement a_k and the prediction of the image displacement $l_k = \mathbf{J}\Delta\mathbf{r}$. a_k compromises a large control variable for fast convergence with linear regime of $\mathbf{J}$. Nonetheless for eye-in-hand configuration the promising concept of adaptive image Jacobian and trust region control proves to be inapplicable as a translational motion in x (respectively y) is difficult to distinguish from a rotational movement around the y-axis (respectively x-axis), resulting in an almost identical optical flow. In conjunction with small image noise the result is false convergence of the adaptive Jacobian and local minima in control. In order to overcome theses disadvantages [118] presents a completive study between analytical and adaptive Jacobian in 3DOF, incorporating additionally the epipolar constraint in the adaptation. [130] propose a calibration-free Jacobian by re-expressing and online adaptation of focal length and scale in each control cycle. The concept of adaptive image Jacobian is well suited for fixed camera systems observing the robot's motion and the object or applications with reduced degrees of motion of the camera, but has no practical importance for mobile robots.

Hybrid visual servoing: In order to improve visual servoing [26] propose two and one-half-dimensional visual servoing in order to exploit the advantages of IBVS and PBVS. $2\frac{1}{2}$D visual servoing decouples rotational and translational velocity control by primarily estimating the rotation between the current and desired object view, e.g. from decomposition of the homography $\mathbf{H}$ (cf. section 2.1). The control for the rotational motions is expressed as:

$$\omega = -\mathbf{k}\xi u_\xi, \tag{2.19}$$

whereas ξ and u_ξ correspond to the angle and axis of the rotation parametrization, respectively. Contrary to the rotational motions the translational error is corrected directly in the image space:

$$v = -\mathbf{J}_{vt}^{+}(\mathbf{k}\Delta\mathbf{f}_t + \mathbf{J}_{v\omega}\omega), \tag{2.20}$$

$\mathbf{J}_{vt}$ and $\mathbf{J}_{v\omega}$ correspond to separated Jacobians for the translational and rotational motions and $\mathbf{J}_{v\xi u_\xi}$ is the separated Jacobian for angle and axis of rotation parametrization. $\Delta\mathbf{f}_t$ is defined by the error in the image space caused by a translational deviation from the desired view. The control law is defined analogous to equation 2.15 as follows:

$$\dot{\mathbf{r}} = -\mathbf{k}\mathbf{J}^{+}\Delta\mathbf{f} \quad \text{with} \quad \Delta\mathbf{f} = (\mathbf{f}_t, \xi u_\xi) \quad \text{and} \quad \mathbf{J} = \begin{bmatrix} \mathbf{J}_{vt} & \mathbf{J}_{v\omega} \\ 0 & \mathbf{J}_{v\xi u_\xi} \end{bmatrix}. \tag{2.21}$$

Note that $\mathbf{J}_{vt}$ and $\mathbf{J}_{v\omega}$ are expressed similar to the general $\mathbf{J}$ with the major advantage that the distance z to the object is expressed in terms of the ratio $\frac{t}{d}$ obtained by the homography decomposition. Image-based visual servoing based on Jacobian shows local asymptotic stability, whereas $2\frac{1}{2}$D visual servoing achieves even global asymptotic stability.

Visual servoing with decoupled image moments: This is referred to as "*partitioned visual servo*" in the classification from [25]. Visual servoing by image moments is investigated by [45, 69] using the distance between two image points as well as their orientation.

Moments of higher order based on projected image regions are introduced by [151]. The motivation for decoupled image moments is to find an interaction matrix, which establishes a one-to-one relationship between image moments and their degrees of motion, resulting in a simple linear control problem. Analogous to the hybrid visual servoing the rotational and translational degrees of freedom should be completely decoupled resulting in a smooth convergence in the 3D task space. [143] describes image moments using coplanar closed contours, which enable a decoupled control scheme if the object is orientated parallel to the image plane. Recently these ideas where extended by [145] showing a dependence of the few remaining off-diagonal couplings with the object shape. [9] presents visual servoing by photometric moments describing the global appearance. The authors in [116] employ visual servoing with decoupled image moments for controlling the position and orientation of a quadrotor relative to observed landmarks on the ground.

Visual servoing on epipoles: Visual servoing based on epipolar geometry is first introduced by [124]. Using the epipolar constraint the error is defined as the distance of an interest point to its appropriate epipolar line in terms of a_i and b_i, corresponding to the u- and v-direction, respectively. The control law is defined as:

$$\dot{\mathbf{r}} = -K{\mathbf{J}_\mathbf{e}}^{+}\Delta\mathbf{f} \quad \text{with} \quad \mathbf{J}_\mathbf{e} = [a_i, b_i]\mathbf{J}. \tag{2.22}$$

[120] extends this idea by employing multi-view visual servoing directly on the epipoles, which benefits from three images taken during a training stage. The set of reference images consists of a desired image and two additional reference views taken from two distinct vantage points. The visual servoing control law defines the error in terms of the difference between the epipoles from the desired view $[\mathbf{e}^1_{\text{ref}}, \mathbf{e}^2_{\text{ref}}]^T$ and the epipoles $[\mathbf{e}^1_a, \mathbf{e}^2_a]^T$ from the actual image onto the two additional reference views (projection of the optical center of the first camera onto the second camera). Therefore two essential matrices $\mathbf{E}$ are estimated during each control cycle. Nonetheless the control completely decouples translational and rotational motions, in which the rotational control is primarily used to keep the features in the field of view. This approach is less promising regarding service robotics because three reference images are required in conjunction with sequential estimations of $\mathbf{E}$.

Appearance based visual servoing: Appearance based visual servoing (direct visual servoing) is classified as an image-based visual servoing method, whereas the appearance of the object (cf. section 2.2) is directly provided as input to the controller instead of extracted point features. An approach based on angular color cooccurrence histograms in conjunction with reinforcement learning satisfying the continuous action and state space requirement is successfully demonstrated by [80]. An agent learns online the optimal policy $\boldsymbol{\pi}(s, a)$ which is defined as: $\boldsymbol{\pi}(s, a) = \text{argmax}_a \mathbf{Q}(s, a)$, whereas the action value function $\mathbf{Q}(s, a)$ contains the mapping from object appearances s (angular color cooccurrence histograms) to the control action a in order to reach the grasping pose. [37] captures the appearance of an object by a PCA (**P**rinciple **C**omponent **A**nalysis) in order to reduce the high dimensionality of the intensity image followed by an offline training stage for the interaction matrix. These methods initially require an object recognition step as well as a

continually accurate object segmentation during visual servoing for object manipulation. The latter is difficult to achieve in textured environments. A novel approach referred to as luminance based visual servoing is presented in [30], performing the visual servoing directly on the image intensities. The error $\Delta\mathbf{f}$ is thereby defined by the difference of all intensities between the current image $\mathbf{I}$ and the reference image $\mathbf{I_{ref}}$ while the interaction matrix $\mathbf{J_a}$ is determined in terms of the 2D motion constraint from equation 2.13:

$$\frac{\partial \mathbf{I}(u,t)}{\partial t} = -\nabla \mathbf{I}(u,t)\mathbf{v} \quad \Rightarrow \quad \mathbf{J_a} = -(\nabla \mathbf{I_u}\mathbf{J}(\dot{\mathbf{u}}) + \mathbf{I_v}\nabla \mathbf{J}(\dot{\mathbf{v}})). \tag{2.23}$$

By reformulating the visual servoing as an optimization problem, the control law using the Levenberg-Marquardt optimization algorithm is defined as follows:

$$\dot{\mathbf{r}} = -\mathbf{k}(\mathbf{J_a}^T\mathbf{J_a} + \mu\,\mathrm{diag}(\mathbf{J_a}^T\mathbf{J_a}))^{-1}\mathbf{J_a}^T(\mathbf{I} - \mathbf{I_{ref}}), \tag{2.24}$$

where the parameter μ is chosen in dependence of the cost function in order to switch between steepest descent and Gauss-Newton optimization. Although luminance based visual servoing is quite promising, it requires object recognition and segmentation. This renders the approach solely suitable for mobile navigation. Nonetheless appearance based visual servoing is a novel recently emerging branch within the field of visual control e.g. for micropositioning of microelectromechanical structures [144] and is a promising avenue as it is presumably close to human object recognition and manipulation.

Visual servoing with structured light: Visual servoing with structured light rather describes the exploitation of active visual sensors than actually representing novel visual servoing concepts. The authors in [117] propose a camera setup with four laser pointers (structured light) for visual servoing relative to planar objects. The structured light not only eases the feature extraction stage, enabling the control also for objects with homogeneous surfaces, but additionally allows for decoupled visual servoing close to the goal position by image moments resulting in a good task-space trajectory. This methodology therefore falls under the category of visual servoing with decoupled image moments. The task of automatic seam filling in the context of aircraft construction is solved by [63] with a hybrid visual servoing scheme with structured light. Hybrid control with structured light combines position-based visual servoing, which locally reconstructs the pose between tool and workpiece to regulate the robot perpendicularly to the workpiece's surface, with image-based visual servoing used for centering and tracking the seams to be filled. The concept of enforcing texture by structured light onto homogeneous object regions is promising, as it transforms the passive sensor camera into an active sensor system. Utilizing structured light in the visible range is questionable for service robotics and therefore not followed in the context of this thesis. **T**ime-**o**f-**F**light (ToF) [84] cameras additionally provide depth information for the 2D image plane, but at the cost of low camera resolution and high power consumption. Because of the low resolution and the 3D information these cameras are suited for low demanding PBVS [122] and might have their main application area in autonomous navigation.

The characterization of the different visual servoing concepts leads to the appraisal of visual servoing concepts stated in section 1.3 and summarized in figure 1.3. Visual servoing with decoupled image moments excellently complies with the service robotic specifications. The three-stage design methodology for vision and control applications presented in [113] is applied for systematically development of visual servoing with decoupled image moments, which is conform with design methodology for mechatronic systems [150].

2.5 Experimental systems for visual servoing, navigation and localization

This section describes the fundamentals and the set-ups of the different experimental systems used in this thesis. The mobile platform Pioneer 3-DX from MobileRobots Inc. in conjunction with a 5 DOF manipulator Katana 6M from Neuronics is used to achieve mobile manipulation in indoor office environments. To evaluate the visual object manipulation in 6 DOF, the proposed visual servoing schemes are applied to an industrial manipulator RV 20-16 from Reis which is introduced as well.

A mobile platform of the type Pioneer 3-DX is employed in the course of this thesis for the transition of vision guided to visual navigation as presented in chapters 3 and 4. It possesses a ring of eight forward and rear sonar sensors, because sonar is affordable and robust. These sensors are indispensable as a back-up sensor in the case that the visual perception fails. The experimental set-up is equipped with a Sick Laser ranger LMS 200 (cf. figure 2.3). The robot is additionally equipped with the Pan-Tilt-Zoom camera (PTZ camera) VC-C4 from Canon as well as an omnidirectional camera system consisting of a camera DFK-31AF03 from Imaging Source and a D40 optic. The platform Pioneer 3-DX is a two-wheel differential-drive robot with an additional castor wheel for stabilization. The robot kinematics is non-holonomic as it possesses fewer local degrees of freedom than its global state space. The motion of the differential drive robot is restricted to translation along its current heading and rotations around the vertical axis, but it is unable to move sideways. The robot state is defined by $[x_R, z_R, \theta_R]^T$ in order to comply with the usual camera coordinate frame. The z-axis is along the robot's direction of motion, the x-axis is horizontally orientated and the y-axis of robot and camera rotation is vertically orientated. The differential drive robot motion is described by a velocity motion model as

$$\begin{aligned} x(k+1)_R &= x(k)_R + v_R \Delta t \sin(\theta_R) \\ z(k+1)_R &= z(k)_R + v_R \Delta t \cos(\theta_R) \\ \theta(k+1)_R &= \theta(k)_R + \omega_R \Delta t, \end{aligned} \tag{2.25}$$

where v_R and ω_R denote the translational and rotational velocity of the robot. Alternatively an odometry motion model is applied in this thesis, which uses the odometer measurements in order to compute the robot's pose.

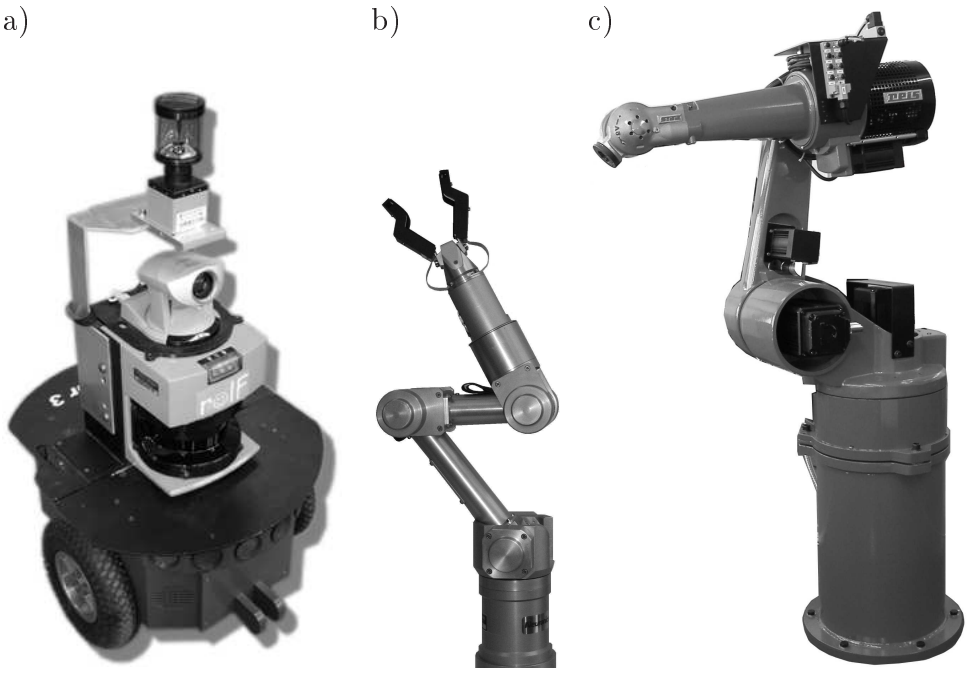


Figure 2.3: a) Mobile Platform equipped with sonar rings, laser range finder, omnidirectional and PTZ camera; b) Katana 6M from Neuronics; c) RV 20-16 from Reis.

The robot arm Katana 6M is employed in this thesis for developing a novel visual servo controller for object manipulation. It is a five degree of freedom (DOF) serial-link manipulator as shown in figure 2.3 b). The arm is equipped with a two-finger gripper at the end-effector with five integrated infrared (IR) proximity sensors and force sensors. The forward kinematics, which determines the position and orientation $\mathbf{x}$ of the end-effector as a function of the joint angles $\mathbf{q}$ is defined as: $\mathbf{x} = \mathbf{k}(\mathbf{q})$. $\mathbf{x}$ is defined as $[x, y, z, \alpha, \beta, \gamma]$, whereas $[x, y, z]$ describes the pose in Cartesian coordinates and $[\alpha, \beta, \gamma]$ the rotation around the x, y and z axes (roll, pitch, yaw). The inverse kinematics consists of the determination of the appropriate joint angles to a specified end-effector position: $\mathbf{q} = \mathbf{k}^{-1}(\mathbf{x})$. As most of the thesis is concerned with visual servoing, which implies velocities in the task space, the differential kinematics are essentially defined by: $\dot{\mathbf{x}} = \mathbf{J}_{\mathrm{dk}}(\mathbf{q})\dot{\mathbf{q}}$. The implemented forward, inverse as well as the differential kinematics including a collision detection are state of the art and therefore assumed to be known. These functionalities and terms are used in the thesis without further explanations.

The industrial robot RV20-16 from Reis is a six degree of freedom manipulator with a maximum payload of 16 kg. The absolute positioning error is about one to two mm and the repeatability is specified with 0.05 mm. The RV20-16 is depicted in figure 2.3 c). The Katana 6M as well as the industrial robot RV20-16 are used for visual servoing for object gripper alignment as described in chapter 5 and 6. Additionally the grasping and manipulation of daily objects with the Katana 6M for service robotic applications is described in [81] using the presented control schemes.

Chapter 3

From vision guided to visual navigation of mobile robots

This chapter investigates the possibility to represent the set of required behaviors for topological navigation in unstructured indoor environments solely due to an omnidirectional camera. Starting point for the investigations is the vision guided navigation scheme based on sonar and laser presented in [114] with a topological map without prior structuring of the environment. As a replacement for the distance sensors the omnidirectional image provides the stimuli for a novel obstacle avoidance by means of several reconstructed perspective views, from which a confidence rated time to contact is extracted [112]. A visual door passing behavior is treated in a coherent purely vision-based framework [121].

Examples and an overview for visual behaviors are provided in section 2.3. Inspirational for the following investigations is the fundamental work by [27] of implementing visual behaviors based on a monocular camera. The target of the following investigations is to represent as far as possible all required visual behaviors for indoor office navigation solely by omnivision (cf. section 2.1) due to its inherent advantages such as its 360° field of view contrary to other approaches e.g. [29, 28]. Additionally the design of the behaviors is mandatory to be model-free to operate in unstructured environments. Using a more sophisticated camera system a complete framework for visual navigation is provided in the following requiring no artificial structuring of the environment.

This chapter is organized as follows: Vision guided navigation is introduced as a starting point in section 3.1 with topological localization using omnivision and reactive behaviors with distance sensors. The door passing behavior is described in section 3.2. Visual behaviors for obstacle avoidance and corridor centering using omnivision are described in section 3.3 and investigated during an experimental evaluation.

3.1 Vision-guided navigation

Vision-guided navigation pursues a learning by demonstration scheme for the topological navigation. According to the hybrid control architecture depicted in figure 1.2 the stimuli for the reactive layer is provided by range sensors, whereas the planning layer perceives its local environment by means of an omnidirectional camera. The experimental platform is a Pioneer 3DX mobile robot equipped with a 2D laser scanner, an omnidirectional camera system and a ring of sonar sensors (cf. figure 2.3 a)). The distance sensors (laser, sonar) capture the local environment of the robot and provide the stimuli for the obstacle avoidance, the corridor centering, the door detection and the door passing behavior. The robot localizes itself within a topological map based on detected correspondences between omnidirectional views.

3.1.1 Planning

The presented method for vision-guided navigation requires a learning phase in which a graph is created offline (figure 3.1). A topological map represented by a directed graph forms the basis of path planning and navigation. In this representation of the office environment nodes in the graph constitute waypoints and are associated with distinctive visual features. The visual features are later recognized in the current camera image and enable a unique association with nodes in the topological map by finding correspondences with stored features. During the learning phase the robot is guided manually through the environment, and at relevant locations for navigation the corresponding SIFT features are extracted and added as nodes to the topological map. Neighboring nodes are connected via edges in the graph. It is assumed that navigation between connected waypoints does not require localization but is entirely accomplished by means of reactive behaviors such as corridor following and door passing. Depending on the type of neighboring relationship the planning layer engages the combination of reactive behaviors that is suitable in the current context, e.g. door passing and obstacle avoidance for a pair of waypoints connected through a door. Topological path planning is reduced to graph search, which is solved by the common Dijkstra algorithm [40]. The context of an edge depends on the type of connection between waypoints, and the coordination layer, depending on the context provided by the planning layer, activates the following subsets of reactive behaviors:

- Corridor: obstacle avoidance, constant velocity and corridor centering
- Door passing: obstacle avoidance, constant velocity, door passing and homing
- Open spaces (in analogy to corridor): obstacle avoidance, constant velocity and, if required, homing

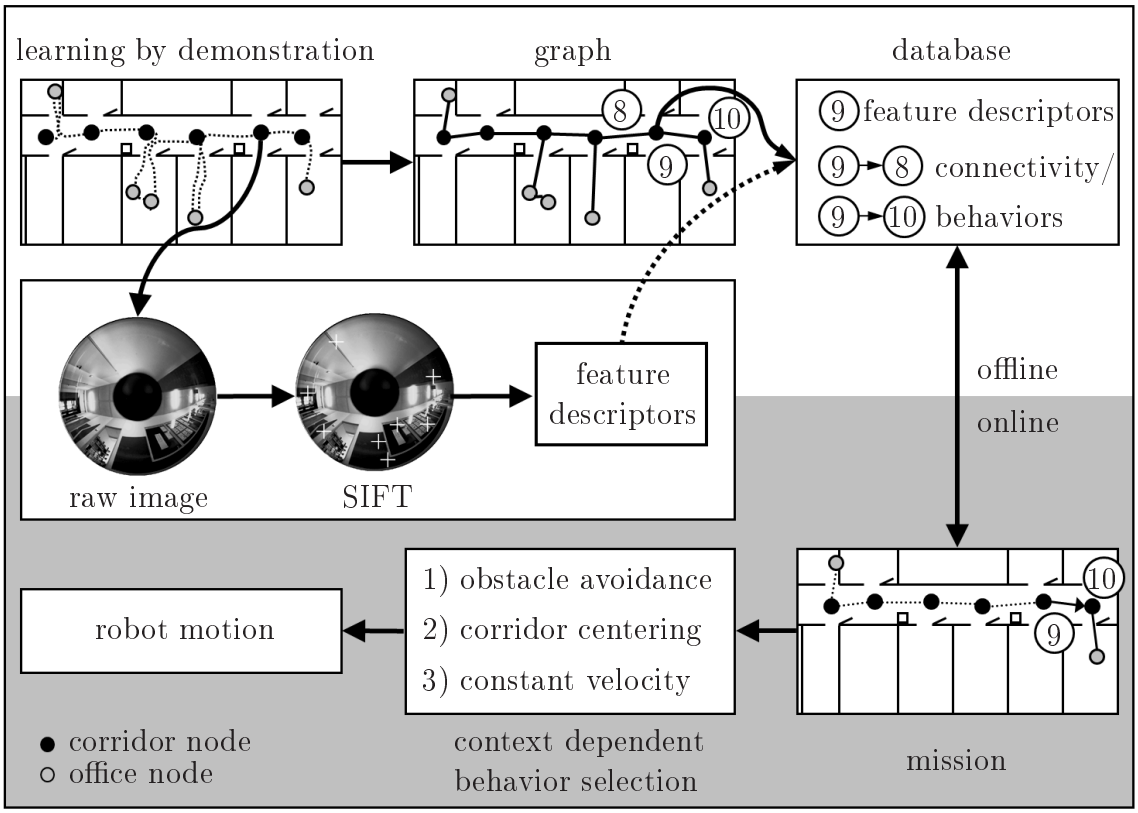


Figure 3.1: Vision-guided navigation: learning by demonstration offline (white background) and context dependent choice of behavior online (grey background).

3.1.2 Topological localization

In order to generate a topological map the robot is guided through its environment in a demonstration while reference images of the local surroundings are captured at distinctive locations such as doors or junctions. Waypoints are unambiguously described by the SIFT features detected in the omnidirectional image of the corresponding environment similar to [2]. Figure 3.2 depicts an omnidirectional image with the corresponding detected SIFT features. Neighboring waypoints in the image are associated with nodes connected via edges in the graph. Localization in the topological map is achieved by similarity of the current view with the reference images captured during demonstration. The high density of SIFT features in natural environments allows to introduce waypoints at arbitrary locations in the desired density without explicit reference to specific landmarks. The SIFT features detected in the current view are compared with those stored in the database. The measure of similarity between two locations is expressed by the relative frequency of corresponding SIFT features in the current and reference image. To activate a node, ten topologically next neighbors of the last traversed node are compared with the SIFT features of the current camera image. This neighbor search enables a continuing localization, even if the next node

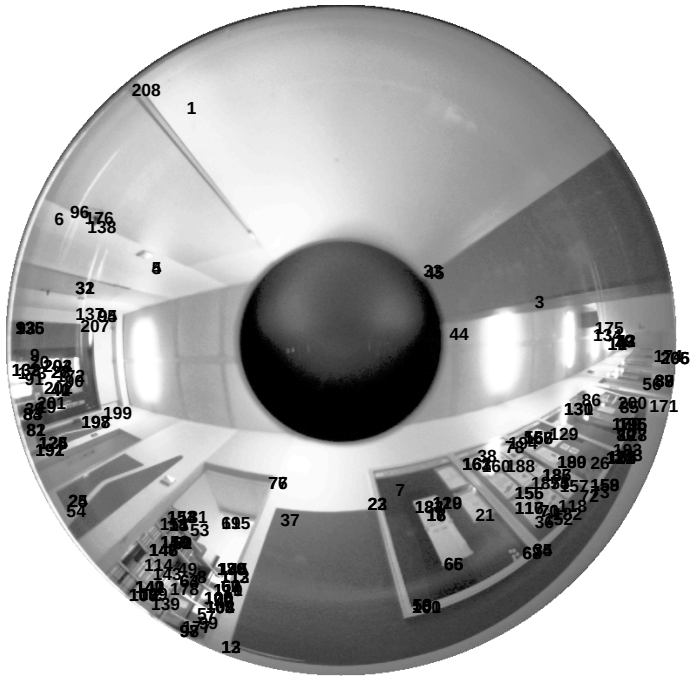


Figure 3.2: Omnidirectional camera image with blind spot and extracted SIFT features.

according to the plan is not identified or missed. However, this rare non-identification of a topological node in practice only occurs if a large majority of the features is occluded at the waypoint. The method is also suited without restriction for global localization of the robot, whereas the computing time for initial comparison of the current view with all stored views in the database increases accordingly. The measure of similarity between SIFT features is obtained as explained in chapter 2.2. A waypoint is recognized if at least 20% of the features in the current view are in agreement with those of a reference view in the database. If several topological nodes exceed this threshold at the same time, the node with the highest correlation is selected. Simultaneous exceeding of this threshold value only occurs in case of very small distances of the nodes, i.e. below 30 cm. If the correspondence is larger than 10% but smaller than 20%, a node is recognized if the similarity of the node with the highest correlation is at least twice as high as the one with the second highest correlation. In order to evaluate the performance and robustness of the proposed scheme the robot travels along a corridor passing ten previously demonstrated waypoints as depicted in figure 3.3. The waypoints are distributed in front of the doors in order to activate the door passing behavior if the appropriate door to be passed according to the plan is reached. The nodes are sequentially activated in the right order, indicated by the filled out dots in figure 3.3 b), c) and d), during a navigation through the corridor under different light conditions. The local resolution of the localization is obviously based on the high activation level of the nodes only in the vicinity of the waypoints. The robustness of the method is evident, as all passed waypoints are recognized reliably while the corresponding door hypotheses are created.

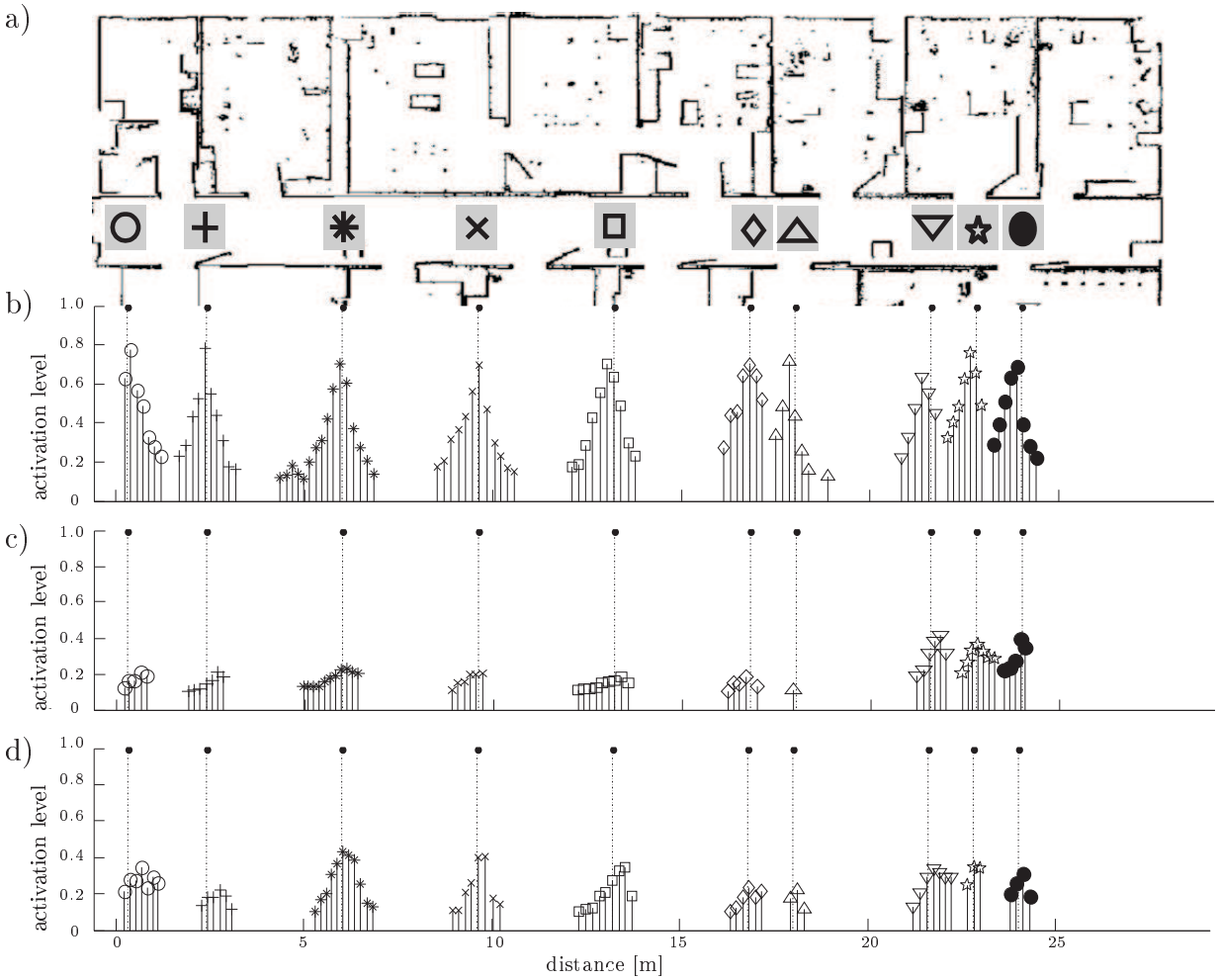


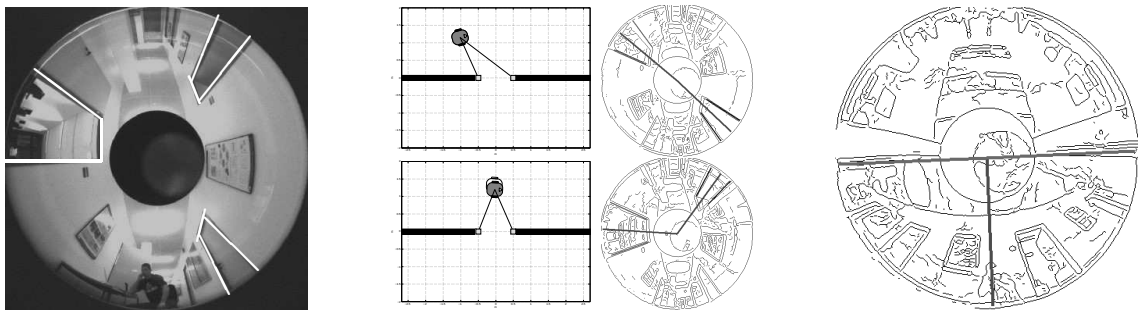
Figure 3.3: Specificity of SIFT features for varying illumination and surroundings: a) Metric map of the corridor with waypoints in front of the doors; b) Node activation for identical illumination conditions; c) Node activation for different illumination conditions and lateral distance to the capture point of one fourth of the corridor width; d) Node activation for strongly changed illumination conditions (different time of day, closed doors).

Reactive behaviors with distance sensors: A set of configurable fuzzy behaviors is designed according to [128]. The goal-oriented navigation results from the interaction of constant velocity, homing, corridor centering, obstacle avoidance and door passing behavior. Details of the fuzzy behavior representation of these behaviors can be found in [61] as well as the behavior coordination. The design of a reactive door passing behavior as well as a detailed description of the performance and robustness of vision-guided navigation is discussed in [114]. A video of the experimental evaluation of vision-guided navigation including localization, reactive behavior coordination and door passing can be downloaded from the website of the department [126]. The transition from vision-guided to vision-based navigation is subject of the following chapter in which the behaviors are represented

instead of distance measurements solely by means of visual perception primitives such as optical flow and texture.

3.2 Visual behavior for door passing

Visual door traversal is a vital skill for autonomous mobile robots operating in indoor environments. An omnidirectional view offers the advantage that an initial scan of the environment for doors by rotating the robot base becomes obsolete. In addition the omniview guarantees that the door remains visible throughout the entire door traversal whereas with a conventional perspective camera the door eventually leaves the field of the view such that the final stage of door traversal is performed in open loop control. The omniview also offers an advantage in scenarios with semi open doors in which the robot still detects the door in the rear view after it has passed the door leaf. The objective of the approach presented in [121] is to provide a robust solution of the entire door detection and navigation problem relying on omnidirectional vision only. The vision-based door recognition and traversal problem is structured into the three steps of door detection (a), door localization and tracking (b) and door traversal (c) as shown in figure 3.4, which are shortly summarized.



a) Door detection. b) Door tracking. c) Door passing.

Figure 3.4: Visual door passing behavior [121].

a) Door detection: Images contain a large amount of information which necessitates the filtering, extraction and interpretation of those image features that are relevant to the visual door detection. The door detection is composed of three subsequent fundamental steps: image processing, line processing and door frame recognition. The image processing involves edge detection, thinning, gap bridging, pruning and edge linking. Afterwards the line processing aggregates individual edge segments into categorized lines by means of line approximation, line segmentation, horizontal and vertical line selection as well as line merging. Similar to other approaches in the past the door detection scheme relies on a door frame model composed of two vertical door posts in conjunction with a horizontal top segment. The final step in the door detection comprehends the matching between plausible

combinations of vertical and horizontal lines with multiple potential door frame patterns. These door patterns are inspired by the work of [103], which defines simple and double door frames.

b) Door localization and tracking: Door localization estimates the robot's current pose (x_R, z_R, θ_R) with respect to the door coordinate system. In case of monocular cameras the robot pose is usually recovered by triangulation of features across multiple captures taken from different viewpoints. In the literature this localization scheme is known as bearing only localization. The built-in odometry estimates the relative robot motion between consecutive viewpoints. Since both the measurement and the motion are subject to noise and errors, the robot position with respect to the door is estimated with an **E**xtended **K**alman **F**ilter (EKF) [152]. The state prediction of the EKF relies on the odometry motion model, which describes the relative robot motion between two consecutive poses [147]. The initial two consecutive door detections are used to initialize the states and covariances of the Kalman filter. Afterwards the door localization is based on sequential prediction and update steps.

c) Door passing: Typically the door is first detected in the image once the robot is about two to three meters away from the door. The door is tracked continuously by means of the Kalman filter and the robot continues its motion parallel to the corridor until the robot is located laterally with respect to the door center. At this instance the robot stops and turns 90° towards the door, continuously tracking its relative orientation. Before initiating the traversal the open door state is verified from a sequence of images by extraction of the time to contact and the texture. A homogeneous texture indicates a closed door, whereas random texture implies an open door. A large time to contact guarantees safe traversal. The robot traverses the door at constant velocity by centering itself with respect to the continuously tracked door posts. The visual servoing controls the robot's turn rate such that both door posts remain equilateral in the omnidirectional view. The Kalman filter is no longer applied as the depth information becomes unreliable at close range and is not needed for guiding the robot through the door.

The algorithm is tested on 1000 manually labeled images taken from video sequences captured in the office environment of the department. Most images contain multiple doors, such as the scenario in figure 3.4 with three successfully detected and validated doors. False positives in single images for doors amount to 3%, false negatives occur 5% of the time [121]. To render the detection algorithm even more robust, the door frames are tracked over consecutive images during the motion. Initial false positives are eventually rejected in subsequent captures. This validation step is of particular importance for the Kalman filter localization.

3.3 Visual behaviors for collision-free navigation

3.3.1 Corridor centering

Corridor centering behavior has a strong resemblance with the visual-motor behavior of honey bees. Bees fly by balancing the optical flow generated in the lateral portion of the optic array of both eyes. This strategy enables them to fly exactly in the center of a tunnel. The corridor following behavior balances the magnitude of the optical flow generated on both the left and right hemisphere of the omnidirectional camera to drive the robot towards the center of the corridor [32]. A simple but robust control law results from the comparison of the magnitude (its maximum) of the optical flow in the left and right hemisphere $[\dot{\mathbf{u}}, \dot{\mathbf{v}}]^T_{\text{left}}$ and $[\dot{\mathbf{u}}, \dot{\mathbf{v}}]^T_{\text{right}}$, respectively:

$$\Delta\Theta_R = k(\max([\dot{\mathbf{u}}, \dot{\mathbf{v}}]^T_{\text{left}}) - \max([\dot{\mathbf{u}}, \dot{\mathbf{v}}]^T_{\text{right}})). \tag{3.1}$$

It is based on the assumption that the region which generates a larger optical flow contains objects in closer proximity to the robot than the opposite side. In equation 3.1 Θ_R describes the rotation and k is a proportional gain factor. The corridor following with the omnidirectional camera detects the optical flow across an angular region of 45° to 135° on both sides of the translation direction for balancing.

3.3.2 Obstacle avoidance by optical flow

To avoid obstacles in front of the robot the time to contact is estimated from the divergence of the optical flow onto an image grid from the reconstructed perspective frontal view. Time to contact estimates are fused with the confidence in the respective visual information, namely the local variance of optical flow and statistical analysis [112]. Both together determine the desirability and safety of traveling in the corresponding direction. Oscillatory movements of the robot are prevented by reconstruction of two additional peripheral views for which the time to contact is measured solely by optical quantities, e.g. the object's viewing angle and its derivative. A general overview of methods for optical flow extraction based on a time sequence of images is given in chapter 2.3. According to the performance evaluation of optical flow techniques for indoor navigation with a mobile robot, the approach from Lucas and Kanade achieves the best results in terms of accuracy, efficiency and robustness [96]. Therefore the Lucas-Kanade pyramid algorithm from [17] is used which provides a sparse but more accurate optical flow field compared to other differential, intensity based methods, even for large optical flow.

The obstacle avoidance behavior guides the robot reliably towards obstacle-free regions and circumnavigates objects in the vicinity of the robot or regions afflicted with uncertainty concerning visual perception. Figure 3.5 a) shows the omnidirectional camera view of a

corridor. In the case of obstacle avoidance as well as turn around behavior to avoid uncertain and potentially critical spaces initially three perspective views are reconstructed as shown in figure 3.5 b) [119]. The frontal view accomplishes collision avoidance with objects in front of the robot, whereas the peripheral views contribute to the general stabilization of the robot's motion. The opening angle of the omnidirectional camera for perspective construction is approximately 75°. The resolution of the reconstructed frontal view is 200 by 90 pixels. The field of view is partitioned into a grid of 10 by 5 windows corresponding to different viewing directions of the robot (cf. figure 3.5 b). The upper number in the cell view corresponds to the time to contact estimates in seconds and the lower number is the confidence into the time to contact estimate.

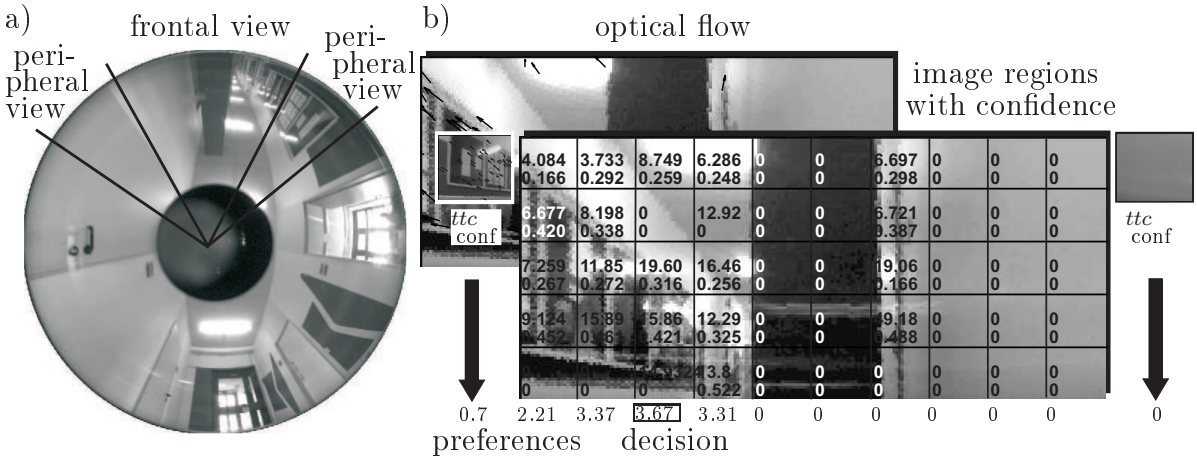


Figure 3.5: a) Omnidirectional image; b) Reconstructed perspective views and orientation preferences based on the time to contact (*ttc*) and confidence.

The stimulus of the obstacle avoidance behavior is the time to contact (*ttc*), which is estimated from the optical flow vectors in the image. From the divergence of the optical flow field the time to contact is derived [31] according to:

$$ttc = \frac{z}{v_{R_z}} = \frac{2}{\nabla(\dot{\hat{u}}, \dot{\hat{v}})} \quad \text{with} \quad \nabla(\dot{\hat{u}}, \dot{\hat{v}}) = \frac{\partial \dot{\hat{u}}}{\partial \hat{u}} + \frac{\partial \dot{\hat{v}}}{\partial \hat{v}}. \tag{3.2}$$

This equation indicates that estimating the time to contact only depends on the optical flow, but requires no knowledge or estimation of the scene depth z and v_{R_z}. [31] uses different symmetrical divergence templates to calculate the divergence of the optical flow for a dense optical flow field. This method fails when the optical flow field is not dense enough to calculate the derivative along the normal direction. Contrary to the analysis above, the proposed approach does not assume that the divergence is determined for the projection center, but that the lateral change of motion is small compared to the depth of the scene, thus allowing for the image-segmented estimation of the divergence and consequently the time to contact as well. A derivation of equation 3.2 for these specifications is found in

appendix A. Hence, an approach for sparse optical flow fields is proposed where all the optical flow pairs found within a confined neighborhood contribute to the calculation of the divergence of the particular flow vector under consideration. The neighborhood around a single flow vector is defined by a window of finite size formally referred to as a pairing window. The pairing window approach involves estimation of divergence of an optical flow vector by calculating the average divergence across all individual flow field vectors within a neighborhood window. The size of the pairing window and the density of the flow field determine the accuracy of the divergence estimate. A pairing window corresponds to one single image region depicted in figure 3.5 b). Equation 3.3 computes the divergence for each pairing window ∇_{pw}, indicated by the subscript pw:

$$\nabla_{\mathrm{pw}}(\dot{u},\dot{v}) = \frac{1}{(n-1)!}\sum_{i=1}^{(n-1)!}\left(\frac{\partial \dot{u}_i}{\partial u_i}+\frac{\partial \dot{v}_i}{\partial v_i}\right) = \frac{1}{n}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left(\frac{\dot{u}_i-\dot{u}_j}{\Delta u_{ij}}+\frac{\dot{v}_i-\dot{v}_j}{\Delta v_{ij}}\right). \tag{3.3}$$

n is defined by the number of optical flow vectors, resulting in $(n-1)!$ different pairings to be considered. For each pair the individual divergence is calculated and aggregated into the divergence of the pairing window ∇_{pw}. The individual divergence is calculated from normal optical flow vectors $\dot{u}_i$, $\dot{v}_i$ and $\dot{u}_j$, $\dot{v}_j$ along their respective directions u and v.

