WS 2011-2012, MT-MA3, No. DISAL-MP15 20.09.2011–16.03.2012

Rafael Mosberger

Vision-based Tracking of Humans wearing a Reflective Vest using a Single Camera System

Master Thesis, 2012, Microengineering

Supervision: Mobile Robotics and Olfaction Lab, MR&O
Centre for Applied Autonomous Sensor Systems, AASS
Department of Technology, Örebro University
Professor: Achim J. Lilienthal / Assistant: Henrik Andreasson

Co-Supervision: Distributed Intelligent Systems and Algorithms Lab, DISAL
École Polytechnique Fédérale de Lausanne, EPFL
Professor: Alcherio Martinoli / Assistant: Amanda Prorok

Contents

Abstract

This thesis presents a possible solution for people detection and tracking in industrial environments shared between machines and humans. Addressing safety critical applications, we make the basic assumption that people wear reflective vests. In order to detect these vests and to discriminate them from other reflective materials, we propose an approach based on a single camera system equipped with an infrared flash and an infrared bandpass filter.

The camera acquires pairs of images, one with and one without IR flash, in short succession. The image pairs are related to each other through feature detection and tracking, which allows to identify a set of interest points for which the relative intensity difference is high and which are thus believed to originate from a reflective vest. The local neighborhood of these features is then further observed. Based on a local image descriptor, a Random Forest classifier is applied to discriminate between features caused by a reflective vest and features caused by other reflective materials. For features classified as a reflective vest, the distance between camera and vest is estimated by a Random Forest regressor, again on the basis of the local image descriptor. The distance estimates combined with the intrinsic camera model allow to estimate the 3D position relative to the camera for every vest feature. Finally, a particle filter incorporates the single position estimates and keeps track of the position of a reflective vest over time.

The proposed system is evaluated in several indoor and outdoor environments and under different weather conditions. The results indicate good classification performance and promising accuracy in position estimation and tracking.

Symbols

$\boldsymbol{I}' = (I'_f, I'_{nf})$	Raw input image pair
$\boldsymbol{I} = (I_f, I_{nf})$	Unwrapped input image pair
$\boldsymbol{u} = [u, v]^T$	Image coordinate pair
f_a	Image pair acquisition rate
t_a	Time delay between acquisition of I'_f and I'_{nf}
f	Visual image feature
$\mathcal{F}$	Set of image features
$\boldsymbol{r} = [r_1, ..., r_{N_r}]^T$	Image feature descriptor
$\mathcal{R}$	Set of image feature descriptors
$\hat{p}_{vest}$	Probability that a feature represents a reflective vest
$\hat{c}$	Random Forest class estimate
$\tilde{c}$	Ground-truth class label
$\hat{d}$	Random Forest distance estimate
$\tilde{d}$	Ground-truth distance label
$\hat{\boldsymbol{p}}$	3D position estimate
$\mathcal{P}$	Set of 3D position estimates
$\mathcal{S}_t$	Set of particles at time t
$\boldsymbol{s}_t$	System state at time t
$Bel(\boldsymbol{s}_t)$	Belief distribution over state $\boldsymbol{s}_t$

Chapter 1
Introduction

People detection is an important task in both autonomous machines and human operated vehicles equipped with driver assistant technology. Especially when it comes to applications where machines operate in industrial workspaces shared with humans, it plays a crucial role towards improved safety for the operators and their co-workers. Different sensor modalities are commonly used in people detection, including laser scanners and vision-based systems with visible light and thermal imaging sensors. All approaches suffer from certain drawbacks in safety critical applications. Conventional 2D laser scanners represent the de-facto safety standard equipment for automated guided vehicles (AGVs) that operate in indoor applications on flat ground. In uneven terrain, 3D laser scanners can be employed but they come with a very high price. Thermal cameras are also expensive and their use depends on the ambient temperature. Systems based on conventional cameras usually offer an inexpensive solution but require that the ambient illumination is neither too strong nor too weak. Yet, for the application in safety systems dedicated to industrial environments, reliable people detection in a variety of different conditions is critical.

In many industrial workplaces such as manufacturing areas, construction sites, warehouses or storage yards, the wearing of a reflective safety vest (cf. Figure 1.1) is a legal requirement. In contrast to more general approaches, the work presented in this thesis therefore takes advantage of the enhanced visibility of a person due to the reflective vest to facilitate the detection. Andreasson et al. [1] introduced a people detection system based on a single camera unit which is able to detect humans wearing a reflective vest by detecting reflective material. Its core principle is to take two images in short succession, one with and one without infrared (IR) flash, and to process them as a pair. The algorithm identifies regions with a significant intensity difference between the two images in order to detect locations where reflective material appears.

Figure 1.1: Reflective Safety Vest

The goal of the underlying project is to optimize the existing camera system and extend it towards position estimation and temporal tracking of persons wearing reflective vests, based only on visual input.

1.1 Project Outline

The camera system proposed in [1] allows the detection of people wearing a reflective vest. The system was tested in indoor and outdoor environments and the results confirmed that the approach is promising. Yet, in its current state the system is unable to distinct between reflective vests and other reflective materials. A first part of the project is therefore dedicated to solve this shortcoming by performing binary classification of the detected reflective objects. Machine learning techniques shall be combined together with a robust image feature descriptor, extracted from the image regions where vests are suspected, to obtain the model of the classifier.

A fundamental extension of the system is then envisaged. In addition to the detection of reflective vests, the system shall be enabled to estimate the distance of a detected vest relative to the camera. Again, the proposed method consists in applying machine learning techniques. The performance of different image feature descriptors in combination with an appropriate regressor model will be evaluated. Once a distance estimate is obtained, a corresponding position estimate of the detected reflective vest in 3D space can easily be inferred using the intrinsic camera model.

The final goal of the project is the integration of the obtained position estimates into a recursive state estimation filter. The filter is supposed to keep track of a reflective vest as the position of an observed person evolves over time. A probabilistic approach shall be adopted, taking into account that the individual position estimates are prone to errors.

The individual parts of the system will be evaluated in different scenarios including indoor and outdoor environments and different weather conditions. The results will allow to identify possible weaknesses of the system and form the basis for further improvements of both hardware and software.

1.2 Motivation

Vision-based people detection for non-stationary environments has been extensively studied for applications in robotic vehicles, (semi-)autonomous cars, driver assistant systems and surveillance. Solutions on purely visual input are interesting from an economic point of view as standard cameras represent an inexpensive sensor type. Yet, the performance of vision-based techniques heavily depends on the presence of good visible structures in the images, and thus on a sufficient illumination of the observed scene. Their application is usually not suitable for dim or completely dark environments. Also, vision-based approaches typically struggle in cases where people have little contrast with the background. For these reasons, existing people detection approaches are not directly applicable in safety critical applications that are supposed to operate under challenging conditions, such as rain, snow or direct exposure to sunlight.

To overcome this shortcomings, cameras are commonly used in combination with other sensor modalities and a large amount of scientific work deals with sensor fusion between cameras and laser scanners for people detection [2]. However, to the best of the author's knowledge, there exists no people detection system which makes use of the beneficial properties of a reflective vest in the detection process. The system presented in this paper focuses on the detection of people in industrial environments where the condition that workers wear a reflective vest is fulfilled.

Instead of analyzing single images as it is done in most of the related work, our system processes a pair of images, one of which is taken with an IR flash and one without. The proposed algorithm exploits the fact that the IR flash is very strongly reflected by the vest reflectors to detect locations in the image where a large intensity difference exists between the two images. It has been shown in [1] that especially at higher ranges where spatial resolution decreases rapidly in the image, the approach based on an image pair and the use of an IR flash outperforms a state-of-the-art people detection algorithm (Histogram of Oriented Gradient) applied to a single image that is acquired without active illumination.

1.3 Report Outline

This report is organized as follows. Chapter 2 introduces the hardware used for image acquisition and discusses the individual processing steps of the vest detection and tracking algorithm. In Chapter 3, the performance of the different parts of the system is evaluated in various environments. The evaluation results are discussed in Chapter 4 and conclusions are drawn in Chapter 5. Finally, Chapter 6 gives an outlook on further work in perspective of future improvements of the system.

Chapter 2
System Description

The reflective vest detection and tracking system presented in this report consists of a single camera unit and an ensemble of processing steps that compare two input images, one acquired with IR flash and one taken without, to estimate the position of a person wearing a reflective vest. In this chapter, the hardware components as well as the individual processing steps of the algorithm will be discussed in detail and in the order they are applied. Figure 2.1 depicts a schematic overview of the complete algorithm and shows how the individual steps are related.

The input of the system is a pair of raw images, one taken with IR flash and one without. The hardware components of the camera system that acquires the two images will be subject of Section 2.1 while the according intrinsic camera model is introduced in Section 2.2. Section 2.3 is dedicated to the acquisition process of a raw image pair $\boldsymbol{I}' = (I'_f, I'_{nf})$, where I'_f denotes the image acquired with IR flash and I'_{nf} the image taken without flash. The area of interest in is then extracted from the raw images and undistorted in a processing step referred to as *image unwrapping*, discussed in Section 2.4. The resulting pair of unwrapped images is denoted $\boldsymbol{I} = (I_f, I_{nf})$. Given the fact that the emitted IR flash is strongly reflected by the reflectors of a safety vest, the regions where such a vest appears in the images have distinctly higher intensity values in I_f compared to I_{nf}.

As it is discussed in Section 2.5, a feature detector is then applied to the image I_f taken with an IR flash, in order to identify a set $\mathcal{F}_{raw}$ containing high intensity blob-like interest points, referred to as *features*. If a reflective vest is visible in the image, one or several of these features will be detected in high intensity regions produced by the reflective material of the vest. However, the set potentially includes additional interest points representing other high-intensity regions in the image. Thus, several subsequent processing step aim at removing these non-vest points from the initial features set.

The features detected in image I_f are tracked in I_{nf} and, based on the output of the tracker, a subset of features is discarded as not belonging to reflective material and thus not originating from a reflective vest. Features are discarded if they are successfully tracked and if the intensity difference between the two images at the corresponding locations is below a defined threshold. This pre-selection step, described in Section 2.6, ideally results in a set $\mathcal{F}_{reflex}$ of features that represent reflective materials in the camera's field of view. To discriminate between reflective vests and other reflective materials, a binary Random Forest classifier is additionally applied. To do so, a feature descriptor $\boldsymbol{r}$ is extracted from image I_f for every feature in $\mathcal{F}_{reflex}$, according to Section 2.7. The descriptor represents a characteristic set of variables describing the visual content in the neighborhood of a feature in a much more robust way than the raw intensity values. The classification procedure as well as the supervised learning process applied to obtain the Random Forest classifier are subject of Section 2.8. The result of the feature detection process is a feature set $\mathcal{F}_{vest}$ in which all features are considered as to originate from a reflective vest.