For the peripheral view (figure 3.5) the time to contact is estimated in a different way, as the focus of expansion lies outside the peripheral view. Therefore the alternative approach inspired by [87] determines the time to contact based on the temporal evolution of the direction of the optical flow. The advantages of this approach consist on the one hand in the estimation of the time to contact solely via optical quantities such as the angle and its temporal derivative. On the other hand the work space is complementary to determine the time to contact by means of the divergence of the optical flow. The time to contact is derived by differentiating the geometric relation: $d = z\tan(\phi)$, in which ϕ denotes the orientation of the obstacle in robocentric coordinates, z the distance component to the obstacle in driving direction of the robot and d the distance component perpendicular to the driving direction. The temporal derivative yields $\mathrm{d}/\mathrm{d}t(d) = z\dot{\phi}\cos^{-2}(\phi)+\dot{z}\tan(\phi) = 0$. Thus, the time to contact for obstacles in both peripheral fields of view is given by:

$$ttc = \frac{\cos\phi\,\sin\phi}{\dot{\phi}}. \tag{3.4}$$

[92] employs the same definition to detect obstacles, to obtain their range and to model the environment by means of corners in the perspective camera image.

The information content and the reliability of the pure time to contact is increased by estimating the confidence of the current perception. For each single cell of the grid a degree of confidence is determined by means of the variance of the individual measurements of the optical flow within a window according to:

$$\mathrm{conf}_{\mathrm{seg}(i,j)} = \frac{1}{n}\sum_{nv=1}^{m}\begin{cases} 1-\frac{1}{\sigma}\mathrm{abs}(ttc_{nv}-\mu_{(i,j)}) & \text{, if } (ttc_{nv}-\mu_{(i,j)})<\sigma \\ 0 & \text{otherwise} \end{cases}. \tag{3.5}$$

Here ttc_{nv} corresponds to one of the m total time to contact estimates that are computed from the corresponding flow vectors in the window $\text{conf}_{\text{seg}(i,j)}$ with the row and column position (i, j) of the cell. $\mu_{(i,j)}$ designates the mean of the time to contact values in a segment and σ is the standard deviation of the data set. If there are no time to contact values or optical flow in the cell under consideration, then the time to contact is assigned to zero. Grids with a time to contact value equal to the mean time to contact are also given zero confidence, implying the absolute confidence about the presence of an obstacle. Due to the adaptive fitting of σ homogeneous fields obtain distinctly more confidence than inhomogeneous cells. For a homogeneous field of consistent measurements the confidence tends to one, whereas for the opposite case of large and spurious values, representing noise, it tends to zero. If $ttc_{nv} - \mu_{(i,j)} < \sigma$ is not valid, the confidence of the corresponding window is reduced to zero.

The time to contact and its corresponding confidence values are fused within the grid columns representing the driving directions. From these aggregated information first preferences for obstacle-free directions are created and from these the turn rate and translational velocity of the robot are computed. Furthermore, the final recommendation for the direction Θ_{R_k} is influenced by the angular acceleration compared to the previous rotation $\Delta\Theta_{R_{k-1}}$ in order to guarantee a smooth rotation by averaging:

$$\Theta_{R_k} = \text{argmax}_j \left(ttc_{\text{avg}_j} \text{conf}_{\text{avg}_j} - \Delta\Theta_{R_{k-1}} \right), \tag{3.6}$$

in which $\Delta\Theta_{R_{k-1}}$ denotes the difference between the new direction and the robot's current heading, ttc_{avg_j} and $\text{conf}_{\text{avg}_j}$ are defined as the mean of the time to contact and confidence values, respectively, of the j-th column. Apart from scaling, the time to contact is calculated in the same way for the peripheral views.

The turn around behavior is responsible for the detection of dead end situations, in which the robot cannot circumnavigate the obstacle but has to track back. This behavior initiates a 180° turn in dead ends or so-called tricky corners for which the optical perception indicates no save passage. This situation is recognized by an inconsistent or nonexistent field of flow with low confidence. On the other hand this behavior assumes the robot's control in case of large-area objects without texture, because the optical flow does not provide information due to the lack of contrast. Planes without texture are abstracted as obstacles and stored in a local spatial memory until visual stimuli evaluated with sufficient confidence reemerge for this region.

3.3.3 Experimental results: Navigation with omnivision

The behavior coordination involves activation of the correct behavior at the right instance. Behavior selection strongly depends on the current context of the robot [18]. In this case a subsumption architecture is employed. The main reason is that the visual information is

not as reliable as the distance measurements. In textureless environments it is difficult to assess the distance of obstacles from visual measurements. The subsumption architecture shown in figure 3.6 has a layered structure, in which each layer is associated with a specific behavior. The higher layers have the authority to subsume (indicated by s) and inhibit behaviors in the lower layer.

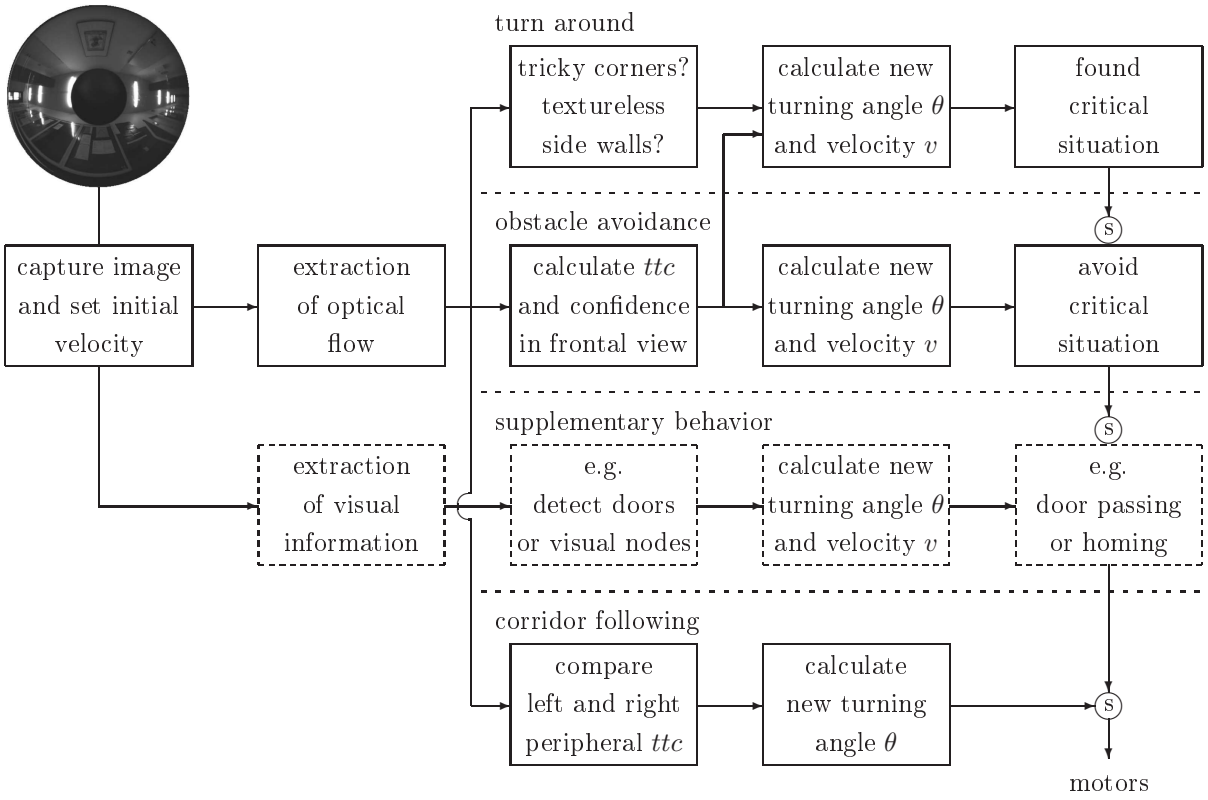


Figure 3.6: Subsumption architecture with corridor following, obstacle avoidance and turn around behavior. Supplementary visual behaviors are indicated by dashed blocks.

The most basic behavior is the corridor centering behavior in the lowest layer. It compares the optical flow magnitude of the left and right hemisphere of the omnidirectional camera. This behavior maintains a constant translational velocity and only changes the turning direction. The intermediate layer consists of the obstacle avoidance behavior which is activated if the time to contact falls short of a threshold. The obstacle avoidance behavior commands a turning direction and a translational velocity. The translational velocity is proportional to the time to contact. Once the obstacle avoidance triggers, it subsumes the lower layer and overrides its output with its translational and rotational velocities v_R and ω_R. The input for the motor commands is replaced by the recommendations of the obstacle avoidance. The turn around behavior has the highest priority. This behavior responds to corners which represent dead ends of the corridor or textureless wall segments and initiates a 180° turn with a subsequent wandering into a new direction. Furthermore, this behavior

includes a remember-sidewall component that remembers side walls or obstacles detected at previous instances and avoids them over the next n control cycles. The turn around behavior is based on the same measurements as the obstacle avoidance layer. If a tricky corner (i.e. a textureless region filling the complete field of view) is detected, it blocks the output of obstacle avoidance from reaching the motors and suppresses the output of corridor following. One major advantage of the subsumption architecture is the ease of integrating supplementary visual behaviors such as door passing and visual homing as shown later on indicated by the additional dashed block in figure 3.6.

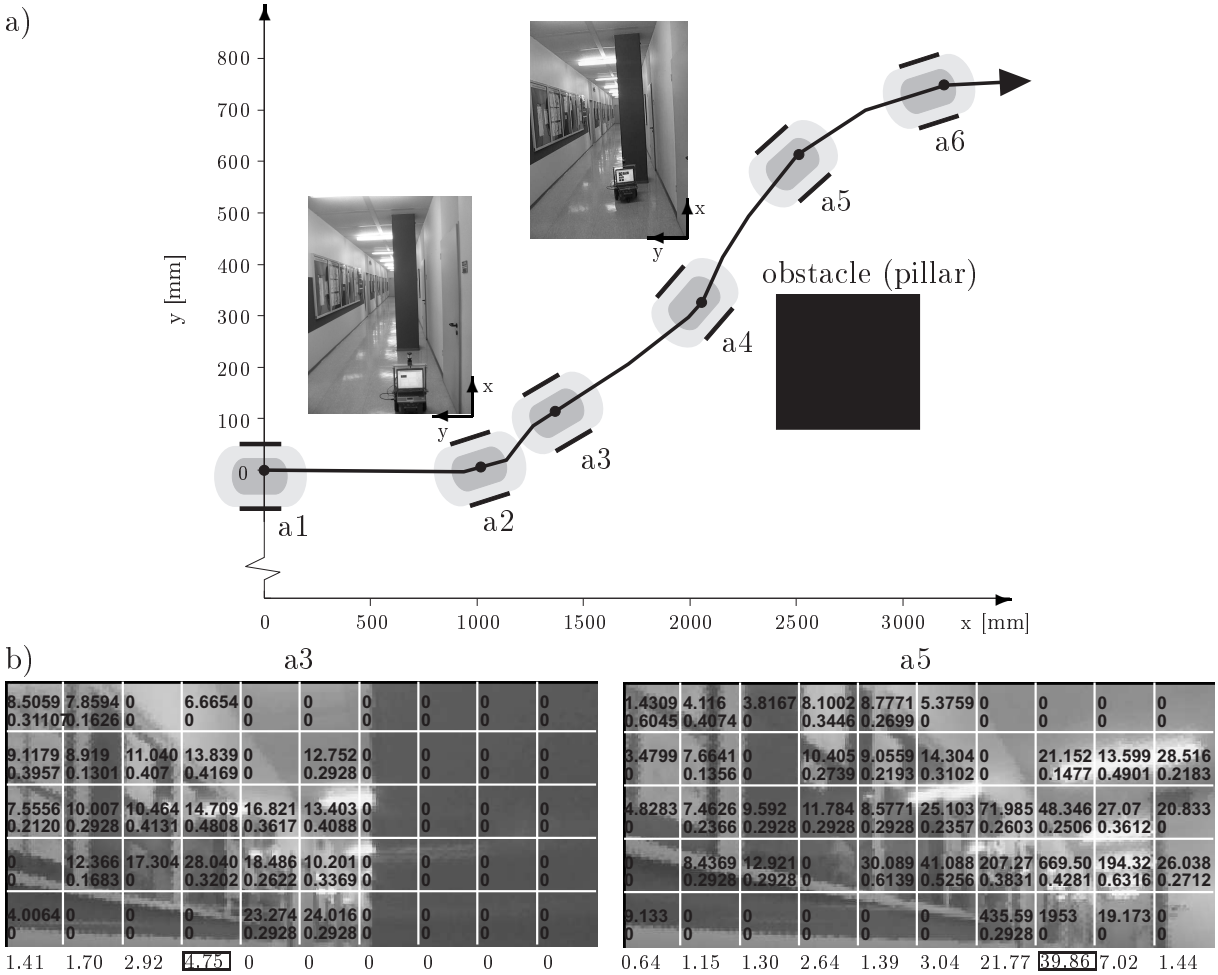


Figure 3.7: a) Obstacle avoidance: trajectory traversed by the robot; b) *ttc* and confidence grids with final decision.

The experimental evaluation of the visual navigation is carried out with the same robot configuration and in same locations as in the experiments described in section 3.1 for vision-guided navigation (cf. also [114]). In the following a prototypical scenario is presented which points out the operation of obstacle avoidance (cf. figure 3.7) by means of difficult situations imposed on the robot. At the beginning of the experiment the robot is located

close to the right wall of the corridor at a distance of 2 m in front of a pillar which is situated in driving direction, thus blocking a part of the corridor, as depicted in figure 3.7 a) which shows the environment and the actual path traveled by the robot (a1-a6). The images on the left side of the robot are the perspective peripheral images from one side of the traveling direction (for sake of clarity only the left side is shown). The robot successfully evades the obstacle without colliding with the corridor wall or the pillar. Figure 3.7 b) additionally shows the frontal visual perception of the robot with the direction recommendation for the two locations a3 and a5. The entire frontal image view is partitioned into grids of size 20 by 20 pixels in which each grid estimates an average time to contact (upper value in figure 3.7 b) and an associated confidence (lower value, according to equation 3.5). For every column the average of the time to contact and confidence is computed. Finally, the gains of the two lateral planes on the left and the right contribute towards the final gain which provides the decision variables for turning. The robot turns in the direction of the region with the highest overall gain value. At the waypoint a3 in figure 3.7 the turn around behavior inhibits the output of the obstacle avoidance behavior, as confidence in the four right columns is low due to the lack of texture of the corridor wall. In case more than three columns on one side exhibit a confidence of zero, the presence of a side wall or a wall without texture is signalled. This procedure completely immunes the optical flow on the right side thereby avoiding the side wall and the pillar with the neighborhood region around it. The recommendations of the two neighboring columns are also set to zero, so that only the remaining directions of the left hemisphere influence the final selection of the heading direction. From the set of candidate headings the fourth column of the left has the highest preference, causing a subsequent evasive maneuver of the robot to the corridor center. At the waypoint a5 all directions possess sufficient confidence in visual information, so that again the obstacle avoidance behavior obtains sole control over the robot. It advocates its maximal recommendation for the third column to the right with the highest time to contact, in robocentric view pointing towards the free corridor for the current alignment of the robot. The corridor centering behavior is sensitive towards inhomogeneous texture distribution in the right and left lateral field of view which may lead to a lateral displacement of the robot compared to the center of the corridor, resulting in an oscillation of the robot motion around the corridor center. The stability of the corridor centering behavior can be improved by appropriate controller design to suppress disturbances caused by varying texture. Furthermore not all possible scenarios for regions with low texture can be controlled robustly by the turn around behavior.

In this chapter visual behaviors via an omnidirectional camera for mobile robot navigation in unstructured environments are introduced. The next chapter describes visual homing in order to complete the transition from vision guided to goal-orientated visual navigation.

Chapter 4

Global visual homing by visual servoing

As visual homing in a topological map is much more complex than homing in a metric map with distance sensors this chapter especially focusses on the design of visual homing to complete the visual navigation. The homing behavior with decoupled navigation and gaze control with a virtual camera plane is first presented in [105]. The topological localization from section 3.1 is extended to automatically select optimal reference images for the visual homing behavior. Thereby the advantages of visual input from a pan-tilt in conjunction with an omnidirectional camera are combined for global visual homing. The time-optimal reference feature and image selection is provided by the omnidirectional camera system [108] using the information from the topological map, whereas local pose convergence is achieved by the pan-tilt camera.

The presented work obeys the paradigm of topological map-based navigation by a directed graph. Visual homing follows approaches from visual servoing presented in section 2.4 with the major difference that global convergence towards the reference image is required even if the features from the reference view are far away from the actual pose of the robot. Especially the decoupling strategy for navigation and gaze control as well as the synergy of omnidirectional and monocular camera are milestones for global visual homing for office environments with minimal texture offering additional freedom for visual homing compared to the schemes cited in section 2.3.

This chapter is organized as follows: The general concept for visual homing is described in section 4.1 motivating the advantages of a decoupled navigation and gaze control by the virtual camera plane. The virtual camera plane is described in section 4.2 and the difference between a vertical and horizontal virtual camera plane is pointed out. In order to utilize the virtual camera plane, the required gaze control is summarized in section 4.3. Different approaches for visual navigation control as well as their experimental evaluation are described in section 4.4. The visual navigation behavior emerging from the individual visual behaviors is presented in section 4.5, which concludes with an comparison of vision-

guided and vision-based navigation.

4.1 General concept

Figure 4.1 depicts the scheme of visual homing behavior as well as the integration in the hybrid control architecture (cf. figure 1.2). The visual homing for the subordinate reactive layer is later on integrated in the subsumption architecture from figure 3.6 for fully visual navigation of mobile robots. The major components of the visual homing behavior by large

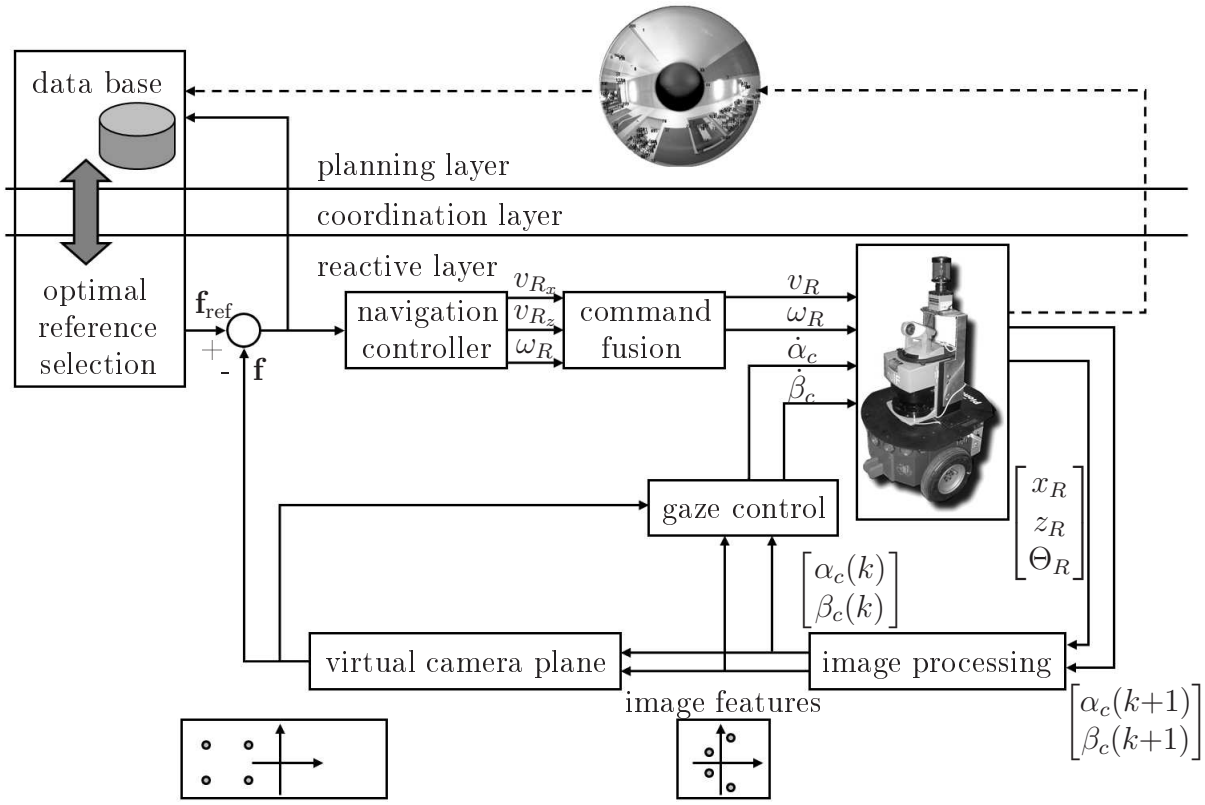


Figure 4.1: Large view visual servoing integrated in the hybrid control architecture.

view visual servoing are: gaze control, virtual camera plane, visual navigation control as well as the command fusion. The gaze control guarantees that the image features remain in the camera's field of view. The virtual camera plane represents the mutual control space thereby decoupling gaze from navigation control. As gaze is controlled independently of the robot motion and the features are defined in a virtual camera plane, the visual controller uses the same landmarks over a larger range of motion. Therefore, fewer visual landmarks are required to describe a smooth path through the environment. The optimal exploitation of landmarks even enables visual navigation in environments with little texture e.g. office environments and consequently only few natural landmarks.

Figure 4.2 demonstrates the advantages of a swiveling versus a fixed camera for visual navigation. In figure 4.2 a) the standard scenario from literature is depicted, where the camera is fixed relative to the robot motion, restricting the field of view which results in a limited convergence area of a landmark. In this case this limitation is resolved by the independent gaze and navigation control resulting in a larger convergence area of the same landmark as depicted in figure 4.2 b) and c). An additional extension of this approach compared to standard approaches enables the robot to navigate towards (figure 4.2 b)) as well as parallel to a landmark (figure 4.2 c)) which is particularly useful for traversing confined indoor spaces such as corridors. In order to control the robot motion independently of the gaze the observed features are transformed from image into virtual camera plane.

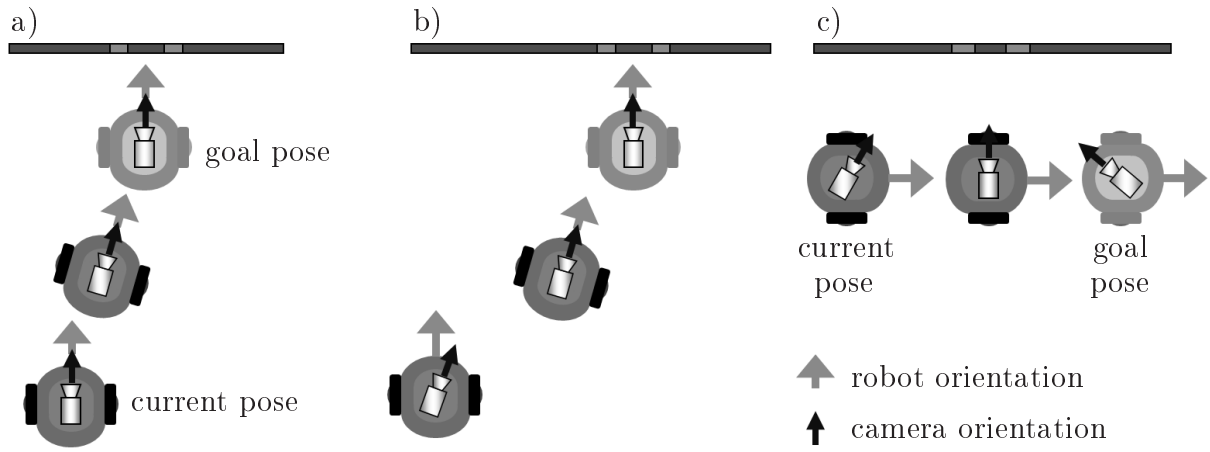


Figure 4.2: Visual servoing for navigation of mobile robots: a) Fixed camera; b) Decoupled gaze and navigation control for navigation towards a landmark using a swiveling camera; c) Decoupled control for navigation parallel to a landmark.

Three different approaches for navigation control on the virtual camera plane are investigated: (a) image Jacobian, (b) image moments and (c) a combination of partial scene reconstruction in conjunction with image moments. These approaches differ by the strategy to decouple rotational and translational velocity components. As the first approach (a) was originally intended as case study for visual homing, artificial landmarks in the height of the robot's camera necessitate a vertical virtual camera plane. The second (b) and third approach (c) rely on natural landmarks, thus using a horizontal virtual camera plane as described in the following. The optimal reference images are tracked and selected by the omnidirectional camera as depicted in figure 1.2 using the features for topological localization from section 3.1. Optimality of a reference image is defined by the number of visual features as well as in terms of their traceability in the required workspace. Visual homing with omnidirectional cameras using rectification of omnivision, SIFT and a scale based image Jacobian is presented for room nodes [88]. Nonetheless our experiments indicate that an omnidirectional camera alone is not suited for visual homing especially in narrow corridor environments, as feature recognition over a large area is attenuated by image distortion, thus the pan-tilt camera is additionally required for global visual homing.

4.2 Virtual camera plane

As the camera has two degrees of freedom (pan α_c, tilt β_c) the reference image is not uniquely related to the robot's pose but depends on the camera orientation as well. There are two ways to properly capture and define reference features for vision-based navigation with a rotating camera. The straightforward solution is to capture an entire set of reference images at different camera orientations α_c, β_c for a particular pose. In addition to the large memory requirements this approach is not feasible as for most camera orientations the actual reference features are not visible in the captured image but rather need to be computed based on a geometric reconstruction of the scene.

A better solution is to define a transformation from the camera image to a virtual camera plane that is independent of the camera's actual pan and tilt. This methodology has the advantage that only a single reference image at a single camera orientation needs to be captured. The features are projected onto a virtual camera plane, thus allowing for correspondence between image features in the current pose of the robot and the reference pose independently of the camera's viewing direction. In principle it is possible to transform features either onto a vertical or a horizontal virtual camera plane. [105] presents a planar robot motion decoupled from gaze control via transformation of the image points onto a vertical virtual camera plane with image Jacobian (a), while in [108] a horizontal virtual camera plane is mostly preferred for approach (b) and (c). The transformation to the virtual plane assumes a calibrated camera system and undistorted image features, thereby large radial distortions of the image are the major source of discrepancies between the theoretical feature coordinates in the virtual camera plane and the calculated ones. In order to transform features from the real image to the virtual camera plane, the following assumptions are made:

- Assumption 1: The rotation axes of the camera for the pan and tilt angle coincide with the focal point of the camera.
- Assumption 2: The rotation axis of the robot along the vertical axis coincides with the virtual camera axis normal to the horizontal virtual camera plane or in the case of the vertical camera plane with its coplanar vertical axis.

Assumption 1 might be considered too restrictive as the precise alignment of the rotational axis of the camera with its focal point is difficult to achieve. Nonetheless this assumption allows for rendering the transformation of the current camera plane to the virtual camera plane (equation 4.3) solely as a function of the known focal length λ, pan α_c and tilt β_c.

In the **vertical virtual camera plane**, the optical flow $\dot{u}, \dot{v}$ in the image is caused by the robot's translatory motion v_{R_x} and v_{R_z} and its rotation ω_R. The image Jacobian for point features in equation 4.1 not only depends on the pixel coordinates of the features u_i and v_i, but also on their unknown depth z_i and the camera's focal length λ. Therefore the

controlled variables are highly coupled and non-linear in z_i, which additionally varies with each individual feature and the robot's pose relative to the feature 3D coordinates:

$$\begin{aligned} \dot{u_i} &= \frac{\lambda}{z_i} v_{R_x} + \frac{-u_i}{z_i} v_{R_z} + \frac{-\lambda^2 - {v_i}^2}{\lambda} \omega_R, \\ \dot{v_i} &= 0 \quad + \frac{-v_i}{z_i} v_{R_z} + \frac{-u_i v_i}{\lambda} \omega_R. \end{aligned} \tag{4.1}$$

The coupling is reduced by the definition of a **horizontal virtual camera plane** rather than a vertical. For indoor applications the mobile robot is restricted to planar movements. For vision-based navigation with the vertical camera plane, the orientation of the reference plane depends on the type of waypoint and the location of the features relative to the designated robot path. The reference feature plane is either perpendicular to the robot's heading or parallel to it, depending on whether the robot is supposed to approach or to pass by the waypoint. The horizontal camera plane and the plane in which the robot moves are coplanar, and the orientation of the horizontal plane is solely defined by the reference pose independent of the robot's designated path. The optical flow in the horizontal camera plane is related to the robot movement according to:

$$\begin{aligned} \dot{u_i} &= \frac{\lambda}{z_i} v_{R_z} - v_i \omega_R = k_i v_{R_z} - v_i \omega_R, \\ \dot{v_i} &= \frac{\lambda}{z_i} v_{R_x} + u_i \omega_R = k_i v_{R_x} + u_i \omega_R. \end{aligned} \tag{4.2}$$

This detrimental coupling of feature motions in the vertical plane is avoided in the horizontal virtual camera plane if one considers the following observations and assumptions:

- The distance of the feature points is invariant to the robot motion v_{R_x}, v_{R_z} and thereby constant.
- The depth of the scene is small compared to the distance of the focal point to the features, yielding a weak perspective camera model.

The depth z_i of features no longer depends on the planar robot motion and is replaced by a constant k_i.

The transformation from the actual to the virtual camera plane is stated as:

$$[u_V, v_V, 0, 1]^T = \mathbf{T}_{C_R}^{C_V}(\lambda, \alpha_c, \beta_c) \, [u, v, 0, 1]^T , \tag{4.3}$$

where u and v denote the pixel coordinates in the current camera plane and u_V and v_V those in the virtual camera plane. In case the first assumption is violated, the transformation $\mathbf{T}_{C_R}^{C_V}$ also depends on the extrinsic camera parameters and the coordinates of the feature point in the world frame, thus making the transformation infeasible. The features from the reference as well as from the current view are transformed to the virtual camera plane according to equation 4.3 in order to calculate the control error in the image space.

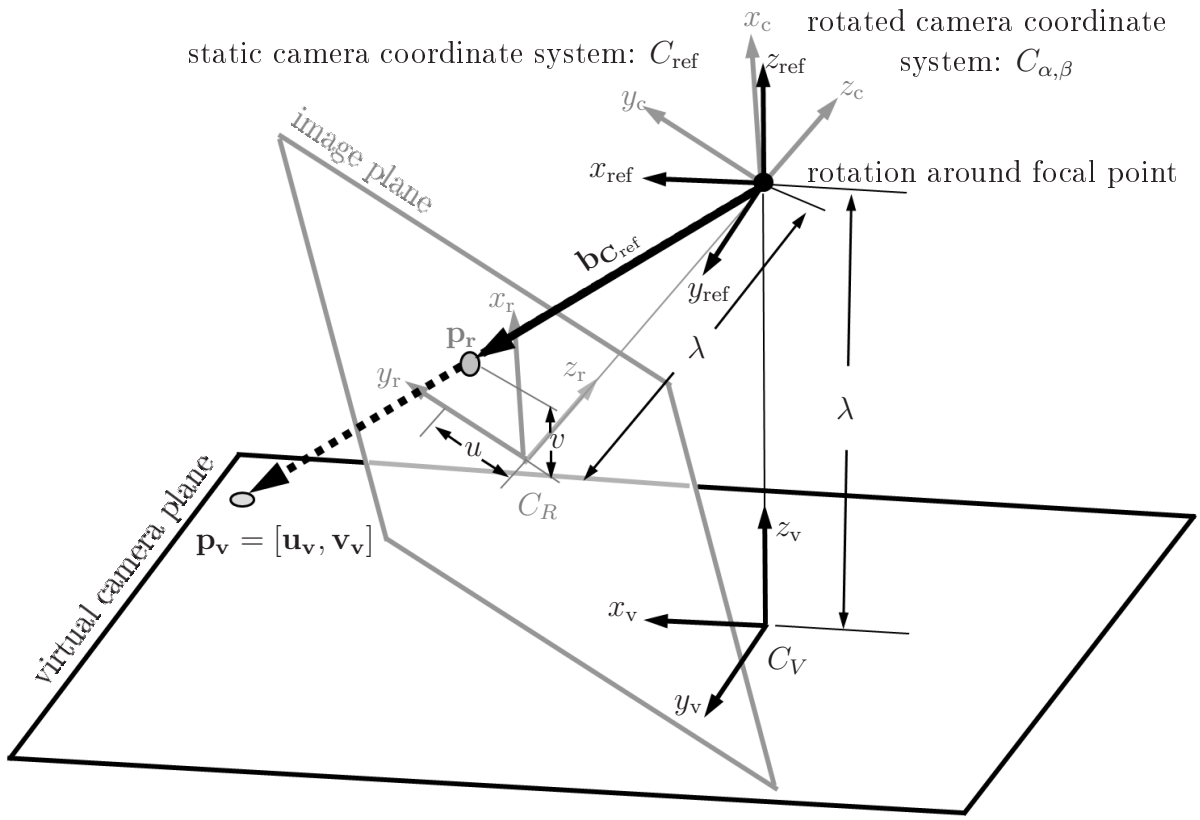


Figure 4.3: Transformation of the real camera images onto the virtual camera plane.

The schematics of the transformation from the image plane to the virtual camera plane (in this case horizontal) is depicted in figure 4.3. The detailed transformation $\mathbf{T}_{C_R}^{C_V}$ from the camera plane to the horizontal virtual camera plane is as follows:

$$[u_V, v_V, 0, 1]^T = \mathbf{T}_{C_{\text{ref}}}^{C_V} \frac{\lambda}{-\mathbf{b}_{C_{\text{ref}}} \mathbf{z}_V} \mathbf{T}_{C_{\alpha,\beta}}^{C_{\text{ref}}} \mathbf{T}_{C_R}^{C_{\alpha,\beta}} [u, v, 0, 1]^T , \tag{4.4}$$

with

$$\mathbf{b}_{C_{\text{ref}}} = \mathbf{T}_{C_{\alpha,\beta}}^{C_{\text{ref}}} \mathbf{T}_{C_R}^{C_{\alpha,\beta}} [u, v, 0, 1]^T \tag{4.5}$$

and

$$\mathbf{T}_{C_R}^{C_{\alpha,\beta}} = \mathbf{T}_{C_{\text{ref}}}^{C_V} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \lambda \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{C_{\alpha,\beta}}^{C_{\text{ref}}} = \begin{bmatrix} \cos\beta_c & 0 & \sin\beta_c & 0 \\ \sin\alpha_c \sin\beta_c & \cos\alpha_c & -\sin\alpha_c \cos\beta_c & 0 \\ -\cos\alpha_c \sin\beta_c & \sin\alpha_c & \cos\alpha_c \cos\beta_c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.6}$$

The transformations $\mathbf{T}_{C_R}^{C_{\alpha,\beta}}$ from the camera coordinate system C_R to the coordinate system $C_{\alpha,\beta}$ as well as the transformation $\mathbf{T}_{C_{\text{ref}}}^{C_V}$ from the fixed reference frame C_{ref} centered at the focal point to the virtual camera plane C_V only depend on the focal length λ. The complete

transformation $\mathbf{T}_{C_R}^{C_V}$ is therefore constructed from equations 4.4, 4.5 and 4.6 where $\mathbf{b}_{C_{\text{ref}}}$ represents the image features in the reference frame. Thus, a point $\mathbf{p_i}$ in the image plane is transformed onto a point $\mathbf{p_v}$ in the virtual camera plane as depicted in figure 4.3 via the following steps: First $\mathbf{p_i}$ is transformed from the camera coordinate system C_R to a rotated camera coordinate system $C_{\alpha,\beta}$ via $\mathbf{T}_{C_R}^{C_{\alpha,\beta}}$ solely by translation with λ, then this rotated camera coordinate system $C_{\alpha,\beta}$ is rotated around the focal point with $\mathbf{T}_{C_{\alpha,\beta}}^{C_{\text{ref}}}$ into the static camera coordinate system C_{ref}. The intersection point of $\mathbf{b}_{C_{\text{ref}}}$ with the virtual camera plane is determined by means of the theorem of intersecting lines: The ratio between $\lambda|\mathbf{b}_{C_{\text{ref}}}|/(-\mathbf{b}_{C_{\text{ref}}}\mathbf{z}_V)$ and λ is equal to the ratio of $|\mathbf{b}_{C_{\text{ref}}}|$ and $\mathbf{b}_{C_{\text{ref}}}(-\mathbf{z}_V)$ (i.e. the component of $\mathbf{b}_{C_{\text{ref}}}$ in direction of $-\mathbf{z}_V$), resulting in the factor $\lambda/(-\mathbf{b}_{C_{\text{ref}}}\mathbf{z}_V)$ given in equation 4.5. As the intersection point is expressed in the static camera coordinate system C_{ref}, in a final step it is transformed to the virtual camera coordinate system C_V via $\mathbf{T}_{C_{\text{ref}}}^{C_V}$.

Whether the features are transformed to the vertical or horizontal virtual camera plane highly depends on the feature 3D coordinates. In case the feature templates are at the same height as the robot's camera only a transformation to the vertical virtual camera plane makes sense e.g. approach (a) as the pixel coordinates become infinite for a transformation to the horizontal plane. The 3D placement of the artificial landmarks above the robot enables the transformation to the horizontal virtual camera plane, but poses higher computational burden on the landmark detection due to large affine distortions of the landmarks in the camera view. Because of the transition from artificial to natural landmarks the affine distortions are handled by the feature extraction (e.g. SIFT) and pose thereby no restrictions to the orientation of the virtual camera plane. In man-made environments the texture is normally on vertical planes, especially in the height of the view of humans. Thus a transformation to the horizontal is preferred due to the 3D locations of the texture and the previously explained advantages of the horizontal virtual camera plane. Finally it is stated that for a limit above 45° of the pan angle the horizontal virtual camera plane is numerically preferred whereas below this threshold a vertical camera plane is desirable. The transformation demands a camera gaze control in order to center the image features.

4.3 Camera gaze control

The objective of the camera gaze control is to regulate the camera orientation independent of the robot's motion in order to track a landmark or features and to center them in the image. Standard camera systems have a limited field of view of about 60° in the horizontal plane and about 40° in the vertical plane. This field of view is extended by a pan-tilt unit in conjunction with a gaze controller for tracking the feature points. The gaze control is the connecting step between the proposed virtual camera plane and the navigation control in the virtual camera plane as described in the following. Different approaches are known for gaze control. As the gaze control is only a tool for the navigation with the virtual camera plane, a description of gaze control using the virtual camera plane and homography is

depicted in figure 4.4 and used for the approach (b) with image moments and (c) with a combination of partial scene reconstruction in conjunction with image moments.

Gaze control is composed of a feed-forward path that predicts the feature motion in the virtual camera plane based on the known robot motion command. The feedback control based on a virtual centroid compensates disturbances. The feedback control projects the feature centroid of reference features into the current view according to the homography $\mathbf{H}$, referred to as the virtual centroid. The transformation of the centroid of the features from the goal view $\hat{u}_{\text{cog}}, \hat{v}_{\text{cog}}$ into the current view $u_{\text{cog}}, v_{\text{cog}}$ is achieved via:

$$[u_{\text{cog}}, v_{\text{cog}}, 1]^T = \mathbf{T}_{C_R}^{C_V}(\lambda, \alpha_c, \beta_c)\mathbf{H}\mathbf{T}_{C_V}^{C_R}(\hat{\lambda}, \hat{\alpha_c}, \hat{\beta_c})\,[\hat{u}_{\text{cog}}, \hat{v}_{\text{cog}}, 1]^T\,. \tag{4.7}$$

The angular errors of the required camera rotations $\Delta\alpha_c$ and $\Delta\beta_c$ of the PTZ camera are given by:

$$\Delta\alpha_c = -\frac{\Delta v_{\text{cog}}}{\lambda}, \quad \Delta\beta_c = -\frac{\Delta u_{\text{cog}}}{\lambda}. \tag{4.8}$$

The structure for the gaze control for the approach (a) with image Jacobian is similar to the

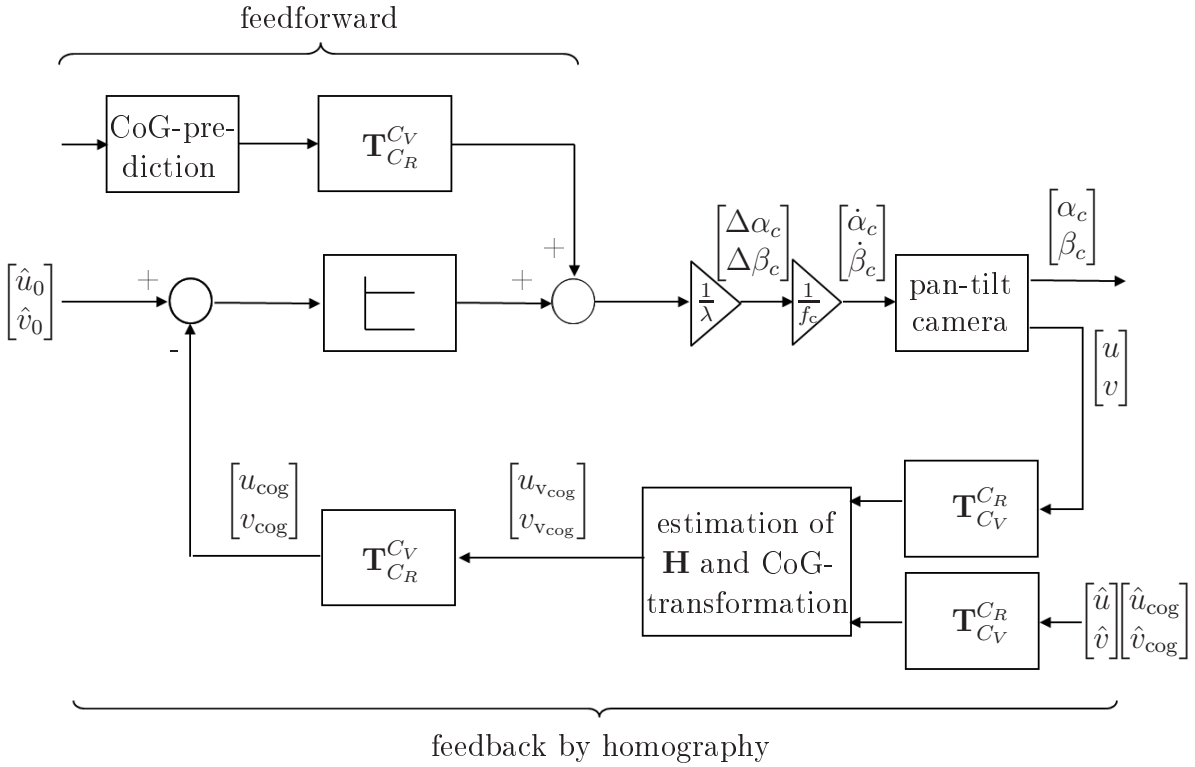


Figure 4.4: Gaze control.

gaze control in figure 4.4 consisting of a feedforward and a feedback part. Nonetheless the utilization of artificial landmarks simplifies the design of the gaze control as the templates in the 3D task space are located at the same height as the camera and in addition the unknown

depth of the features is estimated using projective geometry based on the knowledge of the intrinsic camera parameters and the known dimensions of the rectangular arrangement of the templates. The optical motion of the pixel coordinates in the image plane is predicted based on the known motion action of the robot using image Jacobian and the appropriate counter rotations of the camera $\dot{\alpha}_c$, $\dot{\beta}_c$ that cancel the optical flow by robot motion is calculated. Further details for gaze control with image Jacobian are described in [105].