Following the detection of reflective vests in the input images, the system estimates the 3D position of the reflective vest markers that caused the appearance of the corresponding features $\mathcal{F}_{vest}$ in the image I_f. As discussed in Section 2.9, a Random Forest regressor model that is again obtained by supervised learning allows to predict the distance $\hat{d}$ for all the features in $\mathcal{F}_{vest}$, based on the same local image descriptors that were previously used for classification. Section 2.10 then illustrates how a distance estimate is used in combination with the intrinsic camera model to obtain a position estimate $\hat{\boldsymbol{p}}$ in 3D space.

Finally, the vest tracking algorithm is introduced in Section 2.11. The vest tracker considers the scenario where image pairs are repeatedly acquired. In this scenario, a reflective vest not only needs to be detected in the individual image pairs but has to be tracked over a sequence of input images. The vest tracking algorithm provides a filtering mechanism that continuously incorporates the single vest position estimates $\hat{\boldsymbol{p}}$ to produce a final estimate of the system's state $\boldsymbol{s}_t$. The state $\boldsymbol{s}_t$ comprises the position and speed of a reflective vest as observed by the camera. The uncertainty about the exact location of a reflective vest is represented by a probability distribution over $\boldsymbol{s}_t$ that is approximated by a set of particles provided by a particle filter.

Due to multiple reasons that will be discussed in detail, the individual position estimates $\hat{\boldsymbol{p}}$ are subject to errors. The vest tracking algorithm therefore provides a probabilistic model that takes the uncertainty of the measurements into account. Furthermore, the tracker models the uncertainty that arises from the fact that the motion of an observed person can only be predicted vaguely.

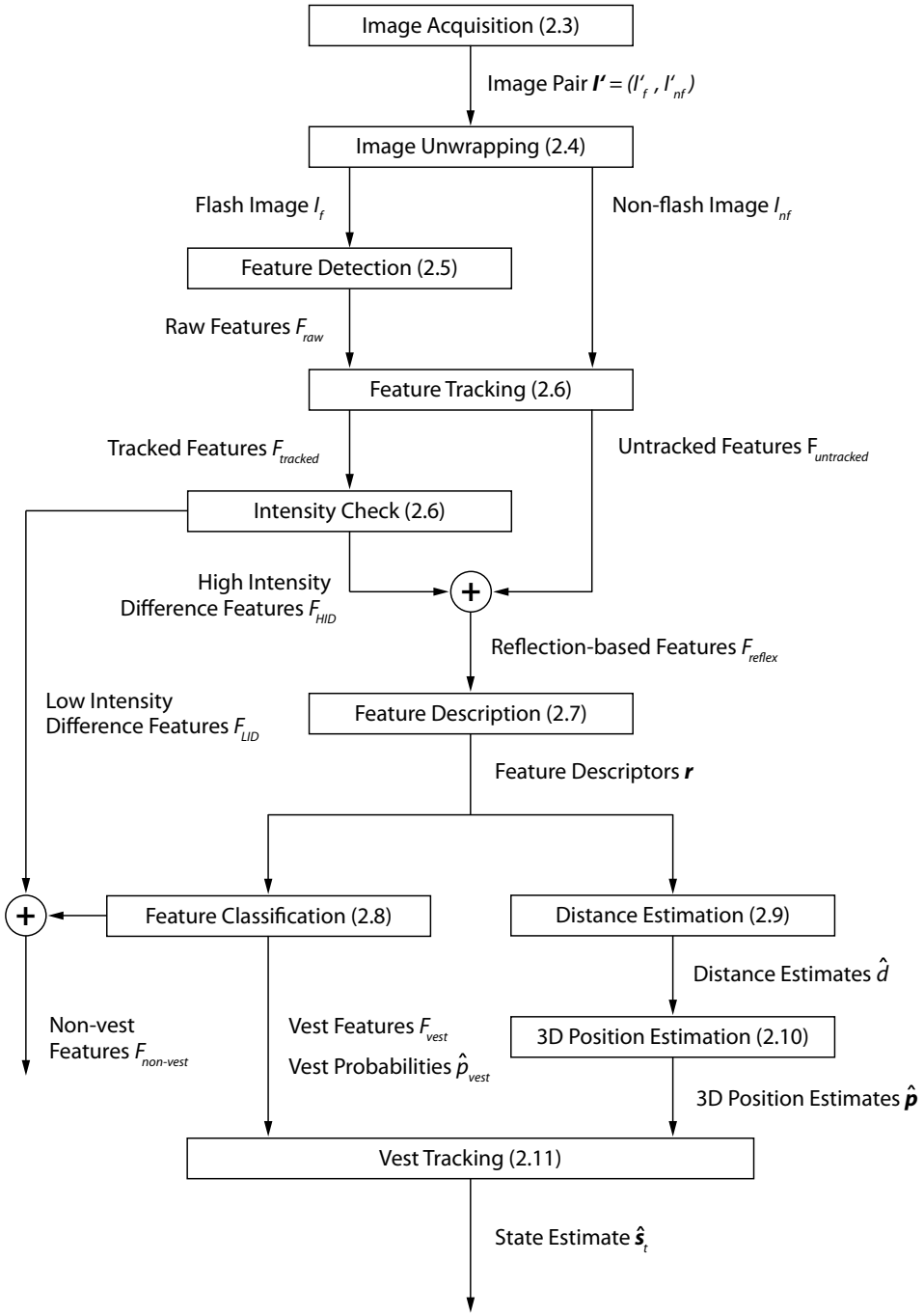


Figure 2.1: The figure shows an overview of the reflective vest detection and tracking system and indicates the data flow between the individual processing steps. The sections in which the different parts are discussed are indicated in brackets.

2.1 Hardware

The camera unit (cf. Figure 2.2) consists of a standard monochrome CMOS sensor (IDS imaging USB UI-1228LE) with a resolution of 752×480 pixels and a fish-eye lens with an approximate field of view (FOV) of $180\,^{\circ}$.

8 IR LEDs with a wavelength of 850 nm are placed in a ring around the lens to form an IR flash system. The characteristic emission of the LEDs reaches its maximum in the direction normal to the LED and is at 50 % at an angle of $60\,^{\circ}$. The arrangement of the LEDs assures a wide and relatively uniform illumination of the camera's FOV.

A bandpass filter with a center wavelength of 852 nm and a full width at half maximum of 10 nm is mounted between the lens and the sensor. The filter corresponds to the dominant IR wavelengths of the IR LEDs. Thus, it prevents all wavelengths that do not correspond to the narrow band emitted by the IR LEDs from entering the camera.

2.2 Camera Model

The usual pinhole camera model is not suitable to describe the perspective projection in a camera system featuring a fish-eye lens. Thus, the general parametric model for omnidirectional cameras introduced in [3] is adopted. The model defines three distinct references, the camera image plane (u', v') in pixel coordinates and the sensor plane (u'', v'') and the camera reference frame (x, y, z) in metric coordinates. The camera reference frame has its origin in the optical center O of the lens and its z-axis pointing in the direction of the optical axis of the lens.

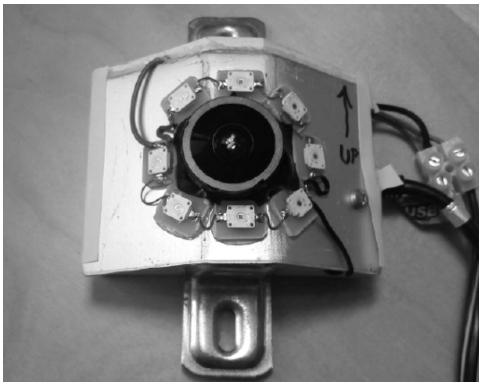


Figure 2.2: The single camera system used for image acquisition consists of a standard monochrome CMOS sensor, an infrared bandpass filter (not visible in the image), a fish-eye lens and a ring of 8 IR-LEDs.

Let us consider a scene point $Q = [x, y, z]^T$ in the camera reference frame and a unit vector $\boldsymbol{e}_Q \in \mathbb{R}^{3x1}$, located in O, and pointing in the direction of point Q. By introducing the image projection function $\boldsymbol{g} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$, the omni-directional camera model reads

$$\boldsymbol{e}_Q = \boldsymbol{g}(A\boldsymbol{u'} + \boldsymbol{t}) \tag{2.1}$$

with

$$\boldsymbol{g}(\boldsymbol{u''}) = \begin{pmatrix} u'' \\ v'' \\ g_z(u'', v'') \end{pmatrix} \tag{2.2}$$

where $g_z : \mathbb{R}^2 \rightarrow \mathbb{R}$ is a non-linear function, rotationally symmetric with respect to the sensor axis. The affine transformation $A\boldsymbol{u'} + \boldsymbol{t}$ accounts for possible axes misalignments between the sensor and image plane. Here, we adopt the approach introduced in [4] where the assumption is made that the function $g_z(u'', v'')$ can be described by a Taylor series expansion. Thus, for $\rho = \sqrt{u''^2 + v''^2}$:

$$g_z(u'', v'') = a_0 + a_1\rho + a_2\rho^2 + ... + a_N\rho^N \tag{2.3}$$

The projection equation (2.1) allows to reconstruct the direction of a 3D scene point corresponding to a given image coordinate pair $\boldsymbol{u'} = [u', v']^T$. We also introduce the inverse projection equation that assigns an image coordinate pair $\boldsymbol{u'}$ to every unit vector $\boldsymbol{e}_Q$, located in the optical center O of the lens and pointing to an arbitrary point Q in the camera reference frame:

$$\boldsymbol{u'} = A^{-1}[\boldsymbol{g}^{-1}(\boldsymbol{e}_Q) - \boldsymbol{t}] \tag{2.4}$$

The model parameters $[A, \boldsymbol{t}, a_0, a_1, a_2, ..., a_N]$ are obtained by intrinsic calibration of the camera according to [4].

2.3 Image Acquisition

The image acquisition involves taking a pair of images, one with IR flash and one without. The time increment t_a between the acquisition of the two images is kept as short as possible in order to minimize the difference between the two images due to changes in viewpoint and changes in the observed scene. The result of the image acquisition is a raw image pair $\boldsymbol{I'} = (I'_f, I'_{nf})$, consisting of the image I'_f taken with flash, and the image I'_{nf} taken without flash. We denote f_a the rate at which raw image pairs $\boldsymbol{I'}$ are acquired. Figure 2.3 shows an example of a raw image pair for the case of a reflective vest appearing in the field of view of the camera. The reader may take note that the very low average brightness of the images is a result of the infrared bandpass filter included in the camera hardware and is a desired property to simplify the vest detection process.

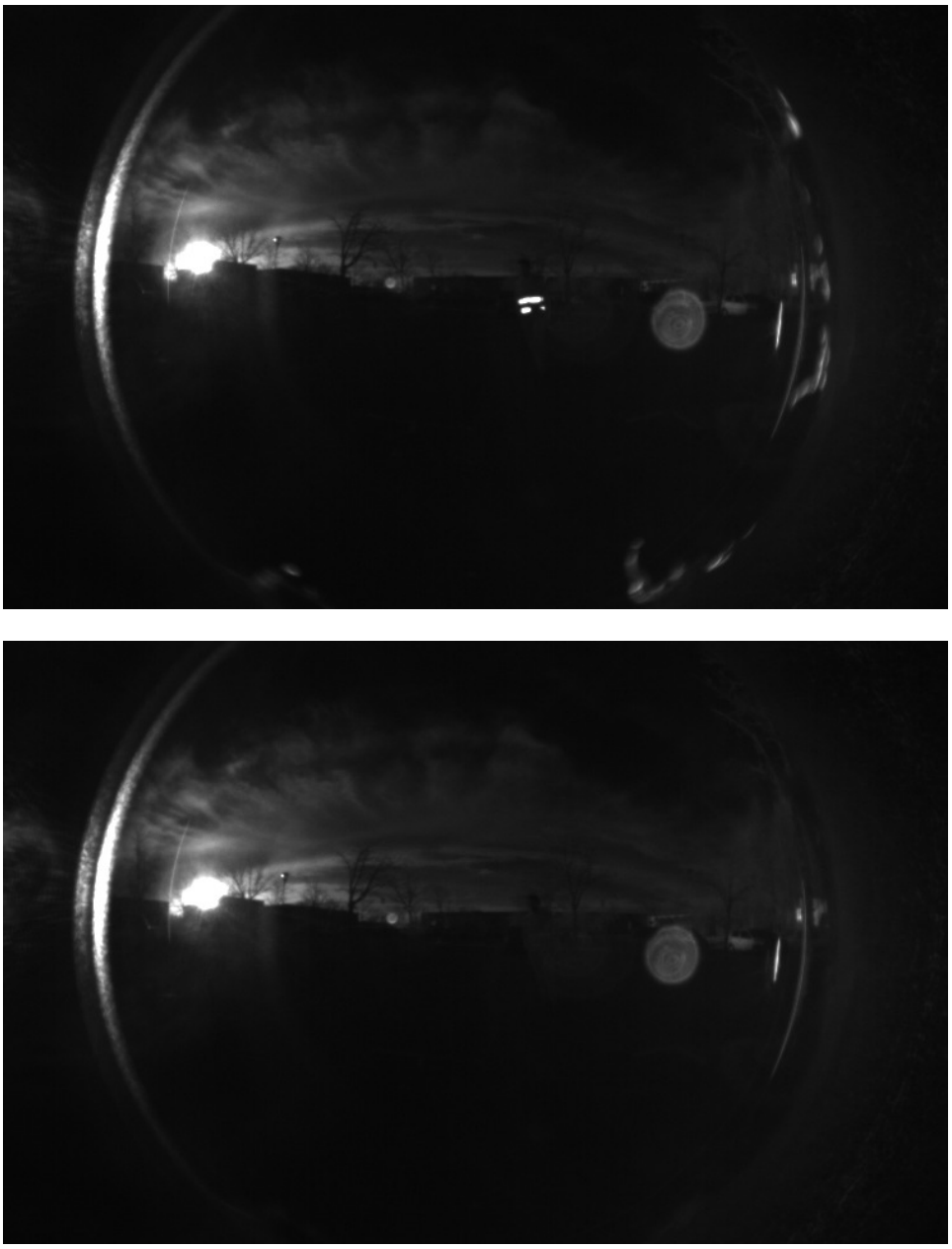


Figure 2.3: Example of a raw image pair $\boldsymbol{I'}$ taken in short succession. The image I'_f (above) was taken with IR flash and the image I'_{nf} (below) without. The difference in intensity values at the location where a reflective vest appears is clearly visible. The filled white circle on the center right represents a lens artifact originating from direct sunshine into the camera. It may be noted that the overall brightness of the images is very low due to the use of the IR bandpass filter in the camera system.

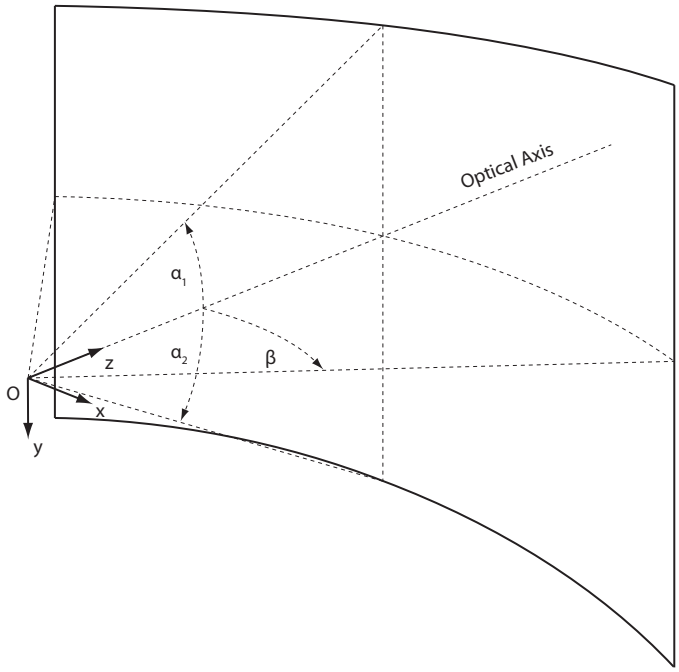


Figure 2.4: Parametrization of the virtual cylinder used to define the panoramic field of view during the unwrapping of the raw fish-eye images. The reference (x, y, z) indicates the orientation of the coordinate system attached to the camera.

2.4 Image Unwrapping

The raw fish-eye images I'_f and I'_{nf} are unwrapped to create an undistorted panoramic view containing the area of interest for the reflective vest detection. Figure 2.4 shows the parametrization of a virtual cylinder with unit radius used to create the panoramic images. The figure further defines the orientation of the camera reference frame (x, y, z) with its origin O lying in the optical center of the lens. The pair of images resulting from unwrapping will be named $\boldsymbol{I} = (I_f, I_{nf})$ and the corresponding image coordinates $\boldsymbol{u} = [u, v]^T$. The images are of width W and height H in pixels, related through the following relation to obtain undistorted images:

$$H = \frac{tan(\alpha_1) - tan(\alpha_2)}{2\beta} W \tag{2.5}$$

An image projection function $\boldsymbol{h} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ is defined for the panoramic image, allowing the construction of a unit length scene vector pointing in the direction corresponding to a pair of given image coordinates $\boldsymbol{u} = [u, v]^T$.

$$\boldsymbol{h}(\boldsymbol{u}) = \begin{pmatrix} cos(\phi)sin(\theta) \\ -sin(\phi) \\ cos(\phi)cos(\theta) \end{pmatrix} \tag{2.6}$$

with

$$\phi = \alpha_1 - \frac{v}{H-1}(\alpha_1 - \alpha_2) \tag{2.7}$$

and

$$\theta = \beta\left(\frac{2u}{W-1} - 1\right) \tag{2.8}$$

Thus, the intensity values $I_f(\boldsymbol{u})$ and $I_{nf}(\boldsymbol{u})$ of the unwrapped images are obtained by projecting the image coordinate pair $\boldsymbol{u}$ into the camera reference frame using Eq. 2.6 before projecting the obtained scene vector on the corresponding raw fish-eye image using Eq. 2.4 of the camera model to obtain the fish-eye coordinate pair $\boldsymbol{u'} = [u', v']^T$:

$$I_{f,nf}(\boldsymbol{u}) = I'_{f,nf}(A^{-1}[\boldsymbol{g}^{-1}(\boldsymbol{h}(\boldsymbol{u})) - \boldsymbol{t}]) \tag{2.9}$$

Figure 2.5 shows the panoramic image pair $\boldsymbol{I}$ resulting from unwrapping of the raw image pair $\boldsymbol{I'}$ shown in Figure 2.3.

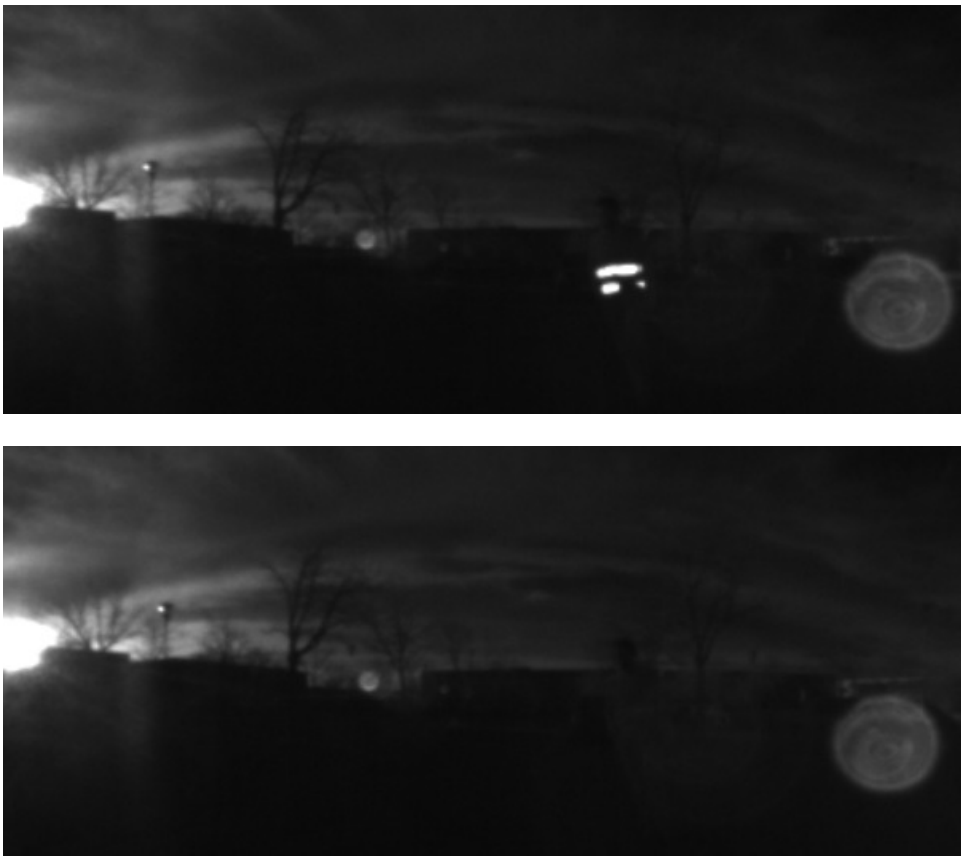


Figure 2.5: Example of an unwrapped image pair $\boldsymbol{I}$ corresponding to the raw image pair $\boldsymbol{I'}$ shown in Figure 2.3 with flash image I_f (top) and non-flash image I_{nf} (bottom).