4.4 Visual navigation control

4.4.1 Control by image Jacobian

The control objective is to regulate the robot's turn rate based on visual feedback in such way that the robot maintains the same orientation and distance to the feature plane as in the demonstrated reference pose. The task space error is constituted by the lateral Δx, longitudinal Δz and orientational error $\Delta\Theta_R$. The visual servo controller design relates the task space errors $\Delta z, \Delta x, \Delta\Theta_R$ with the feature errors in the image. γ_V defines the angle between the virtual camera plane and the orientation of the robot Θ_R. The approach for decoupling rotational ω_R and lateral motion v_R meaningfully transfers the concept from [33] by defining an angular criterion in the virtual plane. Considering the desired task space motion a trapezoidal distortion is specified with the reference angle φ_{ref} between the horizontal line and the straight line defined by the two upper features of the artificial template. The feature error between the reference φ_{ref} and current distortion φ is denoted as $\Delta\varphi$. For small errors $\Delta\varphi$ no rotation of the robot is needed and the longitudinal motion alone reduces the image error as the robot moves towards the reference pose. For the corridor landmarks the virtual plane is oriented parallel to the template plane. As long as the robot is oriented parallel to the feature plane the error $\Delta\varphi$ disappears independent of the lateral error and does therefore not contribute to the robot control ω_R. The independence of the angular feature error from the lateral displacement is only fulfilled if the virtual and the template plane are collinear ($\gamma_t = 90°$). For angles that differ from $\gamma_t = 90°$ the angular error $\Delta\varphi$ varies with the lateral error as well, which substantially complicates the design of a visual servoing controller. This property suggests to design the controller for a virtual image plane at $\gamma_t = 90°$. From geometric considerations it is intuitive that for the reference pose the virtual plane and template plane should be parallel, as the trapezoidal distortion only results from the rotation but not from the translation. The visual control scheme takes advantage of the decoupling between rotation and translation. The angular criterion expressed by the distortion error $\Delta\varphi$ determines the rotational component of control. A rotational component directly computed from the feature position error causes the robot to head directly towards the feature plane. The translational component is calculated based on the residual positional feature error, corrected by feature motion caused by the anticipated rotation.

Table 4.1: Control scheme for visual navigation on virtual camera plane by Jacobian.

1. *Decoupling of the rotational component of the image Jacobian J_ω from the translational $\mathbf{J}_{xz}$-component.*
2. *Computation of the distortion error $\Delta\varphi$ based on the difference $\varphi_{\text{ref}} - \varphi$ in the virtual image plane.*
3. *Computation of the gain:*
$$k = \frac{1}{1 + (\frac{\varphi_0}{|\Delta\varphi| + \zeta})^2}. \tag{4.9}$$
4. *Computation of the rotational control:*
$$\omega_\varphi = -k\Delta\varphi. \tag{4.10}$$
5. *Prediction of the motion of image features Δf_ω:*
$$\Delta f_\omega = J_\omega \omega_\varphi. \tag{4.11}$$
6. *Translational velocity control:*
$$\begin{pmatrix} \dot{x}_C \\ \dot{z}_C \end{pmatrix} = \mathbf{J}_{xz}^{+}(\mathbf{f}_{\text{ref}} - \mathbf{f} - \Delta\mathbf{f}_\omega). \tag{4.12}$$
Transformation of the control commands from the virtual camera plane into the robot's local reference frame:
$$\begin{aligned} v_{R_x} &= \cos(\gamma_V)\dot{x}_C + \sin(\gamma_V)\dot{z}_C, \\ v_{R_z} &= \cos(\gamma_V)\dot{z}_C - \sin(\gamma_V)\dot{x}_C. \end{aligned} \tag{4.13}$$
7. *Motor command fusion using the constants η_1 and η_2, the gains k_1, $k_2 = \frac{1}{1+\frac{|v_{R_x}|}{\eta_2}}$ and $k_3 = \frac{\eta_1}{1+\frac{|v_{R_z}|}{\eta_2}}$:*
$$v_R = k_1 v_{R_z}, \quad \omega_R = k_2\omega_\varphi + k_3 v_{R_x}. \tag{4.14}$$
8. *Control saturation:*
$$|v_R| = \min(|v_{\max}|, |v_R|), \quad |\omega_R| = \min(|\omega_{\max}|, |\omega_R|). \tag{4.15}$$

The overall control scheme for the visual navigation is detailed in table 4.1. The control takes place in the vertical virtual camera plane, therefore the feature error $\mathbf{f}_{\text{ref}} - \mathbf{f}$ between the desired $\mathbf{f}_{\text{ref}}$ and the current image features $\mathbf{f}$ is calculated accordingly in the virtual camera plane. In the first step the image Jacobian $\mathbf{J}$ is decomposed into its rotational and translational component J_ω and $\mathbf{J}_{xz}$, in order to decouple the corresponding controls for the robot motion. The distortion error $\Delta\varphi$ reflects the robot's heading error and modulates the magnitude and sign of the rotational motion. The gain for the rotational component k

in equation 4.9 is reduced for small distortion errors $\Delta\varphi$ in the virtual camera plane with the tuning parameter $\varphi_0 = 0.01$. The gain k for the orientation correction varies smoothly from 0 to 1 with increasing distortion error $\Delta\varphi$. The term ζ avoids numerical instability caused by the division by zero. In addition the sign of the distortion error determines whether the robot turns towards or away from the feature. The commanded rotational velocity ω_φ is proportional to the gain and the angular error $\Delta\varphi$, thus stabilizing the trapezoidal distortion in the image. The translational component regulates the residual positional feature error not yet compensated by the rotation. Therefore, the feature error is corrected by the predicted motion of the image features $\Delta\mathbf{f}_\omega$ due to the rotational velocity command. Prior to the calculation of the translational control this prediction $\Delta\mathbf{f}_\omega$ is subtracted from the observed image error $\mathbf{f}$ according to equation 4.12 to obtain the residual error. Based on the image Jacobian $\mathbf{J}_{xz}$ the translational velocities for lateral and longitudinal motion are calculated using its pseudoinverse. This calculation includes the transformation from the virtual camera frame back to the robot's local reference frame. In case of an omnidirectional drive robot with three local degrees of freedom the control v_{R_x}, v_{R_z} and ω_R is directly converted into appropriate motor controls. However, as the robot pioneer 3DX is a non-holonomic robot with only two degrees of mobility the lateral component v_{R_x} and rotational component ω_R are fused into a single turn rate ω_R. The approach for merging the motion commands proposed in [57] is adapted here. In order to determine the amount of v_{R_x}, v_{R_z} and ω_R to the final motor commands the design parameters k_1, η_1 and η_2 are determined empirically. Finally the motor commands v_R and ω_R are restricted to the velocity limits $v_{R_{\max}}$ and $\omega_{R_{\max}}$, which depend on the frame rate of the visual servoing loop. A control saturation is particularly required for the longitudinal component as initial feature errors are large. Due to the control saturation the robot moves at constant translational velocity for large longitudinal errors and finally slows down as it approaches the reference pose. Once the residual feature error falls below a threshold, the supervisory controller switches to the reference image of the next landmark. The control scheme distinguishes between landmarks such as the docking station which the robot approaches head on and landmarks (2-5 in figure 4.7) which the robot passes parallel to the feature plane. Based on the choice of γ_V the robot navigates towards or parallel to the template plane. In floor sections γ_V is set either to 90° or to -90° depending on the navigation direction. In order to reach the charging station the robot moves toward the template plane in the direction of the virtual camera axis (γ_V= 0°).

4.4.2 Control with image moments and primitive visual behaviors

The image coordinates of the current view $[u, v, 1]^T$ are transformed onto the virtual image plane $[u_V, v_V, 1]^T$ according to equation 4.3. A transformation onto the horizontal camera plane is pursued in the following as it establishes a one-to-one correspondence between the planar robot motion and feature velocities, which facilitates the visual controller design. The transformed feature coordinates $[u_V, v_V, 1]^T$ are control variables of the robot motion

Table 4.2: Control scheme for visual navigation on virtual camera plane by image moments.

1. *Estimation of homography* $\hat{\mathbf{H}}$ *on the horizontal virtual camera plane and decomposition of* $\hat{\mathbf{H}}$ *(cf. equation 2.7) into rotation* $\mathbf{R}$ *and robot rotation* $\Delta\Theta_R$.
2. *Alignment of the image features on the the horizontal virtual camera plane* $[u_V, v_V, 1]^T$ *with the image features in the reference view according to:*

$$\begin{bmatrix} u_V \\ v_V \end{bmatrix} = \begin{bmatrix} \cos(\Delta\Theta_R) & -\sin(\Delta\Theta_R) \\ \sin(\Delta\Theta_R) & \cos(\Delta\Theta_R) \end{bmatrix} \begin{bmatrix} u_V \\ v_V \end{bmatrix}. \tag{4.16}$$

3. *Calculation of image moments for*

$$f_x = \frac{\sum_{i=1}^{n} u_V(i)}{n}, \quad f_z = \frac{\sum_{i=1}^{n} v_V(i)}{n}. \tag{4.17}$$

4. *Definition of image error for*

$$\Delta f_\Theta = \Delta\Theta_R, \quad \Delta f_x = f_{\mathrm{ref}_x} - f_x, \quad \Delta f_z = f_{\mathrm{ref}_z} - f_z. \tag{4.18}$$

5. *Calculation of behavior output:*

   B1 *Behavior for longitudinal alignment*

$$v_{R_{\mathrm{Left}_{B_1}}} = v_{R_{\mathrm{Right}_{B_1}}} = \Delta f_z \left| \frac{\Delta f_z}{\Delta f_x \Delta f_\Theta} \right|. \tag{4.19}$$

   B2 *Behavior for orientational alignment*

$$v_{R_{\mathrm{Left}_{B_2}}} = -\Delta f_\Theta \left| \frac{\Delta f_\Theta}{\Delta f_x \Delta f_x} \right|, \quad v_{R_{\mathrm{Right}_{B_2}}} = -v_{R_{\mathrm{Left}_{B_2}}}. \tag{4.20}$$

   B3 *Behavior for lateral alignment*

$$v_{R_{\mathrm{Left}_{B_3}}} = v_{R_{\mathrm{Right}_{B_3}}} = -\frac{\Delta f_x \Delta f_\Theta}{|\Delta f_z \Delta f_z|}. \tag{4.21}$$

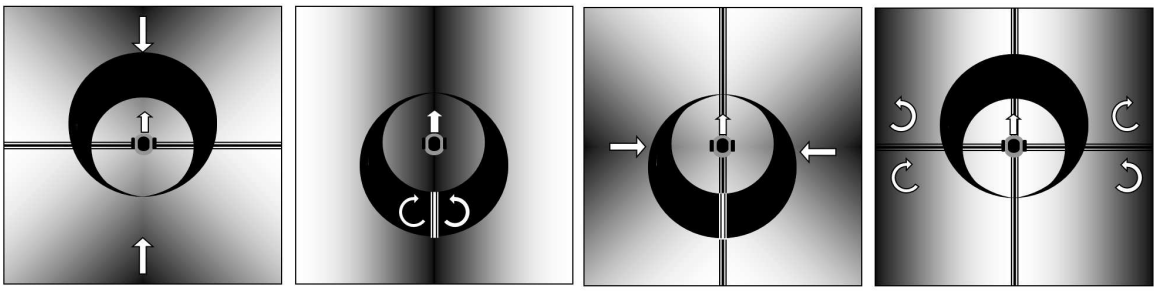
   B4 *Behavior for lateral alignment*

$$v_{R_{\mathrm{Left}_{B_4}}} = -\mathrm{sign}(f_z) \frac{\Delta f_x}{\Delta f_\Theta} \left| \frac{\Delta f_x}{\Delta f_\Theta} \right|, \quad v_{R_{\mathrm{Right}_{B_4}}} = -v_{R_{\mathrm{Left}_{B_4}}}. \tag{4.22}$$

6. *Motor command fusion* $v_{R_{\mathrm{Left,Right}}} = \sum_{i=1}^{4} v_{R_{\mathrm{Left,Right}_{B_i}}}$.

controller as well as the gaze controller. The overall control scheme for visual navigation on a virtual camera plane by image moments is detailed in table 4.2. The rotation $\Delta\Theta_R$ of the

robot around its vertical axis between the reference view and the current view is estimated by the decomposition of the homography $\hat{\mathbf{H}}$. Furthermore, this decomposition also yields the direction vector between the views. However, this information is not sufficient for image-based control of the robot motion, as the direction vector is only defined up to a scale. First the current virtual camera plane is back rotated by $\Delta\Theta_R$. The centroid is calculated by means of the rotation corrected pixel coordinates, which are linearly related to the longitudinal and lateral displacement of the robot relative to the goal pose. The image moments f_x and f_z are described by the centroid components. The image errors $\Delta f_\Theta, \Delta f_x, \Delta f_z$ are defined by equation 4.18. The decoupled image moments calculated from corresponding features in the current and reference view and the corresponding image errors control the motion of the robot in three degrees of freedom. The transformation of the free motion onto the two local degrees of motion of the robot is realized by fusion of the motor commands issued by four concurrent behaviors. The behavior representation as well as the navigation behavior resulting from the fusion of the individual behaviors is designed to first eliminate the error of the lateral position. The commanded wheel velocities $v_{R_{\text{Left}}}, v_{R_{\text{Right}}}$ are computed by the aggregated recommendations of the behaviors. Figure



behavior B_1 behavior B_2 behavior B_3 behavior B_4

Figure 4.5: Situational behaviors for visual servoing. Dark areas and thick lines correspond to robot configurations with dominant behavior activation. White arrows indicate the corresponding action proposed by the behaviors in the respective configuration. The intensity of the grey values indicate the level of activity according to longitudinal and lateral offset.

4.5 illustrates the operation mode of the four situational behaviors for visual servoing of a non-holonomic robot. Behavior B_1 compensates the longitudinal error assuming that the lateral and angular errors have been compensated beforehand. The corresponding visual feature Δf_z for the longitudinal motion is scaled by the inverse gain $\Delta f_x \Delta f_\Theta$. For small residual errors in Δf_x and Δf_Θ the robot approaches the goal position straight on as indicated in the left image for behavior B_1 of figure 4.5. In principle the robot could also drive backwards to the goal position. The second behavior B_2 regulates the orientation and thereby the wheel velocities of the robot as a function of the rotational and lateral error. It corrects the orientation of the robot in situations in which the lateral error is already compensated. Behaviors B_3 and B_4 are for compensating the lateral error. The third behavior triggers in case of a lateral displacement and remains dominant as long as the longitudinal error remains small. Behavior B_4 turns the robot in order to compensate

the lateral error. This is essential as the robot cannot drive towards the goal position if its current orientation coincides with the goal direction in case of a lateral displacement. The responses of the four behaviors are aggregated into a total response in order to obtain the wheel velocities. If the commanded wheel velocities are outside the admissible range, the velocities are reduced in a proportional manner.

4.4.3 Control with homography

Another promising approach for the navigation control instead of using primitive visual behaviors based on image moments is the estimation and decomposition of the homography (cf. chapter 2.1) between the features in the virtual horizontal camera plane, i.e. in the reference view as well as in the current view. The reduced degrees of freedom of the mobile robot simplify the decomposition of the homography considerably as the estimated homography obeys additional constraints. The overall control scheme for visual navigation on a virtual camera plane by homography is detailed in table 4.3. The linear control law is expressed as a function of the polar coordinates ρ and α as well as the orientation error $\Delta\Theta_R$ of the robot between the current and goal position which is directly extracted from the partial pose estimation of the homography. As the homography decomposition yields a scaled translation vector which becomes unreliable for $\rho = 0$, ρ is scaled by a deviation from the pixel coordinates in the actual and reference view. t_x and t_z are the elements of the translation vector $\mathbf{t} = [t_x, t_z]^T$ from the decomposition of the homography. $\hat{u}_{\text{vcog}}$ and u_{vcog} denote the centroid of the u-coordinates of the reference and current view expressed in the horizontal virtual camera plane after the feature rotation about $\Delta\Theta_R$. Similarly, the centroid of the v-coordinates is denoted by $\hat{v}_{\text{vcog}}$ and v_{vcog}. The motor command fusion from equation 4.14 is adapted to the information extracted from the homography in the horizontal virtual camera plane. The stability for the control is guaranteed by a proper choice of the proportional control parameters according to [6].

4.4.4 Experimental results

(a) Large view visual servoing with a pan camera: In the experiments the non-holonomic robot navigates between rooms in order to reach a charging station as it tracks a sequence of properly located visual landmarks. To achieve this task, the robot has to exit the initial room, travel along the corridor and finally enter the room with the docking station as shown in figures 4.6 and 4.7. Governed by the visual servoing scheme the real robot successfully completes the mission in several experiments. The robot docks to the battery charger unit with a lateral accuracy of less than two centimeters, which is accurate enough to establish electrical contact between the robot's and the charger's contacts. The same mission is completed with two different camera configurations and control schemes. The first experiment runs with a standard visual servo controller and a static camera

Table 4.3: Control scheme for visual navigation on virtual camera plane by homography.

1. *Estimation of homography* $\hat{\mathbf{H}}$ *on the horizontal virtual camera plane and decomposition of* $\hat{\mathbf{H}}$ *(cf. equation 2.7) into rotation* $\mathbf{R}$ *and translation* $\mathbf{T}$.
2. *Calculation of the rotational control by extracting the rotation of the robot around its vertical axis* $\Delta\Theta_R$ *from* $\mathbf{R}$.
3. *Calculation of polar coordinate* α *:*

$$\alpha = \arctan(t_z/t_x). \tag{4.23}$$

4. *Calculation of scaled polar coordinate* ρ *:*

$$\rho = \sqrt{(t_x(\hat{u}_{\text{vcog}} - u_{\text{vcog}}))^2 + (t_z(\hat{v}_{\text{vcog}} - v_{\text{vcog}}))^2}. \tag{4.24}$$

5. *Motor command fusion using the the gains* k_1*,* k_2 *and* k_3*:*

$$v_R = k_1\rho, \quad \omega_R = k_2\Delta\Theta_R + k_3\alpha. \tag{4.25}$$

aligned with the direction of motion. The second experiment takes advantage of the gaze control of the pan-tilt camera and relies on the visual navigation on the virtual camera plane by Jacobian. The same mission is accomplished with fewer landmarks that are more conveniently mounted to the corridor walls and a smoother trajectory. Figure 4.6 shows the position and orientation of visual landmarks and the path followed with the standard visual servo controller described by [70]. As the translational and rotational velocities are coupled and, more important, camera and robot heading are aligned, the robot is only able to move directly towards a visual landmark at an angle of 90° between feature plane and camera axis. The mission is accomplished with seven properly distributed landmarks but particular in the corridor section for landmarks 2-5 the resulting path is jagged and suboptimal. In order to traverse the corridor it is necessary to install extra boards inside the corridor to accommodate landmarks three and five. For service robotic tasks such manipulation of the environment is not acceptable as visual navigation should be only based on landmarks that occur naturally in the environment. Figure 4.7 shows the landmark locations and robot path for the visual controller with gaze control. With only six landmarks the robot travels along the shortest path in the center of the corridor. In contrast to the fixed camera scenario, additional landmarks along the corridor become obsolete. The camera control switches to the next landmark once the visual control converges to the previous reference image. Figure 4.7 depicts the reference positions for landmark three and four, referred to as reference position three and four. Although landmark four is longitudinally ahead of the robot's start pose (reference position three), the landmark four is behind the reference position four, which is achieved by a wide-angle camera tracking of the landmark. Further results regarding the positioning performance are provided in [105].

Even though these experiments are based on artificial templates, the following major ad-

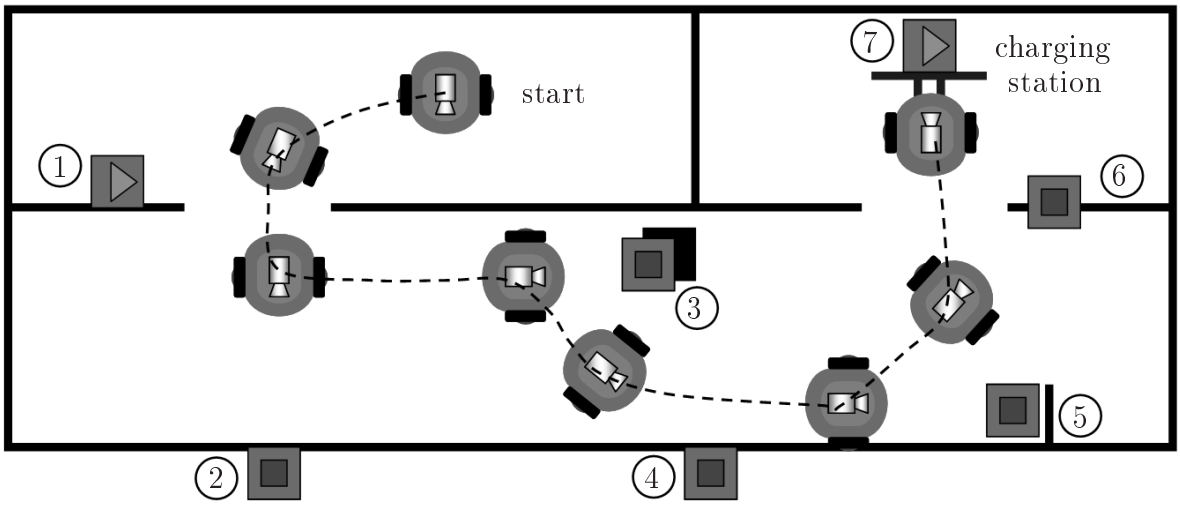


Figure 4.6: Suboptimal robot trajectory due to the limited field of view and inconvenient location of landmarks.

vantages of the visual servoing scheme with virtual camera plane are proven:

- Visual paths require fewer landmarks, especially useful for sparse textured areas.
- Reduced risk of loosing features during control.
- Motion parallel with respect to the feature plane is feasible.

(b) and (c) Large view visual servoing with a pan-tilt camera and omnivision: In order to achieve visual navigation in unstructured environments, the navigation control on the virtual camera plane by image moments and by homography is applied to real world requirements. Therefore features and reference views are extracted from the texture of the office environment and integrated into the topological map from section 3.1. Contrary to many approaches in literature the proposed scheme requires no connectivity of the perspective reference views, but solely the connectivity of the features in the omnidirectional views. This procedure is more flexible and robust, as omnidirectional perception guarantees the visibility of existing texture across a large region of the workspace.

The planning layer automatically generates a topological map in form of a directed graph from omnidirectional views captured during the demonstration run. This map subsequently serves for localization and path planning as well as for dynamical selection of the current optimal reference view for image-based navigation. Each node contains apart from the omnidirectional view also the monocular reference image for local navigation. The planning layer generates a sequence of reference views with overlapping combinations of features in the omnidirectional views, leading from the current to the goal view in the image and workspace. During navigation, in case of sufficient feature visibility of the next waypoint, the image-based control switches to the next monocular reference view, thus allowing for global navigation of the robot. The dynamic switching to the subsequent visible monocular reference view is achieved by the corresponding features stored in the omnidirectional

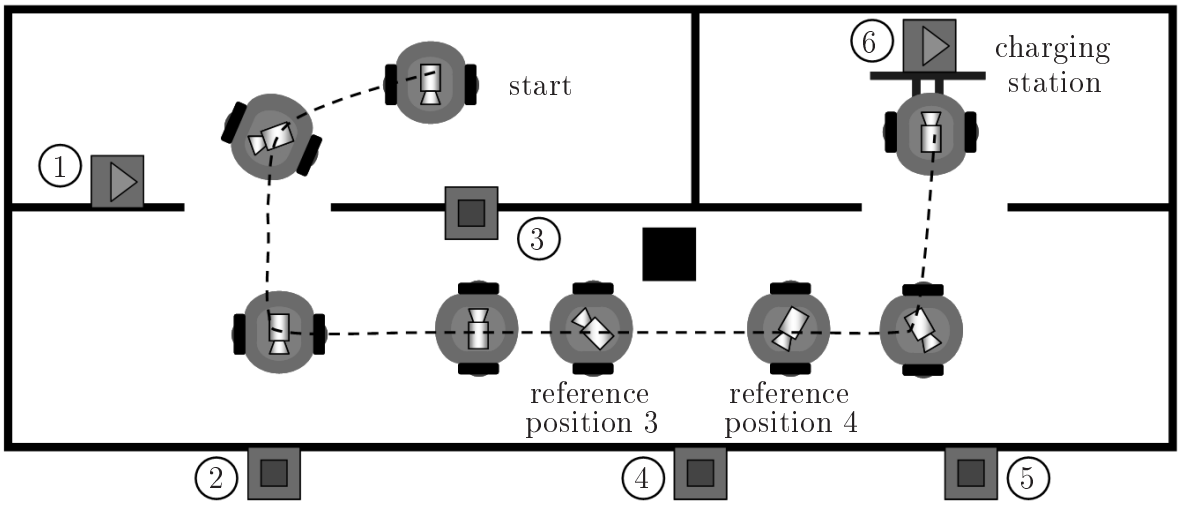


Figure 4.7: Desirable robot trajectory by efficient exploitation of visual landmarks.

view from which also the initial angle for the PTZ camera is calculated. The image-based navigation and localization operates with distinct specific SIFT features. Figure 4.8 illustrates the evolution of the quality of features in several reference views for a local section of the navigation graph. The quality of a monocular reference view is determined by the number of available features as well as their continuing visibility along a longer path. The dynamic selection of the most favorable reference view for the current situation is carried out by means of the two specified criteria.

Figure 4.9 shows the characteristics of visual navigation on horizontal virtual camera plane for the reference views CV3 and CV8 that are extracted by means of the feature distribution in figure 4.8. The initial compensation of the lateral error and the subsequent alignment of the robot by means of the context depending behaviors is evident. Initially the robot is dislocated by an offset of 50 cm laterally and 3 m longitudinally to the goal position. After 2 m the next set of features is detected in the omnidirectional view and the control switches to the next reference image. The residual position error is about 5 cm along both spatial directions. The approaches for visual homing (b) and (c) on the virtual horizontal camera plane exhibits a similar performance as the approach introduced beforehand with navigation on a vertical virtual camera plane with Jacobian. The insights for visual homing as the results of this work under the assumption of planar surfaces in office environments can be summarized as follows: The virtual camera plane allows for decoupled navigation and gaze control. Efficient exploitation of existing texture by omnidirectional preselection of monocular reference views in conjunction with the virtual camera plane enables visual homing within unstructured environments with minimal texture without the urge of 3D modelling. The limited field of view of the pan-tilt is compensated by the omnidirectional camera, whereas the low resolution of the omnidirectional is avoided by the pan-tilt camera. Dynamic environments with natural texture are dealt with by generic representation of moments as well as local feature extraction such as SIFT, ORB or SURF (cf. section 2.2).

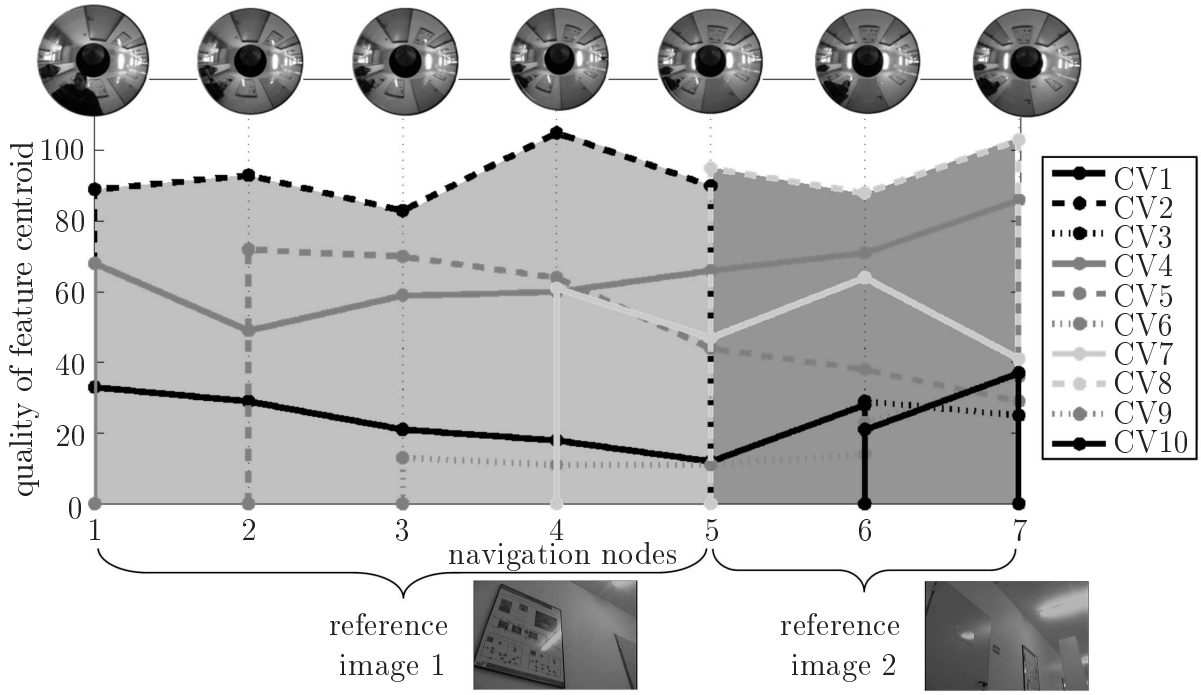


Figure 4.8: Quality of reference views generated from tracked feature centroids over part of the sequence of omnidirectional image navigation nodes.

4.5 Comparison of vision guided and visual navigation

Visual reactive behaviors for homing, centering, door passing and obstacle avoidance are successfully designed and implemented equivalently to behaviors based on distance information. In order to achieve goal-oriented visual navigation in an office environment the visual behaviors door passing and visual homing are integrated into the subsumption architecture as previously described in figure 3.6.

Figure 4.10 details the image-based navigation with the topological map of the department's office environment including the visual nodes and edges of the graph in the upper half and the definition of the three different types of visual nodes in the lower half. The topological representation is analogous to figure 3.1 and the description in section 3.1, however, with the important difference that the stimuli for the reactive behaviors are solely provided by the visual perception. The lower layers of the subsumption architecture are adapted according to the type of the next node instructed by the planning layer. Accordingly for the corridor node the behaviors are ordered in priority from the highest layer to the lowest: turn around, obstacle avoidance, visual homing and corridor centering, thus allowing for traversal of the corridor regions with minimal texture as the corridor centering requires significantly less texture than the visual homing. In the standard corridor scenario the robot is driven by visual homing, which subsumes the corridor centering, as long as the

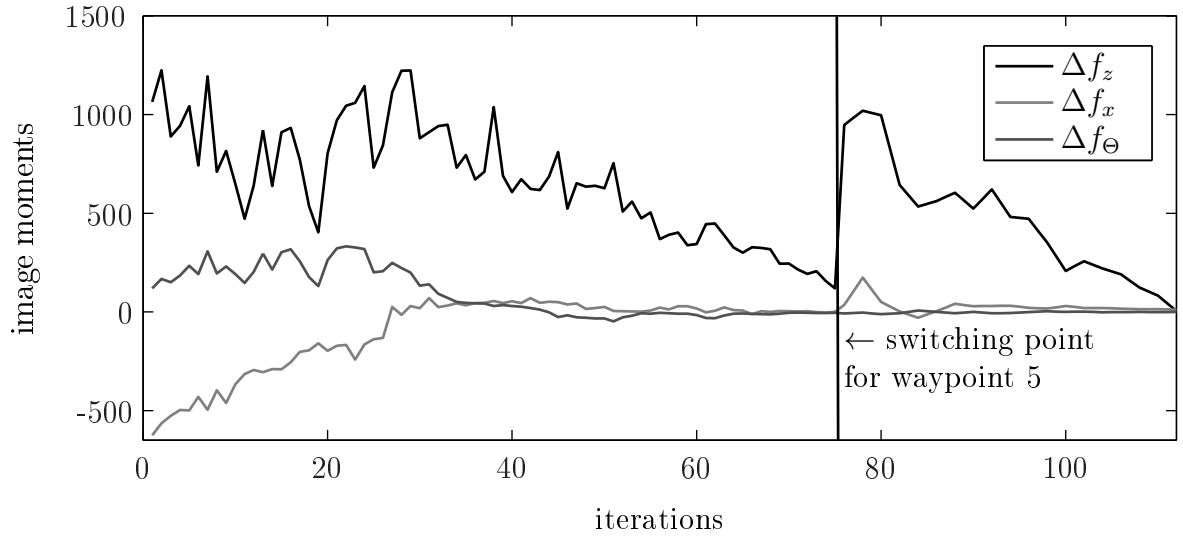
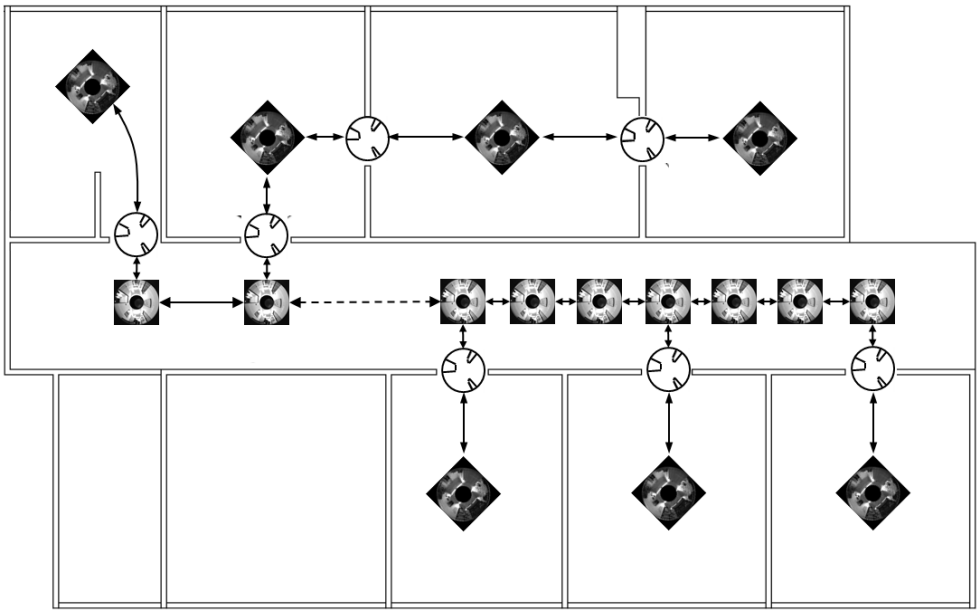


Figure 4.9: Visual homing for reference views CV3 and CV8 from figure 4.8.

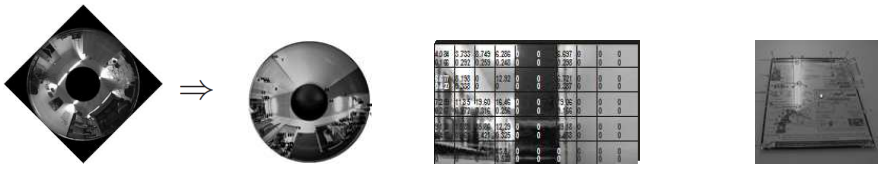
obstacle avoidance is not activated by nearby objects in front of the robot (cf. figure 3.6). The localization with omnivision runs parallel in order to monitor the execution of the plan by the coordination layer. For reaching the door node, only the door passing as well as a modified obstacle avoidance behavior are required. The modification of the obstacle avoidance for the door node is mandatory, such that the door posts represent no obstacles for the robot. The third visual node handles room scenarios, which require turn around, obstacle avoidance and visual homing, stated again from the highest to the lowest layer of the subsumption architecture. As expected from the individual evaluation of the visual behaviors, visual navigation successfully fulfills its objectives during the experiments in the office environment.

In regions with sufficient visual clues both stimuli, vision and distance, demonstrate a similar performance. Nonetheless the overall visual navigation is subject to the same shortcomings as described in section 3.3.3 caused by textureless environments which provide no stimulus for obstacle avoidance and turn around behavior. Compared to the results in section 3.3.3 the corridors are now traversed on a straight line in the same manner as shown in figure 4.7 because the homing behavior subsumes the corridor centering. The critical situations such as static (potentially textureless) obstacles and tricky corner situations handled by the turn around behavior and imposed on the robot in the experiments in the previous sections do not occur during the experiment as the goal-oriented navigation avoids these situations beforehand during the graph generation. The experiments clarify that visual stimuli alone are not sufficient to capture all relevant aspects of the environment for robust navigation mainly due to large textureless office regions. These regions are identified by means of the confidence in the optical flow and avoided by the turn-around behavior even if they are traversable. Therefore the vision guided navigation outperforms the visual navigation in these particular cases. Nonetheless a proper fusion of visual stimuli with sonar

measurements constitute an economic alternative to laser sensors for robot navigation. The visual stimuli of the turn around behavior are replaced by sonar measurements or are directly fused with the confidence rated time to contact.

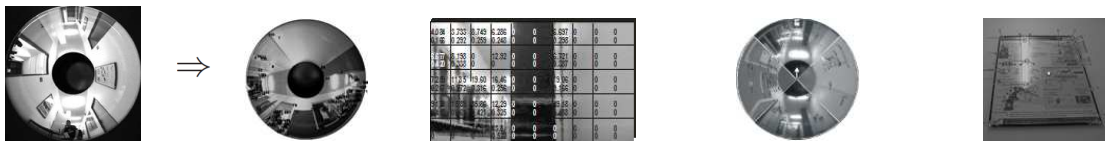


a) Navigation overview with room, door and corridor nodes.



b) Room node: localization obstacle avoidance visual homing.

c) Door node: door passing obstacle avoidance.



d) Corridor node: localization obstacle avoidance corridor centering visual homing.

Figure 4.10: Image-based navigation (analogous to figure 3.1, but with vision as stimuli).

Chapter 5

Local visual servoing with generic image moments

This chapter introduces a novel 6 DOF visual servoing scheme for end-effector to object alignment that relies on the pixel coordinates, scale and orientation of augmented point features such as SIFT. The visual servoing scheme for augmented features enables the use of a large variety of local feature extractions such as ORF, SURF, or GLOH (cf. section 2.2). The control is based on geometric moments computed over a dynamic set of redundant augmented feature correspondences between the current and the reference view [64]. The method is generic as it does not depend on a geometric object model but automatically extracts augmented features from images of the object. The foundation of visual servoing on generic augmented features renders the method robust with respect to loss of redundant features caused by occlusion or changes in viewpoint. The moment based representation establishes an approximate one-to-one relationship between image moments and degrees of motion [109]. This property is exploited in the design of a decoupled controller that demonstrates superior performance in terms of convergence and robustness compared with an inverse image Jacobian controller.

The presented work follows the paradigm of decoupled image moments extending the ideas presented in [143] to overcome the known shortcomings of visual servoing schemes stated in section 2.4, cf. [24, 70]. Visual servoing with decoupled image moments falls into the category of "*partitioned visual servo*" according to the nomenclature of [25]. The proposed approach guarantees a sensitivity matrix with fewer off-diagonal couplings as shown in [143] or [145] for uncalibrated visual servoing, even for markerless visual servoing by explicitly exploiting the additional information contained in augmented features such as SIFT. Using explicitly the complete information of the feature extraction for the first time the proposed solution for decoupled visual servoing differs also significantly from other known approaches [26, 33, 120] explained in detail in section 2.4.

The chapter is organized as follows: Section 5.1 defines augmented point features, which include the pixel coordinates u_i and v_i, the canonical orientation of the keypoint ϕ_i and scale σ_i for a single feature $\mathbf{f}_i$. The automatic feature identification is essential for proper convergence of the visual control towards the reference view. Section 5.2 explains the aggregation of the augmented point features into image moments for visual servoing. A correlation analysis between the image moments and degrees of motion based on the corresponding image Jacobian is provided to establish their approximate one-to-one relationship. Section 5.3 presents visual servoing in 4 DOF and section 5.4 in 6 DOF with augmented point features. In section 5.5 visual servoing on a virtual camera plane is described as an alternative. The chapter is summarized with an evaluation and conclusion in section 5.6.

5.1 Augmented point features

A single augmented point feature $\mathbf{f}_i$ such as SIFT contains four attributes, namely the pixel coordinates u_i and v_i, the canonical orientation of the keypoint ϕ_i and its scale σ_i. In the following, the desired appearance of augmented features in the reference position is denoted by $\mathbf{f}_{\mathrm{ref}_i} = [u_{\mathrm{ref}_i}, v_{\mathrm{ref}_i}, \phi_{\mathrm{ref}_i}, \sigma_{\mathrm{ref}_i}]$ and the current augmented features are denoted by $\mathbf{f}_i = [u_i, v_i, \phi_i, \sigma_i]$. Scale and keypoint orientation are ideal to control the distance to the object and the rotation around the camera axis as they are at large insensitive to translation and rotation along the other axes [64].