2.5 Feature Detection

The reflection of the IR light by the reflectors of a vest results in high intensity blob-like regions at locations where the vest appears in the image I_f. Shape and size of the high intensity regions depend heavily on the distance between the camera unit and the person wearing the vest as well as on the body pose of the person. Especially at short distances, the reflective markers of a vest appear as elongated regions rather than as circular blobs. Furthermore, a vest can be partly occluded by objects between the person and the camera. Figure 2.6 depicts a selection of the variety of different patterns that the reflection of the IR light on the reflective vest produces in the image I_f.

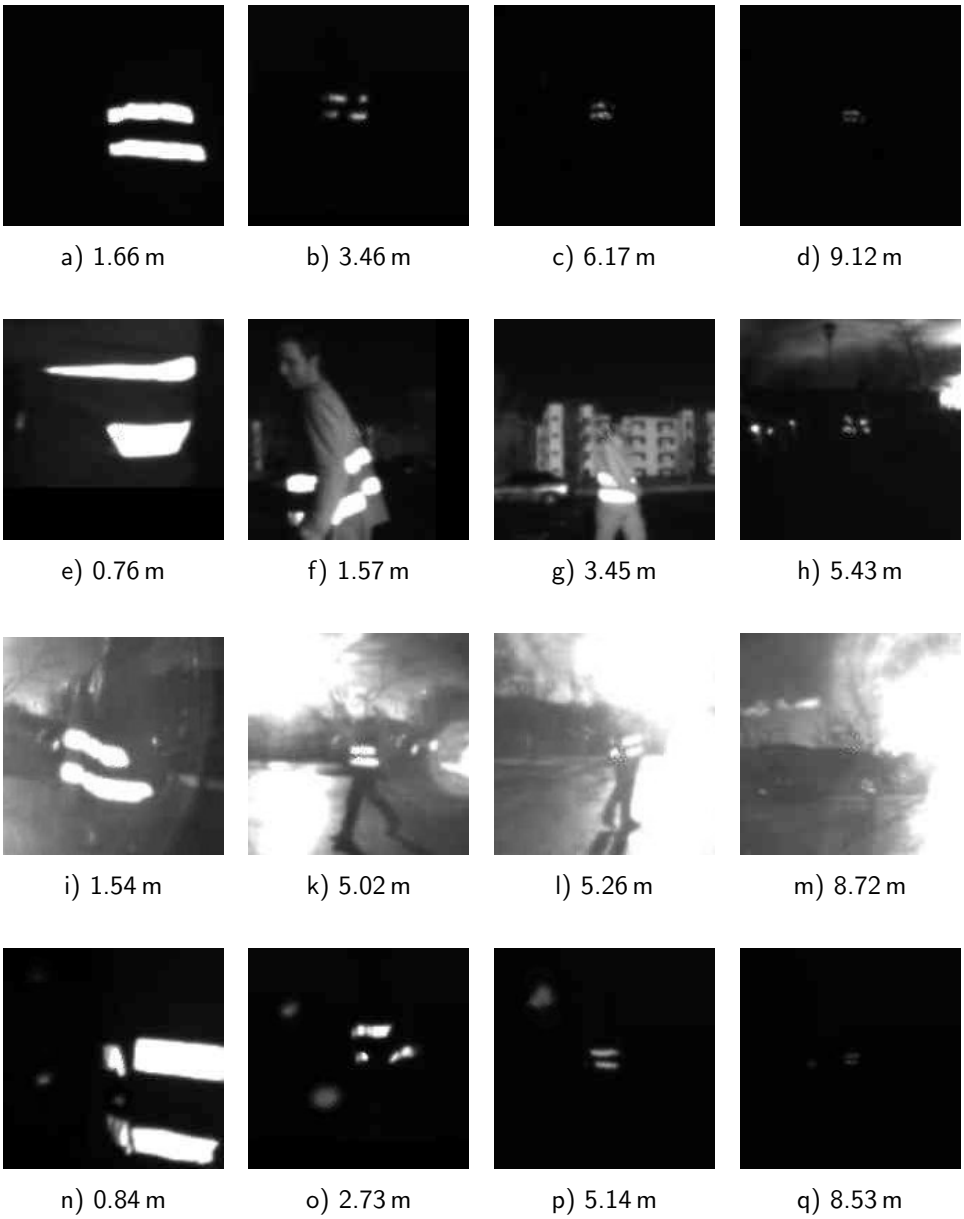
Based on the above observations, the assumption is made that the high-intensity image patterns produced by a reflective vest can be represented by a single blob at higher distances and by several individual blobs at near distances. Based on this assumption, the first step in the vest detection process consists in identifying in the image I_f a set of interest points at locations where such high intensity regions appear.

A large variety of interest point detectors exists that can be regrouped mainly into edge detectors, corner detectors and blob detectors, according to the type of image features that are detected. Popular blob detectors include the Laplacian of Gaussian(LoG), Difference of Gaussians (DoG), Maximally Stable Extremal Regions (MSER) or grey-level blobs. The choice of a suitable blob detector for our application is limited by real-time constraints as well as by the need for scale-invariance, a property which is important in order to detect reflective vest features of different size. Our application uses the STAR algorithm by Konolige et al. which is a speeded-up version of the Center Surround Extrema (CenSurE) feature detector [5]. The STAR algorithm is computationally efficient and complies with our scale-invariance requirements. In our application, we slightly modified the detector in order to respond only to positive intensity peaks.

The result of the STAR detector applied to the image I_f is a set $\mathcal{F}_{raw}$ of N_f interest points, referred to as *features* f_i, $i = 1, ..., N_f$, where every feature is described by its scale s and its image coordinate pair $\boldsymbol{u}_f$ indicating the location in the image I_f where the feature was detected:

$$\mathcal{F}_{raw} = \left\{ f^{[i]} = \left\langle s^{[i]}, \boldsymbol{u}_f^{[i]} \right\rangle \mid i = 1, ..., N_f \right\} \tag{2.10}$$

An exemplary result of the feature detection is given by the ensemble of black circles in Figure 2.7. The example illustrates that under the influence of the IR illumination from the flash and the sun, the detected feature set $\mathcal{F}_{raw}$ includes many features that do not originate from a reflective vest. Also it is worth mentioning that due to the STAR algorithm's sensitivity to circular shapes, one reflective vest marker can be detected more than once, especially when its shape appears elongated.



a) 1.66 m b) 3.46 m c) 6.17 m d) 9.12 m

e) 0.76 m f) 1.57 m g) 3.45 m h) 5.43 m

i) 1.54 m k) 5.02 m l) 5.26 m m) 8.72 m

n) 0.84 m o) 2.73 m p) 5.14 m q) 8.53 m

Figure 2.6: The figure shows examples of image patterns that result from the reflection of the IR flash light on the reflective vest material during the acquisition of image I_f. The corresponding distance between the camera and the vest is indicated for each example. The image patches show the variety of patterns that is encountered, namely **a-d)** indoors without any other IR light source than the flash, **e-h)** outdoors under the influence of sunlight, **i-m)** outdoors with direct sunshine into the camera, and **n-q)** outdoors with perturbing reflections on snowflakes.

Figure 2.7: The figure illustrates the result of the feature detection process applied to an image I_f taken with IR flash. A reflective vest is visible in the center of the image. Detected features are drawn as black circles, the size of which indicates the feature scale s. The figure shows that under the influence of the IR light emitted by the sun, the feature set $\mathcal{F}_{raw}$ includes many features that do not originate from a reflective vest.

2.6 Feature Tracking and Intensity Check

The detected features in the set $\mathcal{F}_{raw}$ originate either from a reflective material or from another bright object in the FOV of the camera. As the images I_f and I_{nf} were taken in short succession, the appearance of non-reflective features changes little from one image to another. In contrast, this assumption is not valid for features originating from reflective material since the intensity values in the vicinity of such features differ considerably for the image pair $\boldsymbol{I}$, as it has been illustrated in Figure 2.3. Based on this property, the first processing step to eliminate non-vest features consists in tracking every raw feature $f \in \mathcal{F}_{raw}$, detected in image I_f, in the corresponding image taken without IR flash, I_{nf}, and in evaluating the intensity difference for successfully tracked features.

Let's consider the image pair $\boldsymbol{I} = (I_f, I_{nf})$ and the set $\mathcal{F}_{raw}$ of raw features detected in image I_f. Given the location $\boldsymbol{u}_f$ of a feature f in the image I_f, the goal of the LK feature tracker is to determine the corresponding location $\boldsymbol{u}_{nf} = \boldsymbol{u}_f + \boldsymbol{\Delta}$ in the image I_{nf}, so that $I_f(\boldsymbol{u}_f)$ and $I_{nf}(\boldsymbol{u}_{nf})$ are similar in a defined local neighborhood. The vector $\boldsymbol{\Delta} = [\Delta_u, \Delta_v]^T$ is referred to as the *displacement vector* or the *optical flow* at location $\boldsymbol{u}_f$. The algorithm achieves its goal by minimizing the function $\epsilon_{LK}(\boldsymbol{u}_{nf})$, defined as

$$\epsilon_{LK}(\boldsymbol{u}_{nf}) = \sum_{m=-\omega_{LK}}^{\omega_{LK}} \sum_{n=-\omega_{LK}}^{\omega_{LK}} (I_f(u_f + m, v_f + n) - I_{nf}(u_{nf} + m, v_{nf} + n))^2 \tag{2.11}$$

which represents the squared sum of intensity differences in a square neighborhood of size $(2\omega_{LK} + 1)$ of $I_f(\boldsymbol{u}_f)$ and $I_{nf}(\boldsymbol{u}_{nf})$.

The tracking is performed using a pyramidal implementation of the iterative Lucas-Kanade (LK) feature tracking method [6]. The LK algorithm is based on three major assumptions. The first, named *temporal persistence*, implies that the time increment between the two images is small enough such that the location of a feature changes little from one image to another. This is assured by the fact that the images I_f and I_{nf} are taken in very short succession. Secondly, *spatial coherence* is assumed, meaning that neighboring points in the first image (here I_f) belong to the same surface and therefore have similar motion and stay neighboring points in second image (here I_{nf}). The third and final assumption, referred to as *brightness constancy*, stands for the property that an object does not change in appearance from one image to another and its brightness therefore remains similar.