In the following the accuracy of the rotation estimate and its robustness is analyzed with respect to changes in viewpoints caused by camera rotations along the other axes using SIFT as augmented point features. The camera is rotated around the optical axis over the entire range -180° to 180°. The distribution of the error between the estimated mean computed over all SIFT features and the true rotation is shown in figure 5.1. The graph shows the distribution of the error ε_γ across the 128 rotation steps. The mean absolute error amounts to $|\varepsilon_\gamma| = 0.52°$ and the standard deviation σ_γ of the error distribution ε_γ is about 0.4°. Notice, that the absolute error in the estimated orientation is smaller for rotations close to the reference orientation which eventually determines the residual orientation error for the visual control. This accuracy in orientation is confirmed in the closed-loop control visual servoing experiments. The average keypoint orientation coincides with the camera orientation, which guarantees a unique minimum and the stability of visual control of γ. Even if the image and feature plane are not parallel the perspective distortion of the SIFT feature caused by a camera rotation along an orthogonal axis hardly deteriorates the rotation estimate which still accurately captures the camera orientation. Table 5.1 shows that orthogonal rotations along α only have a minor effect. Rotations of more than 30° cause affine deformations for which the SIFT keypoint descriptors in different views are no longer compliant. For rotations of up to 30° the mean absolute error increases to $|\varepsilon_\gamma| = 1.13°$ which is still accurate enough for the application at hand.

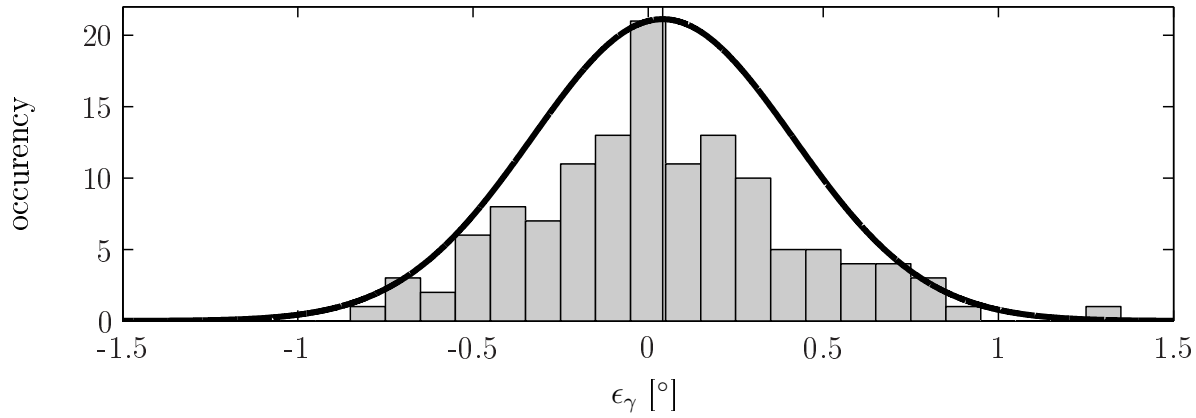


Figure 5.1: Estimation error for γ across absolute orientations from -180° to 180°.

Table 5.1: Error of the rotation estimate as a function of camera rotation $\Delta\alpha$ along the orthogonal axis. (All units in °.)

$\Delta\alpha$	0	5	10	15	20	25	30
$\lvert\varepsilon_\gamma\rvert$	0.52	0.26	0.36	1.05	0.88	0.92	1.13
σ	1.13	1.50	1.10	2.60	3.11	3.44	4.59

Figure 5.2 depicts the variation of scale σ for typical SIFT features as a function of the distance z between the object and the camera. The scale of SIFT features is given by $\frac{k}{z}$. The constant gain k depends on the focal length of the camera multiplied by the initial scale of the feature.

SIFT features in the current image are matched with their corresponding reference features in the goal image by comparison of their distinctive keypoint descriptors. Keypoint descriptors of the same feature in different views are, although similar, not exactly identical, which might result in false correspondences between features. In addition a SIFT feature present in the reference image might not appear in the current image and vice versa. Therefore, the objective of the automatic feature selection is to establish reliable correspondences between the same features that are robust across different views in order to avoid false correspondences. Candidates for stable and unambiguous SIFT features are identified according to the following criteria:

- similarity
- angular criterion
- epipolar constraint

The list of candidate reference features is composed of all features originally detected in the goal image. Feature selection proceeds in three stages, of which the first two stages operate offline and reject features in the reference image, whereas the last online stage analyzes

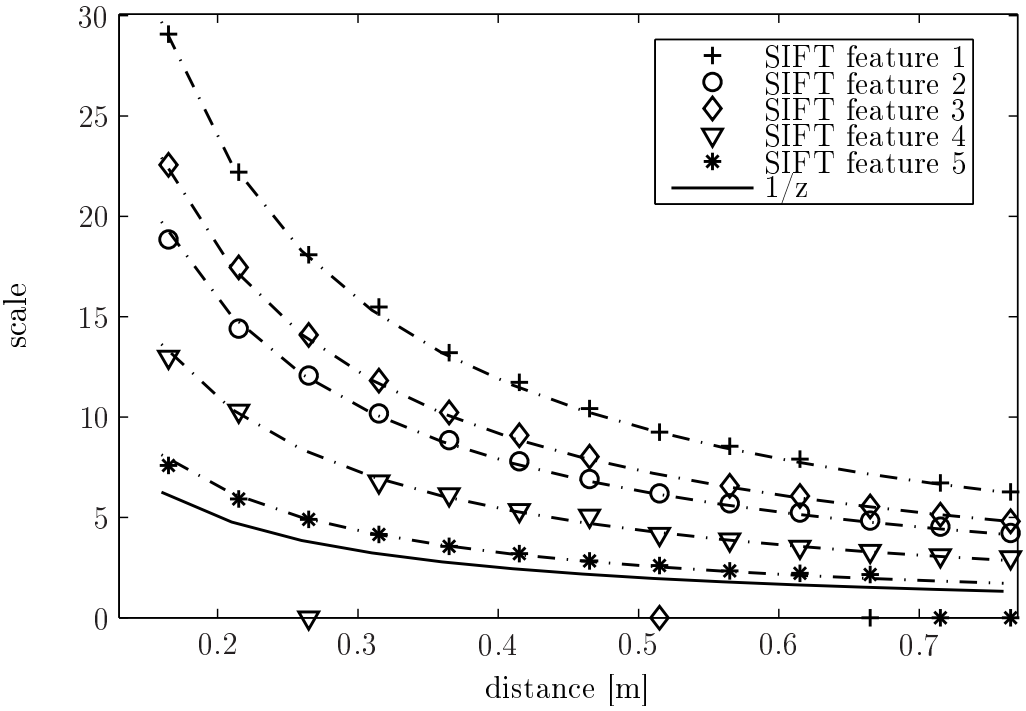


Figure 5.2: Scale versus distance.

the features in the current image. The first stage only compares SIFT features in the reference image with each other. Similar SIFT features that strongly resemble each other are immediately rejected to avoid later confusion among them. In the second stage SIFT features are matched across multiple views taken from different camera poses distributed across the entire workspace. In this case those reference keypoints are rejected for which the matched keypoint violates the angular criterion and the epipolar constraint. In the third online stage, the angular criterion is applied once more to the features detected in the current image. Only those features that pass all of the above tests are finally considered within the visual control scheme.

5.2 Generic moments

5.2.1 Moments for rotation

A camera rotation around its optical axis by γ induces an inverse rotation of equal magnitude of the keypoint orientations ϕ_i. The averaged keypoint rotation f_γ regulates the

camera rotation:

$$f_\gamma = \frac{1}{n}\sum_{i=1}^{n}\phi_i. \tag{5.1}$$

The point features u_i, v_i are first aligned with the camera orientation in the reference view according to the observed feature f_γ. The correction along γ is therefore defined as $\Delta f_\gamma = f_{\mathrm{ref}_\gamma} - f_\gamma$. The visual features are rotated by Δf_γ such that the current feature locations u_i and v_i are aligned with the camera orientation in the reference view. The new feature locations u_i' and v_i' are determined as follows:

$$\begin{bmatrix} u_i' \\ v_i' \end{bmatrix} = \begin{bmatrix} \cos(\Delta f_\gamma) & -\sin(\Delta f_\gamma) \\ \sin(\Delta f_\gamma) & \cos(\Delta f_\gamma) \end{bmatrix} \begin{bmatrix} u_i \\ v_i \end{bmatrix}. \tag{5.2}$$

In the following the corrected pixel coordinates u_i' and v_i' are used for the computation of the remaining image moments and for better comprehensibility are denoted as u_i and v_i. In order to control the rotations around α and β two additional image moments corresponding to the rotations about the x- and y-axis, f_α and f_β, are defined. The image moments f_α and f_β capture the perspective distortions of lines connecting pairs of features caused by rotations:

$$f_\alpha = \sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{(-v_{\mathrm{ref}_i} - v_{\mathrm{ref}_j})\left\|\mathbf{p}_j - \mathbf{p}_i\right\|}{\sum_{k=1}^{n}\sum_{l=k+1}^{n}\left\|\mathbf{p}_k - \mathbf{p}_l\right\|}, \tag{5.3}$$

$$f_\beta = \sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{(-u_{\mathrm{ref}_i} - u_{\mathrm{ref}_j})\left\|\mathbf{p}_j - \mathbf{p}_i\right\|}{\sum_{k=1}^{n}\sum_{l=k+1}^{n}\left\|\mathbf{p}_k - \mathbf{p}_l\right\|}. \tag{5.4}$$

The term $\left\|\mathbf{p}_j - \mathbf{p}_i\right\|$ denotes the length of the line connecting the two pixels:

$$\left\|\mathbf{p}_j - \mathbf{p}_i\right\| = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2}, \tag{5.5}$$

which are weighted by the factor $(-v_{\mathrm{ref}_i} - v_{\mathrm{ref}_j})$. Its sign indicates whether the line is above or below the u-scan line through the camera's principal point. The absolute magnitude of the weight increases with the vertical distance from the image center. The image moment f_β represents the equivalent effect of dilations and compressions of lines caused by rotations along the y-axis. Figure 5.3 illustrates the effect for a square configuration of four feature points that form six lines. Figure 5.3 a) depicts the image of the square for parallel feature and image plane, whereas in figure 5.3 b) the image with the camera is tilted around the x-axis and the shift along the y-direction is compensated. The distortion increases the length of line 1 and simultaneously decreases the length of line 3. This dilation and compression of lines is captured by the equation 5.3.

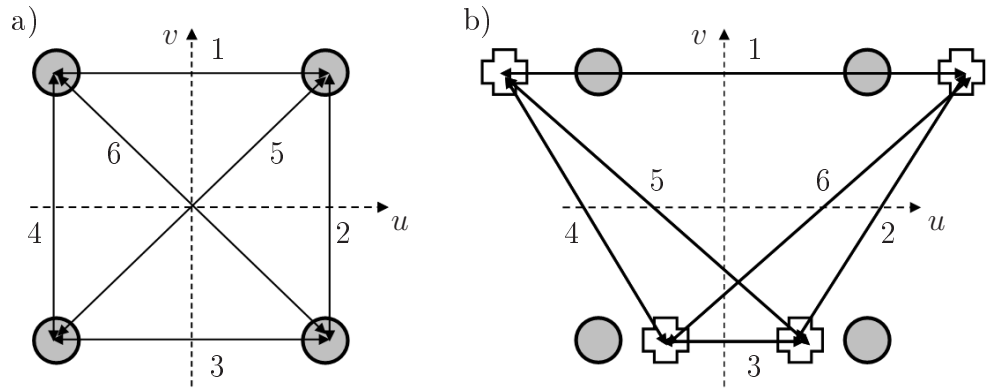


Figure 5.3: Perspective distortion caused by camera tilt α captured by the image moment f_α.

5.2.2 Moments for translation

The translation along the camera axis is governed by the image moment f_z defined as the average scale of the augmented features:

$$f_z = \frac{1}{n}\sum_{i=1}^{n}\sigma_i. \tag{5.6}$$

Alternatively f_z can be expressed as the distance between point features according to:

$$f_{zd} = \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n}\sqrt{(u_i-u_j)^2+(v_i-v_j)^2}}{\frac{n}{2}(n-1)}, \tag{5.7}$$

that captures the average scale of the scene in a similar manner. Nonetheless computing the image moment for z from the distance between point features f_{zd} is not invariant with respect to perspective distortions caused by rotations along the other two axes. Therefore, the inherent scale of augmented features defined by f_z is preferred to the alternative definition f_{zd}.

The moment based controller in [64] operates with the geometric centroid of point features for regulating translations along the x- and y-axis. The image moments are defined by the centroid of matched features according to:

$$f_x = \frac{1}{n}\sum_{i=1}^{n}u_i, \quad f_y = \frac{1}{n}\sum_{i=1}^{n}v_i. \tag{5.8}$$

However, the geometric centroid primarily captures the horizontal translation of the camera, but suffers from a sensitivity to motions along the remaining degrees of freedom, especially from motions in z, α and β.

In the following the image Jacobian $\mathbf{J}_{f_x,f_y}$ under the assumption of prior backrotation around the optical axis is derived in order to analyze the remaining couplings (cf. appendix B). The centroid feature $[f_x, f_y]^T$ behaves similar to a virtual point feature and the Jacobian is obtained by averaging the individual point feature Jacobians stated in [70]:

$$\mathbf{J}_{f_x,f_y} = \frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix} \frac{\lambda}{z} & 0 & \frac{-u_i}{z} & \frac{-u_i v_i}{\lambda} & \frac{\lambda^2+u_i^2}{\lambda} \\ 0 & \frac{\lambda}{z} & \frac{-v_i}{z} & \frac{-\lambda^2-v_i^2}{\lambda} & \frac{u_i v_i}{\lambda} \end{bmatrix}, \tag{5.9}$$

in which λ denotes the focal length and z denotes the distance between the camera and the feature plane. The main difference with respect to a point feature is the simplifying assumption that all augmented features share approximately the same depth z. This assumption is reasonable as long as the depth of the scene is small compared to the distance to the camera (weak perspective projection).

The visual features f_x and f_y capturing translations along the x- and y-axis are expressed as the weighted aggregation of the matched feature locations [109]:

$$f_x = \sum_{i=1}^{n} w_i u_i, \quad f_y = \sum_{i=1}^{n} w_i v_i. \tag{5.10}$$

By proper selection of the weights w_i attributed to individual point features $[u_i, v_i]$ it is possible to decouple f_x, f_y from the remaining degrees of freedom. With this objective in mind the image moments f_x and f_y are supposed to only depend on v_x and v_y, respectively.

Decoupling for 4 DOF: The following analysis for the proper dynamic weight focuses on the element $\tilde{J}_{f_x,z}$ related to the u-component. $\tilde{J}_{f_x,z}$ defines the non-linear coupling of the feature motion in dependence on task space motions in z. Notice, that for the decoupling of the visual feature f_y the same methodology is applied only using v_i instead of u_i. The undesired off-diagonal element of the sensitivity matrix $\tilde{J}_{f_x,z}$ is eliminated if the dynamic weights w_i satisfy the constraint:

$$\tilde{J}_{f_x,z} = \sum_{i=1}^{n} w_i \frac{-u_i}{z} = 0. \tag{5.11}$$

For an arbitrary set of point features $[u_i, v_i]$, this constraint is violated for the geometric centroid calculation with equal weights $w_i = 1/n$. In order to maintain the similarity with the conventional centroid a minimal variation Δw_i of the original weights $w_i = 1/n + \Delta w_i$ is endeavored that satisfies the constraint. This optimal variation is obtained by minimizing the following cost function in conjunction with a Lagrange multiplier λ_1:

$$F = \frac{1}{2}\sum_{i=1}^{n}\left(w_i - \frac{1}{n}\right)^2 + \lambda_1 \sum_{i=1}^{n} w_i u_i. \tag{5.12}$$

In order to solve the optimization problem the partial derivative of equation 5.12 with respect to w_i and the Lagrange multiplier λ_1 are computed as:

$$\begin{aligned} \frac{\partial F}{\partial w_i} &= \left(w_i - \frac{1}{n}\right) + \lambda_1 u_i = 0 \\ \frac{\partial F}{\partial \lambda_1} &= \sum_{i=1}^{n} w_i u_i = 0, \end{aligned} \tag{5.13}$$

which in turn yields the least squares solution:

$$w_i = \frac{1}{n} - \frac{u_i \bar{u}}{\sum_{i=1}^{n} u_i^2}, \quad \bar{u} = \sum_{i=1}^{n} \frac{u_i}{n}. \tag{5.14}$$

Intuitively, the weight of features whose pixels possess the opposite sign as the geometric centroid $\bar{u} = \sum_i u_i/n$ is increased, whereas those with the same sign are down-weighted. Notice, that by definition the weighted centroid is always located at the origin of the current image, thus $f_x = f_y = 0$. However, the reference features $f_{\mathrm{ref}_x} = \sum_i w_i u_{\mathrm{ref}_i}$ and $f_{\mathrm{ref}_y} = \sum_i w_i v_{\mathrm{ref}_i}$ are no longer constant, but depend indirectly on the current image via the dynamic weights w_i and are therefore implicitly susceptible to motions along multiple degrees of freedom. In order to verify the decoupling of the weighted visual features f_x and f_y from a motion v_z, it is necessary to show that the weights for an identical set of feature points remain indeed independent of the distance z. The following considerations assume that the image plane is oriented parallel to the feature plane. The perspective projection of a world point on the image plane is given by $u_i = x_i \frac{\lambda}{z}$, whereas the same point displaced by Δz is projected to $u_{i,\Delta z} = x_i \frac{\lambda}{z+\Delta z}$. Assuming that the weights w_i fulfill the constraint $\sum_{i=1}^{n} u_i w_i$, the weighted sum of the feature points u'_i at distance $z + \Delta z$ is thus given by:

$$\sum_{i=1}^{n} u_{i,\Delta z} w_i = \frac{z}{z+\Delta z} \sum_{i=1}^{n} u_i w_i = 0. \tag{5.15}$$

Due to fact that $\frac{z}{z+\Delta z}$ is a proportional factor common to all features, the optimal weights for the first set of feature points also transform the virtual centroid of the second set of feature points to zero. Nonetheless, the weights are effectively down-scaled by the factor $\frac{z}{z+\Delta z}$. In order to render the weights themselves and not only the centroid independent of the distance z, the weights are normalized according to:

$$w_{i,\mathrm{norm}} = w_i \frac{1}{n} \sum_{k=1}^{n} |w_k|. \tag{5.16}$$

Assuming that the weighted sum of the feature points is initially zero for the reference view, the weights $w_{i,\mathrm{norm}}$ transform the virtual centroid of every other view to the image center independent of the distance z (cf. [109]). Figure 5.4 depicts two projections of the

same features, in which the corresponding viewpoints differ by a displacement Δz. The weights w_i are optimized according to the first set of feature points, but are also applied to weight the second set. Both virtual centroids are located in the image center even though the weighted feature points for the two sets differ. The example demonstrates that neither the weights nor the visual features f_x and f_y change with a camera motion along z. Notice, that in the weighted scheme the role of current and reference features is reversed. Typically, the reference features are constant and the current features change with the motion of the camera. However, in the weighted scheme the current centroid ($f_x = 0, f_y = 0$) is constant per definition and always coincides with the principal point due to equation 5.11. Instead the reference features $f_{\mathrm{ref}_x}, f_{\mathrm{ref}_y}$ change over time as the weights w_i change with the current view, even though the point features $u_{\mathrm{ref}_i}, v_{\mathrm{ref}_i}$ themselves are constant.

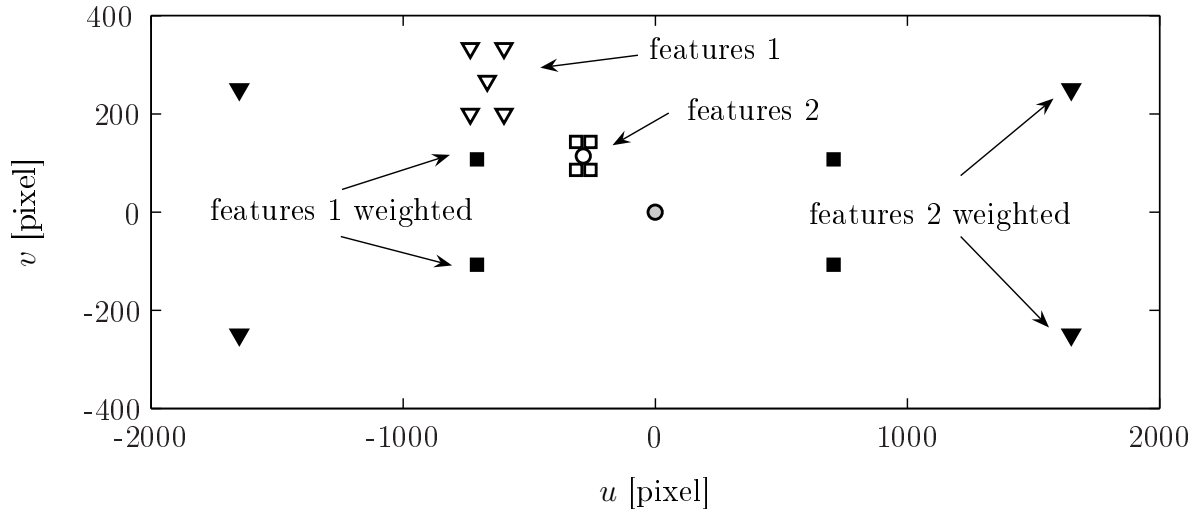


Figure 5.4: Virtual centroid for two sets of feature points at different distances between camera and feature plane.

Figure 5.5 depicts the image projections of the conventional and weighted centroid u-components' view as a function of the lateral task space error Δx. The two horizontal lines correspond to the constant conventional reference centroid and the constant weighted current centroid u-components. In this example, the reference image is deliberately chosen such that the majority of features is located in the right half plane. Therefore, the conventional centroid is shifted along the u-component by 120 units from the principal point. The current conventional centroid depends linearly on the lateral error and intersects the reference centroid for a zero lateral error $\Delta x = 0$. However, the offset and slope depend on the longitudinal distance between camera and feature plane. Current and reference features interchange their role for the weighted scheme, in that the former remains constant and the later changes with the lateral error in a non-linear fashion. In this case the dependency of the centroid on the lateral error remains the same independent of the longitudinal pose error in z. Again, the reference and current centroid intersect for zero lateral error. Even

though the slopes for the weighted and conventional centroid exhibit opposite signs, the actual image error $f_{\mathrm{ref}_x} - f_x$ has the same sign for both schemes. Figure 5.5 also reveals a slight asymmetry of the weighted reference centroid for positive and negative lateral errors. For large positive lateral errors the image error even increases slightly with decreasing lateral error. This asymmetry is caused by the inhomogeneous distribution of point features u_i, which in turn effects the adaptation of the weight factors through equation 5.14. It should be noted, that the asymmetry and slope inversion do not effect the stability or convergence of the visual control and only occur if the point feature distribution in the reference image is significantly skewed.

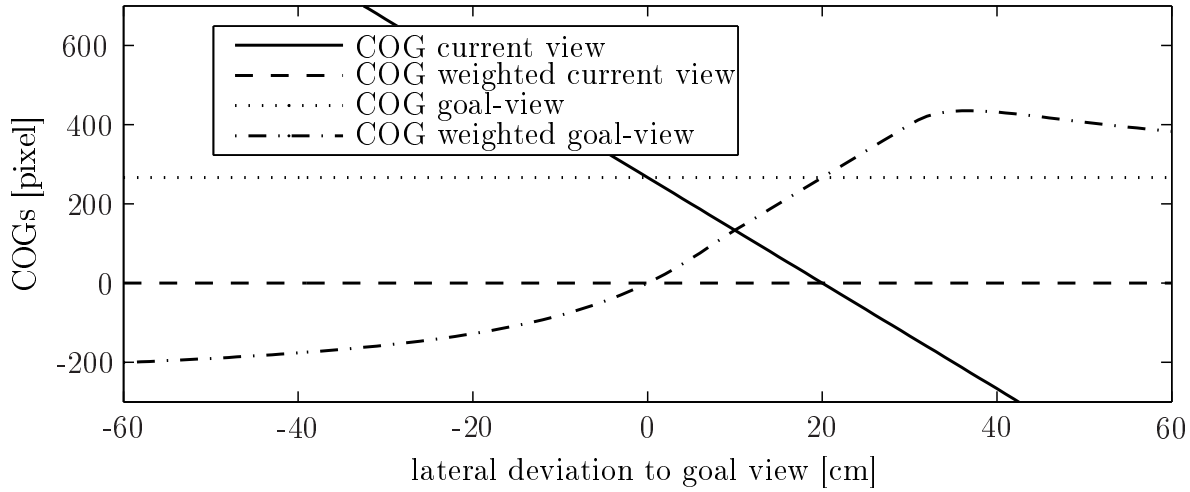


Figure 5.5: Centroid u-component as a function of the lateral error for the conventional and weighted centroid in the current and reference view.

Decoupling for 6 DOF: The aim is to extend the decoupling to visual servoing for 6 DOF. This requires that the features f_x and f_y do not only become independent on the motion v_z but also on the rotations ω_α and ω_β. Again, the constraints emerge from an analysis of the Jacobian in equation B.1. The optimization problem for decoupling the visual feature f_x from the motions v_z and ω_α according to the Jacobian in equation 5.3 is stated as follows:

$$F = \frac{1}{2}\sum_{i=1}^{n}\left(w_i - \frac{1}{n}\right)^2 + \lambda_1 \sum_{i=1}^{n} w_i u_i + \lambda_2 \sum_{i=1}^{n} w_i u_i v_i. \tag{5.17}$$

Theoretically, it is possible to cancel the component $\sum_{i=1}^{n} w_i \left(\lambda^2 + u_i^2\right)$ of the Jacobian related to the motion ω_β as well. If this additional constraint is included, the minimization problem in equation 5.17 is algebraically still solvable (cf. appendix B). Unfortunately, the inclusion of this constraint substantially reduces the sensitivity of the feature f_x with its associated motion v_x, resulting in a deterioration of the visual control. Therefore, the

weight optimization problem only includes constraints for the cancelation of elements $\tilde{J}_{f_x\alpha}$ and $\tilde{J}_{f_y\beta}$ in the sensitivity matrix based on the image Jacobian in equation 5.21. The cost function F is partially differentiated with respect to w_i and the Lagrangian multipliers λ_i:

$$\begin{aligned} \frac{\partial F}{\partial w_i} &= w_i - \frac{1}{n} + \lambda_1 u_i + \lambda_2 u_i v_i, \\ \frac{\partial F}{\partial \lambda_1} &= \sum_{i=1}^{n} w_i u_i, \quad \frac{\partial F}{\partial \lambda_2} = \sum_{i=1}^{n} w_i u_i v_i. \end{aligned} \tag{5.18}$$

Changing to vector notation by substituting $\mathbf{k} = [\frac{1}{n}, \ldots, \frac{1}{n}]$, $\mathbf{u} = [u_1, \ldots, u_n]$ and $\mathbf{p} = [u_1 v_1, \ldots, u_n v_n]$ a set of linear equations is obtained: $\mathbf{w} + \lambda_1 \mathbf{u} + \lambda_2 \mathbf{p} = \mathbf{k}$, with the constraints $\mathbf{w}\mathbf{u}^T = 0$, $\mathbf{w}\mathbf{p}^T = 0$. In order to determine the Lagrangian multipliers, the weights w are eliminated by multiplying with the transpose of u and p. This results in a system of linear equations, which can be solved in closed form:

$$\begin{bmatrix} \mathbf{u}^{\mathrm{T}}\mathbf{u} & \mathbf{u}^{\mathrm{T}}\mathbf{p} \\ \mathbf{p}^{\mathrm{T}}\mathbf{u} & \mathbf{p}^{\mathrm{T}}\mathbf{p} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} \mathbf{u}^{\mathrm{T}}\mathbf{k} \\ \mathbf{p}^{\mathrm{T}}\mathbf{k} \end{bmatrix}. \tag{5.19}$$

The corresponding weights of the point features are determined as $\mathbf{w} = \mathbf{k} - \lambda_1 \mathbf{u} - \lambda_2 \mathbf{p}$ and are subsequently normalized according to equation 5.16.

5.2.3 Coupling analysis of the sensitivity matrix

The image Jacobian for the visual feature f_α is given by

$$\mathbf{J}_{f_\alpha} = \frac{\sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{p_{ij} v_{\mathrm{ref}_{ij}} \lambda}{\sqrt{8} z^2} \begin{bmatrix} 0 & 0 & -\frac{2\lambda}{z} & -v_{ij} & +u_{ij} \end{bmatrix}}{\sum_{k=1}^{n}\sum_{l=i+1}^{n} \|\mathbf{p}_k - \mathbf{p}_l\|}, \tag{5.20}$$

in which $v_{ij} = (v_i - v_j)/2$, $u_{ij} = (u_i - u_j)/2$, $v_{\mathrm{ref}_{ij}} = (v_{\mathrm{ref}_i} + v_{\mathrm{ref}_j})/2$, $u_{\mathrm{ref}_{ij}} = (u_{\mathrm{ref}_i} + u_{\mathrm{ref}_j})/2$ and the length $p_{ij} = \|p_i - p_j\|$. In the Jacobian for the analogous visual feature f_β the u- and v-components are interchanged. The resulting full 6 DOF Jacobian matrix (sensitivity) exhibits the following block structure:

$$\begin{bmatrix} \dot{f}_x \\ \dot{f}_y \\ \dot{f}_z \\ \dot{f}_\alpha \\ \dot{f}_\beta \\ \dot{f}_\gamma \end{bmatrix} = \begin{bmatrix} J_{f_x,x} & 0 & 0 & 0 & \tilde{J}_{f_x,\beta} & 0 \\ 0 & J_{f_y,y} & 0 & \tilde{J}_{f_y,\alpha} & 0 & 0 \\ 0 & 0 & J_{f_z,z} & 0 & 0 & 0 \\ 0 & 0 & 0 & J_{f_\alpha,\alpha} & \tilde{J}_{f_\alpha,\beta} & 0 \\ 0 & 0 & 0 & \tilde{J}_{f_\beta,\alpha} & J_{f_\beta,\beta} & 0 \\ 0 & 0 & 0 & 0 & 0 & J_{f_\gamma,\gamma} \end{bmatrix} \begin{bmatrix} T_x \\ T_y \\ T_z \\ \omega_\alpha \\ \omega_\beta \\ \omega_\gamma \end{bmatrix}. \tag{5.21}$$

However, some residual couplings remain through the non-zero off-diagonal elements ($\tilde{J}_{f_x,\beta}$, $\tilde{J}_{f_y,\alpha}$, $\tilde{J}_{f_\alpha,\beta}$, $\tilde{J}_{f_\beta,\alpha}$) in the Jacobian. These terms capture the effect of a rotation α along the x-axis on the motion of the v-components of point features, and vice versa a rotation β along the y-axis on the u-component. The complete analysis of the sensitivity matrix is in appendix B and a summary of visual servoing with generic image moments in table B.1.

5.3 Positioning in 4 DOF with augmented point features

Determining optimal control parameters for the image-based controller is not so easy to accomplish with conventional methods of controller generation such as LQR design due to the lack of an object model, model uncertainties of image processing and non-linearities of the control path. A sufficiently exact model of the closed-loop control, consisting of robot and image processing and considering dynamical properties of the manipulator, calibrations, variable latency time and pixel noise as well as the non-linear dependence of the image features on the camera pose, can only be generated with large effort and results in a complex and domain-comprehensive model. Thus, an approach for automatic hardware-in-the-loop (HIL) optimization of the image-based controller gains by evolutionary optimization is introduced. In the following first the determination of control parameters and successively the experimental results are presented.

5.3.1 Controller optimization

The moments for rotation and translation along the camera axis for 4 DOF visual servoing correspond to the image moments introduced in equation 5.1, 5.6 and 5.8. Notice that in the first step the coupled image moments f_x and f_y are employed as this is the most complex case resulting in the following sensitivity matrix given in equation 5.22. For 6 DOF visual servoing with decoupled image moments only four off-diagonal couplings remain. Later on the optimal control parameters for the decoupled moments in 6 DOF using the definition in equation 5.10 are determined in the same manner.

$$\begin{bmatrix} \dot{f}_x \\ \dot{f}_y \\ \dot{f}_z \\ \dot{f}_\gamma \end{bmatrix} = \begin{bmatrix} J_{f_x,x} & 0 & \tilde{J}_{f_x,z} & 0 \\ 0 & J_{f_y,y} & \tilde{J}_{f_y,z} & 0 \\ 0 & 0 & J_{f_z,z} & 0 \\ 0 & 0 & 0 & J_{f_\gamma,\gamma} \end{bmatrix} \begin{bmatrix} T_x \\ T_y \\ T_z \\ \omega_\gamma \end{bmatrix} \tag{5.22}$$

The closed-loop control system consists of the coupling of four decoupled PD controllers:

$$\begin{aligned} [v_x, v_y, v_z, \omega_\gamma]^T &= [k_x, k_y, k_z, k_\gamma]^T [\Delta f_x, \Delta f_y, \Delta f_z, \Delta f_\gamma]^T + \\ &\quad [k_{Dx}, k_{Dy}, k_{Dz}, k_{D\gamma}]^T [\Delta \dot{f}_x, \Delta \dot{f}_y, \Delta \dot{f}_z, \Delta \dot{f}_\gamma]^T, \end{aligned} \tag{5.23}$$

whereas k_{Dx}, k_{Dy}, k_{Dz} and $k_{D\gamma}$ denote the gains of the D (differential) parts. An integral part in the control path is omitted due to the integrating nature of the plant.

The control quality and stability of the image-based control as well as the characteristics of the image features are determined by the selection of the control parameters of the PD controller. The initial control parameters are obtained manually or by means of simplifying dynamic first order models. Subsequently an automatic hardware-in-the-loop (HIL)

optimization of the image-based controller is performed by evolutionary optimization taking into account several performance criteria. The term "evolving hardware" denotes a methodology connecting evolutionary algorithms with the design and optimization of mechanic and electronic systems [89]. In the context of HIL optimization of image-based controllers time-consuming evaluation of controllers on the experimental system proves to be problematic, as they need to consult several motions from initially different robot poses in order to guarantee sufficient robustness for covering the complete workspace. If evolutionary optimization procedures are applied in this context, the available time frame for the fitness evaluations has to be exploited as efficiently as possible.

Thus, the proposed approach utilizes a model-based evolution strategy which initially evaluates all generated offspring by means of an online-learned fitness model [62], such that only the most promising candidates according to the estimated fitness are subject to the actual fitness evaluation on the real robot. According to the model-assisted evolution strategy (MAES) out of the λ_p offspring created within a generation only the estimated λ_e most promising candidates undergo the real test [149]. From those the μ_p best solutions are selected as parents. The achievable progress by MAES within the evolutionary optimization does not strongly depend on the actual model error but more on the capability of predicting correctly the ranking order of the population during pre-selection. Therefore MAES already offers advantages if the model-based pre-selection is better than a purely random choice in terms of ranking. λ-CMAES (controlled MAES) improves the MAES as the number of actually evaluated individuals λ_e is fitted dynamically with the quality of the fitness model in order to guarantee a constant selection quality. λ-CMAES from [62] is employed for the HIL optimization of the visual controller because the evolutionary progress by means of the number of actual fitness evaluations is superior compared to standard evolution strategies. In the following the results of the evolutionary HIL optimization of the image-based controller by means of λ-CMAES ($\lambda_p = 30$, $\lambda_e \in [6, 15]$, $\mu_p = 5$) are presented. The cost function minimizes the quadratic image error

$$F = \sum_{i=1}^{4} \int_0^T \Delta f_i^2(t) dt \tag{5.24}$$

without an explicit penalty of the control effort. The velocities v proposed by the controller are limited by the saturation of the actuating variable during transfer to the robot control. The control behavior is observed for a $T = 15\,\mathrm{s}$ control deviation and evaluated by means of the quadratic control error. To guarantee a robust performance, each controller is evaluated for four different initial displacements of the robot arm and the mean for the costs of all four runs is calculated. Altogether, one HIL quality evaluation of a controller requires one minute on a 1.8 GHz Pentium 4 system.

Before the actual optimization of the controller on the real robot the robustness and efficiency of the method is analyzed in a simulated virtual reality. Optimization in a virtual environment offers the advantage of exactly reproducible behavior which is not subject to disturbances due to variable illumination, dynamical constraints and variable latency as

can be observed in the real system. The gain factors of an image-based controller for a flexible camera are optimized by means of the cost function in equation 5.24.

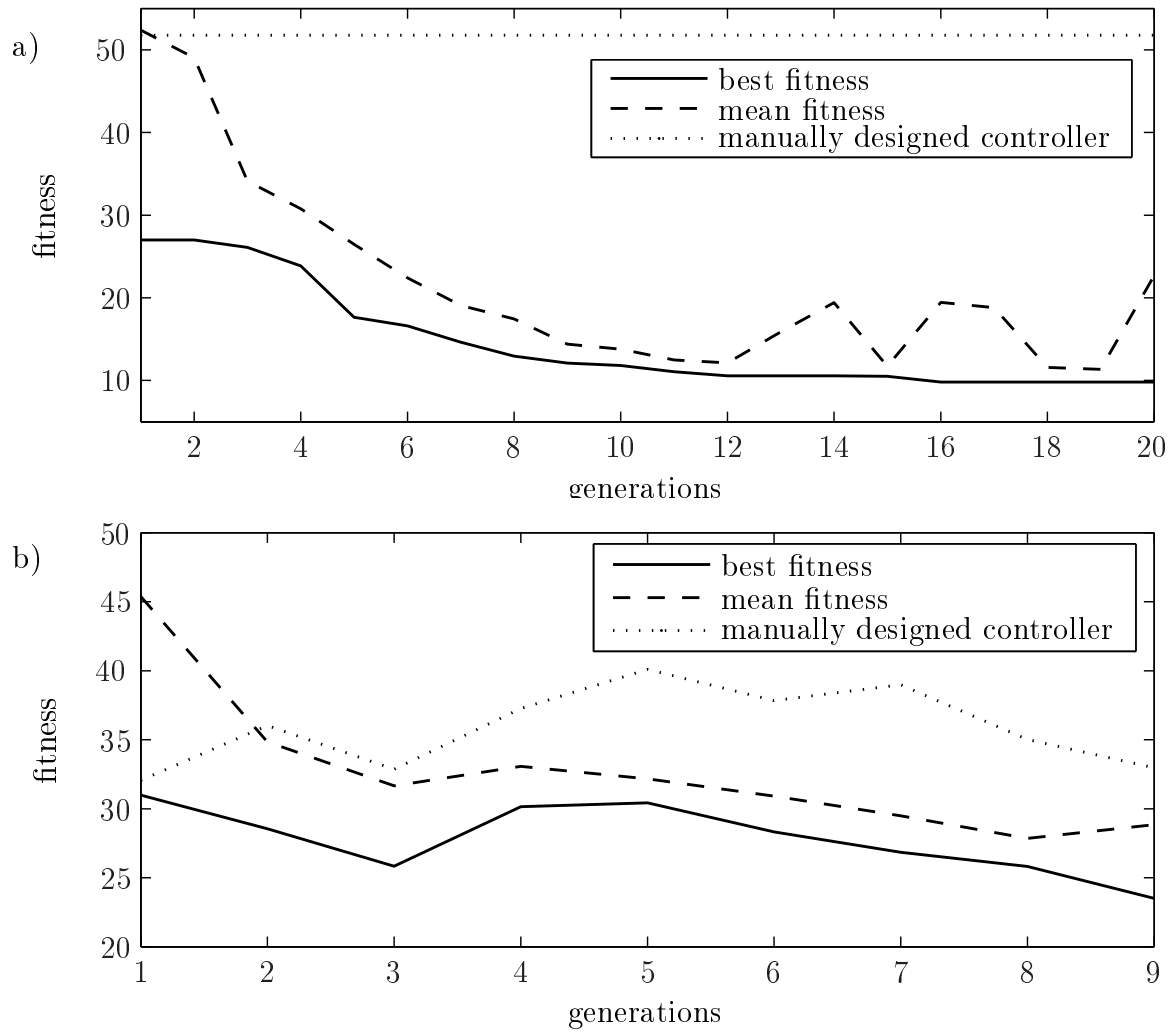


Figure 5.6: a) Development of the mean and best fitness for evolutionary optimization of the image-based controller in virtual reality compared to the empirically tuned controller; b) Evolutionary HIL optimization of the image-based controller on the target system.

Figure 5.6 a) depicts the development of the mean and best fitness within a progress of 20 generations. The optimization results in a significant improvement of the quadratic image error of a factor five compared to the empirically tuned controller. The intermittent increase of the mean fitness towards the end of the evolution is caused by individual unstable controllers with extremely bad quality, which however do not influence the development of the best individuals.

The HIL optimization on the target system shown in figure 5.6 b) proceeds over nine

generations in total, corresponding to an expenditure of time of about 3.5 hours. By using λ-CMAES the fitness evaluation time is reduced by 20%.

For comparison the empirically tuned controller is tested during the continuing optimization under the same conditions as the current generation. The quality variations of up to 20% during the evolution in spite of identical control parameters illustrate the influence of external disturbances on the control behavior. The trend of this influence is reflected in a similar manner for the empirically tuned controller in the fitness progress of the best controller of one generation. It becomes apparent that the optimized controller exhibits a consistently better behavior and the quadratic image error is reduced towards the end of the evolution in average of approximately ten units compared to the empirically tuned controller.

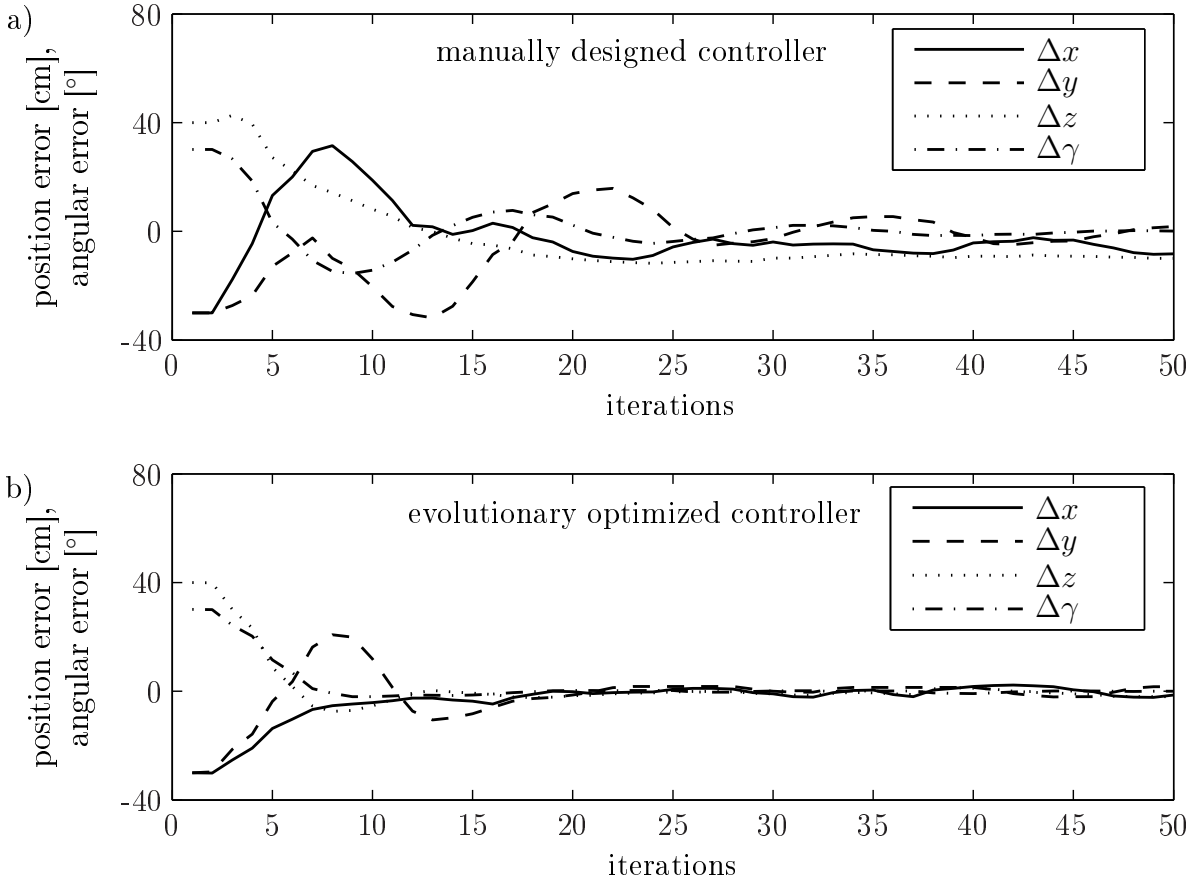


Figure 5.7: Comparison of the error evolution in the work space for the empirically tuned (a) and evolutionary optimized (b) controller.