If the above key assumptions hold true for an image pair $\boldsymbol{I}$ and a feature $f \in \mathcal{F}_{raw}$, the tracker usually succeeds to track the feature in image I_{nf}. We collect these successfully tracked features in a subset of the raw features, named $\mathcal{F}_{tracked}$:

$$\mathcal{F}_{tracked} = \{f \in \mathcal{F}_{raw} \mid f \text{ is successfully tracked}\} \tag{2.12}$$

For successfully tracked features, their locations $\boldsymbol{u}_f$ in image I_f and $\boldsymbol{u}_{nf}$ in image I_{nf} are known and the neighborhoods of both locations can be compared to determine how similar they are. Very alike intensity values in both neighborhoods indicate that the feature does not originate from reflective material. In return, higher intensity values in the neighborhood of $\boldsymbol{u}_f$ suggest that the feature represents reflective material. We therefore submit every tracked feature to an *intensity difference check* by evaluating the mean value of the absolute intensity differences in a square neighborhood of size $(2\omega_{ID} + 1)$ around $\boldsymbol{u}_f$ in I_f and $\boldsymbol{u}_{nf}$ in I_{nf}, according to

$$\epsilon_{ID} = \sum_{m=-\omega_{ID}}^{\omega_{ID}} \sum_{n=-\omega_{ID}}^{\omega_{ID}} \frac{I_f(u_f + m, v_f + n) - I_{nf}(u_{nf} + m, v_{nf} + n)}{(2\omega_{ID} + 1)^2} \tag{2.13}$$

and we define a set of high-intensity difference features $\mathcal{F}_{HID}$ as follows:

$$\mathcal{F}_{HID} = \{f \in \mathcal{F}_{tracked} \mid \epsilon_{ID} > \lambda_{ID}\} \tag{2.14}$$

Tracked features for which instead $\epsilon_{ID} \leq \lambda_{ID}$ are considered as to originate from an area without reflective material and will not be further processed in the detection of reflective vests. We collect them in a set of low-intensity difference features $\mathcal{F}_{LID}$, according to:

$$\mathcal{F}_{LID} = \mathcal{F}_{tracked} \setminus \mathcal{F}_{HID} \tag{2.15}$$

The described approach of comparing the intensity values of the images I_f and I_{nf} relies on the successful tracking of features. If, in contrast, one

or several of the key assumptions mentioned above are not verified for a feature $f \in \mathcal{F}_{raw}$, the LK feature tracker usually fails to track it. In the case of features originating from reflective material, the *brightness constancy* assumption is clearly violated as the intensity values in the neighborhood of such a feature are much higher in the image I_f than in I_{nf}, due to the reflection of the IR flash. In most of the cases, the tracker is therefore unable to find any suitable match in the image I_{nf} and the feature is labeled as untracked. We collect these untracked features in a subset of $\mathcal{F}_{raw}$, named $\mathcal{F}_{untracked}$:

$$\mathcal{F}_{untracked} = \mathcal{F}_{raw} \setminus \mathcal{F}_{tracked} \tag{2.16}$$

We finally define a set of beliefed reflection-based features $\mathcal{F}_{reflex}$ including the untracked features as well as the tracked features that show a high intensity difference in both images:

$$\mathcal{F}_{reflex} = \mathcal{F}_{untracked} \cup \mathcal{F}_{HID} \tag{2.17}$$

It is worth noting that in contrast to the standard application of a feature tracker, we in our detection scheme are not only interested in features that can be successfully tracked. In fact, we also specifically identify features that cannot be tracked as possible vest features, assuming that the reason for the inability to track them is the violation of the brightness constancy assumption.

However, there are two more reasons for which a feature might not be tracked. First, the movement of an object in the image relative to the background leads to the violation of the *spatial coherence* assumption, as neighboring points in the image can have non-similar motion in this case. Features that are detected near the border between such an object and the background usually fail to be tracked and are therefore mistakenly included in the set $\mathcal{F}_{reflex}$ of reflection based features. Second, a feature that is detected near the border of the image I_f can possibly be invisible in the image I_{nf} as it moves out of the field of view of the camera. This effect can be minimized by limiting the feature detection process to the area of pixels that has at least a distance of b pixels to the image border. Finally, extreme camera movements that cause strong motion blur can result in a high number of detected features that cannot be successfully tracked in the image I_{nf} and that also undesirably end up in the set $\mathcal{F}_{reflex}$.

Figure 2.8 shows the result of the feature tracking process and illustrates that by constructing the feature set $\mathcal{F}_{reflex}$ using the procedure described above, the major part of raw features that do not correspond to reflective vest features are eliminated.

Figure 2.8: The figure illustrates the result of the feature tracking process. The image brightness has been adapted to increase readability. Locations where a feature has been detected are indicated by a cross in image I_f (above). The detection area in I_f is restricted to the white bounding box to assure that detected features are still in the camera's FOV when taking image I_{nf} (below), even under fast movement.
Features that have been successfully tracked (1–5, 7, 8 and 21) are represented in white color in image I_f and the tracked locations are indicated by a corresponding white cross in image I_{nf}. All tracked features in the above example show very low intensity difference and are therefore not considered as reflection based features.
Features that failed to be tracked (6 and 9–20) are marked as black crosses in image I_f. Features 6, 11–15 and 20 failed to be tracked due to violation of the brightness constancy assumption and are correctly identified as reflection based features (11–15 represent a reflective vest and 6 and 20 a reflective metallic surface on a car). Features 9, 10 and 16–19 could not be tracked due to the violation of the spatial coherence assumption and will mistakenly be included in the set of reflection based features.

2.7 Feature Description

As presented in the last section, the feature set $\mathcal{F}_{reflex}$ primarily contains features that originate from the reflection of the IR light on a reflective material. However, some cases have been discussed and illustrated in Figure 2.8 in which non-reflective features were mistakenly included. Furthermore, among the reflective materials that can appear in the scene it will be important to distinct between the reflective vest markers and other reflective objects such as metallic surfaces, windows, mirrors or different types of reflective markers typically present in an industrial environment. Therefore, a classifier will be introduced in the next section, able to compute the probability that a feature f in $\mathcal{F}_{reflex}$ belongs to a reflective vest.

The classifier will not directly evaluate the raw intensity values of the image. Instead, a local image descriptor is computed for every feature in $\mathcal{F}_{reflex}$. The image descriptor is a vector of N_r descriptor variables and is extracted from a square image patch P of size $\omega_P(s)$ centered around the location $\boldsymbol{u}_f$ where a feature was detected in image I_f. The patch size $\omega_P(s)$ of the image patch is a function of the feature scale s and is chosen to be:

$$\omega_P(s) = s + \omega_{P_0} \tag{2.18}$$

This ensures that the size of the patch from which the descriptor is extracted linearly scales with the effective feature scale s but a minimum patch size of ω_{P_0} is guaranteed even for very small features.

Requirements for an appropriate descriptor include robustness to illumination changes, motion blur, viewpoint changes and noise as well as computational efficiency of the extraction process. Popular feature descriptors are therefore often based on local intensity differences. State-of-the-art feature descriptors that were found appropriate include SURF [7], BRIEF [8] and BRISK [9].

2.7.1 SURF Descriptor

For the extraction of the SURF descriptor, the local image patch P is divided into 4×4 square subregions of size $\omega_P/4$. The responses d_u and d_v of a horizontal and a vertical Haar wavelet of size $\omega_P/10$ are computed at 5×5 regularly spaced locations inside every subregion (cf. Figure 2.9a). The wavelet responses are then summed up to obtain a vector of four descriptor variables per subregion,

$$\boldsymbol{r}_{sub} = \left[\sum d_u, \sum d_v, \sum |d_u|, \sum |d_v|\right] \tag{2.19}$$

and the complete descriptor is expressed as the concatenation of the vectors of all 4×4 subregions, resulting in the final SURF descriptor of $N_r = 64$ variables:

$$\boldsymbol{r}_{SURF} = [\boldsymbol{r}_{sub1}, \boldsymbol{r}_{sub2}, ..., \boldsymbol{r}_{sub16}] \tag{2.20}$$

In dividing the patch to be analyzed into subregions, the SURF descriptor focuses on the description of the spatial distribution of intensity gradients. The descriptor is invariant to an image intensity offset following higher illumination and invariance to changes in contrast can be achieved by turning the descriptor $\boldsymbol{r}_{SURF}$ into a unit vector.

In addition, the SURF descriptor is designed to be rotation and scale invariant. Yet, this property only holds true if the SURF descriptor is used either in combination with the corresponding SURF feature detector or with an alternative detector providing scale and orientation of detected features. As the STAR feature detector (cf. Section 2.5) employed in our application only provides a feature scale s but no orientation, our version of SURF lacks rotation invariance. In literature, this unoriented SURF version is referred to as upright SURF or U-SURF.

2.7.2 BRIEF Descriptor

The BRIEF descriptor was designed for very efficient computation and relies on simple, pairwise image intensity comparisons. BRIEF is non-rotation invariant and acts on a square patch of fixed size ω_F. To obtain a quasi scale invariant version of the descriptor, the patch P is scaled by the factor ω_F/ω_P to obtain the patch P' whose size is adapted for the extraction of the BRIEF descriptor. The method then defines a binary intensity test τ_F, acting on a smoothed version P'_σ of the scaled image patch P', obtained by applying a Gaussian kernel with standard deviation σ:

$$\tau_F(P'_\sigma, (\boldsymbol{u}_1, \boldsymbol{u}_2)) := \left\{ \begin{array}{ll} 1 & \text{if } P'_\sigma(\boldsymbol{u}_1) < P'_\sigma(\boldsymbol{u}_2) \\ 0 & \text{otherwise} \end{array} \right\} \tag{2.21}$$

with $(\boldsymbol{u}_1, \boldsymbol{u}_2)$ the pair of sampling locations of which the intensity values are compared. For a set of N_r different test location pairs $(\boldsymbol{u}_1, \boldsymbol{u}_2)_i$, the BRIEF descriptor is then defined as the string of binary variables according to:

$$\boldsymbol{r}_{BRIEF} = \sum_{i=1}^{N_r} 2^{i-1}\, \tau_F(P'_\sigma, \boldsymbol{u}_{1,i}, \boldsymbol{u}_{2,i}) \tag{2.22}$$

The precomputed test locations are sampled from an isotropic Gaussian distribution centered in the middle of the patch, according to Figure 2.9b. The number N_r of test locations is typically 128, 256 or 512. For the same reasons as described in the discussion of SURF, the BRIEF descriptor lacks rotation invariance.

2.7.3 BRISK Descriptor

Like BRIEF, the BRISK descriptor is based on pairwise image intensity comparisons. But in contrast to BRIEF, the sampling locations are not ran-

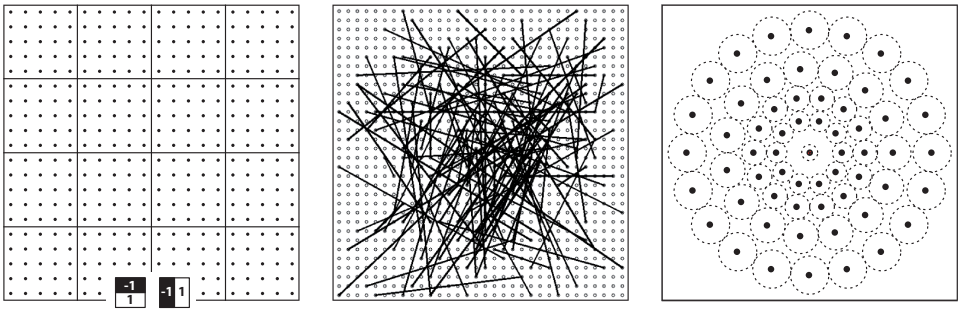


Figure 2.9: Sampling patterns of the different feature descriptors: **a)** The SURF pattern divides the image patch into 4×4 square subregions and the responses of horizontal and vertical Haar wavelets are computed at 5×5 equally spaced locations in every subregion. **b)** The BRIEF sampling pattern defines N_r precomputed random test location pairs, shown by a line connecting the two points, at which the intensity values are compared. Source: Calonder et al., 2010 **c)** The BRISK sampling pattern: black points indicate the sampling locations while the dashed circles indicate one standard deviation of the Gaussian kernel used to smoothen the intensity values at the sampling locations. Source: Leutenegger et al., 2011

domly distributed on the patch to be described. BRISK introduces a sampling pattern consisting of several concentric circles, centered in the middle of the patch P, as illustrated in Figure 2.9c. A total of N_{BK} sampling locations $\boldsymbol{u}_i$ are distributed and equally spaced on the circles. Gaussian smoothing is applied to the intensity values at all locations, with a standard deviation σ_i proportional to the distance between two points on the respective circle. The smoothed intensity value at point $\boldsymbol{u}_i$ is denoted $P(\boldsymbol{u}_i, \sigma_i)$. A set $\mathcal{A}$ of all $N_{BK}(N_{BK}-1)/2$ sampling point pairs is defined as:

$$\mathcal{A} = \{(\boldsymbol{u}_i, \boldsymbol{u}_j) \mid i < N_{BK} \wedge j < i\} \tag{2.23}$$

Based on $\mathcal{A}$, a subset $\mathcal{A}_S$ of short-distance pairings and a subset $\mathcal{A}_L$ of long-distance pairings are constructed, according to

$$\begin{aligned} \mathcal{A}_S &= \{(\boldsymbol{u}_i, \boldsymbol{u}_j) \in \mathcal{A} \mid \|\boldsymbol{u}_i - \boldsymbol{u}_j\| < \delta_{max}\} \\ \mathcal{A}_L &= \{(\boldsymbol{u}_i, \boldsymbol{u}_j) \in \mathcal{A} \mid \|\boldsymbol{u}_i - \boldsymbol{u}_j\| > \delta_{min}\} \end{aligned} \tag{2.24}$$

with fixed thresholds δ_{max} and δ_{min}. The algorithm first computes a characteristic direction $\boldsymbol{g}_{BK}$ for the image patch P to be described, based on the long-distance pairs:

$$\boldsymbol{g}_{BK} = [g_{BR,u}, g_{BR,v}]^T = \frac{1}{|\mathcal{A}_L|} \cdot \sum_{(\boldsymbol{u}_i, \boldsymbol{u}_j) \in \mathcal{A}_L} \boldsymbol{g}_{BK}(\boldsymbol{u}_i, \boldsymbol{u}_j) \tag{2.25}$$

with

$$\boldsymbol{g}_{BK}(\boldsymbol{u}_i, \boldsymbol{u}_j) = (\boldsymbol{u}_j - \boldsymbol{u}_i) \cdot \frac{P(\boldsymbol{u}_j, \sigma_j) - P(\boldsymbol{u}_i, \sigma_i)}{\|\boldsymbol{u}_j - \boldsymbol{u}_i\|^2} \tag{2.26}$$

The pattern is rotated by the angle $\gamma = \arctan2(g_{BR,u}, g_{BR,v})$ around the center of patch P, resulting in the rotated sampling point pairs $(\boldsymbol{u}_i^\gamma, \boldsymbol{u}_j^\gamma)$. An intensity comparison test τ_K is then defined by

$$\tau_K(P, \boldsymbol{u}_i^\gamma, \boldsymbol{u}_j^\gamma) := \left\{ \begin{array}{ll} 1 & \text{if } P(\boldsymbol{u}_j^\gamma, \sigma_j) > P(\boldsymbol{u}_i^\gamma, \sigma_i) \\ 0 & \text{otherwise} \end{array} \right\} \tag{2.27}$$

and the final BRISK descriptor defined as the bit string resulting from the concatenation of all binary short-distance test responses:

$$\boldsymbol{r}_{BRISK} = \sum_{(\boldsymbol{u}_i^\gamma, \boldsymbol{u}_j^\gamma) \in \mathcal{A}_S} 2^{i-1} \, \tau_K(P, \boldsymbol{u}_i^\gamma, \boldsymbol{u}_j^\gamma) \tag{2.28}$$

In contrast to BRIEF and SURF, the BRISK descriptor is designed to be rotation invariant even if the feature detector does not provide any feature orientation, which is the case for the STAR algorithm.

2.8 Feature Classification

Based on the feature descriptors extracted according to Section 2.7, the subsequent processing step aims at classifying the features $f \in \mathcal{F}_{reflex}$ into a set of vest features and a set of non-vest features. More precisely, we wish to predict a probability $\hat{p}_{vest}$ that a given feature $f \in \mathcal{F}_{reflex}$ originates from a reflective vest and classify it in either as vest or non-vest feature according to $\hat{p}_{vest}$ and a given threshold λ_{vest}. To do so, a binary classifier is trained by a supervised learning approach. Supervised learning is a machine learning technique in which a set of training samples are labeled with the desired output value and fed to the learning algorithm during the training session. In our case, the training samples are feature descriptors and the desired output values are known class indexes that indicate whether the descriptor corresponds to a reflective vest or not. Once trained, the classifier is then expected to *generalize*, that is, to accurately predict the class output for an unseen feature descriptor where no label is available.

We choose to employ a Random Forest [10] classifier, motivated by several of its advantages compared to other classification techniques. First of all, Random Forests can not only deal with classification but also regression problems, a property that we will exploit in Section 2.9 where we aim at estimating the distance between the camera and a given feature f based on its feature descriptor $\boldsymbol{r}$. Their application is also motivated by the computational efficiency in predicting an output value once supervised learning is completed. Finally, Random Forests have shown high performance in image classification [11, 12] and they have the potential for parallel implementation. The latter can become very important when it comes to accelerate the supervised learning process, especially if the training material contains a large number of samples.

A Random Forest is an ensemble of decision trees [13]. It obtains a prediction of the output variable by averaging together the predictions of the individual trees in the forest. All the decision trees are different from each other and every tree is only a weak classifier that tends to overfit the training data used during supervised learning. But by aggregating the predictions of all the individual trees, the forest achieves much higher accuracy in predicting the output variable compared to single decision trees. In machine learning terms, *variance* is reduced while *bias* is kept low when comparing the Random Forest to the individual decision trees.

2.8.1 Training the Random Forest

Let $\mathcal{R}$ be a data set of $N_{\mathcal{R}}$ local image feature descriptors $\boldsymbol{r}^{[i]}$, $i = 1, ..., N_{\mathcal{R}}$ with corresponding binary class labels $\tilde{c}^{[i]}$ that take the value 1 if the respective descriptor corresponds to a reflective vest feature and 0 otherwise. Every feature descriptor $\boldsymbol{r}^{[i]}$ is a vector of N_r descriptor variables $r_j^{[i]}$, with $j = 1, ..., N_r$. A descriptor variable can take a numerical value, as in the case of the SURF descriptor, or be categorical as in the case of the binary BRIEF and BRISK descriptors.

Supervised learning of a Random Forest is performed by recursively growing N_{tr} individual decision trees based on the training data. Randomization of the trees is accomplished by two means, first by a random excerpt of the training data used to train an individual tree and by the random selection of a subset of all descriptor variables that a tree might split the data on. A different training set $\mathcal{R}_m$ of feature descriptors is created for every tree m, by randomly sample $N_{\mathcal{R}}$ elements from the dataset $\mathcal{R}$ with replacement. Using this technique, referred to as *bootstrapping*, only about two-third of the elements in $\mathcal{R}$ are included in the training set $\mathcal{R}_m$ of the m-th tree, some of them with multiple copies. The remaining one-third of labeled feature descriptors is used as a test set and serves to estimate the classification error during supervised learning.

A training algorithm is then used to grow the individual trees of the forest. At each node k in tree m, the corresponding set of image descriptors $\mathcal{R}_{m,k}$ is split into two subsets $\mathcal{R}'_{m,k}$ and $\mathcal{R}''_{m,k}$, corresponding respectively to the left and right child node. Whether during the split a descriptor $\boldsymbol{r} \in \mathcal{R}_{m,k}$ is placed in $\mathcal{R}'_{m,k}$ or $\mathcal{R}''_{m,k}$ depends on the value that one of its N_r descriptor variables take. The choice of the variable index j to base the split on is limited to a randomly chosen subset of candidates from all N_r descriptor variables, usually of size $\sqrt{N_r}$, whereby the subset is different for every tree in the forest. The best choice among the candidates is then chosen by evaluating the variable leading to the highest *information gain*

$$\Delta E = -\frac{|\mathcal{R}'_{m,k}|}{|\mathcal{R}_{m,k}|}\,\xi(\mathcal{R}'_{m,k}) - \frac{|\mathcal{R}''_{m,k}|}{|\mathcal{R}_{m,k}|}\,\xi(\mathcal{R}''_{m,k}) \tag{2.29}$$

where $|\cdot|$ denotes the number of elements in a set and $\xi(\mathcal{R}'_{m,k})$ and $\xi(\mathcal{R}''_{m,k})$ impurity measures for the sets $\mathcal{R}'_{m,k}$ and $\mathcal{R}''_{m,k}$. Three different measures are commonly adopted in binary classification problems to measure the impurity of a set $\mathcal{R}_{m,k}$ at a given node k, namely the *entropy*,

$$\xi_E(\mathcal{R}_{m,k}) = -\sum_{n=0}^{1} q_n \cdot log_2(q_n) \ , \tag{2.30}$$

the *gini index*,

$$\xi_G(\mathcal{R}_{m,k}) = 1 - \sum_{n=0}^{1} q_n^2 \ , \tag{2.31}$$

and the *misclassification error*,

$$\xi_M(\mathcal{R}_{m,k}) = 1 - \max\{q_0, q_1\} \tag{2.32}$$

where q_n denotes the fraction of descriptors $\boldsymbol{r}$ in set $\mathcal{R}_{m,k}$ whose class label $\tilde{c}$ is n. An impurity measure of $\xi(\mathcal{R}_{m,k}) = 0$ implies that the set of descriptors $\mathcal{R}_{m,k}$ contains only elements with the same class label. Figure 2.10 depicts the characteristics of the three different impurity measures. In our work, we employ the gini index to measure the impurity of a feature set.