Figure 5.7 compares the empirically tuned PD controller with the evolutionary optimized controller in terms of regulation of the task error. For the degrees of freedom x, y and γ

the integrated quadratic control error and overshoot are reduced, whereas the regulation of the z-component is inferior. The quadratic control error of the empirically tuned controller is reduced from $F = 32$ to $F = 23$ according to equation 5.24 by means of the HIL optimization. The slower convergence is partly due to the coupling of the features for the x and y-component with motions in z-direction. The perspective camera projection causes a magnification of the image with approaching camera and therefore a simultaneous increase of the feature errors f_x and f_y. As the total costs depend on the aggregation of the individual errors, a slower convergence along the z-axis still leads to a reduced total quadratic control error.

5.3.2 Simulation and experimental results

The optimized 4 DOF visual controller is evaluated for a free moving camera in virtual reality and in experiments on a 5 DOF KATANA manipulator with an eye-in-hand configuration. For the evaluation of the decoupled visual features for 4 DOF visual servoing, the manipulator is dislocated from the reference pose by an initial displacement $\Delta x = 20\,\text{mm}$, $\Delta z = 40\,\text{mm}$ and $\Delta\gamma = 25°$. Substantially larger displacements are not feasible in the experiments due to the restricted workspace of the KATANA manipulator and the eye-in-hand constraint of keeping the object in view of the camera. Figure 5.8 compares the evolution of the task space error and the image error for the original coupled controller (a, b) and the decoupled controller (c, d). In case of the coupled controller the z-error introduces a dynamic shift of the current feature f_x. Although the image error itself is compensated after 12 iterations, the corresponding residual task space error in x remains until complete regulation of Δz after about 25 iterations. The decoupled controller eliminates the impact of v_z on the feature f_x such that image and task space error in the x-component converge simultaneously after 12 iterations. Even though there is no initial displacement along the y-axis, the inherent coupling with v_z induces an undesired motion in task space of $\Delta y \approx 5\,\text{mm}$. Again, the decoupled controller eliminates this disturbance and Δy is not effected by the motions v_x or v_z. The residual task space error of the decoupled controller is less than 0.5 mm for the position and 0.5° for the rotation.

The potential of decoupled visual servoing in the context of object manipulation in the context of service robotics is investigated by [81], which utilizes the proposed visual servoing for gripper-object alignment in conjunction with a subsequent grasping stage. The object manipulation is realized by a two-stage approach, in which the object-gripper alignment is achieved by the proposed visual servoing and subsequently a two-finger grasping strategy is applied in order to manipulate the object without slippage and damage. The approach is advantageous for service robot manipulation as it necessitates only one reference image of the pre-grasping pose and an approximate estimate of the object's weight. In order to demonstrate the effectiveness of the approach a textured object is successfully picked up, moved and released at a novel position and orientation in twenty consecutive trials without human intervention.

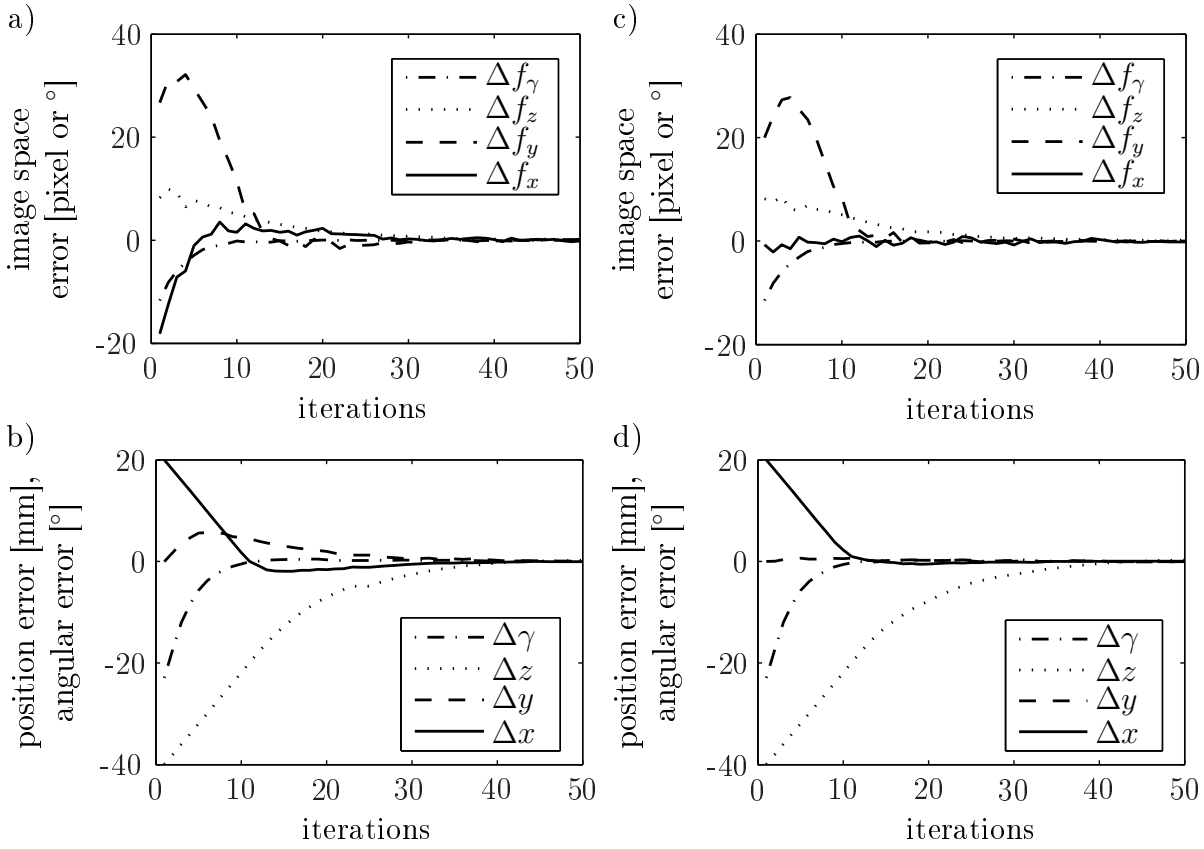


Figure 5.8: Image error (a) and position and angular error in task space (b) for visual servoing in 4 DOF; Image error (c) and position and angular error in task space (d) for decoupled visual servoing in 4 DOF.

5.4 Positioning in simulations in 6 DOF with augmented point features

Figure 5.9 shows the task space error for visual servoing for 6 DOF in a virtual reality simulation. Notice the substantial coupling among the degrees of freedom in the task space for the conventional controller (a). The translational motions in x, y and z demonstrate a significant overshoot caused by the couplings of the conventional centroids f_x, f_y with T_z, ω_α and ω_β. The task space errors in x and z initially increase and only start to converge after stabilization of the other errors. For the weighted centroid control (b) with the partially decoupled Jacobian the disturbances are significantly reduced and the six task space errors converge smoothly and largely independent of each other. The residual overshoot in the x- and y-components is caused by the remaining coupling with ω_β and ω_α, respectively. The results clearly demonstrate that the weighted features result in a more favorable task space motion of the camera.

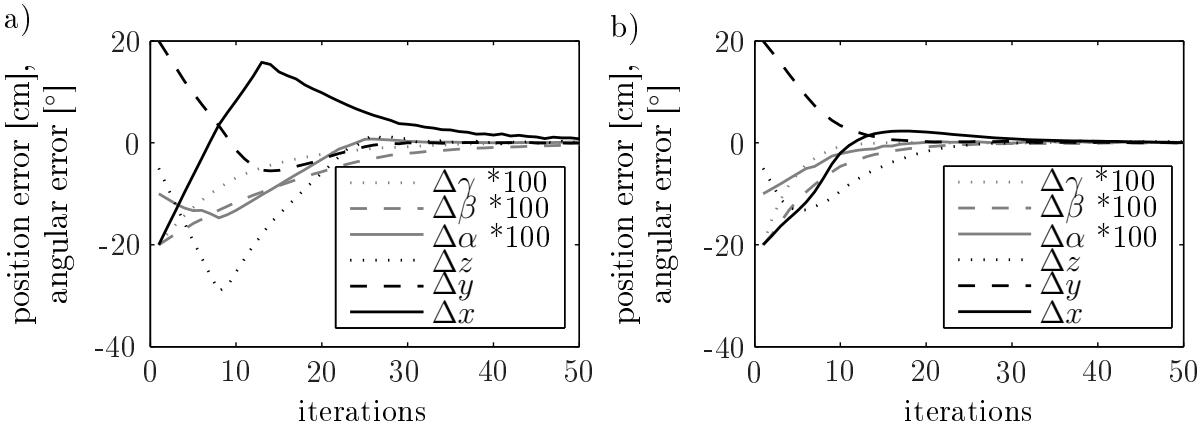


Figure 5.9: a) Position and angular error in task space for visual servoing in 6 DOF; b) Position and angular error in task space for decoupled visual servoing.

Experimental results are provided in the following section in a competitive analysis with an alternative approach. F

5.5 Alternative: Visual servoing on a virtual camera plane

The concept of a virtual camera plane is inspired by the rectification stage in stereovision. The virtual camera plane was first introduced by [105] in order to control the motion of a mobile robot independent of its gaze as explained in section 4.2. The main idea is to transform the features in the current view onto a virtual camera view, which is coplanar with the reference view in combination with the decoupling from section 5.2, cf. [113]. Figure 5.10 illustrates the geometric relationship between the reference, actual and virtual camera plane. The current camera frame exhibits a translational and rotational offset with respect to the reference camera frame. The origin of the virtual frame coincides with the current view, whereas its orientation is identical to the reference frame. The features in the current view are back-rotated onto the virtual camera plane, thus allowing the unbiased observation of the visual feature errors receptive to camera translations independent of the camera's orientation.

The control scheme for visual servoing on a virtual camera plane is detailed in table 5.2. The rotation between the actual view and the reference view is estimated by the proper decomposition of a homography into a rotational and translational part. The homography establishes a point to point transformation between two camera views for a set of features that lie on a plane according to equation 2.5. The estimation of the homography requires at least four feature correspondences, whereas for the sake of robustness and accuracy a two-

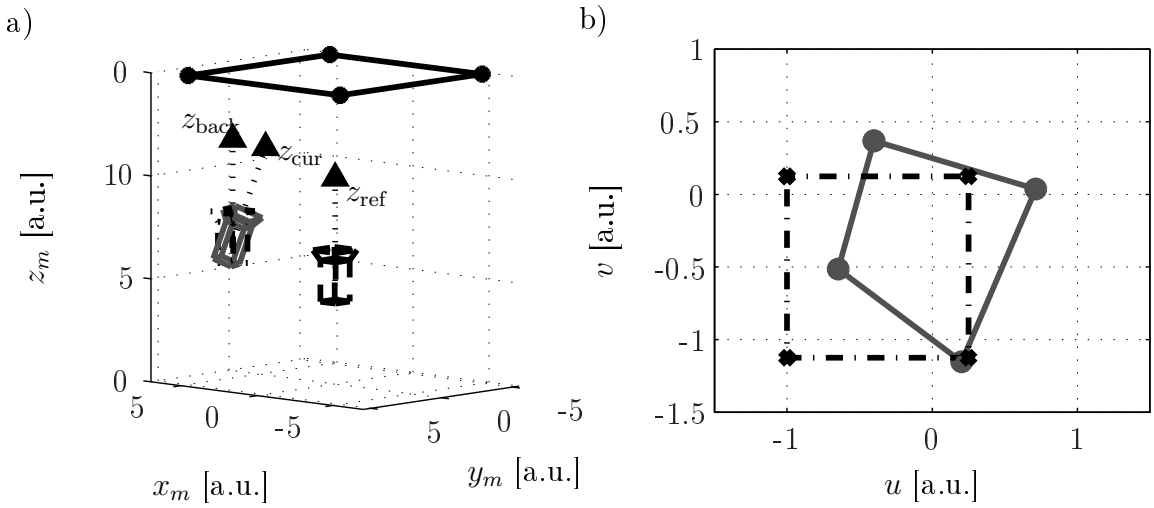


Figure 5.10: a) Camera configuration; b) Transformation of the actual image coordinates onto the virtual camera plane.

stage estimation process is applied for elimination of outliers by means of the **RAN**dom **SA**mple **C**onsensus algorithm (RANSAC) [49] and a subsequent least squares estimation. The homography is decomposed into the rotation matrix $\mathbf{R}$, the ratio $\mathbf{t}/d$ and the normal vector $\mathbf{n}$. It is assumed that the normal vector $\mathbf{n}$ of the feature plane in the reference configuration is roughly known in order to resolve the ambiguity of multiple possible solutions to equation 2.5. The rotational part of the image Jacobian only depends on the intrinsic camera parameters and is therefore independent of the camera pose. This property enables the homogeneous transformation of features from the actual to the virtual camera plane (step 2 in table 5.2) without effecting the feature error. As the virtual and reference frame are coplanar the residual feature error is solely attributed to the translational error in task space.

The estimated rotation constitutes the feedback signal to regulate the robot orientation in task space. The translational degrees of freedom are regulated by image moments similar to section 5.2. The third step consists of determining the moments f_x and f_y by the weighted centroid according to equation 5.10. The homogeneous transformation already accounts for the dependency on rotations around the x- and y-axis, respectively. The cost function F only contains a single Lagrange multiplier to achieve decoupling of the z-component:

$$F = \frac{1}{2}\sum_{i=1}^{n}\left(w_i - \frac{1}{n}\right)^2 + \lambda_i \sum_{i=1}^{n} w_i \hat{u}_{v_i}. \tag{5.25}$$

Note that equation 5.25 differs from the cost function introduced in equation 5.12 by replacing u_i by $\hat{u}_{v_i}$, thus allowing for completely decoupled visual servoing in 6 DOF. The cost function F is again minimized by computing the partial derivative of equation 5.25 with respect to w_i and the Lagrange multiplier λ_i, which yields the least squares solution stated in step 3 of the control scheme. In the fourth step the moment f_z is based upon the

average distance between two point features in order to regulate translations along z . The fifth step is optional as it is carried out only once for a camera manipulator setup in order to determine the optimal control parameters. Finally the controller setpoint is determined by means of the image moments f_x, f_y, f_z and the estimated rotations.

Table 5.2: Visual servoing on a virtual camera plane.

1. *Estimation and decomposition of the homography into rotation and translation:*

$$\hat{\mathbf{H}} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T \quad \Leftrightarrow \quad \mathbf{\Lambda} = \mathbf{U}^T(d\mathbf{R} + \mathbf{t}\mathbf{n}^T)\mathbf{V} \quad \text{(cf. section 2.1).}$$

2. *Homogeneous transformation of the actual image coordinates onto the virtual camera plane similar to equation 4.3 with an additional $\boldsymbol{T}_{\text{ext}}$ equal to the estimated rotation matrix $\boldsymbol{R}$ via:*

$$\begin{bmatrix} \hat{u}_v \\ \hat{v}_v \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{u} \\ \hat{v} \\ 1 \end{bmatrix}.$$

3. *Determination of the image moments f_x and f_y onto the virtual camera plane according to:*

$$f_x = \sum_{i=1}^{n} w_i \hat{u}_{v_i} \quad \text{with} \quad w_i = \frac{1}{n} - \frac{\hat{u}_{v_i}\overline{\hat{u}_v}}{\sum_{i=1}^{n}(\hat{u}_{v_i})^2},$$

$$f_y = \sum_{i=1}^{n} w_i \hat{v}_{v_i} \quad \text{with} \quad w_i = \frac{1}{n} - \frac{\hat{v}_{v_i}\overline{\hat{v}_v}}{\sum_{i=1}^{n}(\hat{v}_{v_i})^2} \quad \text{(cf. equation 5.10 to 5.14).}$$

4. *Determination of the image moments f_z onto the virtual camera plane according to:*

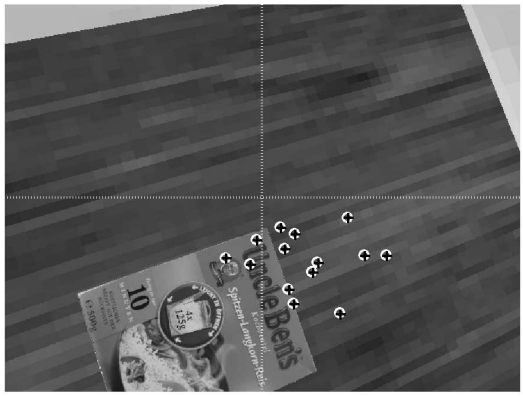
$$f_z = \left(\frac{n}{2}(n-1)\right)^{-1} \sum_{i=1}^{n}\sum_{j=i+1}^{n} ||p_j - p_i|| \quad \text{with} \quad p_{i,j} = [\hat{u}_{v_i}\ \hat{v}_{v_i}] \quad \text{(cf. equation 5.7).}$$

(5.) *Singular computation of the gains: HIL optimization of the controller by λ-CMAES (cf. section 5.3.1).*

6. *Determination of controller setpoint by means of the image moments f_x, f_y, f_z and the estimated rotations.*

Figure 5.11 compares the image location of features for a camera actually aligned with the virtual frame and the estimated location of features in the virtual image plane according to the current view. Despite the substantial translational and rotational displacement between both frames the estimated features back-rotated onto the virtual camera plane

only deviate from their actual positions by a small error of 1.6 pixel in the u-coordinate and of 1.4 pixel in the v-coordinate.



Δx	100 cm
Δy	30 cm
Δz	50 cm
$\Delta\alpha$	15°
$\Delta\beta$	10°
$\Delta\gamma$	20°
Δu	1.61 pixel
Δv	1.38 pixel

Figure 5.11: Virtual camera plane: White circles correspond to the original view, which is displaced but coplanar with the reference view. The black markers indicate the features transformed from the current view to the virtual plane by back-rotation by means of the estimated homography.

The visual servoing scheme on a virtual camera plane as well as the visual servoing with image moments using SIFT are evaluated in virtual reality and in experiments on a 6 DOF Reis manipulator with an eye-in-hand configuration. For the comparison of the two control designs the manipulator is dislocated from the reference pose in the virtual reality by an initial displacement $\Delta x = 40\,\text{cm}$, $\Delta y = 80\,\text{cm}$, $\Delta z = 60\,\text{cm}$, $\Delta\alpha = 5°$, $\Delta\beta = 30°$, $\Delta\gamma = -30°$. Figure 5.12 compares the evolution of the task space error for the original controller via image moments and the alternative controller based on the virtual camera plane. The undesired coupling of the visual moment f_x with the error in β results in an initial increase of the position error in x for visual servoing with image moments. Based on the remaining couplings the pose errors converge to zero under the condition that the other position errors are already eliminated. The results clearly demonstrate that the visual servoing with a virtual camera plane results in an even more favorable task space motion. The convergence is much faster as each error component converges independent of each other. Therefore the gains are tuned separately for each DOF, enabling a more efficient HIL optimization. The residual task space error of the visual control with the virtual camera plane is less than 0.5 mm for position and 0.05° for rotation, which is approximately one order of magnitude smaller than the visual servo control with image moments.

Notice, that the potentially incorrect correspondences are detected online for the visual servoing with decoupled image moments based upon the consistency of the main orientation of the individual SIFT features. In the case of the virtual camera plane, false and noisy correspondences are eliminated based on the robust estimation of the homography with RANSAC. The price for the increase in performance of the visual controller with the virtual

camera plane is the additional effort to properly calibrate the camera model. The scheme depends in particular on an accurate estimate of the transformation from the tool center point to the focal point.

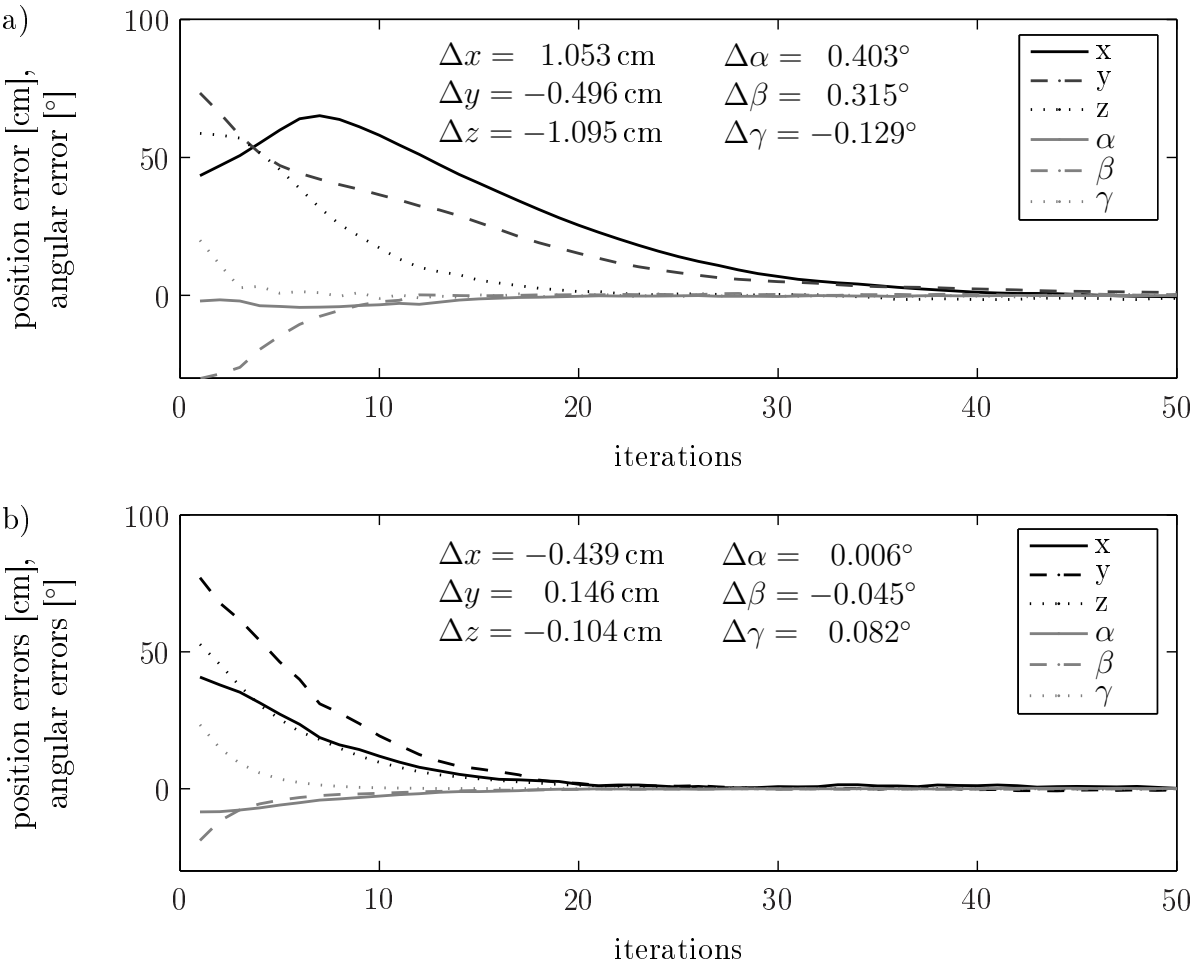


Figure 5.12: a) Task space error for visual servoing with image moments; b) Task space error for visual servoing using the virtual camera plane. The task space error in the virtual reality is determined by the feed-forward kinematics from the calculated joint angles.

Figure 5.13 compares the evolution of the task space error for both visual servoing controllers during experiments on the 6 DOF industrial robot. Notice, that the final accuracy which is achieved by the decoupled image moments using SIFT is superior to the visual servoing with the virtual camera plane. The larger residual error is due to the imperfect calibration of the transformation between the robot's tool center point and the camera frame whose origin is located in the focal point. However, the rate of convergence of the visual servoing on the virtual camera plane is substantially higher than for the controller with image moments. Achieving the level of accuracy demonstrated in the virtual reality on the real system requires a much more advanced and precise calibration.

5.6 Analysis and conclusion

This chapter presents a novel approach for visual servoing based on decoupled image moments using augmented point features such as SIFT features. The properties of local features render the approach universally applicable for manipulation of daily-life objects that exhibit texture. Local feature extraction (e.g. SIFT, ORB, SURF, GLOH, GF-HOG) for visual servoing applications offer the further advantage that object recognition and pose alignment of the manipulator rely on the same object representation. For 4 DOF visual servoing a set of completely decoupled image moments is derived that results in robust and independent convergence of the corresponding task space errors. Problems of the classical Jacobian based visual servoing scheme such as the camera retreat problem and local minima are resolved. A novel sensitivity matrix for 6 DOF visual servoing is introduced, which has only four off-diagonal coupled components between the visual features and the degree of motion. The visual control with the proposed methodology causes the pose errors to converge largely independent of each other resulting in a smoother task space motion of the camera. As an alternative to visual servoing based on decoupled image moments, the idea of the virtual camera plane for decoupled navigation and gaze control introduced in section 4 is transferred to the domain of visual servoing for object manipulation, named "visual servoing on a virtual camera plane".

Table 5.3 summarizes the main characteristics of the two controller designs. Contrary to visual servoing on a virtual camera plane, visual servoing with image moments requires neither an intrinsic nor an extrinsic calibration for transformation between the robot's tool center point and the camera frame. Additionally, for achieving its high performance, visual servoing on a virtual camera plane partially reconstructs the scene by means of the homography, whereas the estimation of the rotation matrix necessitates the knowledge of the normal vector $\mathbf{n}$ on the object's surface in the reference view. This method has the advantage that no off-diagonal couplings between the visual features and the degree of motion remain. Four non-collinear features are required to estimate the homography. Visual servoing with image moments needs only three features that span a large area around the focal point. Visual servoing on a virtual camera plane slightly outperforms visual servoing with image moments in terms of position accuracy and convergence, however, under the condition of proper intrinsic, extrinsic calibration and decomposition of the estimated homography. As the performance increase is not justified by the additional effort, visual servoing with image moments is preferable due to its model- and calibration-free design and its efficient implementation. Following this argumentation, the object manipulation with the Katana arm presented in [81] utilizes the proposed visual servoing with decoupled image moments for gripper-object alignment.

The next chapter describes global visual servoing. Due to the limited visibility and perceptibility of features across different views, it becomes necessary to introduce additional intermediate reference views to navigate across the entire view hemisphere.

Table 5.3: Characteristics of the two different visual servoing controllers, visual servoing on a virtual camera plane (cf. table 5.2) versus image moments (cf. table B.1).

	Visual servoing with image moments	Visual servoing on a virtual camera plane
intrinsic calibration	-	required
extrinsic calibration	-	essential for performance
model knowledge	-	normal vector $\mathbf{n}$
remaining couplings	4	0
minimal number of features	3	4
performance	very good	excellent

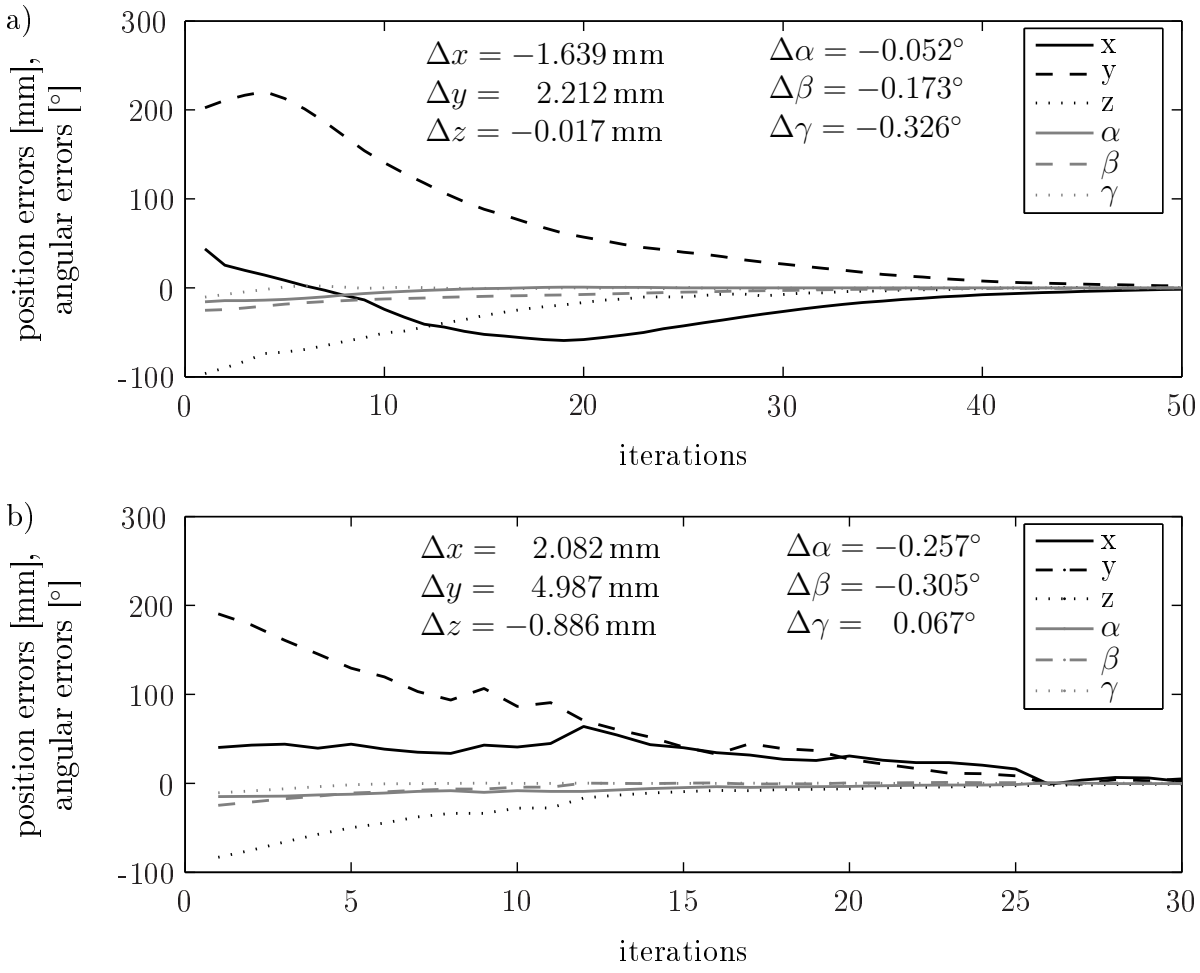


Figure 5.13: a) Task space error for visual servoing with image moments during experiments on a 6 DOF industrial robot; b) Task space error for visual servoing using the virtual camera plane. The task space error for the 6 DOF industrial robot is determined by the feed-forward kinematics from the measured joint angles.

Chapter 6

Global visual servoing with dynamic feature sets

This chapter presents a novel approach to global visual servoing in the context of object manipulation, also stated as large view visual servoing in [110]. In many scenarios the features extracted in the reference pose are only perceivable across a limited region of the work space. The limited visibility of features necessitates the introduction of additional intermediate reference views of the object and requires path planning in view space. Figure 6.1 depicts exemplarily such a scenario in which the features from the current pose and the reference pose do not intersect. The visual control is based on decoupled image moments using augmented point features such as SIFT features [109, 64] as defined in section 5.2. The approach is generic in the sense that the control operates with a dynamic set of feature correspondences rather than a static set of geometric features. The additional flexibility of dynamic feature sets enables flexible path planning in the image space and online selection of optimal reference views during servoing to the goal view. The time to convergence to the goal view is estimated by a neural network considering the residual feature error and the quality of the feature distribution. The transition among reference views occurs on the basis of this estimated cost which is evaluated online based on the current set of visible features. The dynamic switching scheme achieves robust and nearly time-optimal convergence of the visual control across the entire task space. The effectiveness and robustness of the scheme is confirmed in a virtual reality simulation and on two different experimental setups on industrial robot manipulators with an eye-in-hand configuration [85].

Following the model-free paradigm of this thesis already perused in visual servoing with decoupled image moments as well as visual navigation, the global visual servoing scheme is designed without any object models contrary to e.g. [142]. Optimal motion control for visual servoing to a static reference view has been discussed in chapter 5, whereas this chapter addresses the issue of global visual servoing with extraction and matching of dynamic sets of SIFT features. The view space is partitioned by an entire set of intermediate,

Figure 6.1: Application example for global visual servoing: start configuration (left); Disjunctive feature distribution in current and reference view (centered); Goal pose (right).

partially overlapping reference views of the object. The authors in [97] integrate a path planner in the image space with a visual controller based on potential fields in order to obtain visual navigation for large displacements. The work in [123] extends these concepts by qualitative visual servoing based on objective functions that capture the progression along the path, the feature visibility and camera orientation. This chapter provides a contribution to optimal path planning in the image space considering the residual feature error in conjunction with the quality of the feature distributions in alternative reference views. The additional flexibility of dynamic feature sets allows for adaptive online switching among reference views while navigating towards the goal view.

The chapter is organized as follows: Section 6.1 provides a stability analysis motivated by the feature distribution in the image space. Due to the limited feature visibility across different views it is necessary to introduce intermediate reference views. Time-optimal reference selection to accomplish global visual servoing is introduced in section 6.2. Navigation in image space is described in section 6.3. Section 6.4 demonstrates simulations in virtual reality on a sphere as well as on industrial robot arms and analyzes the convergence behavior of alternative switching strategies. An alternative to global visual servoing is introduced in section 6.5. Within a two-stage approach first a model-free pose estimation with viewpoint interpolation for a look-then-move strategy is applied, followed by local visual servoing close to goal pose. The chapter concludes with a summary in section 6.6.

6.1 Stability analysis depending on feature distribution

The local stability of the visual control loop requires that the feature error has a unique minimum at the reference pose. Even though a single SIFT feature suffices in principle

for coupled 4 DOF visual servoing, the computation of weighted centroids requires at least two non-coincident point features for decoupled 4 DOF visual servoing. Visual servoing in 6 DOF depends on at least three non-collinear features. Convergence of the control to the reference pose is achieved under the assumption of continuous visibility and perceptibility of this minimum number of correspondences. As stated in [48], three feature points which ideally form a large-area triangle enclosing the origin are optimal for visual control. Three features are minimal as the distortion in features f_α and f_β is observed relative to the average length between the points. However, not all configurations of three feature points are suitable for control. Stable visual control of the rotations requires that the three features are widespread and that the formed triangle encloses the origin. A too small separation of the three point features causes a change of sign in the moments f_α and f_β resulting in an unstable control. Figure 6.2 illustrates this phenomenon as it shows the point distributions for five triangular sets of different separation (figure 6.2 a) and the corresponding variation of the moment f_α for the five sets with respect to rotation about the x-axis (figure 6.2 b). In case of the widespread feature set the feature error f_α has a unique root at the origin [111]. However, the feature set closest to the origin induces two roots of f_α with non-zero rotational error to the left and right of the origin. These additional roots cause the visual control to converge to an equilibrium state that differs from the reference pose. Figure 6.3 shows the development of the feature moment f_x during a lateral movement for

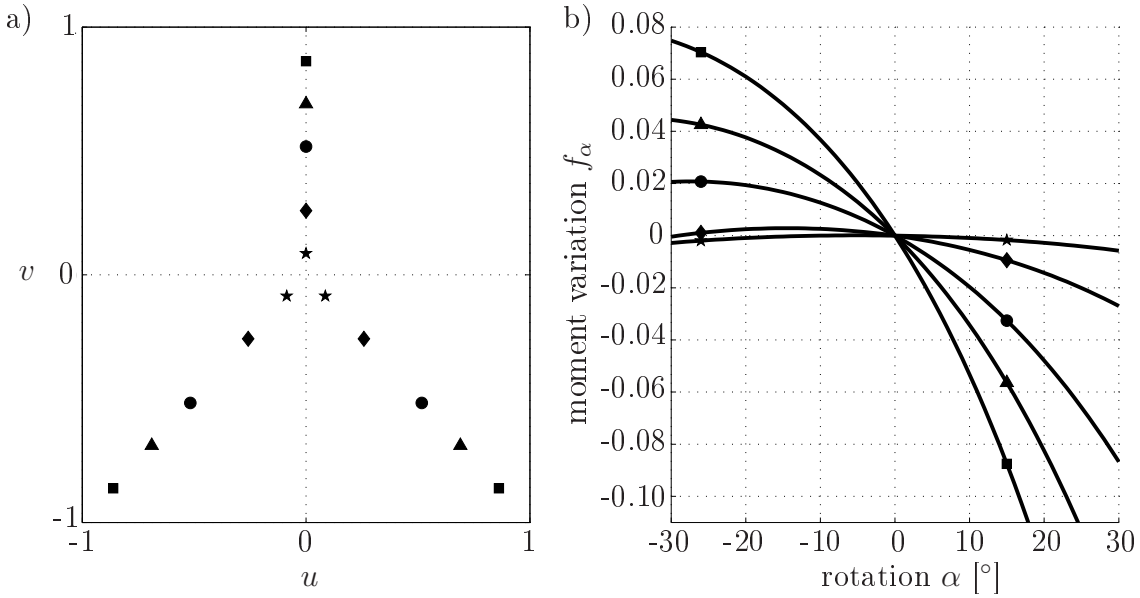


Figure 6.2: Feature distribution in image plane (a) and impact on rotation moments (b).

a randomly chosen subset of features. The feature configurations depicted in the insets of the figure demonstrate the effect of an extreme feature occlusion on the calculated moment. The configurations that are used for all other displayed moment developments represent feature occlusions with a randomly changing distribution in the image plane as well as in the number of features. The camera is laterally displaced by -40 cm to +40 cm

while the features at a distance of 75 cm are projected onto a normalized image plane. The figure demonstrates the impact of feature occlusions on the visual moment. The two envelopes marked by triangles and rectangles correspond to the extreme, but highly unlikely scenario in which all features in either the left or right half-plane are occluded resulting in a highly asymmetric configuration. The dotted lines correspond to random feature occlusions. In all cases the unique equilibrium point is globally stable. In case of the two extreme distributions the weighted feature moment does not evolve monotonically with the lateral displacement, due to the effect of skewed weights which increase in absolute magnitude with the asymmetry of the feature distribution. Even though this phenomenon effects the rate of convergence global stability is still guaranteed.

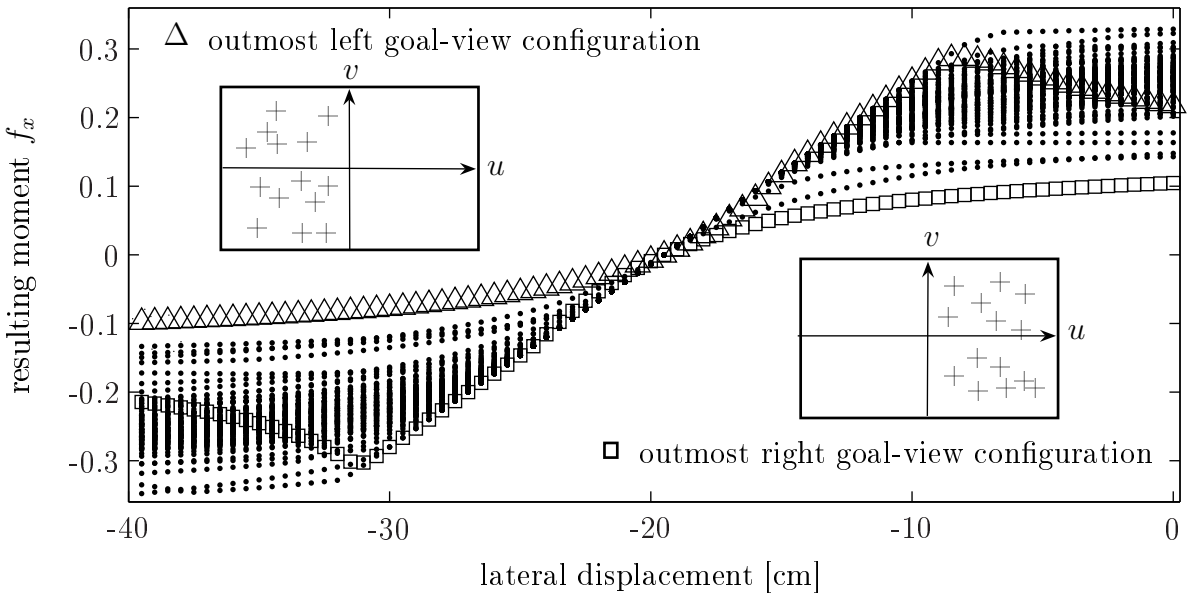


Figure 6.3: Feature distribution in image plane (insets) and impact on rotation moments.

In contrast to [48] the presented approach does not select the subset of optimal features online, but rather utilizes all available features matched between the current and the reference view in order to maximize robustness and accuracy. The general definition of visual features in terms of statistical moments renders the scheme robust with respect to occlusion or partial loss of perceptibility of features. Notice, that the reference features are recomputed online with respect to the subset of matched features. Typically, the number of matched features varies between 5 and 40, depending on the camera pose and the amount of useful texture in the current image. The visibility of individual features is limited by the camera's field of view, occlusion by the object and changes in perspective. Therefore, global visual servoing requires multiple reference images in order for the camera to navigate across the entire view hemisphere. Intermediate reference images are captured across the entire work space in x-, y- and z-direction. It is assumed that the object always remains in view of the camera, which naturally restricts the orientation of the camera along the x- and y-axis. The point of departure is a set of overlapping intermediate reference views

with partially shared features among neighboring images. The objective of this thesis is to generate a time-optimal and robust visual control across the entire task space by proper switching among neighboring reference images. For that purpose, the cost of the current view is compared with respect to all overlapping reference images, and the control switches to the reference image with minimal cost. A crucial step is to estimate the cost in terms of the time to reach the reference pose from the feature error and geometric configuration of features. Based on the estimated cost the optimal path is determined by shortest path graph search.