The way a set $\mathcal{R}_{m,k}$ is split into two subsets depends on the type of the descriptor variable the split is based on. If the type is numerical (e.g. SURF descriptor), a random threshold λ_k is chosen at node k and a split on the j-th descriptor variable is performed by:

$$\begin{aligned} \mathcal{R}'_{m,k} &= \{\boldsymbol{r} \in \mathcal{R}_{m,k} \mid r_j > \lambda_k\} \\ \mathcal{R}''_{m,k} &= \{\boldsymbol{r} \in \mathcal{R}_{m,k} \mid \boldsymbol{r} \notin \mathcal{R}'_{m,k} \ \} \end{aligned} \tag{2.33}$$

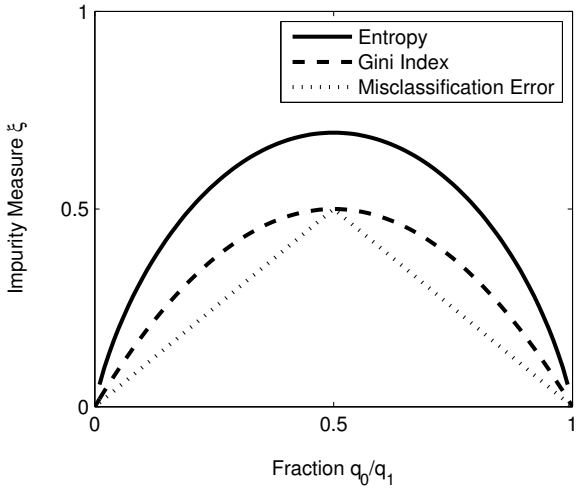


Figure 2.10: Impurity measures used in the Random Forest classifier

If instead the descriptor variable is categorical, a random subset $\mathcal{Q}_k$ of all values that r_j can take is chosen and the split on the j-th variable is accomplished by:

$$\begin{aligned} \mathcal{R}_{m,k}^{'} &= \{\boldsymbol{r} \in \mathcal{R}_{m,k} \mid r_j \in \mathcal{Q}_k\} \\ \mathcal{R}_{m,k}^{''} &= \{\boldsymbol{r} \in \mathcal{R}_{m,k} \mid \boldsymbol{r} \notin \mathcal{R}_{m,k}^{'}\} \end{aligned} \tag{2.34}$$

Using this approach, the tree is grown by iteratively splitting the dataset until either a specified depth is reached or one of the created subsets $\mathcal{R}'_{m,k}$ and $\mathcal{R}''_{m,k}$ is empty.

2.8.2 Predicting with the Random Forest

Once the model of the classifier is established through supervised learning, the classification of a feature $f \in \mathcal{F}_{reflex}$ is performed by propagating its descriptor $\boldsymbol{r}$ down every tree of the forest until it is placed in a leaf node. The propagation path is given by the learned model of the tree. For each node k, the model specifies the index j of the descriptor variable and the threshold λ_k (numerical variables) or class subset $\mathcal{Q}_k$ (categorical variables) on which the decision to propagate the sample to the left or right branch is based. After the sample reached a leaf node, the class prediction $\hat{c}_m$ of the m-th tree is given by the majority of class labels of the training samples that were placed in the same leaf node during the learning phase.

The classification of a feature with the Random Forest classifier provides N_{tr} individual class votes $\hat{c}_m$, one per each tree in the forest. A vote $c_m = 1$ indicates that the m-th tree votes for a reflective vest feature while $\hat{c}_m = 0$ means that the tree votes against a vest. A probability that a descriptor $\boldsymbol{r}$ represents a reflective vest can then be inferred from the N_{tr} individual class votes by dividing the number of trees voting for a reflective vest by the total number of trees N_{tr} in the forest:

$$\hat{p}_{vest} = \frac{1}{N_{tr}} \sum_{m=1}^{N_{tr}} \hat{c}_m \tag{2.35}$$

Finally, we classify the features $f \in \mathcal{F}_{reflex}$ with a high probability $\hat{p}_{vest}$ in a set $\mathcal{F}_{vest}$, according to:

$$\mathcal{F}_{vest} = \{f \in \mathcal{F}_{reflex} \mid \hat{p}_{vest} > \lambda_{vest}\} \tag{2.36}$$

Simultaneously, we collect all other features, the ones with low probability $\hat{p}_{vest}$, together with the set $\mathcal{F}_{LID}$ of previously rejected features (see Figure 2.1 and Eq. 2.15) in a set of non-vest features $\mathcal{F}_{non-vest}$ that will not be further processed:

$$\mathcal{F}_{non-vest} = (\mathcal{F}_{reflex} \setminus \mathcal{F}_{vest}) \cup \mathcal{F}_{LID} \tag{2.37}$$

2.9 Distance Estimation

The same local feature descriptors $\boldsymbol{r}$ used for feature classification described in the last section are employed to estimate the distance between a feature $f \in \mathcal{F}_{vest}$ and the camera. Again, supervised learning is performed, this time to train a Random Forest regressor on a set of descriptors that are labeled with the ground-truth distance between the camera and the reflective vest that caused the appearance of a given vest feature. The trained regressor model is then applied to obtain a distance estimate $\hat{d}$ for descriptors of unseen features.

Let us again call $\mathcal{R}$ a training data set consisting of $N_{\mathcal{R}}$ feature descriptors $\boldsymbol{r}^{[i]}$. The training set only contains feature descriptors that actually correspond to reflective vest features. Every descriptor $\boldsymbol{r}^{[i]}$ is assigned a ground-truth distance label $\tilde{d}^{[i]}$ indicating the distance between the camera and the reflective vest that caused the appearance of feature $f^{[i]}$ in image I_f.

The supervised learning algorithm for the Random Forest regressor is similar to the one applied to train the classifier described in Section 2.8. Yet, the impurity measure ξ of a data set $\mathcal{R}_k$ at node k has to be adapted to the case of regression, where the variance of the distance labels $\tilde{d}^{[i]}$ of all descriptors in $\mathcal{R}$ is used, according to:

$$\xi(\mathcal{R}_k) = \frac{1}{|\mathcal{R}_k|} \sum_{\boldsymbol{r}^{[i]} \in \mathcal{R}_k} (\tilde{d}^{[i]} - \bar{d})^2 \quad \text{with} \quad \bar{d} = \frac{1}{|\mathcal{R}_k|} \sum_{\boldsymbol{r}^{[i]} \in \mathcal{R}_k} \tilde{d}^{[i]} \tag{2.38}$$

With the regressor successfully trained, the distance estimation for an unseen feature f is performed by propagating its descriptor $\boldsymbol{r}$ down every tree of the forest until it is placed in a leaf node. The distance estimate $\hat{d}_m$ of the m-th tree is given by the average value computed from the labels of all feature descriptors that were placed in the same leaf node during the learning phase. The final distance estimate $\hat{d}$ of the forest is the average value of all individual tree estimates, $\hat{d}_m$:

$$\hat{d} = \frac{1}{N_{tr}} \sum_{m=1}^{N_{tr}} \hat{d}_m \tag{2.39}$$

2.10 3D Position Estimation

In Section 2.4, a projective function $\boldsymbol{h}(\boldsymbol{u})$ was introduced (see Eq. 2.6) that maps a pair of image coordinates $\boldsymbol{u}$ of the unwrapped image I_f to a unit vector in the camera reference frame. This unit vector points into the direction of the object that caused the intensity value $I_f(\boldsymbol{u})$. Thus, for every detected vest feature we can obtain an estimate of the 3D position in the camera reference frame by projecting the location $\boldsymbol{u}_f$ where the feature f was detected in I_f to a unit vector in 3D space and by multiplying the length of the vector by the corresponding distance estimate $\hat{d}$.

$$\hat{\boldsymbol{p}} = \hat{d} \cdot \boldsymbol{h}(\boldsymbol{u}_f) \tag{2.40}$$

The position estimation is carried out for all features that were classified as vest features and collected in the set $\mathcal{F}_{vest}$.

2.11 Vest Tracking

Until now, the detection of a reflective vest focused on the processing of a single image pair $\boldsymbol{I} = (I_f, I_{nf})$, consisting of an image taken with IR flash and an image taken without flash. The result of the detection process is a feature set $\mathcal{F}_{vest}$ in which for every feature f a 3D position estimate $\hat{\boldsymbol{p}}$ was estimated.

As stated in the introduction, the ultimate goal of our application is to keep track of the position of a person wearing a reflective vest, relative to the camera. Yet, this quantity generally evolves over time and cannot be measured directly with a single camera setup. Rather, the algorithm has to rely on the position estimates $\hat{\boldsymbol{p}}$ that were obtained by regression with a Random Forest. The regressor tries to model the process by which certain image patterns are generated in the acquired images when the camera system observes a reflective vest. This process of image acquisition is corrupted with noise, resulting in random variation of brightness information in the acquired image material that the regressor is unable to learn. Furthermore, a finite number of observations, namely the training material, is given to learn the image formation process. Both circumstances lead to the fact that the position estimates $\hat{\boldsymbol{p}}$ obtained by regression are subject to uncertainty.

For the named reasons, it is important to represent uncertainty when it comes to incorporating the available information into a tracking algorithm. Here, we will adopt the statistical approach provided by *Bayesian* filtering and introduce the recursive *Bayesian* filter in Section 2.11.1. The recursive Bayesian filter builds the basic framework of the *particle filter* that is employed in our application to perform tracking of reflective vests and we will discuss its application in Section 2.11.2.