6.2 Optimal reference image selection

For global visual servoing intermediate views are defined to navigate across the entire view hemisphere. It becomes desirable to switch between intermediate views in a stable, robust and time-optimal manner. The cost in terms of number of control cycles to converge from the current view to the reference image is estimated in order to compute the optimal path. Crucial for this purpose is the proper definition of performance criteria for approximation of the cost function and the analysis of their correlation with the cost [110, 111]. In this case, an artificial neural network learns the relationship between control criteria and costs in a supervised manner. The training data is obtained from observations of the actual number of control cycles required for transitions between neighboring reference views.

6.2.1 Control criteria

Feature error: The overall feature error $\Delta\mathbf{f}(\mathbf{I})=[\Delta f_x,\ \Delta f_y,\ \Delta f_z,\ \Delta f_\alpha,\ \Delta f_\beta,\ \Delta f_\gamma]$ constitutes the most significant performance criterion for the estimation of the cost. A single feature error alone does not provide a good estimate of cost, because the actual time to convergence depends on the feature error with the slowest task space motion, usually associated with the translational degrees of freedom. The rotational errors are bounded by the visibility constraint and are usually stabilized within a few control steps. Each element of $\Delta\mathbf{f}(\mathbf{I})$ is normalized to the interval $[0, 1]$ according to its maximum range. The total feature error is the sum of normalized errors

$$\Delta\hat{f}(\mathbf{I}) = \sum_{i=1}^{6} \left|\Delta\hat{f}_i(\mathbf{I})\right|. \tag{6.1}$$

The feature error already attributes to a substantial amount of variation in the cost, nevertheless the cost estimate is improved by inclusion of additional criteria that capture the quality and robustness of visual control.

Number of correspondences: The robustness and the control performance increase significantly if more than the minimal number of correspondences is established. The

redundancy of multiple features reduces the noise level and contributes to the beneficial widespread dispersion of features in the image space. A small number of features might cause a compact distribution of point features, which causes poor or even unstable control in the image space as shown in section 6.1. The number of matched features also provides an estimate of the geometric distance of the current view to the reference pose. Distant poses only share a subset of mutually visible features, whereas the number of correspondences naturally increases with the proximity of both viewpoints. The criterion $C(\mathbf{I}) = n$ is defined as the absolute number of feature correspondences between the current and the reference view. The criterion

$$C_n(\mathbf{I}) = \begin{cases} 0 & n < n_{\min} \\ \frac{n}{n_{\max}} & n_{\min} < n < n_{\max} \\ 1 & n_{\max} < n \end{cases} \tag{6.2}$$

normalizes $C(\mathbf{I})$ as it requires a minimal number of features $n_{\min}$ and saturates at the upper limit $n_{\max} = 40$ at which no further improvement of the control performance is observed. The parameter $n_{\max}$ is independent of the object and not crucial for approximate cost estimation. The absolute number of visible features alone is not a unique indicator of the expected cost as it also depends on the distribution of these features defined in terms of their entropy and variance around the centroid.

Entropy: Entropy measures the order or disorder in a distribution. Therefore two control criteria are introduced, whereas $E_u(i)$ and $E_v(i)$ capture the distribution along the two axes of the image coordinate. The image is partitioned into $N = 10$ vertical and horizontal equally spaced columns and rows. The entropy along the two axes is calculated as

$$E_u(I) = -\sum_{i=1}^{N} H_u(i) \log_N \left(H_u(i)\right) \tag{6.3}$$

$$E_v(I) = -\sum_{i=1}^{N} H_v(i) \log_N \left(H_v(i)\right) \tag{6.4}$$

in which $H_u(i)$ and $H_v(i)$ denote the relative frequency of features in the i-th column, respectively row. The entropy assumes a value in the interval $[0, 1]$, in which a high entropy indicates a uniform distribution. A low entropy reveals an inhomogeneous distribution, which harms the robustness and speed of convergence of visual servoing.

Centroid location: A concentration of the feature points at the image borders bears the inherent risk of loss of features for small camera rotations. The visual features f_α and f_β require a distribution uniformly centered around the principal point in order to capture the distortion of line segments. The deviation of the feature centroid from the origin is expressed by

$$|\bar{u}| = \sum_{i=1}^{n} |\tfrac{u_i - u_0}{n}|, \quad |\bar{v}| = \sum_{i=1}^{n} |\tfrac{v_i - v_0}{n}| \tag{6.5}$$

in which low values represent desirable feature distributions.

Variance of the feature distribution: The variance of the feature positions provides an additional estimate of the quality of the feature distribution. A low variance in particular in conjunction with a dislocated centroid reflects a feature distribution that is suboptimal for visual control and delays the convergence to the reference image. The variances are computed as

$$\sigma_u = \sum_{i=1}^{n} \frac{(u_i - \bar{u})^2}{n}, \quad \sigma_v = \sum_{i=1}^{n} \frac{(v_i - \bar{v})^2}{n}. \tag{6.6}$$

Notice, that entropy reflects the geometric homogeneity of the feature set, whereas variance captures its width.

Correlation between performance criteria and time to convergence: Control experiments from 150 initial positions randomly distributed over the task space are recorded in order to evaluate the correlation between the performance indicators and the time to convergence. Each control step of the individual runs constitutes a training sample for supervisory learning of the neural network. A control run is considered as successfully converged to the reference image if all image errors are reduced to within 10% of their average initial value. The correlation between the performance criteria and the actual time to convergence provides insight into the influence and relevance of the individual indicators. The linear dependency between two stochastic variables X and Y is computed according to Pearson's correlation coefficient:

$$r_{XY} = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}}, \tag{6.7}$$

which assumes values in the interval $[-1, 1]$. $\bar{X}$ and $\bar{Y}$ are the first-order moments of the stochastic variables. Large absolute values indicate strong correlation between the two quantities. However, a correlation coefficient value of zero would demonstrate that no prediction of the costs based on the chosen control criteria can be done. Table 6.1 specifies the correlations between the performance indicators and the cost in terms of time to convergence.

Table 6.1: Pearson correlation between performance criteria and time to convergence.

	Δf_x	Δf_y	Δf_z	Δf_α	Δf_β	Δf_γ	$\hat{f}$	
r_{XY}	0.30	0.14	0.17	0.14	0.13	0.13	0.63	
	$C(I)$	$C_n(I)$	$E_u(I)$	$E_v(I)$	$\lvert\bar{u}\rvert$	$\lvert\bar{v}\rvert$	σ_u	σ_v
r_{XY}	-0.66	-0.72	-0.66	-0.72	0.44	0.32	-0.64	-0.62

Table 6.2: Training and test set error for neural network trained with feature error $\mathbf{f}(\mathbf{I})$ only and with feature error and performance criteria $\mathbf{f}(\mathbf{I}), \mathbf{c}(\mathbf{I})$. RMSE stands for root mean square error.

	RMSE train	RMSE test	correlation
$\mathbf{f}(\mathbf{I})$	0.0149	0.0297	0.75
$\mathbf{f}(\mathbf{I}), \mathbf{c}(\mathbf{I})$	0.0072	0.0092	0.96

The individual feature errors are only slightly correlated with the cost, whereas the normalized summed feature error $\hat{f}$ is indeed a proper indicator for the distance to the reference pose. Notice, that the relative number of matched features $C_n(\mathbf{I})$ correlates even more with the cost than the summed absolute errors $\hat{f}$. The scalar summed error contains less information than the entire error vector $\mathbf{f}(\mathbf{I})$. This is explicable, as the feature errors related to the translational degrees of freedom converge at a slower rate.

In order to predict the time to convergence two neural networks with different input features are trained with the data acquired during the 150 experimental runs. The multi-layer perceptrons are composed of 16 neurons in the hidden layer and are trained with the standard back-propagation algorithm. The first network only uses the six-dimensional feature error $\mathbf{f}(\mathbf{I})$ as input, whereas the second network in addition has access to the performance criteria $\mathbf{c}(\mathbf{I}) = [C_n(I), E_u(I), E_v(i), \bar{u}, \bar{v}, \sigma_u, \sigma_v]$. Figure 6.4 depicts the relation between the estimated costs on the x-axis and the true costs for the full input network. It also shows the linear regression for the partially and fully informed network. The neural network only trained with the feature error $\mathbf{f}(\mathbf{I})$ achieves a correlation of 0.75 between estimated and true cost. This correlation is substantially improved by incorporation of the additional performance criteria to a degree of 0.96. The improvement in prediction accuracy of the fully informed network error compared to the pure feature error based network is confirmed by the reduced training and test set error shown in table 6.2. This demonstrates that a distance metric to the goal view in the image space has a significantly lower correlation with the costs than $\mathbf{f}(\mathbf{I})$ in conjunction with the image distribution indicators $\mathbf{c}(\mathbf{I})$. This observation confirms the convergence analysis in section 6.1, namely that the feature distribution crucially affects the control performance. Furthermore the control features $|\bar{u}|$ and $|\bar{v}|$ only have a small leverage in order to improve the correlation between control criteria and costs. Finally a correlation of 0.96 is obtained using all defined control criteria.

6.3 Navigation in the image space

The approach neither requires a geometric model of the object nor is it aware of the spatial relationship between the reference views, nor does it perform path planning in the task

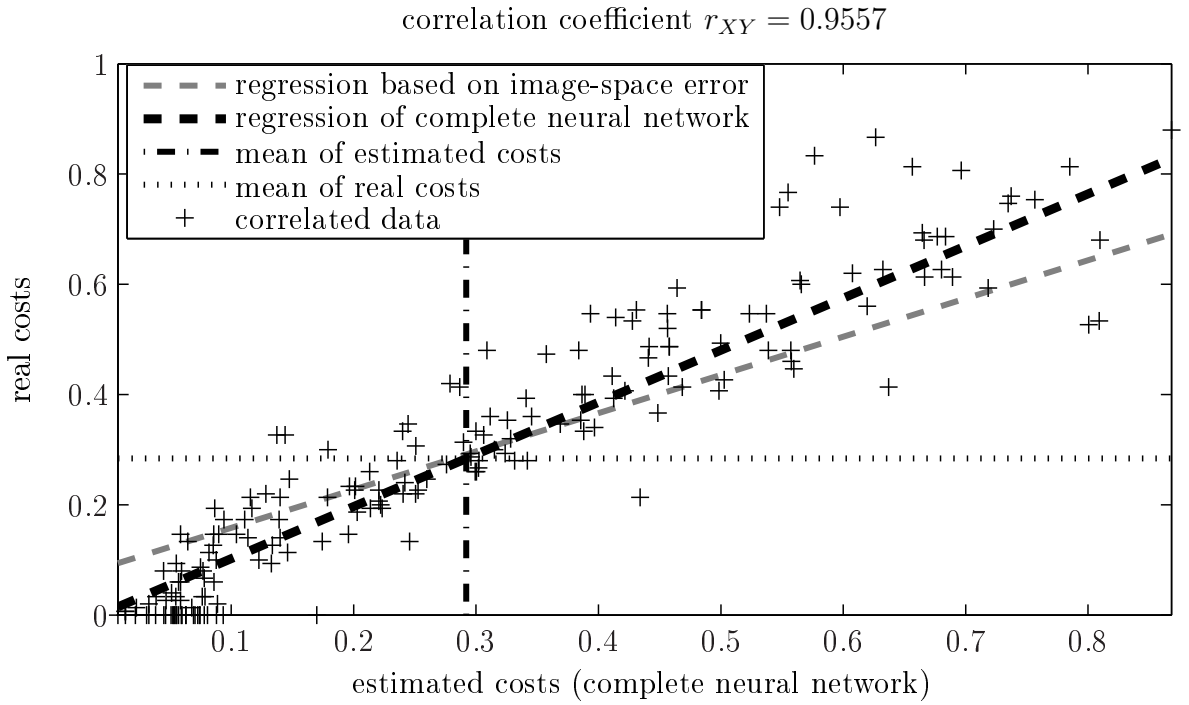


Figure 6.4: Neural network (NN) estimate versus true cost (+) and regression lines for the neural network with $\mathbf{f}(\mathbf{I})$ as input (grey dashed line) and $(\mathbf{f}(\mathbf{I}), \mathbf{c}(\mathbf{I}))$ as input (black dashed line).

space. The optimal path is planned online in the image space rather than in the task space. For that purpose each reference view (RV) represents a node in an undirected graph, in which edges define neighborhood relationships between overlapping views. The cost of an edge connecting two views reflects the transition time between the views expressed in terms of number of iterations to converge from the initial view to the neighboring view. The graph supports the global initial path planning from the start view to the desired goal view, but it also forms the basis for the decision when to switch to the next reference view. The cost estimation within the path planning consists of two major steps, an off-line computation of graph costs between the reference view and an online computation of the cost from the current view to the overlapping reference views. The planner switches between reference views based on a comparison of the accumulated costs of currently feasible reference views.

Initial path planning and cost estimation: The initial cost estimation is based upon the graph constructed from the complete set of reference views which form its nodes. The number of matching features is computed for every possible pair of reference views. An edge is generated between two overlapping views if they share five or more common features. The cost of an edge is estimated by evaluating the set of corresponding features with the neural network described in the previous section. The optimal path from every reference view to the goal view is calculated with the well-known Dijkstra algorithm [40] for finding the shortest path in a weighted graph. This calculation is part of the teach-in-process

in which reference views are captured across the work space and is performed off-line in advance.

Current cost estimation and choice of optimal current reference view: The features extracted from the current view (CV) are continuously compared to those of

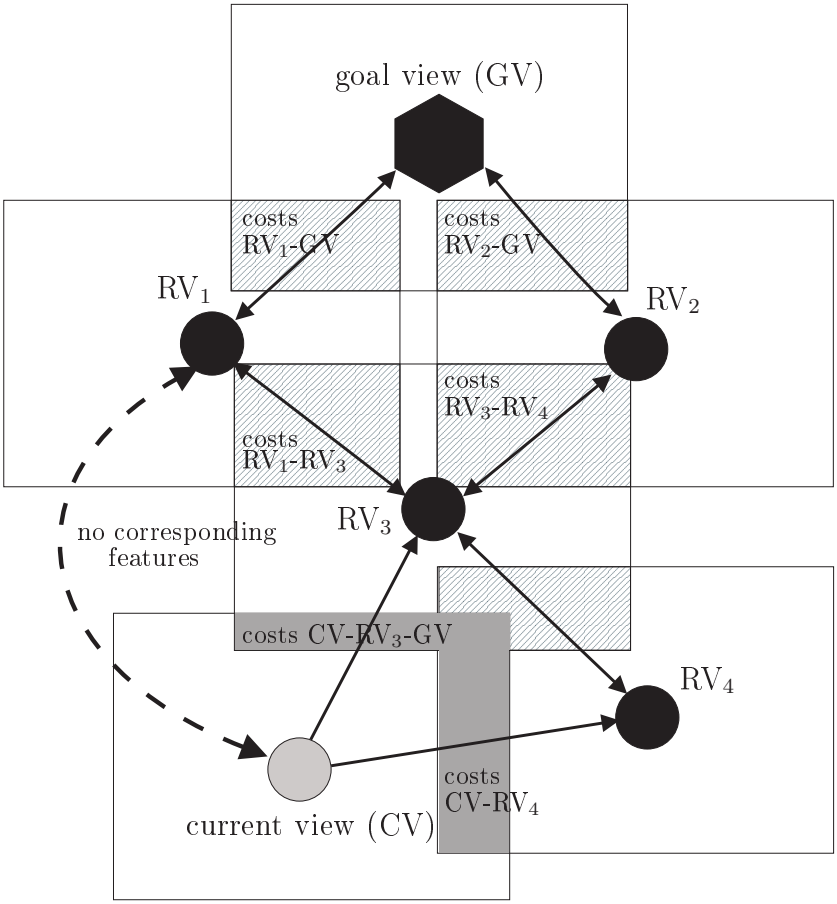


Figure 6.5: Reference-, goal- and current view represented by a graph.

overlapping reference views in order to identify the optimal current reference view online during control. For the potential reference views the time to convergence is estimated in the same way as for the initial generation of the graph. The total costs for reaching a specific reference view plus the already estimated cost for the shortest path from that node to the goal view are compared among all feasible views. The node with minimal cost is selected as the next reference view to be included into the shortest path to the goal. The view evaluation is only performed every fifth control cycle in order to reduce the amount of online computations. Figure 6.5 depicts a section of a graph generated from a set of images with four intermediate reference views $RV_1, \ldots, RV_4$, a goal view GV and the current view CV. The images associated with a view are visualized by rectangles, the hatched areas

represent the overlap between neighboring images which contain common SIFT features. The cost of the transition from the current view to the two feasible reference views RV_3 and RV_4 depends on the number and quality of common features in the grey areas. The current view has no connection to the reference views RV_1 and RV_2 because the subset of common features is empty, as indicated by the dotted line. A hysteresis in the switching scheme avoids the risk of the visual controller getting trapped in a limit cycle around the optimal switching point due to uncertainties in the cost estimate or fluctuations in the matched features. The initially estimated costs of the optimal path from the current view to the goal are weighted by the number of intermediate nodes from the candidate reference views to the goal node. That way, switching to a reference view whose node is closer to the goal node becomes more attractive, whereas the reverse switching to a more distant node is suppressed even if its estimated cost seems more attractive. A transition to a lower cost reference view is only initiated if its superiority is confirmed in two consecutive iterations, thereby gaining additional robustness with respect to cyclic switching.

6.4 Experimental results

The evaluation of global visual servoing is pursued in experiments within a virtual reality environment and on a real 5 DOF robotic arm with an eye-in-hand configuration [110], as well as on a 6 DOF industrial manipulator [85].

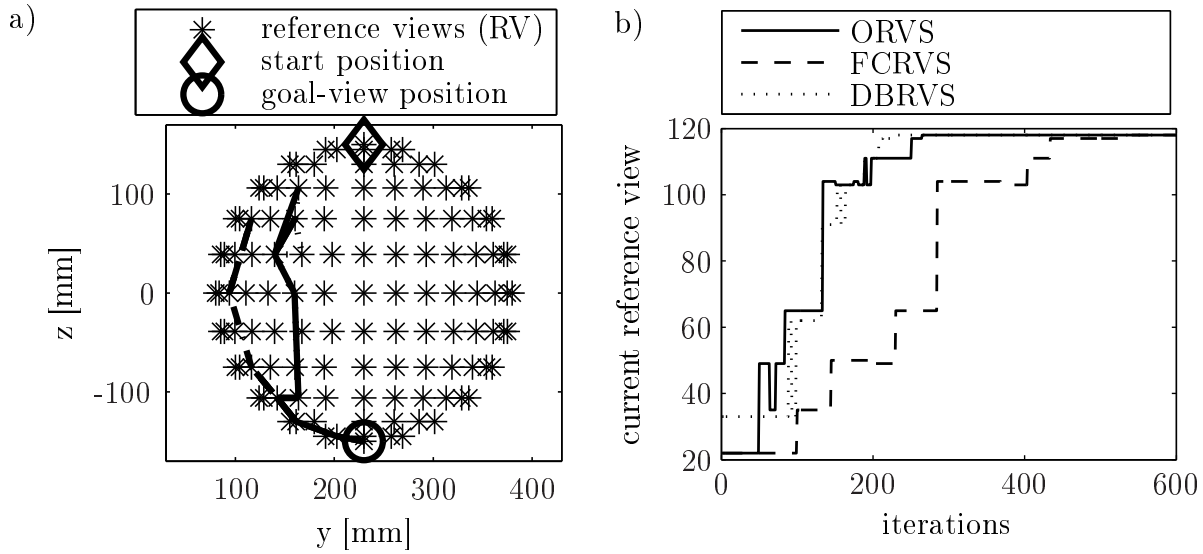


Figure 6.6: Alignment of reference views (RV) - a comparison of chosen sequences from pole to pole on a sphere (a) and as a function of iterations (b) for optimal (ORVS), fixed convergence (FCRVS) and distance-based reference view selection (DBRVS).

In both experimental setups the performance of the cost estimation based switching scheme is compared with two alternative methods. The first method, in contrast to the proposed scheme, assumes that the geometric distance in task space between reference views is known. Once the minimal number of visual features is perceived, it switches to the reference view closest to the goal pose. This switching strategy ignores the perceptibility and quality of the set of matched features and is not sufficiently robust from a control point of view. Nevertheless for the purpose of comparison it provides an upper performance limit. The second method computes an optimal static path that connects the start to the goal node based on the static costs. It is not opportunistic as it does not reestimate the costs online, or replans if other reference views not originally included in the plan suddenly appear more attractive. It switches to the next view outlined in the plan upon convergence of the feature error to a current reference view. This method, although suboptimal, is robust from a control point of view, but could still be improved by relaxing the convergence criterion without sacrificing robustness.

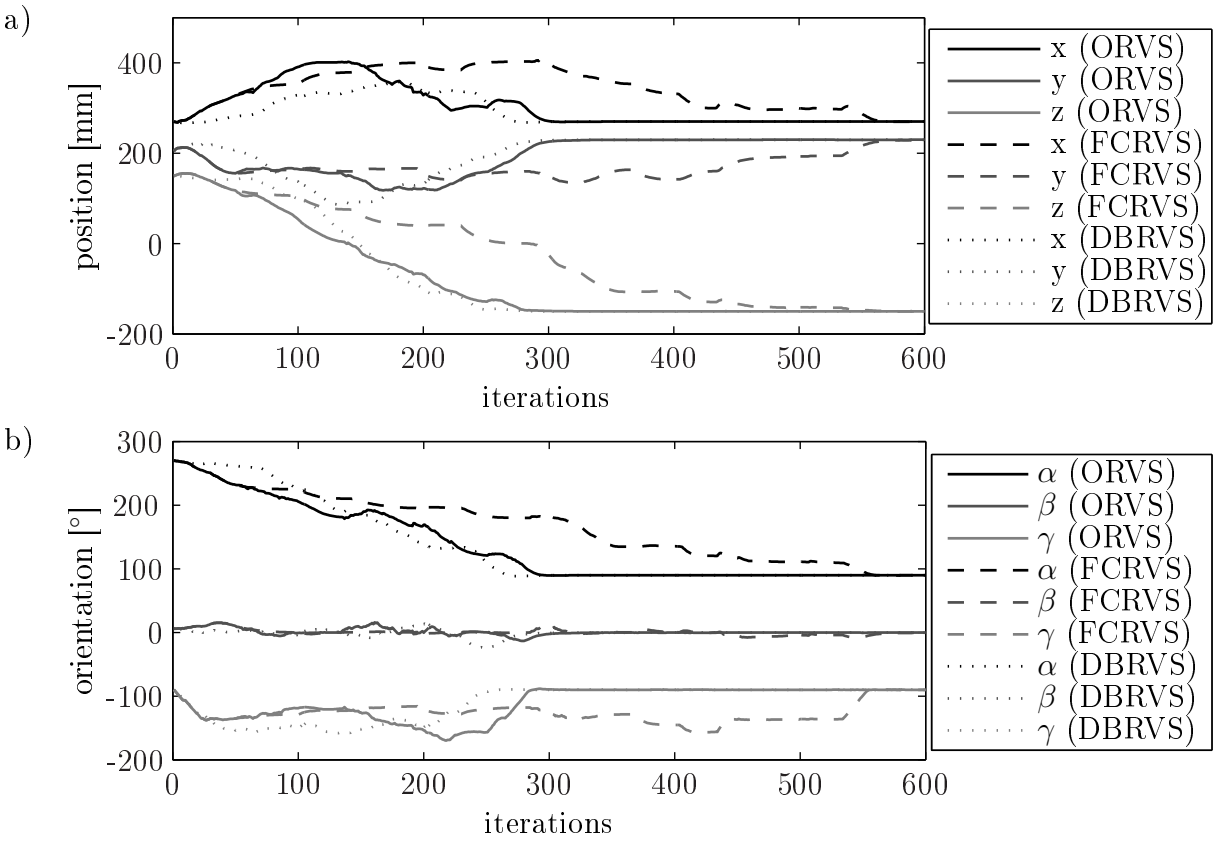


Figure 6.7: Pole to pole trajectories of the compared methods with respect to position (a) and orientation (b) for optimal (ORVS), fixed convergence (FCRVS) and distance-based reference view selection (DBRVS).

6.4.1 Navigation across a sphere within the virtual reality

A virtual reality simulation of a free moving camera allows the verification of the global visual servoing scheme without being constrained by the robot kinematics or workspace. The camera navigates in 6 DOF around a sphere textured with a schematic map of the globe. The reference views are equidistantly located along longitudes and latitudes. The task is to guide the camera visually from the north to the south pole. Figure 6.6 depicts the distribution of reference views together with the path pursued by the three methods under comparison. Even though the camera is initially located above the north pole, all schemes immediately transit to an initial reference view that is already closer to the goal. The distance-based method picks a different large circle route than the other two schemes as it ignores the issue of feature quality. A better rationale is to select the great circle route which guarantees perceptibility of a sufficient number of features for stable traverse to the south pole. This effect is termed the *Pacific problem*, as for the globe example, the equal-distant path either moving over America or Africa contains more features due to the texture and text on the continents than crossing the Pacific with sparse features. The right part of figure 6.6 compares the sequence and progression of reference views followed by the three alternative methods. Figure 6.7 shows the evolution of the task space error in terms of translation and rotation. The number of iterations until convergence is approximately the same for the optimal image-based and the distance-based navigation method. For the former the goal pose is reached within 300 iterations, for the later in about 290 iterations, whereas the static scheme with complete convergence takes about 560 iterations.

6.4.2 Navigation across a semi cylinder with a 5 DOF manipulator

The scheme is also evaluated in an experiment on a 5 DOF Katana robot with an eye-in-hand camera configuration. As the workspace of the manipulator is rather limited, the camera navigates across the inner surface of a semi cylinder with a circumference of 1.8 m and a height of 0.4 m. The inside of the semi cylinder is textured with a panoramic photo of the TU Dortmund campus shown in figure 6.8. This cylindric configuration is optimal with respect to the workspace of the robot as it allows a maximal number of sufficiently distinct reference views. The reference views form a 15×6 grid, horizontally separated by $10°$, vertically by 5 cm. The kinematics of the specific robot limit the camera motion to 5 DOF. At the start pose the camera points at the upper left part of the image and the goal is located in the lower right corner of the cylinder. As shown in figure 6.9, all methods follow at large a similar view sequence. The only significant deviation occurs halfway through the path in a region which mostly contains sky and ground and therefore few distinctive features. The optimal switching scheme takes a small vertical detour in order to exploit the higher concentration of features in the textured band between sky and ground.

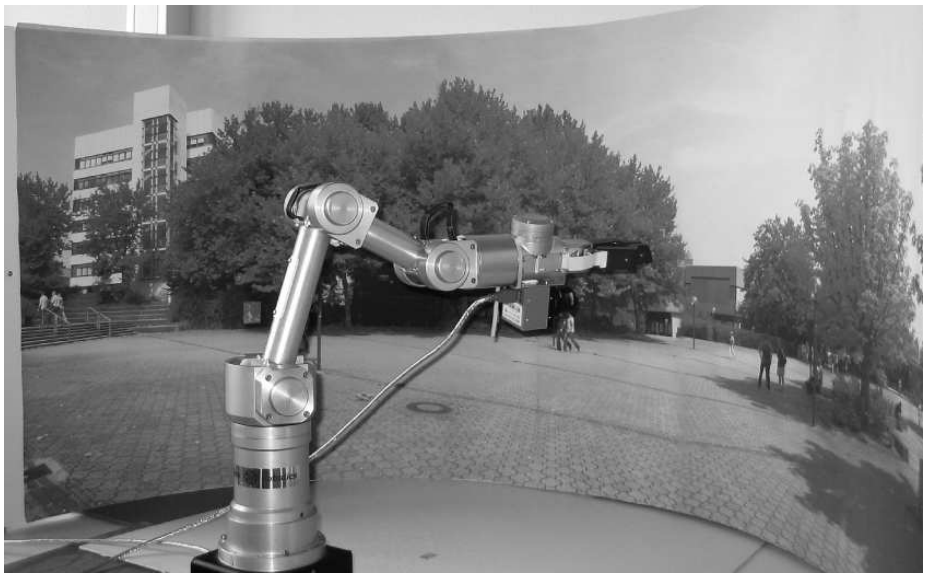


Figure 6.8: Experimental setup for visual servoing in 5 DOF on a semi cylinder.

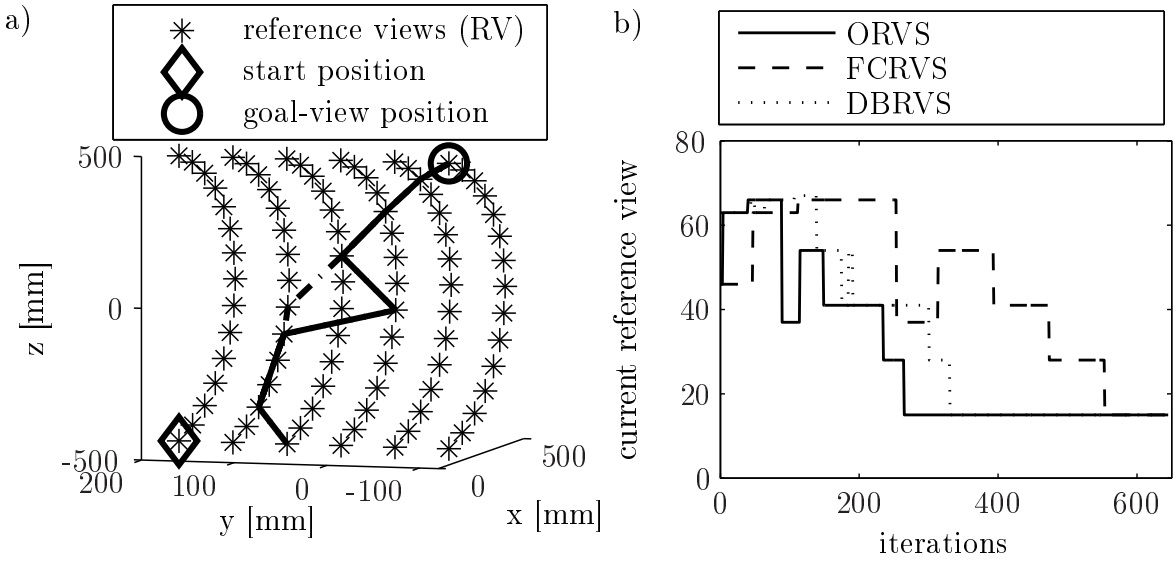


Figure 6.9: Alignment of reference views with the chosen sequences in space (a) and chosen sequences as a function of iterations (b) for optimal (ORVS), fixed convergence (FCRVS) and distance-based reference view selection (DBRVS).

The number of iterations until final convergence is about 300 for the optimal method, 400 for the distance-based approach and 600 for the fixed-convergence method. The difference in time to convergence results from the fact that the two other methods require a much longer time to traverse the region of sparse features as the visual control tends to become unstable due to the poorer quality of feature distributions.

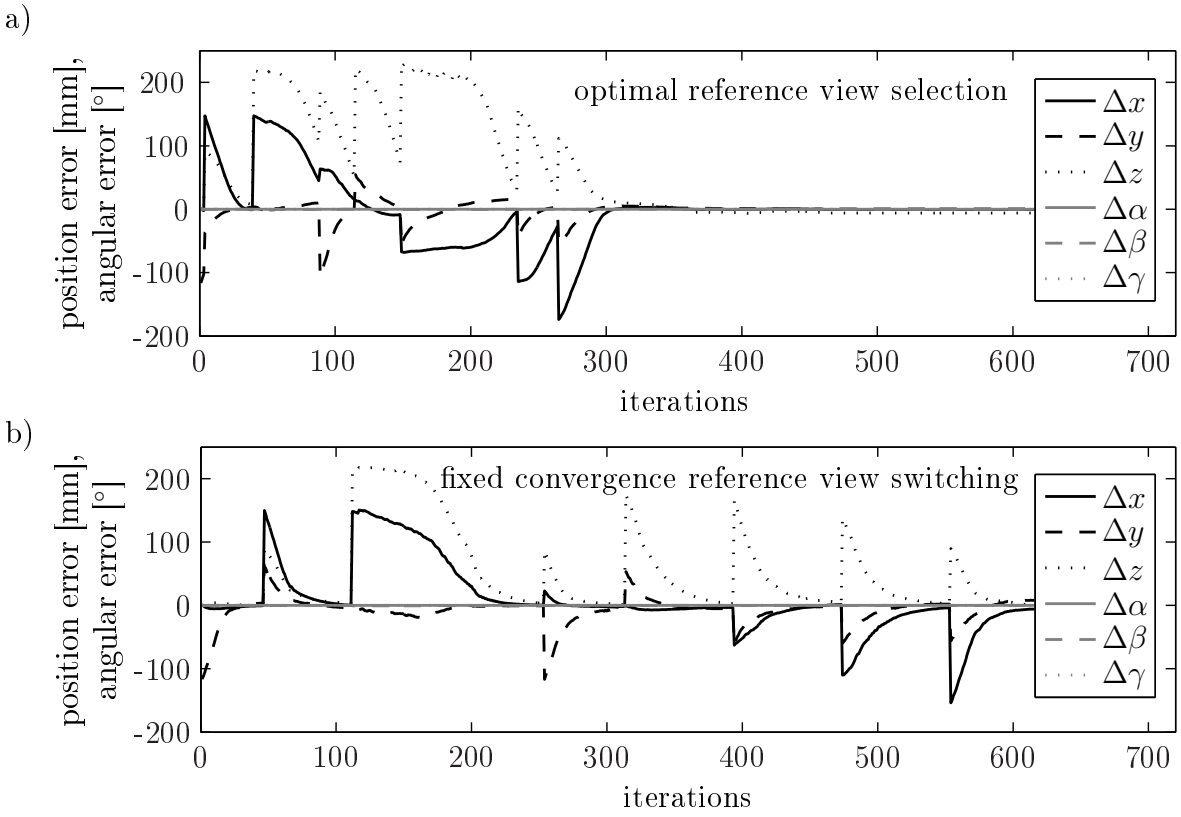


Figure 6.10: Relative task-space error for optimal reference view selection (ORVS, a) and fixed convergence reference view selection (FCRVS, b).

This observation is confirmed by an analysis of the evolution of the relative task space error with respect to the intermediate reference views shown in figure 6.10. Figure 6.10 a) depicts the progression of task space error and switching sequence for the proposed scheme, figure 6.10 b) for the static scheme. The static scheme wastes iterations in situations in which the feature error is already low but not yet fully converged. The optimal cost based scheme avoids delayed transition to the next reference view, as it already switches for substantially larger residual errors without compromising the stability of the control.

6.4.3 Navigation across a cuboid with a 6 DOF manipulator

The results of the experimental realization of image-based visual servoing for a 6-axial industrial robot are presented in the following. A comparison procedure serves for evaluating the time-optimal switching criterion by adjusting the goal position by means of a fixed sequence of reference images and a constant switching threshold. Initially, the static sequence is also generated by the optimal path planning. Figure 6.1 shows the experimental setup consisting of the industrial robot with an eye-in-hand camera configuration.

The robot drives from start to goal position relatively to the object via reference views arranged on a hemisphere around the object. Figure 6.11 shows the temporal devolution of the position error for both procedures. The sequences of reference views utilized in both cases are depicted in figure 6.12, on the left the spatial distribution and on the right the chosen switching points. The graphs in figure 6.12 clarify that the initial path planning is modified by the dynamical choice of references, thus allowing for reaching the goal position earlier. Therefore the control using the static switching criterion is slower: in spite of a time-optimal chosen sequence (and therefore reduced number of reference images) the goal position is only reached after 125 iterations, whereas the time-optimal control converges after around 100 iterations. The control is performed over three reference views, respectively, together covering an elevation angle of around 90°, whereas the dynamical procedure switches faster to the next view. Hereby the slightly different path planning is caused by the varying cost estimation of the dynamical choice of references. In the ideal case the time-optimal sequence corresponds to that with the shortest path in the work space. However, views with unfavorable configurations of features are avoided as they influence the robustness of the control and thus its velocity in a negative manner.

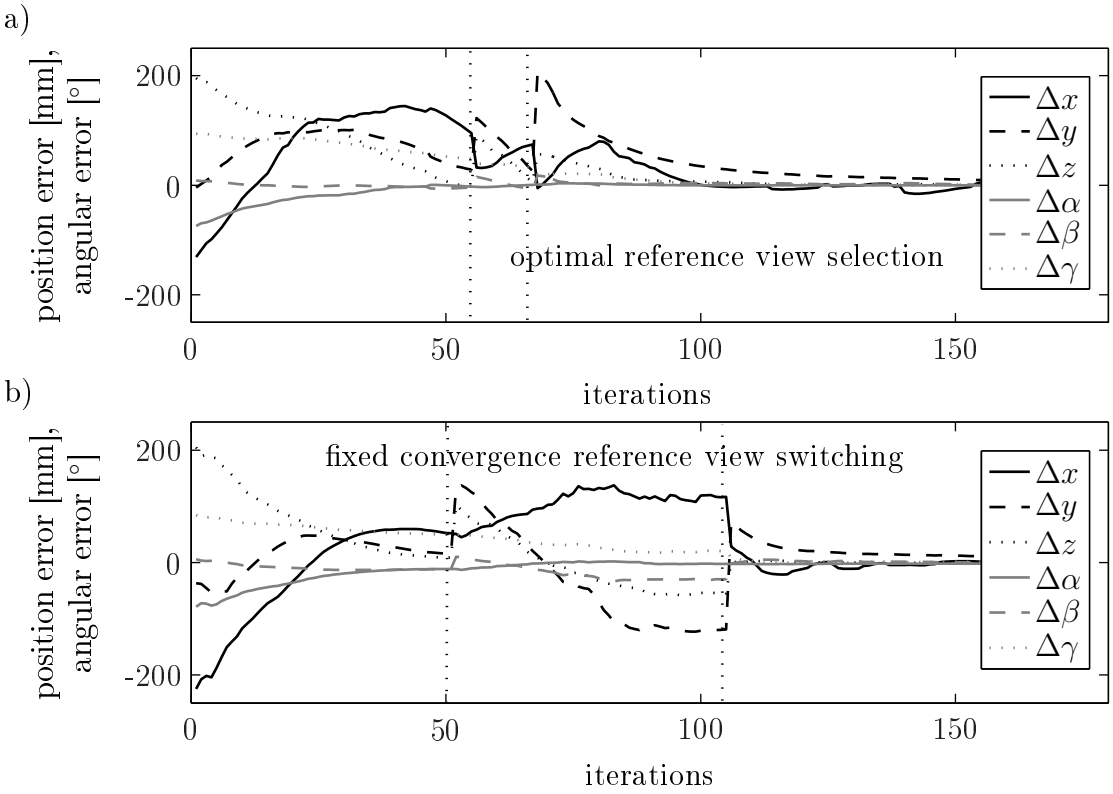


Figure 6.11: Comparison of the trajectories for time-optimized (ORVS, a) and static switching criterion (FCRVS, b).

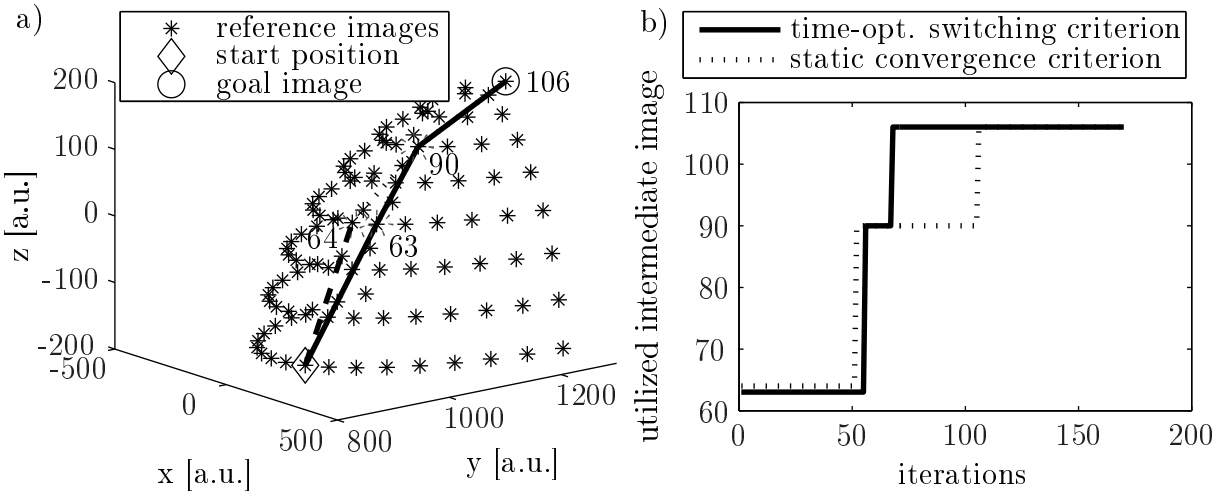


Figure 6.12: Chosen reference images and their pose in the task space and temporal dependence.

6.5 Alternative: Model-free pose estimation with local visual servoing

As an alternative to the presented global visual servoing a more human-like strategy for camera object alignment in the case of limited feature visibility is investigated. It consists of two stages: Initial pose estimation for pre-alignment of the camera relative to the object to speed up the positioning (open loop / look-then-move strategy) and subsequently a refinement by visual servoing using only one reference image instead of a graph of intermediate reference views. A difference to similar approaches in the literature is that these two stages rely on the same object representation. The plenoptic function is already sampled for the set of reference views required for the global visual servoing. This scheme follows the appearance based paradigm [58, 133, 43], such that there is no need for a geometric model and its image correspondences [38]. The approach employs an instance base learning scheme [7] in which the object pose is predicted based on the similarity of the current view with a set of reference views of known pose. Reference views are represented by a set of SIFT features extracted from the corresponding image. Primarily the similarity between reference views and the current view is established based on the frequency of SIFT features matched between both images. The pose is estimated by weighted averaging of the reference poses $[\theta_{\mathrm{az}_i}, \phi_{\mathrm{el}_i}]$ across the N most similar neighbor views so that the estimated azimuth $\hat{\theta}_{\mathrm{az}}$ and elevation $\hat{\phi}_{\mathrm{el}}$ are computed according to:

$$\hat{\theta}_{\mathrm{az}} = \sum_{i=1}^{N} \frac{C_r(\mathbf{I}_i)\theta_{\mathrm{az}_i}}{\sum_{j=1}^{N} C_r(\mathbf{I}_j)}, \quad \hat{\phi}_{\mathrm{az}} = \sum_{i=1}^{N} \frac{C_r(\mathbf{I}_i)\phi_{\mathrm{az}_i}}{\sum_{j=1}^{N} C_r(\mathbf{I}_j)} \tag{6.8}$$

in which the similarity $C_r(\mathbf{I}_i)$ is defined in terms of the absolute number of feature correspondences $C(\mathbf{I}_i)$ between the reference view and the current image divided by the absolute

number of SIFT features in the reference view.

Pose estimation by viewpoint interpolation: Figure 6.13 a) shows a test object in a top reference view and current view, together with the set of extracted SIFT features. Figure 6.13 b) depicts the relative frequency $C_r(\mathbf{I}_i)$ of matched features between the current pose indicated by an open circle and the reference views. The four most similar references

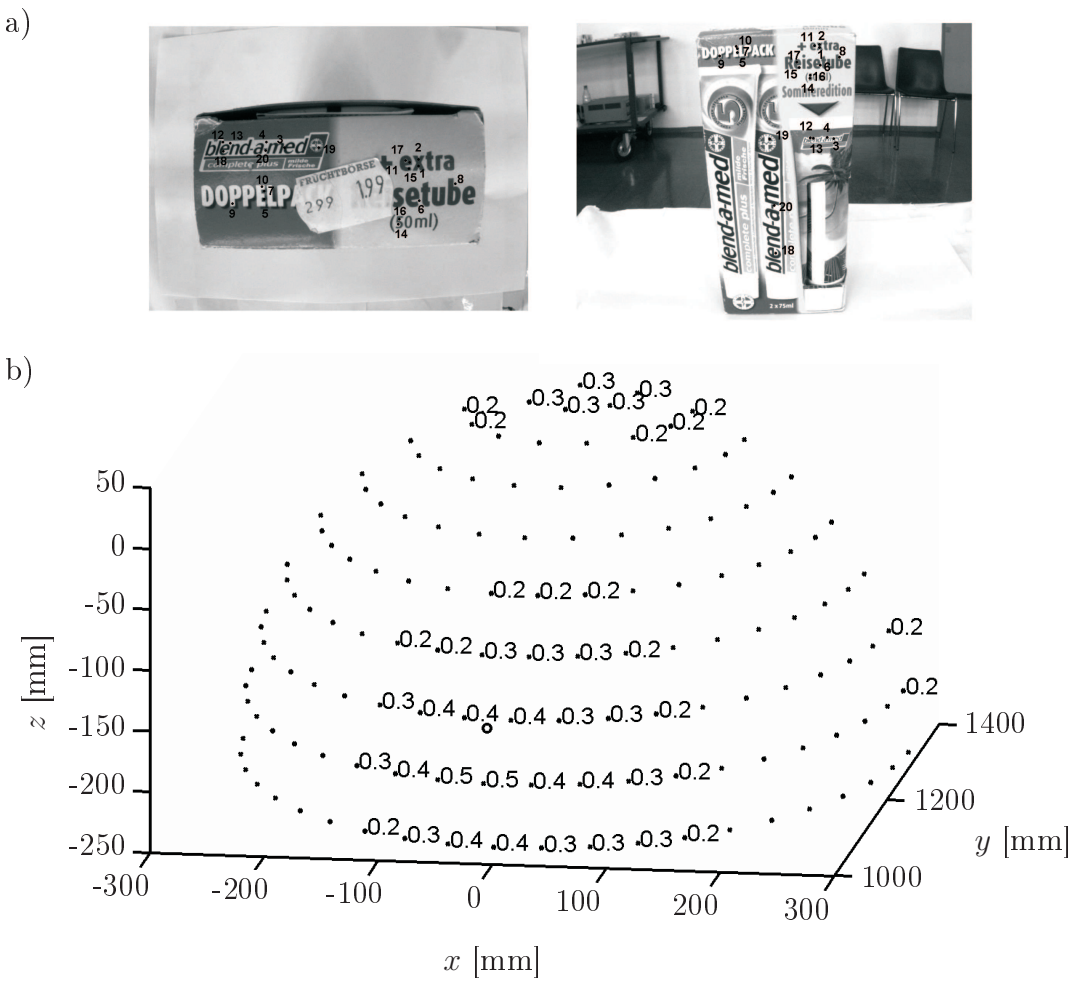


Figure 6.13: a) Confusion of SIFT features between frontal and the top view of the object caused by repetitive texture; b) Similarity based pose estimation according to the relative frequency $C_r(\mathbf{I}_i)$ of matched features. Only those reference views for which more than 15% of features are matched are labeled.

match between 40-50 out of the 100 features in the query view. This ratio drops with increasing distance of viewpoints on the hemisphere from the current viewpoint. Notice, that there is a second region of significant matches at the north pole. These matches originate from a replication of the advertisement text on the frontal and top face of the toothpaste package shown in figure 6.13 a). In order to improve the pose estimation, the initial estimate is refined by inspection of the relative location of matched features across

the current view and two neighboring views. Features are grouped into approximately equilateral triangles. Figure 6.14 depicts the interpolation scheme for the azimuth estimation. The following computations are restricted to the intersection of features matched across the current and the three reference views. These features are grouped into subsets of three features that form a triangle which is characterized by its interior angles. As the camera perspective changes with the viewpoint, the features move accordingly resulting in a variation of the interior angles with the pose. Figure 6.14 illustrates the variation

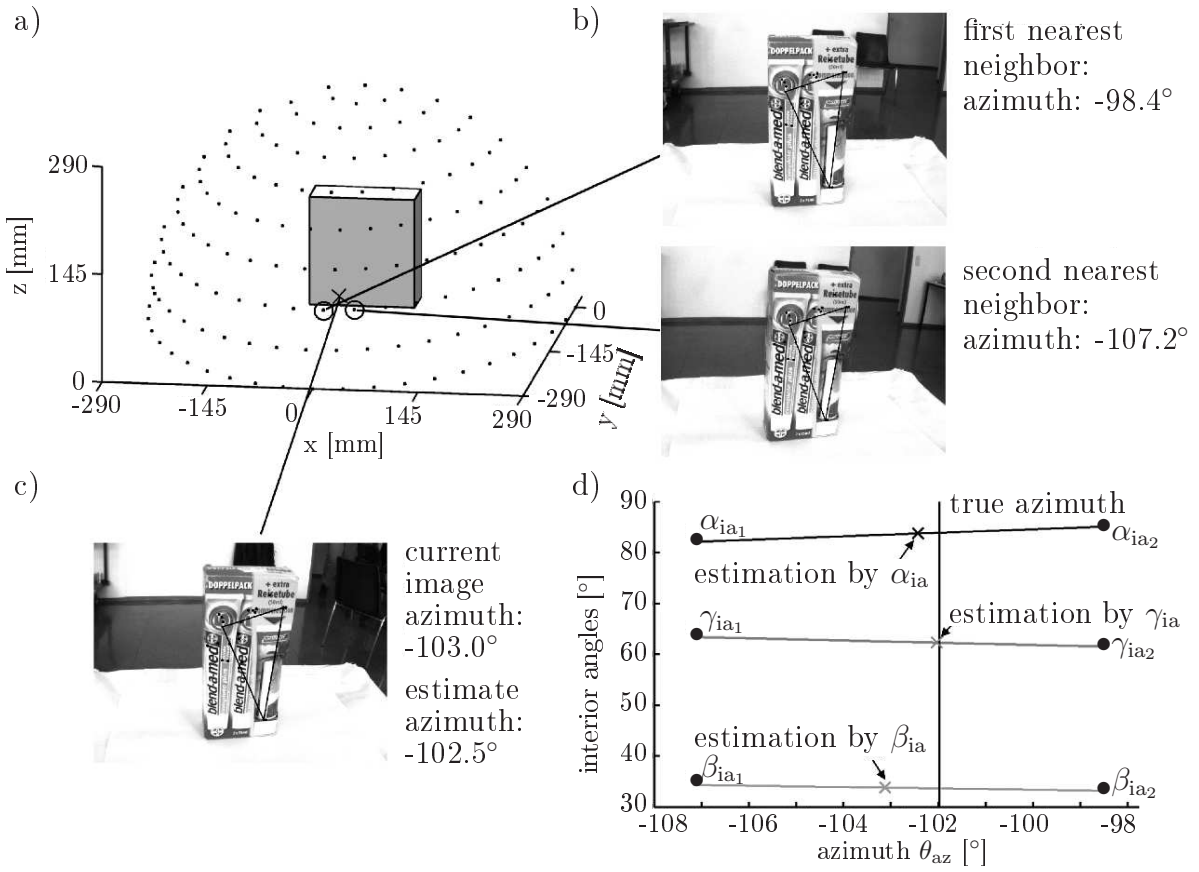


Figure 6.14: a) Current pose and the two nearest neighbors; b) Reference images of the two nearest neighbors form the database; c) Current image; d) Interpolated azimuth due to the interior angles of the triangle.

of the three interior angles α_{ia}, β_{ia}, γ_{ia} between two neighboring reference views with approximate azimuths $\theta_{az_1} = -98°$ and $\theta_{az_2} = -107°$. This variation provides the basis for a local correction of the estimated pose. The relationship between variation in pose and variation of the interior angles is assumed to be linear. The interior angles $\alpha_{ia_q}, \beta_{ia_q}, \gamma_{ia_q}$ in the query view fall in between those of the two reference view triangles. The azimuths

predicted from linear regression with respect to the interior angles are computed as

$$\begin{aligned}
\hat{\theta}_{\mathrm{az}}(\alpha_{\mathrm{ia}}) &= \theta_{\mathrm{az}_1} + (\theta_{\mathrm{az}_2} - \theta_{\mathrm{az}_1}) \frac{\alpha_{\mathrm{ia}_q} - \alpha_{\mathrm{ia}_1}}{\alpha_{\mathrm{ia}_2} - \alpha_{\mathrm{ia}_1}}, \\
\hat{\theta}_{\mathrm{az}}(\beta_{\mathrm{ia}}) &= \theta_{\mathrm{az}_1} + (\theta_{\mathrm{az}_2} - \theta_{\mathrm{az}_1}) \frac{\beta_{\mathrm{ia}_q} - \beta_{\mathrm{ia}_1}}{\beta_{\mathrm{ia}_2} - \beta_{\mathrm{ia}_1}}, \\
\hat{\theta}_{\mathrm{az}}(\gamma_{\mathrm{ia}}) &= \theta_{\mathrm{az}_1} + (\theta_{\mathrm{az}_2} - \theta_{\mathrm{az}_1}) \frac{\gamma_{\mathrm{ia}_q} - \gamma_{\mathrm{ia}_1}}{\gamma_{\mathrm{ia}_2} - \gamma_{\mathrm{ia}_1}}.
\end{aligned} \tag{6.9}$$

The interpolation is performed across multiple triangles, and the correct pose is predicted with an M-estimator that is robust with respect to outliers. The error function of the M-estimator is defined by:

$$\sum_{i=1}^{n} \rho(\varepsilon_i(x_i, \Theta_{\mathrm{m}}), \sigma_{\mathrm{e}}) \tag{6.10}$$

in which Θ_{m} denotes the model parameters and ε_i is the residual error between the model and the data point x_i. The parameter σ_{e} regulates the suppression of outliers and is adapted iteratively to the residual error distribution. The error function

$$\rho(x_i, \Theta_{\mathrm{m}}) = \frac{(x_i - \Theta_{\mathrm{m}})^2}{\sigma_{\mathrm{e}}^2 + (x_i - \Theta_{\mathrm{m}})^2} \tag{6.11}$$

is quadratic for small residual errors but flattens out for large residual errors thus reducing the impact of outliers. The method provides accurate estimates of azimuth and elevation under the assumption that the current view is captured from the same hemisphere as the reference views, at a nominal fixed distance between camera and object. Under the assumption that the object is always centered in the image, the object distance in addition to the azimuth and elevation is sufficient to reconstruct the full 6 DOF pose between the object and the camera. As the distance changes obviously also the triangles are distorted. In order to achieve scale invariance the relationship between interior angles and distance r_{sc} is modeled by an exponential function of the form $\alpha_{\mathrm{ia}}(r_{\mathrm{sc}}) = a_m \exp(b_m r_{\mathrm{sc}}) + c_m \exp(d_m r_{\mathrm{sc}})$ with four unknown parameters $[a_m, b_m, c_m, d_m]$, respectively for $\beta_{\mathrm{ia}}(r_{\mathrm{sc}})$ and $\gamma_{\mathrm{ia}}(r_{\mathrm{sc}})$. The best fit parameters are computed from reference images of the same azimuth and elevation at four different radii. In contrast to the case of constant scale azimuth and elevation estimation in equation 6.9, each interior angle $\alpha_{\mathrm{ia}}, \beta_{\mathrm{ia}}$ or γ_{ia} is now related to an entire manifold of azimuth, elevation and radius. An observation of an interior angle constraints the feasible solution set to a two-dimensional manifold in the three-dimensional azimuth, elevation and radius pose space. For a triplet of interior angles $\alpha_{\mathrm{ia}}, \beta_{\mathrm{ia}}, \gamma_{\mathrm{ia}}$ the three manifolds ideally intersect in isolated unique solutions. As the dataset contains discrete samples $\{[\alpha_{\mathrm{ia}}, \beta_{\mathrm{ia}}, \gamma_{\mathrm{ia}}], [\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}}]\}$ it is difficult to compute the intersection of the underlying manifolds. In practice the problem is transformed into an optimization problem which minimizes the quadratic error between the observed interior angles $[\alpha_{\mathrm{ia}_q}, \beta_{\mathrm{ia}_q}, \gamma_{\mathrm{ia}_q}]$ and the manifolds $\alpha_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}})$, $\beta_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}})$ and $\gamma_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}})$ across the parameters azimuth, elevation

and radius:

$$\begin{aligned}[\hat{\theta}_{\mathrm{az}}, \hat{\phi}_{\mathrm{el}}, \hat{r}_{\mathrm{sc}}] &= \mathrm{argmin}_{\theta_{\mathrm{az}},\phi_{\mathrm{el}},r_{\mathrm{sc}}}((\alpha_{\mathrm{ia}_q} - \alpha_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}}))^2 + \\ &+ (\beta_{\mathrm{ia}_q} - \beta_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}}))^2 + (\gamma_{\mathrm{ia}_q} - \gamma_{\mathrm{ia}}(\theta_{\mathrm{az}}, \phi_{\mathrm{el}}, r_{\mathrm{sc}}))^2). \end{aligned} \tag{6.12}$$

In between the discrete sample points, the manifolds are approximated by equation 6.9 along r_{sc} and a linear function in $[\theta_{\mathrm{az}}, \phi_{\mathrm{el}}]$. Only those parameters $[\theta_{\mathrm{az}}, \phi_{\mathrm{el}}]$ are considered in the minimization that belong to the spherical region spanned by the three nearest neighbors. The estimates $\hat{\theta}_{\mathrm{az}_i}, \hat{\phi}_{\mathrm{el}_i}, \hat{r}_{\mathrm{sc}_i}$ are aggregated over the entire set of triangles by the M-estimator.

Experimental results: In order to evaluate the performance and accuracy of the proposed pose estimation scheme, two types of experiments are conducted. In the first experiment data sets are generated artificially by mapping a set of 3D points in a virtual scene perspectively onto a normalized image plane. The purpose is to evaluate the theoretical limitations of the method, neglecting the negative impact of SIFT feature detection, limited pixel resolution, lens distortion and inherent pose uncertainty of the reference views. The distance between the object and the camera ranges from 100 mm to 700 mm. The second experiment is based on realistic views of the object depicted in figures 6.13 and 6.14. The reference and test images are captured with a robotic arm that moves the camera across the view hemisphere. This data set allows it to assess the accuracy of pose estimation under real world conditions. Due to the limited dexterous workspace of the robot arm, object-camera distances are restricted to the range from 180 mm to 290 mm. At closer distances the object is only partially visible. Due to the limited range, the distance interpolation with four parameters is replaced by a two parameter regression model given by $\alpha_{\mathrm{ia}}(r_{\mathrm{sc}}) = m \exp(b_m r_{\mathrm{sc}})$. The reference data set contains 517 reference views taken at three hemispheres of radius 180 mm, 235 mm and 290 mm. The test set contains 672 images, taken at twelve different radii with 56 images per radius. On average reference images contain between 700 to 1500 SIFT features. In order to accelerate the matching process, initially only the first hundred SIFT features are considered for matching. The preliminary search is sufficient to identify the nearest neighbor candidates. The local vicinity of these candidates is then searched for the nearest neighbor with the complete set of extracted features. Table 6.3 summarizes the results of the simulated data as well as the realistic data set for four different methods, namely single nearest neighbor (SNN), weighted average among three nearest neighbors (WANN), interpolation of azimuth and elevation at a fixed scale (FSI) and scale invariant interpolation of azimuth, elevation and radius (SII). For the interpolation scheme with adaptive scale, the mean error of the radius estimation is reported as well. Three different experiments are performed in order to observe the effects of uncalibrated camera systems compared to calibrated camera systems as well as the improvement achieved by the neighborhood correction step. Compared to the simulated case of the ideal perspective projection the accuracy of all methods is expected to deteriorate on the real world data set. In the nearest neighbor cases the accuracy in the real experiment exceeds the simulated ideal errors. This over-performance is explained by the fact that the simulated recognition rate of SIFT features drops more rapidly with

Table 6.3: Mean absolute error in azimuth and elevation in simulations and experiments. The different results demonstrate the effects of uncalibrated compared to calibrated camera systems as well as the improvement achieved by the neighborhood correction step.

	simulation			experiment1 uncalibrated			experiment 2 calibrated			experiment 3 correction step		
	$\bar{E}(\theta)$	$\bar{E}(\phi)$	$\bar{E}(r)$	$\bar{E}(\theta)$	$\bar{E}(\phi)$	$\bar{E}(r)$	$\bar{E}(\theta)$	$\bar{E}(\phi)$	$\bar{E}(r)$	$\bar{E}(\theta)$	$\bar{E}(\phi)$	$\bar{E}(r)$
SNN	3.0°	2.4°		3.8°	2.7°		3.7°	2.9°		2.2°	1.7°	
WANN	2.4°	1.6°		2.8°	2.6°		3.2°	2.5°		1.7°	1.6°	
FSI	2.0°	1.7°		7.4°	4.6°		6.9°	3.9°		4.0°	1.9°	
SII	0.82°	0.26°	20 mm	2.7°	2.1°	55 mm	2.4°	1.7°	57 mm	1.3°	1.0°	39 mm

change in viewpoint than in the case of the actual object. As the experimental estimates are based on more samples they tend to be more robust. In the case of nearest neighbors the accuracy between simulated and experimental data is comparable. In case of non-scale interpolation small errors in feature locations might result in substantial pose errors, which explains the poor performance on real images afflicted with noise. The large azimuth error is partially explained by the fact that some of the nearest neighbor reference views share too few common features. This in turn causes poor convergence of the M-estimator and inclusion of outliers in the estimate.

The scale invariant interpolation scheme in experiment 1 tends to be more robust, but still does not achieve the theoretically possible accuracy on uncalibrated real world data. It only provides a slight improvement compared to the basic N nearest neighbor scheme. Possible explanations that the scheme falls short of the expected accuracy are limited pixel resolution of feature locations, as the simulated projected images operate with subpixel accuracy, and radial lens distortion. Therefore the second experiment is performed with calibrated camera images that already demonstrates a minor improvement compared to the uncalibrated experiment. The large estimation error is caused by incorrect nearest neighbors. In these cases the interpolation scheme interpolates the wrong neighboring views that do not enclose the true view. In order to prevent false neighbors the interpolation result is verified whether it falls inside the region spanned by the assumed nearest neighbor views. If the interpolated view lies outside the span a new triangle is formed enclosing the extrapolated viewpoint. The results for the third experiment with false neighbor rejection in case of SII are superior to the pure nearest neighbor methods and in reasonable agreement with the ideal simulated errors. SII with correction step only provides a mean angular error of 1.3° in azimuth and 1.0° in elevation and 3.9 cm for camera-object distance.

The full 6 DOF estimation of the relative pose between object and camera requires additional information. First of all the object should always be centered in the image, with the camera axis intersecting the object center. In the case of object manipulation this restriction is achieved by a camera gaze control pointing the camera axis towards the object. Camera gaze control is naturally required in order to keep the object in view. Finally the

rotation of the camera around its optical axis is reconstructed from the keypoint orientation of SIFT features as shown in section 5.1. Under these assumptions the azimuth, elevation and distance estimates are sufficient to reconstruct the full 6 DOF pose between the object and the camera.

6.6 Evaluation and conclusion

This chapter presents a novel approach for optimal global visual servoing based on decoupled image moments with augmented point features in the context of object manipulation considering that the features in the reference image for grasping are not visible in the current object view. It is proven that model-free navigation in image space can be realized by means of a set of overlapping reference views in order to navigate from an arbitrary, unknown start into the goal pose. The switching between reference views occurs on the basis of the estimated time to convergence taken the quality of matched features into account. The cost of reference views is evaluated online throughout progression to the goal view, such that the scheme opportunistically selects the reference view that is optimal in the current context. In principle the work space is arbitrarily extensible under the condition of connectivity of the reference views in image space. The experimental results in virtual reality and on the real robot demonstrate that the approach minimizes the time to convergence without sacrificing the robustness and thereby stability of the visual control. As an alternative to the global visual servoing an appearance-based pose estimation in conjunction with local visual servoing is carried out with the same experimental setup as described in section 6.4.3 for a 6-axial industrial robot. The fundamental idea of initially applying a look-then-move strategy is to achieve even faster convergence to the goal view relying on the same sparse object representation as the global visual servoing. In principle the same accuracy in the reference pose as for the large view visual servoing is achieved as the final step for fine alignment consists of the same visual control scheme. Nevertheless the object representation for initial pose estimation has to be significantly extended requiring a substantially higher amount of computational and memory resources. Therefore it is finally stated that the major advantages of the global visual servoing compared to the look-then-move strategy are firstly that a sparse overlapping object representation sampling the plenoptic function on one radius of the hemisphere is sufficient, secondly that no estimation of the object distance is required, rather the control by the reference images guarantees an equidistance to the object, resulting thirdly in a kind of gaze control keeping the object always centered in the image.

Chapter 7

Conclusions and future work

The objective of this thesis is to advance the development of solely vision-based navigation and manipulation in the context of autonomous service robots.

This thesis demonstrates and emphasizes the potential of **visual reactive behaviors for visual navigation** in unstructured indoor environments. Primarily a vision-guided navigation of a mobile robot is implemented with reactive behaviors using distance sensors for local motion control and omnivision for localization, which provides consecutively the reference for a purely visual navigation. The vision-guided navigation is successfully verified in robotic experiments within office environments achieving the navigation task without collisions. Visual navigation is distinguished from vision-guided navigation by solely relying on the economic vision systems with the ambitious goal to achieve matchable performance in comparison to laser scanners. Thus a set of visual reactive behaviors is designed and implemented equivalently to the behaviors based on proximity information from range sensors. The visual navigation synergizes the comprehensive perception of the local environment of omnivision for localization, obstacle avoidance, optimal reference image selection, etc. with the high precision of a monocular camera in order to design a precise time-optimal homing through several non-overlapping reference images.

Visual homing is achieved by a large view visual servoing scheme comprehending several advantages compared to previous approaches. The concept of a horizontal virtual camera plane allows for decoupled navigation and gaze control and facilitates the derivation of generic moments. Generic moments cope with dynamic environments and lighting conditions and are designed to achieve a direct relationship between image and work space. In addition to localization, the extracted features furthermore provide the selection of reference images for time-optimal visual homing, thereby solving the problem of the limited field of view of the monocular camera and environments with sparse texture.
The set of reactive visual behaviors is completed by a novel obstacle avoidance and turn around behavior by means of several reconstructed perspective views, from which a confi-

dence rated time to contact is extracted. The concept of calculating the time to contact for different traveling directions based on sparse optical flow is introduced as the pairing window approach, enabling different alternatives for robot navigation. The major clue is the additional confidence evaluation of the visual measurements as it allows traveling towards directions that are perceived as obstacle-free.
Door detection, door localization and door traversal are treated for the first time in a coherent purely vision-based framework using omnivision to navigate between rooms and corridors. Notice that due to the overall awareness of the omnivision the door posts remain visible in the omnidirectional view during the door passage which thereby allows a closed-loop control with equidistant passage between the door posts.
Vision-based and visual navigation achieve a similar performance in unstructured office environments with regular texture. The two major advantages of visual navigation consist of low costs and high dimensional data space of cameras allowing for other applications such as person or object recognition. Nonetheless homogeneous office spaces require advanced camera systems such as ToF (Time-of-Flight) cameras [84], which reconstruct for a central field of view additional depth information from time of flight measurements.

Future research is dedicated to learning-by-demonstration, which enables the robot to acquire a behavior or skill through imitation of actions demonstrated by a teacher [131]. This approach allows non-professionals to instruct the robot intuitively without the necessity to program the desired task explicitly. It is sufficient for the teacher to be able to perform the required task. The learning approach extracts the underlying relation between perception and action from the demonstration. The straight forward methodology is a learning-by-demonstration scheme similar to the evolutionary optimized navigation behaviors in order to determine the recommendation of individual behaviors and the overall aggregation.
The future of visual navigation is closely related to cameras with structured light such as the economic Kinetic from Microsoft Cooperation [98], which reconstructs the depth of the scene by a pattern of infrared light points that are invisible to the human eye. An even more promising approach is to fuse omnivision with visual distance information obtained from ToF or triangulation of emitted infrared light, respectively, in order to capture in a single frame the visual perception as well as the depth of the local scene. This leads to 3D VSLAM with scalable abstraction of the map, including maps with dense depth representation, distinctive 3D visual features for instant localization as well as CAD models of the complete environment including objects and texture. Such representations simplify scene understanding, a still unsolved key ability for mobile manipulation, which requires further research in the next decade. The urban challenge also demonstrated that scene understanding is essential to solve complex traffic situations, whereas the close relation between mobile navigation and advanced driver assist systems yields a domination of robotic teams in the classification.

In the second part of this thesis a novel methodology for **image-based visual servoing by decoupled image moments for model-free object manipulation** solely relying on 2D image information is introduced. It relies on the pixel coordinates, scale and orientation of

augmented point features such as SIFT features. The control is based on decoupled image moments, which are generic in the sense that the control operates with a dynamic set of feature correspondences rather than a static set of geometric features. The foundation of visual servoing on generic SIFT features renders the method robust with respect to loss of redundant features caused by occlusion or changes in viewpoint. For 4 DOF visual servoing a set of completely decoupled visual features is introduced, that results in robust and independent convergence of the corresponding task space errors. Problems of the classical Jacobian based visual servoing scheme such as the camera retreat problem and local minima are resolved. A novel sensitivity matrix for 6 DOF visual servoing is introduced, which possesses only four off-diagonal couplings between the visual features and the degrees of motion assuming a valid weak perspective projection model. The visual control with the novel sensitivity matrix causes the pose errors to converge largely independent of each other resulting in a smoother task space motion of the camera. The control parameters of the visual control are automatically tuned in a HIL optimization by a controlled model assisted evolutionary strategy for real time applicability.
Global visual servoing based on decoupled image moments is successfully introduced. The workspace is partitioned into a set of overlapping reference views in order to navigate visually from a start to a goal pose. The switching between reference views occurs on the basis of the time to convergence estimated from the quality and distribution of matched features. The cost of reference views is evaluated online throughout progression to the goal view, such that the scheme opportunistically selects the reference view that is optimal in the current context. The computational demands of SIFT feature extraction, path planning and time-optimal reference selection enable real time visual control. The experimental results in virtual reality and on the real robot demonstrate that the approach minimizes the time to convergence without sacrificing the robustness and thereby stability of the visual control.
As an alternative a look-then-move strategy in conjunction with local visual servoing close to the reference pose is successfully implemented and tested for object manipulation, but it is inferior in terms of convergence time and robustness compared to optimal global visual servoing over multiple reference images.

Future research focuses on the development of a heuristic switching scheme for global visual servoing, that is independent of the object and does not require an offline exploration of the view space for prior cost estimation. An appropriate feature metric captures the distance in view space of features in the current view to the reference view based on the number of intermediate views (degree of separation) and the similarity of keypoint descriptors. Based on the feature distance metric the heuristic selects a reference view with the subset of matched features that is closest to the goal view. The benefit is a robust and continuous navigation in image space without decreasing velocities based on local convergence. Another interesting avenue for visual servoing is to control one agent by multiple cameras [79] or multiple agents by visual servoing [91] similar to cooperative manipulators in industrial manufacturing. To employ the proposed servoing based on decoupled image moments in this context is an interesting topic for future research because of its ease of

implementation and model-free approach.
In order to achieve purely vision-based mobile manipulation the presented object manipulation via optimal global visual servoing with dynamic feature sets only has to be integrated as an additional behavior into the hybrid control architecture containing the set of behaviors for visual navigation. In analogy to AUTOSAR (**AUT**omotive **O**pen **S**ystem **AR**chitecture) in the automotive industry, which provides a framework for the economic reuse of software, [52] recently introduced ROS (**R**obot **O**perating **S**ystem), which consists of an open source framework and construction kit for mobile manipulation applications. The adaptation and integration of the concepts for visual navigation and manipulation presented in this thesis into the ROS framework according to the example from [95] contributes to the ambitious goal of the robotic community to make service robots commonly affordable. In analogy to the well-known ISO26262 [73] for functional safety for road vehicles, a safety standard for mobile service robot applications is recently drafted for the first time as the ISO13482 [72], requiring additional safety validations to fulfill the specifications. The presented methods for obstacle avoidance have to be tested against these specifications e.g. as presented in [74].

Conclusively it can be stated that a purely vision-based navigation using omni and monocular vision is feasible. Experimental validations in an unstructured dynamic indoor environment show the enormous potential of visual navigation. This work demonstrates that it is possible to replace laser sensors by camera systems in the reactive layer. Nonetheless sonar sensors are required as a back-up, which indeed are cost-efficient and light-weight compared to laser sensors. Following the purely vision-based paradigm, it becomes possible to design affordable service robots. As an additional feature, manipulation of daily objects is presented, relying on natural occurring features and converging towards the grasping pose even if these features are not in the current view of the object. Visual navigation in conjunction with global visual servoing for object manipulation achieve the goal of vision-based mobile manipulation outlined in the introduction of this thesis.

Appendix A

Analysis of the grid-based time to contact from optical flow

The derivation of the grid-based time to contact for non-holonomic systems is inspired by [31]. The point of origin for determining the grid-based time to contact (ttc) is the image Jacobian $\mathbf{J}$, which relates differential changes in the camera position $\dot{\mathbf{r}}$ to differential changes in the image feature positions $\dot{\mathbf{f}}$ according to $\dot{\mathbf{f}} = \mathbf{J}\dot{\mathbf{r}}$. Replacing $\dot{\mathbf{f}}$ by the time derivative of the image coordinates u, v (which corresponds to the measured optical flow) and $\dot{\mathbf{r}}$ by the translational v_x, v_y, v_z and rotational velocities $\omega_\alpha, \omega_\beta, \omega_\gamma$, one obtains:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} -\frac{\lambda}{z} & 0 & \frac{u}{z} & \frac{uv}{\lambda} & \frac{-\lambda^2-u^2}{\lambda} & v \\ 0 & -\frac{\lambda}{z} & \frac{v}{z} & \frac{\lambda^2+v^2}{\lambda} & \frac{-uv}{\lambda} & -u \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ \omega_\alpha \\ \omega_\beta \\ \omega_\gamma \end{bmatrix}. \tag{A.1}$$

As the robot motion is planar and the robot is non-holonomic, the velocities v_y, ω_α and ω_γ are equal to zero, thereby equation A.1 simplifies to:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} -\frac{\lambda}{z} & \frac{u}{z} & \frac{-\lambda^2-u^2}{\lambda} \\ 0 & \frac{v}{z} & \frac{-u_i v}{\lambda} \end{bmatrix} \begin{bmatrix} v_{R_x} \\ v_{R_z} \\ \omega_R \end{bmatrix}. \tag{A.2}$$

Assuming a calibrated camera system and therefore normalized image coordinates $\hat{u}$, $\hat{v}$ with focal length $\lambda = 1$, equation A.2 is expressed as:

$$\begin{aligned} \dot{\hat{u}} &= \frac{1}{z}(-v_{R_x} + \hat{u}v_{R_z}) - (1+\hat{u}^2)\omega_R, \\ \dot{\hat{v}} &= \frac{1}{z}\hat{v}v_{R_z} - \hat{u}\hat{v}\omega_R. \end{aligned} \tag{A.3}$$

The rotational part of the optical flow is corrected based on the known egorotation of the robot. The egomotion can be estimated by the integrated wheel encoders or using directly the optical flow by projecting the flow field onto a sphere and optimizing a cost function [54]. By additionally substituting $\frac{1}{z}$ by ρ a set of linear equations is obtained. For reasons of clarity $\dot{\hat{u}}$ is substituted by $\dot{\hat{u}} + (1 + \hat{u}^2)\omega_R$ and $\dot{\hat{v}}$ by $\dot{\hat{v}} + \hat{u}\hat{v}\omega_R$, thus the rotational part ω_R is purged from the measured optical flow:

$$\begin{aligned} \dot{\hat{u}} &= \rho\left(-v_{R_x} + \hat{u}v_{R_z}\right), \\ \dot{\hat{v}} &= \rho\hat{v}v_{R_z}. \end{aligned} \tag{A.4}$$

The divergence of the optical flow is defined by the sum of the partial derivatives:

$$\nabla(\dot{\hat{u}}, \dot{\hat{v}}) = \frac{\partial \dot{\hat{u}}}{\partial \hat{u}} + \frac{\partial \dot{\hat{v}}}{\partial \hat{v}}. \tag{A.5}$$

The partial derivatives are calculated according to:

$$\begin{aligned} \frac{\partial \dot{\hat{u}}}{\partial \hat{u}} &= \frac{\partial \rho}{\partial \hat{u}}(-v_{R_x} + \hat{u}v_{R_z}) + \rho v_{R_z}, \\ \frac{\partial \dot{\hat{v}}}{\partial \hat{v}} &= \frac{\partial \rho}{\partial \hat{v}}\hat{v}v_{R_z} + \rho v_{R_z}. \end{aligned} \tag{A.6}$$

Substituting equation A.6 into A.5 yields:

$$\nabla(\dot{\hat{u}}, \dot{\hat{v}}) = 2\rho v_{R_z} + \frac{\partial \rho}{\partial \hat{u}}(-v_{R_x} + \hat{u}v_{R_z}) + \frac{\partial \rho}{\partial \hat{v}}\hat{v}v_{R_z}. \tag{A.7}$$

Due to the non-holonomic constraint v_{R_x} can be assumed to be zero as the robot cannot move sidewise during small time steps, therefore equation A.7 simplifies to:

$$\nabla(\dot{\hat{u}}, \dot{\hat{v}}) = 2\rho v_{R_z} + \frac{\partial \rho}{\partial \hat{u}}\hat{u}v_{R_z} + \frac{\partial \rho}{\partial \hat{v}}\hat{v}v_{R_z}. \tag{A.8}$$

Solving equation A.8 regarding the time to contact yields:

$$ttc = \frac{z}{v_{R_z}} = \frac{2 + \frac{\partial \rho}{\partial \hat{u}}\frac{\hat{u}}{\rho} + \frac{\partial \rho}{\partial \hat{v}}\frac{\hat{v}}{\rho}}{\nabla(\dot{\hat{u}}, \dot{\hat{v}})}. \tag{A.9}$$

The terms $\frac{\partial \rho}{\partial \hat{u}}\frac{\hat{u}}{\rho}$ as well as $\frac{\partial \rho}{\partial \hat{v}}\frac{\hat{v}}{\rho}$ can be neglected for small values of $\hat{u}$ and $\hat{v}$ resulting in a limited frontal field of view of 75° and the small changes in distance between two consecutive image frames. Therefore the same expression is obtained, but with completely different specifications as the authors in [31]. Equation A.10 indicates that the determination of *ttc* involves only the knowledge of the optical flow field vector divergence, whereas no model knowledge or estimation of z and v_{R_z} is required:

$$ttc = \frac{z}{v_{R_z}} = \frac{2}{\nabla(\dot{\hat{u}}, \dot{\hat{v}})}. \tag{A.10}$$

The authors in [31] develop the time to contact around the optical center $[\hat{u}, \hat{v}] = [0, 0]$ as operating point and determine a single *ttc* in driving direction using divergence templates. They require a dense optical flow, which is not suited for indoor environments. Contrary to their approach *ttc*s are needed for different image regions in order to have alternative course of actions for the robot. Note that their approach is invariant against rotations (responding rotational terms are neglected due to $[\hat{u}, \hat{v}] = [0, 0]$) whereas the optical flow caused by rotation has to be corrected prior to calculating *ttc*.

Conclusively three major differences to [31] can be stated:

+ *ttc* for different image regions and headings of the robot
+ no dense optical flow field required
- rotational parts have to be corrected a-priori

During an experimental evaluation of *ttc* calculation for sparse optical flow fields the robot moves toward a wall while capturing sonar and image snapshots as well as the egomotion. Figure A.1 demonstrates the excellent accordance between the *ttc* measured by a monocular

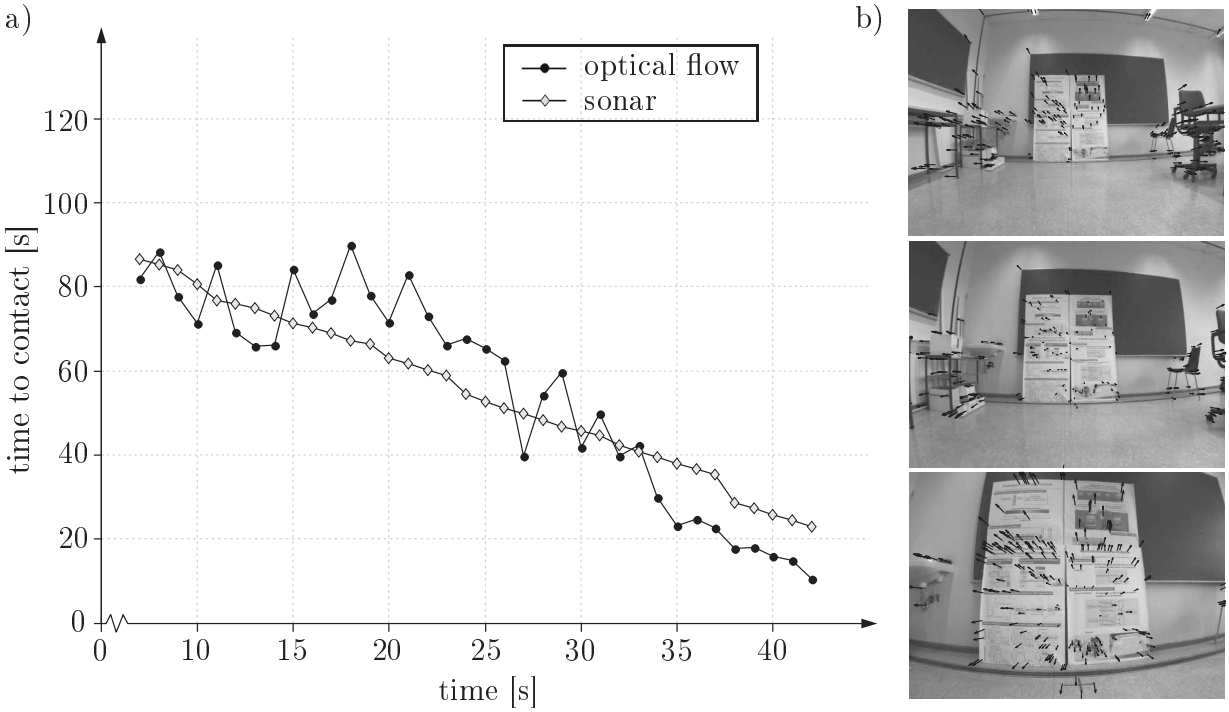


Figure A.1: a) Time to contact as a function of time for a monocular camera versus time to contact due to sonar measurements and known egomotion; b) Sequence of images with sparse optical flow taken during forward motion of the robot.

camera and *ttc* calculated by the division of distance measurements of the robot's sonar by the known egomotion of the robot.

Appendix B

Analysis of the sensitivity matrix

In order to determine the coupling for the choice of moments, i.e. the sensitivity of the moments towards another direction than the intended direction of motion, the image Jacobian is differentiated with respect to these moments. For the moments f_x and f_y this merely signifies to calculate the mean value of the known Jacobian for each feature. As a simplifying assumption all features should have a similar depth, i.e. approximately the same distance to the image plane. This assumption is valid as long as the depth difference is small compared to the distance of the camera to the object. In service robotic applications the above assumption is fulfilled, yielding a valid weak perspective projection model.

$$\mathbf{J}_{f_x,f_y} = \frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix} \frac{\lambda}{z} & 0 & \frac{-u_i}{z} & \frac{-u_i v_i}{\lambda} & \frac{\lambda^2+u_i^2}{\lambda} \\ 0 & \frac{\lambda}{z} & \frac{-v_i}{z} & \frac{-\lambda^2-v_i^2}{\lambda} & \frac{u_i v_i}{\lambda} \end{bmatrix} \tag{B.1}$$

Decoupling $f_{x,y}$ according to equations 5.10 and 5.19 yields

$$\mathbf{J}_{f_x,f_y} = \frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix} \frac{\lambda}{z} & 0 & 0 & 0 & \frac{\lambda^2+u_i^2}{\lambda} \\ 0 & \frac{\lambda}{z} & 0 & \frac{-\lambda^2-v_i^2}{\lambda} & 0 \end{bmatrix}. \tag{B.2}$$

To determine the Jacobian for the moments f_α and f_β, first the feature parameter is transformed and then differentiated with respect to the time in order to obtain the dependence of the change of the feature parameter on the camera velocities $[v_x, v_y, v_z, \omega_\alpha, \omega_\beta]^T$. As the rotation around γ is already compensated as described in equation 5.2, ω_γ does not have to be considered further here. The calculation of $\mathbf{J}_{f_\alpha}$ for f_α (cf. equations 5.3 to 5.5) is now exemplified but is also valid analogously for f_β.

$$f_\alpha = \sum_{i=1}^{n}\sum_{j=i+1}^{n}\frac{(-v_{\mathrm{ref}_i} - v_{\mathrm{ref}_j})\left\|\mathbf{p}_j - \mathbf{p}_i\right\|}{\sum_{k=1}^{n}\sum_{l=k+1}^{n}\left\|\mathbf{p}_k - \mathbf{p}_l\right\|} \tag{B.3}$$

According to the connection between the location in the image and the location of points in real space resulting from the image geometry v and u are replaced by $u_i = \lambda x_i/z$ and

accordingly for u_j, v_i and v_j. Therefore the moment f_α is expressed as:

$$f_\alpha = \frac{\lambda}{z} \frac{\sum_{i=1}^{n} \sum_{j=i+1}^{n} (y_{\mathrm{ref}_i} + y_{\mathrm{ref}_j}) \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(x_k - x_l)^2 + (y_k - y_l)^2}}. \tag{B.4}$$

The transformed moment is differentiated with respect to the time. For sake of clarity only one sum term (m_α) is considered here:

$$\begin{aligned} \dot{m}_\alpha = & - \frac{\lambda}{z^2} \dot{z} (y_{\mathrm{ref}_i} + y_{\mathrm{ref}_j}) \frac{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(x_k - x_l)^2 + (y_k - y_l)^2}} + \\ & + \frac{\lambda}{z} (y_{\mathrm{ref}_i} + y_{\mathrm{ref}_j}) \frac{[(x_i - x_j)^2 + (y_i - y_j)^2]^{-1/2} [(x_i - x_j)(\dot{x}_i - \dot{x}_j) + (y_i - y_j)(\dot{y}_i - \dot{y}_j)]}{\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(x_k - x_l)^2 + (y_k - y_l)^2}} + \\ & - \frac{\lambda}{z} (y_{\mathrm{ref}_i} + y_{\mathrm{ref}_j}) \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \\ & \frac{\sum_{k=1}^{n} \sum_{l=k+1}^{n} [(x_k - x_l)(\dot{x}_k - \dot{x}_l) + (y_k - y_l)(\dot{y}_k - \dot{y}_l)]}{\left[\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(x_k - x_l)^2 + (y_k - y_l)^2} \right]^3}. \end{aligned} \tag{B.5}$$

In analogy to the derivation of the classical image Jacobian $\mathbf{J}$ in [70], due to $\dot{x}_j - \dot{x}_i = -\omega_z (y_j - y_i)$ and $\dot{y}_j - \dot{y}_i = \omega_z (x_j - x_i)$ the expression $(x_i - x_j)(\dot{x}_i - \dot{x}_j) + (y_i - y_j)(\dot{y}_i - \dot{y}_j)$ becomes zero, thus only the first term in equation B.5 remains. Now the variables x_i, x_j, y_i and y_j are back transformed according to $x_{i,j} = u_{i,j} z/\lambda$ and $y_{i,j} = v_{i,j} z/\lambda$, resulting in:

$$\dot{m}_\alpha = -\frac{1}{z} \dot{z} (v_{\mathrm{ref}_i} + v_{\mathrm{ref}_j}) \frac{\sqrt{(u_i - u_j)^2 + (v_i - v_j)^2}}{\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(u_k - u_l)^2 + (v_k - v_l)^2}}. \tag{B.6}$$

Furthermore, inserting $\dot{z} = v_z + \omega_x y - \omega_y x$ with $x = (x_i + x_j)/2$ and $y = (y_i + y_j)/2$ according to [70] yields:

$$\begin{aligned} \dot{m}_\alpha = & \frac{1}{2\lambda} (v_{\mathrm{ref}_i} + v_{\mathrm{ref}_j}) \left(-\frac{2\lambda}{z} v_z - \omega_x (v_i + v_j) + \omega_y (u_i + u_j) \right) \cdot \\ & \cdot \frac{\sqrt{(u_i - u_j)^2 + (v_i - v_j)^2}}{\sum_{k=1}^{n} \sum_{l=k+1}^{n} \sqrt{(u_k - u_l)^2 + (v_k - v_l)^2}}. \end{aligned} \tag{B.7}$$

Thus, the image Jacobian for α is given by

$$\mathbf{J}_{f_\alpha} = \frac{\sum_{\substack{i=1, \\ j=i+1}}^{n} \frac{p_{ij} v_{\mathrm{ref}_{ij}} \lambda}{\sqrt{8} z^2} \begin{bmatrix} 0 & 0 & -\frac{2\lambda}{z} & -v_{ij} & +u_{ij} \end{bmatrix}}{\sum_{k=1}^{n} \sum_{l=i+1}^{n} \|\mathbf{p}_k - \mathbf{p}_l\|}, \tag{B.8}$$

and the image Jacobian for β is expressed accordingly as

$$\mathbf{J}_{f_\alpha} = \frac{\sum_{\substack{i=1, \\ j=i+1}}^{n} \frac{p_{ij} u_{\mathrm{ref}_{ij}} \lambda}{\sqrt{8} z^2} \begin{bmatrix} 0 & 0 & -\frac{2\lambda}{z} & -u_{ij} & +v_{ij} \end{bmatrix}}{\sum_{k=1}^{n} \sum_{l=i+1}^{n} \|\mathbf{p}_k - \mathbf{p}_l\|}. \tag{B.9}$$

The dependencies of the moments of α and β on motions in z-direction are not completely resolved but can be assumed to be nearly zero ($-\frac{2\lambda}{z}$).

For the image moment f_{zd} defined in equation 5.7 again for sake of clarity only one sum term (m_z) is considered here:

$$\begin{aligned} \dot{m}_z &= \frac{d}{dt}\left(\frac{\lambda}{z}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}\right) \\ &= -\frac{\lambda}{z^2}\dot{z}\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}+\frac{\lambda}{z}\frac{(x_i-x_j)(\dot{x}_i-\dot{x}_j)+(y_i-y_j)(\dot{y}_i-\dot{y}_j)}{\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}}. \end{aligned} \tag{B.10}$$

This expression is again simplified because of the relation $\dot{x}_j-\dot{x}_i=-\omega_z(y_j-y_i)$ and $\dot{y}_j-\dot{y}_i=\omega_z(x_j-x_i)$, and the back transformation of $x_{i,j}=u_{i,j}z/\lambda$ and $y_{i,j}=v_{i,j}z/\lambda$ leads to:

$$\dot{m}_z=-\frac{\dot{z}}{z}\sqrt{(u_i-u_j)^2+(v_i-v_j)^2}=-\frac{\dot{z}}{z}||\mathbf{p}_i-\mathbf{p}_j||_2. \tag{B.11}$$

Again, $\dot{z}=v_z+\omega_x y-\omega_y x$ with $x=(x_i+x_j)/2$ and $y=(y_i+y_j)/2$ is inserted, leading to the total sum:

$$\dot{f}_{zd}=\frac{1}{\frac{n}{2}(n-1)}\sum_{i=1}^{n}\sum_{j=i+1}^{n}||\mathbf{p}_i-\mathbf{p}_j||_2\left(-\frac{1}{z}v_z-\frac{1}{2\lambda}(v_i+v_j)\omega_x+\frac{1}{2\lambda}(u_i+u_j)\omega_y\right). \tag{B.12}$$

Thus, the image Jacobian for z is given by

$$\mathbf{J}_{f_{zd}}=\frac{1}{\frac{n}{2}(n-1)}\sum_{i=1}^{n}\sum_{j=i+1}^{n}||\mathbf{p}_i-\mathbf{p}_j||_2\left[\begin{array}{cccccc} 0 & 0 & -\frac{1}{z} & -\frac{1}{2\lambda}(v_i+v_j) & \frac{1}{2\lambda}(u_i+u_j) & \end{array}\right]. \tag{B.13}$$

In order to reduce the couplings furthermore f_z is now replaced by the scaled version $f_{z,\sigma}$. Therefore the sensitivity matrix has the following structure whereas all non-diagonal elements are considered as undesired couplings:

$$\left[\begin{array}{c}\dot{f}_x\\ \dot{f}_y\\ \dot{f}_z\\ \dot{f}_\alpha\\ \dot{f}_\beta\\ \dot{f}_\gamma\end{array}\right]=\left[\begin{array}{cccccc} J_{f_x,x} & 0 & 0 & 0 & \tilde{J}_{f_x,\beta} & 0\\ 0 & J_{f_y,y} & 0 & \tilde{J}_{f_y,\alpha} & 0 & 0\\ 0 & 0 & J_{f_z,z} & 0 & 0 & 0\\ 0 & 0 & 0 & J_{f_\alpha,\alpha} & \tilde{J}_{f_\alpha,\beta} & 0\\ 0 & 0 & 0 & \tilde{J}_{f_\beta,\alpha} & J_{f_\beta,\beta} & 0\\ 0 & 0 & 0 & 0 & 0 & J_{f_\gamma,\gamma}\end{array}\right]\left[\begin{array}{c}T_x\\ T_y\\ T_z\\ \omega_\alpha\\ \omega_\beta\\ \omega_\gamma\end{array}\right]. \tag{B.14}$$

The control scheme for visual servoing with generic image moments in 6 DOF is summarized in table B.1 taking into account the deviation from chapter 5.2 to 5.5 as well as the sensitivity matrix above.

Table B.1: Visual servoing with generic image moments in 6 DOF.

(1.) *Automatic feature selection for the reference view of the object (cf. section 5.1).*

2. *Extraction of augmented point features in the current view* $\mathbf{f}_{\mathrm{ref}_i} = [u_{\mathrm{ref}_i}, v_{\mathrm{ref}_i}, \phi_{\mathrm{ref}_i}, \sigma_{\mathrm{ref}_i}]$ *like SIFT/SURF.*
3. *Determination of the camera rotation* Δf_γ *(cf. equation 5.1) by*

$$\Delta f_\gamma = f_{\mathrm{ref}_\gamma} - f_\gamma \qquad \text{with} \qquad f_\gamma = \frac{1}{n}\sum_{i=1}^{n} \phi_i.$$

4. *Alignment of* u_i *and* v_i *with the image features in the reference view (cf. equation 5.2)*

$$\begin{bmatrix} u_i' \\ v_i' \end{bmatrix} = \begin{bmatrix} \cos(\Delta f_\gamma) & -\sin(\Delta f_\gamma) \\ \sin(\Delta f_\gamma) & \cos(\Delta f_\gamma) \end{bmatrix} \begin{bmatrix} u_i \\ v_i \end{bmatrix}.$$

Redefinition of u_i *equal to* u_i'*, respectively* v_i *equal to* v_i'*.*

5. *Calculation of image moments for camera rotation around* α *and* β *(cf. equations 5.3 and 5.4)*

$$f_\alpha = \sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{(-v_{\mathrm{ref}_i} - v_{\mathrm{ref}_j})\sqrt{(u_i-u_j)^2+(v_i-v_j)^2}}{\sum_{k=1}^{n}\sum_{l=k+1}^{n}\sqrt{(u_k-u_l)^2+(v_k-v_l)^2}},$$

$$f_\beta = \sum_{i=1}^{n}\sum_{j=i+1}^{n} \frac{(-u_{\mathrm{ref}_i} - u_{\mathrm{ref}_j})\sqrt{(u_i-u_j)^2+(v_i-v_j)^2}}{\sum_{k=1}^{n}\sum_{l=k+1}^{n}\sqrt{(u_k-u_l)^2+(v_k-v_l)^2}}.$$

6. *Determination of image moment for regulating translation along camera axis (cf. equation 5.6)*

$$f_z = \frac{1}{n}\sum_{i=1}^{n} \sigma_i.$$

7. *Determination of image moments for* x *and* y *(cf. equation 5.10)*

$$f_x = \sum_{i=1}^{n} w_i u_i, \quad f_y = \sum_{i=1}^{n} w_i v_i.$$

w_i *is determined by minimizing the optimization problem* F *(cf. equation 5.17) through a set of linear equations (cf. equation 5.19).*

(8.) *Singular computation of the gains: HIL optimization of the controller by* λ*-CMAES (cf. section 5.3.1).*

9. *Determination of controller setpoint by overall feature error* $\Delta\mathbf{f}(\mathbf{I}) = [\Delta f_x, \Delta f_y, \Delta f_z, \Delta f_\alpha, \Delta f_\beta, \Delta f_\gamma]^T$ *according to image moments* f_x*,* f_y*,* f_z*,* f_α*,* f_β *and* f_γ*:*

$$[v_x, v_y, v_z, \omega_\gamma, \omega_\alpha, \omega_\beta]^T = [k_x, k_y, k_z, k_\gamma, k_\alpha, k_\beta]^T [\Delta f_x, \Delta f_y, \Delta f_z, \Delta f_\gamma, \Delta f_\alpha, \Delta f_\beta]^T + [k_{Dx}, k_{Dy}, k_{Dz}, k_{D\gamma}, k_{D\alpha}, k_{D\beta}]^T [\Delta \dot{f}_x, \Delta \dot{f}_y, \Delta \dot{f}_z, \Delta \dot{f}_\gamma, \Delta \dot{f}_\alpha, \Delta \dot{f}_\beta]^T.$$

Bibliography

[1] E. H. Adelson and J. R. Bergen. The Plenoptic Function and the Elements of Early Vision. In M. Landy and J. A. Movshon, editors, *Computational Models of Visual Processing*. MIT Press, 1991.

[2] H. Andreasson and T. Duckett. Topological Localization for Mobile Robots using Omni-directional Vision and Local Features. In *Proceedings of the 5th Symposium on Intelligent Autonomous Vehicles (IAV)*, 2004.

[3] A. Angeli, S. Doncieux, J.-A. Meyer, and D. Filliat. Visual topological slam and global localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4300–4305, 2009.

[4] A. A. Argyros, D. P. Tsakiris, and C. Groyer. Biomimetic Centering Behavior. *IEEE Robotics & Automation Magazine*, pages 21–30, 2004.

[5] R. C. Arkin. *Behavior Based Robotics*. MIT Press, Cambridge, Massachusetts, 1998.

[6] A. Astolfi. Exponential stabilization of a mobile robot. In *Proceedings of the Third European Control Conference (ECC)*, pages 181–191, 1995.

[7] C. G. Atkeson, A. Moore, and S. Schaal. Locally weighted learning for control. *Artificial Intelligence Review*, 11(1-5):75–113, 1997.

[8] S. Baker and S. K. Nayar. A Theory of Catadioptric Image Formation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 35–42, 1998.

[9] M. Bakthavatchalam, F. Chaumette, and E. Marchand. Photometric moments: New promising candidates for visual servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5521–5526, 2013.

[10] J. L. Barron, S. S. Beauchemin, and D. J. Fleet. On Optical Flow. In *Proceedings of the 6th International Conference on Artificial Intelligence and Information-Control Systems of Robots (AIICSR)*, pages 3–14, 1998.

[11] J. L. Barron, D. J. Fleet, S. S. Beauchemin, and T. A. Burkitt. Performance of Optical Flow Techniques. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 236–242, 1992.

[12] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. SURF: Speeded Up Robust Features. In *Computer Vision and Image Understanding (CVIU)*, volume 110, pages 346–359, 2008.

[13] H. Bay, T. Tuytelaars, and L. V. Gool. SURF: Speeded Up Robust Features. In *Proceedings of the 9th European Conference on Computer Vision (ECCV)*, pages 404–417, 2006.

[14] H. Becerra, J. Courbon, Y. Mezouar, and C. Sagues. Wheeled mobile robots navigation from a visual memory using wide field of view cameras. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5693–5699, 2010.

[15] G. A. Bekey. *Autonomous robots - From Biological Inspiration to Implementation and Control*. MIT Press, 2005.

[16] F. Bonin-Font, A. Ortiz, and G. Oliver. Visual Navigation for Mobile Robots: A Survey. *Journal of Intelligent and Robotic Systems*, 53:263–296, 2008.

[17] J.-Y. Bouguet. Pyramidal Implementation of the Lucas Kanade Feature Tracker - Description of the Algorithm. Technical report, Inter Corporation, Microprocessor Research Labs, 1999.

[18] R. A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[19] A. R. Bruss and B. K. P. Horn. Passive navigation. *Computer Graphics and Image Processing*, 21:3–20, 1983.

[20] D. Burschka and G. Hager. Vision-Based Control of Mobile Robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1707–1713, 2001.

[21] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: binary robust independent elementary features. In *Proceedings of the 11th European Conference on Computer Vision (ECCV)*, pages 778–792, 2010.

[22] P. Chang and J. Krumm. Object Recognition with Color Cooccurrence Histograms. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 498–504, 1999.

[23] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In D. Kriegman, G. Hager, and A. S. Morse, editors, *The Confluence of Vision and Control*, number 237, pages 66–78. LNCIS Series, Springer, 1998.

[24] F. Chaumette and S. Hutchinson. Visual servo control, Part I: Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90, 2006.

[25] F. Chaumette and S. Hutchinson. Visual servo control, Part II: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, 2007.

[26] F. Chaumette and E. Malis. 2 1/2 D Visual Servoing: A Possible Solution to Improve Image-Based and Position-Based Visual Servoings. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 630–635, 2000.

[27] G. Cheng and A. Zelinsky. Real-Time Visual Behaviours for Navigating a Mobile Robot. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 973–980, 1996.

[28] A. Cherubini and F. Chaumette. Visual navigation of a mobile robot with laser-based collision avoidance. *International Journal of Robotic Research*, 32(2):189–205, 2013.

[29] A. Cherubini, B. Grechanichenko, F. Spindler, and F. Chaumette. Avoiding Moving Obstacles During Visual Navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5227–5232, 2013.

[30] C. Collewet, E. Marchand, and F. Chaumette. Visual servoing set free from image processing. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 81–86, 2008.

[31] D. Coombs, M. Herman, T.-H. Hong, and M. Nashman. Real-Time Obstacle Avoidance Using Central Flow Divergence and Peripheral Flow. *IEEE Transactions on Robotics and Automation*, 14:49–59, 1998.

[32] D. Coombs and K. Roberts. Centering Behavior Using Peripheral Vision. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 440–445, 1993.

[33] P. I. Corke and S. A. Hutchinson. A new partitioned approach to Image-Based Visual Servo Control. *IEEE Transactions on Robotics and Automation*, 17(4):507–515, 2001.

[34] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 886–893, 2005.

[35] A. Dame and E. Marchand. A new information theoretic approach for appearance-based navigation of non-holonomic vehicle. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2459–2464, 2011.

[36] A. Davison. Real-Time Simultaneous Localisation and Mapping with a Single Camera. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1403–1410, 2003.

[37] K. Deguchi. A Direct Interpretation of Dynamic Images with Camera and Object Motions for Vision Guided Robot Control. *International Journal of Computer Vision*, 37(1):7–20, 2000.

[38] D. Dementhon and L. Davis. Model-Based Object Pose in 25 Lines of Code. *International Journal of Computer Vision*, 15(1):123–141, 1995.

[39] G. N. DeSouza and A. C. Kak. Vision for Mobile Robot Navigation: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):237–267, 2002.

[40] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[41] A. P. Duchon, W. H. Warren, and L. P. Kaelbling. Ecological Robotics. In *Adaptive Behavior*, pages 473–507, 1995.

[42] C. Eberst, M. Andersson, and H. I. Christensen. Vision-based door-traversal for autonomous mobile robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 620–625, 2000.

[43] S. Ekvall, D. Kragic, and F. Hoffmann. Object Recognition and Pose Estimation using Color Cooccurrence Histograms and Geometric Modelling. *Image and Vision computing*, 23(11):943–955, 2005.

[44] T. Emter and A. Stein. Simultaneous localization and mapping with the kinect sensor. In *Proceedings of the 7th German Conference on Robotics (ROBOTIK)*, pages 1–6, 2012.

[45] C. Fagerer, D. Dickmanns, and E. D. Dickmanns. Visual grasping with long delay time of a free floating object in orbit. *Autonomous Robots*, 1:53–68, 1995.

[46] Y. Fang, W. E. Dixon, D. M. Dawson, and P. Chawda. Homography-based visual servo regulation of mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics (SMC) - Part B: Cybernetics*, 35(5):1041–1050, 2005.

[47] O. D. Faugeras and F. Lustman. Motion and Structure from Motion in a Piecewise Planar Environment. *Pattern Recognition and Artificial Intelligence (PRAI)*, 2:485–508, 1988.

[48] J. T. Feddema, C. S. G. Lee, and O. R. Mitchell. Weighted selection of image features for resolved rate visual feedback control. *IEEE Transactions on Robotics and Automation*, 7(1):31–47, 1991.

[49] M. A. Fischler and R. C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[50] D. A. Forsyth and J. Ponce. *Computer Vision - A Modern Approach*. Prentice Hall, Pearson Education, New Jersey, 2003.

[51] D. Fox. *Markov Localization: A Probabilistic Framework for Mobile Robot Localization and Navigation*. PhD thesis, University of Bonn, 1998.

[52] B. Gerkey. Building Blocks for Mobile Manipulation. In *Simulation, Modeling, and Programming for Autonomous Robots*, volume 6472, pages 1–1. 2010.

[53] C. Geyer and K. Daniilidis. Catadioptric Projective Geometry. *International Journal of Computer Vision*, 4(3):223–243, 2001.

[54] J. Gluckman and S. K. Nayar. Mobile Ego-Motion and Omnidirectional Cameras. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 999–1005, 1998.

[55] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. V. Gool. Omnidirectional vision-based Topological Navigation. *International Journal of Computer Vision*, 74:219–236, 2007.

[56] I. Gordon and D. Lowe. What and Where: 3D Object Recognition with accurate Pose. In C. S. J. Ponce, M. Hebert and A. Zisserman, editors, *Toward Category-Level Object Recognition*, pages 67–82. Springer, 2006.

[57] G. Hager, D. Kriegman, A. Georghiades, and O. Ben-Shahar. Toward domain-independent navigation: Dynamic vision and control. In *Proceedings of the IEEE Conference on Decision and Control*, pages 3257–3262, 1998.

[58] D. Hall, V. de Verdiere, and J. Crowley. Object Recognition Using Coloured Receptive Fields. In *Proceedings of the 6th European Conference of Computer Vision (ECCV)*, pages 164–178, 2000.

[59] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

[60] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2008.

[61] F. Hoffmann. Fuzzy Behavior Coordination for Robot Learning from Demonstration. In *Proceedings of the North American Fuzzy Information Processing Society (NAFIPS)*, volume 1, pages 157–162, 2004.

[62] F. Hoffmann and S. Hölemann. Controlled Model Assisted Evolution Strategy with Adaptive Preselection. In *Proceedings of the IEEE International Symposium on Evolving Fuzzy Systems*, pages 182–187, 2006.

[63] F. Hoffmann, T. Nierobisch, T. Bertram, M. Castrillon, and H. Apmann. Hybride bildbasierte Regelung mit strukturiertem Licht. In *Sensoren und Meßsysteme 2008, 14. Fachtagung Ludwigsburg, VDI-Berichte Nr. 2011, Düsseldorf: VDI-Verlag*, pages 23–32, 2008.

[64] F. Hoffmann, T. Nierobisch, T. Seyffarth, and G. Rudolph. Visual Servoing with Moments of SIFT Features. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 4262–4267, 2006.

[65] M. Hofmeister, P. Vorst, and A. Zell. A comparison of efficient global image features for localizing small mobile robots. In *Proceedings of the Joint Conference of the 41st International Symposium on Robotics (ISR) and the 6th German Conference on Robotics (ROBOTIK)*, pages 1–8, 2010.

[66] B. K. P. Horn and B. G. Schunck. Determining optical flow. In *Shape recovery*, pages 389–407, 1992.

[67] A. Hornberg. *Handbook of Machine Vision*. Wiley-VCH, Weinheim, 2006.

[68] R. Hu, M. Barnard, and J. P. Collomosse. Gradient field descriptor for sketch based retrieval and localization. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 1025–1028, 2010.

[69] T. S. Huang and A. N. Netravali. Motion and structure from feature correspondences: A review. *Proceedings of the IEEE*, 82(2):252–268, 1994.

[70] S. A. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.

[71] Intel Corporation. Open Source Computer Vision Library Reference Manual. Technical report, 2001.

[72] ISO/TC184/SC2/WG7. Robots and robotic devices; Safety requirements for non-industrial robots; Non-medical personal care robot (Draft), 2012.

[73] ISO/TC22/SC3/WG16. Road vehicles Functional safety, 2011.

[74] T. Jacobs, U. Reiser, M. Haegele, and A. Verl. Development of validation methods for the safety of mobile service robots with manipulator. In *Proceedings of the 7th German Conference on Robotics (ROBOTIK)*, pages 1–5, 2012.

[75] P. Jensfelt. *Approaches to Mobile Robot Localization in Indoor Environments*. PhD thesis, KTH Stockholm, 2001.

[76] P. Jensfelt, D. Kragic, J. Folkesson, and M. Björkman. A Framework for vision based Bearing Only 3D SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2006.

[77] M. Jägersand. Visual Servoing using Trust Region Methods and Estimation of the Full Coupled. In *IASTED Applications of Control and Robotics, Department of Computer Science, University of Rochester, NY 14627*, pages 105–108, 1996.

[78] B. Jähne. *Digitale Bildverarbeitung*. Springer Berlin Heidelberg, 2002.

[79] O. Kermorgant and F. Chaumette. Multi-sensor data fusion in sensor-based control: Application to multi-camera visual servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4518–4523, 2011.

[80] U. Khan, L. A. Khan, and S. Z. Hussain. Reinforcement learning for appearance based visual servoing in robotic manipulation. In *Proceedings of the 8th World Scientific and Engineering Academy and Society (WSEAS) International Conference on Robotics, Control and Manufacturing Technology (ROCOM)*, pages 161–168, 2008.

[81] U. Khan, T. Nierobisch, and F. Hoffmann. Two-Finger Grasping for Vision Assisted Object Manipulation. In K. Kozlowski, editor, *Robot and Motion Control, Lecture Notes In Control and Information Sciences*, volume 360, pages 89–98. Springer, 2007.

[82] G. Klinker, S. Shafer, and T. Kanade. A physical approach to color image understanding. *International Journal of Computer Vision*, 4:7–38, 1990.

[83] M. Knapek, R. S. Oropeza, and D. Kriegman. Selecting promising landmarks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3771–3777, 2000.

[84] A. Kolb, E. Barth, and R. Koch. ToF Sensors: New Dimensions for Realism and Interactivity. In *Proceedings of the Workshop on Time-of-Flight-based Computer Vision (TOF-CV) of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[85] J. Krettek, T. Nierobisch, F. Hoffmann, and T. Bertram. Zeitoptimale bildbasierte Weitbereichsregelung zur Positionierung eines Industrieroboters. In *GMA-Kongress: Automation im gesamten Lebenszyklus, Kongress Baden-Baden, VDI-Berichte Nr. 1980. Düsseldorf: VDI-Verlag*, pages 297–307, 2007.

[86] B. Kuipers and Y.-T. Byun. A Robot Exploration and Mapping Strategy Based on a Semantic Hierarchy of Spatial Representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.

[87] D. N. Lee. The optic flow field. *Philosophical Transactions of the Royal Society of London, Series B, Biological Sciences*, pages 169–178, 1980.

[88] M. Liu, C. Pradalier, F. Pomerleau, and R. Siegwart. Scale-only visual homing from an omnidirectional camera. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3944–3949, 2012.

[89] J. D. Lohn and G. S. Hornby. Evolvable hardware: Using evolutionary computation to design and optimize hardware systems. *IEEE Computational Intelligence Magazine*, 1(1):19–27, 2006.

[90] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981.

[91] G. Lopez-Nicolas, Y. Mezouar, and C. Sagues. Homography-based multi-robot control with a flying camera. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4492–4497, 2011.

[92] T. Low and G. Wyeth. Obstacle detection using optical flow. In *Proceedings of the Australasian Conference on Robotics and Automation (ACRA)*, 2005.

[93] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[94] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[95] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige. The Office Marathon: Robust Navigation in an Indoor Office Environment. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 300–307, 2010.

[96] C. McCarthy and N. Barnes. Performance of optical flow techniques for indoor navigation with a mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5093–5098, 2004.

[97] Y. Mezouar and F. Chaumette. Path planning for robust image-based control. *IEEE Transactions on Robotics and Automation*, 18(4):534–549, 2002.

[98] Microsoft Corporation, Xbox Kinetic, 2013. `http://www.xbox.com/de-DE/Xbox360/Accessories/kinect/Home`.

[99] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005.

[100] MobileRobots Inc, 2013. `http://robots.mobilerobots.com/`.

[101] I. Monasterio, E. Lazkano, I. Rano, and B. Sierra. Learning to traverse doors using visual information. *Transactions of Mathematics and Computers in Simulation*, 60:347–356, 2002.

[102] M. Montemerlo and S. Thrun. *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics*. Springer, 2007.

[103] R. Munoz-Salinas, E. Aguirre, M. García-Silvente, and A. González. Door-detection using computer vision and fuzzy logic. *World Scientific and Engineering Academy and Society (WSEAS) Transactions on Systems*, 10(3):3047–3052, 2004.

[104] H. Najafi, Y. Genc, and N. Navab. Fusion of 3D and Appearance Models for Fast Object Detection and Pose Estimation. In *Proceedings of the Asian Conference on Computer Vision*, pages 415–426, 2006.

[105] T. Nierobisch, W. Fischer, and F. Hoffmann. Large view visual servoing of a mobile robot with a pan-tilt camera. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3307–3312, 2006.

[106] T. Nierobisch and F. Hoffmann. Appearance Based Pose Estimation of AIBO's. In *Proceedings of the International IEEE Conference Mechatronics & Robotics*, volume 3, pages 942–947, 2004.

[107] T. Nierobisch and F. Hoffmann. 2DOF Pose Estimation of Textured Objects with Angular Color Cooccurrence Histograms. In *Proceedings of the 2nd International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 52–59, 2007.

[108] T. Nierobisch, F. Hoffmann, J. Krettek, and T. Bertram. Bildbasierte Navigation eines mobilen Roboters mittels omnidirektionaler und schwenkbarer Kamera. In *20. Fachgespräch Autonome Mobile Systeme, (Informatik Aktuell)*, pages 75–81. Springer, 2007.

[109] T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Weighted moments of SIFT Features for decoupled visual servoing in 6DOF. In *Proceedings of the IEEE Conference on Advances in Cybernetic Systems (AICS)*, pages 193–198, 2006.

[110] T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Optimal Large View Visual Servoing with Sets of SIFT Features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2092–2097, 2007.

[111] T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Time-Optimal Large View Visual Servoing with Dynamic Sets of SIFT. Technical Report Reihe Computational Intelligence, CI 227/07, SFB 531, Universität Dortmund, 2007.

[112] T. Nierobisch, K. K. Narayanan, F. Hoffmann, and T. Bertram. Bildbasierte Navigation mobiler Roboter mittels omnidirektionaler Wahrnehmung. In *Mechatronik 2007, Innovative Produktentwicklung, Tagung Wiesloch, VDI-Berichte Nr. 1971*, pages 435–446. VDI-Verlag Düsseldorf, 2007.

[113] T. Nierobisch, K. Patel, J. Malzahn, F. Hoffmann, and T. Bertram. Rapid Prototyping of Visual Servoing Controllers with Virtual Reality. In *Proceedings of the 6th Polish-German Mechatronic Workshop 2007: System Integration, Ilmenau*, pages 109–120, 2007.

[114] T. Nierobisch, T. Schleginski, and F. Hoffmann. Reactive behaviours for visual topological navigation of a mobile robot. In *Proceedings of the 10th International Conference on Optimisation of Electrical and Electronic Equipment (OPTIM)*, volume 3, pages 113–118, 2006.

[115] D. Nister. An efficient solution to the five-point relative pose problem. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 195–202, 2003.

[116] R. Ozawa and F. Chaumette. Dynamic visual servoing with image moments for a quadrotor using a virtual spring approach. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5670–5676, 2011.

[117] J. Pagès, C. Collewet, F. Chaumette, and J. Salvi. Optimizing plane-to-plane positioning tasks by image-based visual servoing and structured light. *IEEE Transactions on Robotics*, 22(5):1000–1010, 2006.

[118] L. Pari, J. Sebastian, A. Traslosheros, and L. Angel. A comparative study between analytic and estimated image jacobian by using a stereoscopic system of cameras. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6208–6215, 2010.

[119] V. N. Peri and S. Nayar. Real Generation of Perspective and Panoramic Video from Omnidirectional Video. In *Proceedings of the DARPA Image Understanding Workshop*, pages 243–245, 1997.

[120] J. Piazzi and N. J. Cowan. Multi-View Visual Servoing using Epipoles. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 674–679, 2004.

[121] L. F. Posada, T. Nierobisch, F. Hoffmann, and T. Bertram. Image Signal Processing for Visual Door Passing with an Omnidirectional Camera. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISSAPP)*, pages 472–479, 2009.

[122] U. Reiser and J. Kubacki. Using a 3D Time-Of-Flight Range Camera for visual tracking. In *Proceedings of the 6th IFAC Symposium on Intelligent Autonomous Vehicles*, 2007.

[123] A. Remazeilles, F. Chaumette, and P. Gros. 3D navigation based on a visual memory. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2719–2725, 2006.

[124] P. Rives. Visual servoing based on epipolar geometry. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 602–607, 2000.

[125] E. Rosten, R. Porter, and T. Drummond. Faster and better: A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):105–119, 2010.

[126] RST Webpage, 2013. `http://www.rst.e-technik.tu-dortmund.de/cms/de/Forschung/Schwerpunkte/Robotik/Bildbasierte_Navigation/index.html#Navigation`.

[127] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571, 2011.

[128] A. Saffiotti, K. Konolige, and E. Ruspini. A multivalued-logic approach to integrating planning and control. *Artificial Intelligence*, 76(1-2):481–526, 1995.

[129] A. C. Sanderson and L. E. Weiss. Image-based visual servo control using relational graph error signals. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1074–1077, 1980.

[130] A. Santamaria-Navarro and J. Andrade-Cetto. Uncalibrated Image-Based Visual Servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5227–5232, 2013.

[131] S. Schaal. Learning from demonstration. In *Advances in Neural Information Processing Systems*, volume 9, pages 1040–1046. MIT Press, 1997.

[132] B. Schiele and J. L. Crowley. Recognition without correspondence using multidimensional receptive field histograms. In *International Journal of Computer Vision*, volume 36, pages 31–52, 2000.

[133] B. Schiele and A. Pentland. Probabilistic object recognition and localization. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 177–182, 1999.

[134] O. Schreer. *Stereoanalyse und Bildsynthese*. Springer Berlin Heidelberg, 2005.

[135] J. Shi and C. Tomasi. Good features to track. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, 1994.

[136] C. Shu. Automatic Grid Finding in Calibration Patterns using Delaunay Triangulation. Technical Report NRC-46497/ERB-1104, 2003.

[137] N. T. Siebel, O. Lang, F. Wirth, and A. Gräser. Robust Positioning of a Robot by Visual Servoing using a Trust-Region Method. In *Forschungsbericht der Deutschen Forschungsvereinigung für Meß-, Regelungs- und Systemtechnik (DFMRS) e.V.*, volume 1, pages 23–29, 1999.

[138] R. Sim, P. Elinas, and M. Griffin. Vision-based SLAM using the rao-blackwellised particle filter. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) - Workshop on Reasoning with Uncertainty in Robotics*, pages 9–16, 2005.

[139] S. Soatto and R. Brocket. Optimal and Suboptimal Structure from Motion. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 282–288, 1997.

[140] S. A. Stoeter, F. Le Mauff, and N. P. Papanikolopoulos. Real-time door detection in cluttered environments. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 187–192, 2000.

[141] A. Stopp, T. Baldauf, S. Horstmann, and S. Kristensen. Ein Sicherheitskonzept für Roboterassistenten in der Fertigung. *Automatisierungstechnische Praxis 2/2005*, pages 69–73, 2005.

[142] H. Strasdat, J. M. M. Montiel, and A. J. Davison. Scale drift-aware large scale monocular slam. In *Proceedings of Robotics: Science and Systems (RSS)*, 2010.

[143] O. Tahri and F. Chaumette. Point-based and region-based image moments for visual servoing of planar objects. *IEEE Transactions on Robotics and Automation*, 21(6):1116–1127, 2005.

[144] B. Tamadazte, G. Duceux, N.-F. Piat, and E. Marchand. Highly precise micropositioning task using a direct visual servoing scheme. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5689–5694, 2011.

[145] A. Y. Tamtsia, O. Tahri, Y. Mezouar, and E. Tonye. New Results in Image Moments-Based Visual Servoing. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5251–5256, 2013.

[146] Technical Committee on Mobile Manipulation, 2013. `http://mobilemanipulation.org/`.

[147] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. MIT Press, 2005.

[148] M. Tomono. 3D localization based on visual odometry and landmark recognition using image edge points. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5953–5959, 2010.

[149] H. Ulmer, F. Streichert, and A. Zell. Model-assisted steady-state evolution strategies. *Genetic and Evolutionary Computation (GECCO), LNCS*, 2723:610–621, 2003.

[150] VDI. VDI-Richtlinie: Entwicklungsmethodik für mechatronische Systeme. VDI 2206, 2004.

[151] L. Weiss, A. C. Sanderson, and C. P. Neuman. Dynamic Sensor-Based Control of Robots with Visual Feedback. *IEEE Journal on Robotics and Automation*, RA-3(1):404–417, 1987.

[152] G. Welch and G. Bishop. An Introduction to the Kalman Filter. Technical report, 1995.

[153] S. Wenxia and J. Samarabandu. Investigating the Performance of Corridor and Door Detection Algorithms in Different Environments. In *Proceedings of the International Conference on Information and Automation (ICIA)*, pages 206–211, 2006.

[154] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor. Omni-directional Vision for Robot Navigation. In *Proceedings of the IEEE Workshop on Omnidirectional Vision (Omnivis)*, pages 21–28, 2000.

[155] H. Zhang, B. Li, and D. Yang. Keyframe detection for appearance-based visual slam. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2071–2076, 2010.

Previously published contents of this thesis

Parts of the material presented in this thesis has been originally published in conferences and journals by the author. In the following these publications are ordered by chapter. The papers with the major contributions for the scientific community are also cited in the bibliography.

Chapter 3

- F. Hoffmann and T. Nierobisch. Bildgestützte Navigation von mobilen Robotern mit einem omnidirektionalen Kamerasystem. *40. Regelungstechnisches Kolloquium, Kurzfassung der Beiträge, Boppard,* pages 34-35, 2006.
- T. Nierobisch, T. Schleginski, and F. Hoffmann. Reactive behaviours for visual topological navigation of a mobile robot. In *Proceedings of the 10th International Conference on Optimisation of Electrical and Electronic Equipment (OPTIM´06), Brasov, Rumania,* volume 3, pages 113-118, 2006.

Chapter 4

- T. Nierobisch, F. Hoffmann, J. Krettek, and T. Bertram. Bildbasierte Navigation eines mobilen Roboters mittels omnidirektionaler und schwenkbarer Kamera. *20. Fachgespräch Autonome Mobile Systeme, Kaiserslautern, Springer, (Informatik Aktuell),* pages 75-81, 2007.
- T. Nierobisch, K. K. Narayanan, F. Hoffmann, and T. Bertram. Bildbasierte Navigation mobiler Roboter mittels omnidirektionaler Wahrnehmung. *Mechatronik 2007, Innovative Produktentwicklung, Tagung Wiesloch. VDI-Berichte Nr. 1971. Düsseldorf: VDI-Verlag,* pages 435-446, 2007.
- T. Nierobisch, W. Fischer, and F. Hoffmann. Large view visual servoing of a mobile robot with a pan-tilt camera. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2006), Beijing, China,* pages 3307-3312, 2006.

- L.-F. Posada, T. Nierobisch, F. Hoffmann, and T. Bertram. Image Signal Processing for Visual Door Passing with an Omnidirectional Camera. In *Proceedings of the International Conference on Computer Vision Theory and Applications (VISSAPP),* pages 472-479, 2009.

Chapter 5

- T. Nierobisch, K. V. Patel, J. Malzahn, F. Hoffmann, and T. Bertram. Rapid Prototyping of Visual Servoing Controllers with Virtual Reality. In *Proceedings of the 6th Polish-German Mechatronic Workshop 2007 "System Integration", Ilmenau,* pages 109-120, 2007.
- T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Weighted moments of SIFT Features for decoupled visual servoing in 6DOF. In *Proceedings of the IEEE Conference on Advances in Cybernetic Systems (AICS2006), Sheffield Hallam University, United Kingdom,* pages 193-198, 2006.
- F. Hoffmann, T. Nierobisch, T. Seyffarth, and G. Rudolph. Visual Servoing with Moments of SIFT Features. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC 2006), Taipeh, Taiwan,* pages 4262-4267, 2006.
- U. Khan, T. Nierobisch, F. Hoffmann, and T. Bertram. Evolutionäre Hardware-in-the-Loop Optimierung bildbasierter Regler. In J. Gausemeier, F. Rammig, W. Schäfer, A. Trächtler and J. Wallaschek, editors, *5. Paderborner Workshop Entwurf mechatronischer Systeme, HNI-Verlagsschriftenreihe Band 210,* pages 81-97, 2007.

Chapter 6

- T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Optimal Large View Visual Servoing with Sets of SIFT Features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA),* pages 2092-2097, 2007.
- T. Nierobisch, J. Krettek, U. Khan, and F. Hoffmann. Time-Optimal Large View Visual Servoing with Dynamic Sets of SIFT. *Reihe Computational Intelligence, CI 227/07, SFB 531, Universität Dortmund,* 2007.
- J. Krettek, T. Nierobisch, F. Hoffmann, and T. Bertram. Zeitoptimale bildbasierte Weitbereichsregelung zur Positionierung eines Industrieroboters. *GMA-Kongress 2007, Automation im gesamten Lebenszyklus, Kongress Baden-Baden. VDI-Berichte Nr. 1980. Düsseldorf: VDI-Verlag,* pages 297-307, 2007.

Acknowledgements

This thesis results from my activity at the chair for Control and Systems Engineering (Regelungssystemtechnik) at the Technische Universität Dortmund. First of all I would like to thank my supervisors Univ.-Prof. Dr.-Ing. Prof. h.c. Dr. h.c. Torsten Bertram and apl. Prof. Dr. rer. nat. Frank Hoffmann for offering me the opportunity to work in their group and for their encouragement and expertise. I would like to express my gratitude to Mr. Hoffmann for inspiring me to pursue research in the first place and his extensive support and many invaluable discussions. I especially thank Univ.-Prof. Dr.-Ing. Bernd Tibken for taking the time to act as the second examiner for my thesis.

Thank you René Franke und Jörn Malzahn for the critical reading of the manuscript. I am grateful for the discussions about science and general life issues with my former office colleague Johannes Krettek. I would like to thank Jürgen Limhoff for his invaluable technical support and the ability to endure my fast-paced ideas. I am also greatly indebted to a large amount of students who contributed with their work results in the context of project groups, student research projects and diploma and master theses to parts of this manuscript, especially Krishna Kumar Narayanan and Luis Felipe Posada Aristizabal. Thank you Mareike and Gaby for your support and sometimes inconspicuous help, which had nonetheless a great effect on my work. Furthermore, I would like to thank **all other people from RST** which are not named explicitly here for the wonderful working atmosphere.

Special thanks go to my wife Dr. Simone Streit-Nierobisch for her love, support and understanding during the thesis, especially in the last time-consuming weeks (or years) of finishing this work. To my son Tom Vinzenz for his critical comments during the writing even though I didn't completely understand them. Finally, I would like to dedicate this thesis to my parents Renate and Dr. Horst Nierobisch as well as my sister Anne Franziska who have always supported me throughout my life. I am deeply indebted to my mom Renate for the extensive babysitting of Tom at the final stages of this thesis.

Dortmund, December 2, 2015 Thomas Nierobisch