We will now adapt the notation to the scenario where image pairs $\boldsymbol{I}$ are repeatedly acquired and we denote $\boldsymbol{I}_t$ the image pair acquired at time

step $t \in \mathbb{Z}$. Please note that despite being a time index, we will refer to t as the *time*. We further denote $\hat{\boldsymbol{p}}_t^{[i]}$ the estimated position corresponding to feature $f^{[i]} \in \mathcal{F}_{vest}$ at time t and introduce the set $\mathcal{P}_t$ of all position estimates obtained at the same time t, according to

$$\mathcal{P}_t = \left\{ \hat{\boldsymbol{p}}_t^{[i]} \mid i = 1, ..., N_{\mathcal{P}_t} \right\} \tag{2.41}$$

where $N_{\mathcal{P}_t}$ is the number of position estimates obtained at time t. $N_{\mathcal{P}_t}$ simply equals the size of the set $\mathcal{F}_{vest}$ at time t. In the remainder of this chapter, we will refer to $\hat{\boldsymbol{p}}_t$ as a single *observation* and to $\mathcal{P}_t$ as the *set of observations* at time t.

We further introduce the state vector $\boldsymbol{s}_t$, which represents the set of quantities that will be recursively estimated by the vest tracking algorithm. In addition to the position $\boldsymbol{p}_t = [x_t, y_t, z_t]^T$ of a reflective vest at time t, the state also includes its velocity in the camera reference frame, denoted by the ensemble $\dot{\boldsymbol{p}}_t = [\dot{x}_t, \dot{y}_t, \dot{z}_t]^T$:

$$\boldsymbol{s}_t = \begin{bmatrix} \boldsymbol{p}_t \\ \dot{\boldsymbol{p}}_t \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \\ z_t \\ \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \end{bmatrix} \tag{2.42}$$

The estimation of the velocity of an observed reflective vest will allow to make a better prediction of the state transition from $\boldsymbol{s}_t$ to $\boldsymbol{s}_{t+1}$, as it is done in the motion model described in Section 2.11.2.

2.11.1 Recursive Bayesian Filter

A recursive Bayesian filter tries to estimate the state $\boldsymbol{s}_t$ of a system by exploiting all the available observations. In our case, the state $\boldsymbol{s}_t$ contains the position and velocity of a reflective vest in the camera reference frame and the observations are given by the set $\mathcal{P}_t$. The uncertainty over the exact state at time t is modeled by a probability distribution over $\boldsymbol{s}_t$ that we will refer to as the belief $Bel(\boldsymbol{s}_t)$. The belief represents the probabilistic density function (PDF) over the state variable $\boldsymbol{s}_t$, conditioned on all observations that were made until time t, which is in our case:

$$Bel(\boldsymbol{s}_t) = p(\boldsymbol{s}_t | \mathcal{P}_1, \mathcal{P}_2, ..., \mathcal{P}_t) \tag{2.43}$$

Under the *Markov assumption*, the recursive Bayesian filter provides a mechanism to recursively update the belief every time a new set of observations $\mathcal{P}_t$ is available. The *Markov assumption*, also referred to as the *complete state assumption*, postulates that if the state $\boldsymbol{s}_{t-1}$ is known, the observation

$\mathcal{P}_t$ is conditionally independent from all observations obtained until time t. Given the belief $Bel(\boldsymbol{s}_{t-1})$ and a new set of observations $\mathcal{P}_t$, the filter first achieves a predictive belief of the state $\boldsymbol{s}_t$ at time t, named $\overline{Bel}(\boldsymbol{s}_t)$. This step is referred to as the *prediction*:

$$\overline{Bel}(\boldsymbol{s}_t) = \int p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})\ Bel(\boldsymbol{s}_{t-1})\ d\boldsymbol{s}_{t-1} \tag{2.44}$$

The term $p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1})$ is referred to as the *state transition probability* and represents the probabilistic description of the system's *motion model*, $\boldsymbol{s}_t = \boldsymbol{\psi}_{Motion}(\boldsymbol{s}_{t-1})$. The motion model describes how the state $\boldsymbol{s}_t$ of the system evolves over time due to the system's dynamics. The predictive belief $\overline{Bel}(\boldsymbol{s}_t)$ is then corrected by incorporating the set of observations $\mathcal{P}_t$ to obtain the belief $Bel(\boldsymbol{s}_t)$. This step is referred to as either *correction* or *update*.

$$Bel(\boldsymbol{s}_t) = \alpha_t\ p(\mathcal{P}_t|\boldsymbol{s}_t)\ \overline{Bel}(\boldsymbol{s}_t) \tag{2.45}$$

Here, α_t is a normalization factor. The term $p(\mathcal{P}_t|\boldsymbol{s}_t)$ is the *measurement probability* and represents the probabilistic description of a *measurement model* which is of the form $\hat{\boldsymbol{p}}_t = \boldsymbol{\psi}_{Measurement}(\boldsymbol{s}_t)$. The measurement model describes the formation process of a position estimate $\hat{\boldsymbol{p}}_t$ for a given state $\boldsymbol{s}_t$. In contrast, the measurement probability $p(\mathcal{P}_t|\boldsymbol{s}_t)$ represents the likelihood of making a set of observations $\mathcal{P}_t$ under the assumption that the state of the system is $\boldsymbol{s}_t$.

2.11.2 Particle Filter

In this application we employ a *particle filter* which is a non-parametric implementation of the recursive Bayesian filter. In a particle filter, the belief distribution $Bel(\boldsymbol{s}_t)$ is approximated by a set of N_p samples, called particles, according to

$$Bel(\boldsymbol{s}_t) \approx \mathcal{S}_t = \left\{ \left\langle \boldsymbol{s}_t^{[k]}, w_t^{[k]} \right\rangle \mid k = 1, ..., N_p \right\} \tag{2.46}$$

where $\boldsymbol{s}_t^{[k]}$ denotes a state hypothesis and $w_t^{[k]}$ a weight, called *importance factor*. The implementation of the Bayes filter is accomplished using a procedure called *sequential importance resampling (SIR)* [14]. Let us consider the particles $\langle \boldsymbol{s}_{t-1}^{[k]}, w_{t-1}^{[k]} \rangle \in \mathcal{S}_{t-1}$, representing the belief $Bel(\boldsymbol{s}_{t-1})$, and a set of observations made at time t, named $\mathcal{P}_t$. A predictive particle set $\overline{\mathcal{S}}_t$, representing the predicted belief $\overline{Bel}(\boldsymbol{s}_t)$ according to Eq. 2.44, is obtained by applying the motion model to all the state hypotheses $\boldsymbol{s}_{t-1}^{[k]}$ individually:

$$\overline{\boldsymbol{s}}_t^{[k]} = \boldsymbol{\psi}_{Motion}(\boldsymbol{s}_{t-1}^{[k]}) \tag{2.47}$$

The update step, according to Eq. 2.45, is then accomplished in two steps. First, an importance factor $\tilde{w}_t^{[k]}$ is computed for every predicted state $\overline{\boldsymbol{s}}_t^{[k]}$.

The weight $\tilde{w}_t^{[k]}$ represents the likelihood of making the set of observations $\mathcal{P}_t$ given the state $\overline{\boldsymbol{s}}_t^{[k]}$, according to the measurement probability:

$$\tilde{w}_t^{[k]} = p(\mathcal{P}_t|\overline{\boldsymbol{s}}_t^{[k]}) \tag{2.48}$$

The weights $\tilde{w}_t^{[k]}$ are then normalized in order to sum to unity:

$$w_t^{[k]} = \frac{\tilde{w}_t^{[k]}}{\sum_{j=1}^{N_p} \tilde{w}_t^{[j]}} \tag{2.49}$$

Finally, the set of particles $\mathcal{S}_t$ is obtained by resampling with replacement N_p particles from the predicted set $\overline{\mathcal{S}}_t$ according to the importance factors $w_t^{[k]}$.

$$\mathcal{S}_t = \left\{ \left\langle \boldsymbol{s}_t^{[k]}, w_t^{[k]} \right\rangle \mid i = 1, ..., N_p \right\} \quad \text{with} \quad p(\boldsymbol{s}_t^{[k]} = \overline{\boldsymbol{s}}_t^{[k]}) = w_t^{[k]} \tag{2.50}$$

While different techniques exist to perform the resampling procedure, we employ the approach named *low variance resampling* as proposed in [15].

At time $t = 0$, an initial particle set $\mathcal{S}_0$ is generated by uniform distribution of the particles in the state space which has lower and upper limits specified by two vectors $\boldsymbol{s}_{min}$ and $\boldsymbol{s}_{max}$. If during a state transition from $t-1$ to t a particle's state $\boldsymbol{s}_t$ falls out of the bounds, it is re-initialized by sampling again from the same uniform distribution.

Finally, given the particle set $\mathcal{S}_t$ at time t, an estimate of the position of an observed person can be obtained using the weighted mean of the particle states:

$$\hat{\boldsymbol{s}}_t = \sum_{k=1}^{N_p} w_t^{[k]} \boldsymbol{s}_t^{[k]} \tag{2.51}$$

Particles filters show several important advantages over other techniques aiming at representing the belief distribution in recursive state estimation. First of all, particle filters can represent arbitrary belief distributions, due to the fact that the belief is not described by a parametric model but approximated by the density of a set of samples, the particles. In addition, the performance and computational complexity of the algorithm is adjustable by the choice of the number of particles N_p. Particle filters also focus their computational resources on the regions in the state space where states have high probability, which is very beneficial for resource-constrained real-time applications.

Motion Model

As stated above, the motion model is a function $\boldsymbol{s}_t = \boldsymbol{\psi}_{Motion}(\boldsymbol{s}_{t-1})$ that predicts the state at time t based on the known state at time $t-1$ using a description of the system dynamics. In our case, a possible change of the state $\boldsymbol{s}_t$ can be caused by two sources, namely movement of the camera and movement of the observed person wearing the reflective vest. We therefore split the motion model into two parts, representing the motion of the camera and the observed vest respectively.

$$\boldsymbol{\psi}_{Motion}(\boldsymbol{s}_t) = \boldsymbol{\psi}_{Motion,Cam}(\boldsymbol{s}_t) + \boldsymbol{\psi}_{Motion,Vest}(\boldsymbol{s}_t) \tag{2.52}$$

The possible movements of the observed person in an industrial environment are vast and include walking at constant speed, accelerating in any direction or performing abrupt turns or twists. Furthermore, a person can move by means of a vehicle. To successfully keep track of an observed person, the motion model applied in the particle filter needs to be able to represent all these different motion types in a probabilistic way. The model also has to cope with the fact that no sensory input at all is provided that could be used to predict the change in position between two time steps.

In our approach, we approximate the motion of a person that is observed by the system. We assume that between two times steps, a person performs a straight movement at constant speed. To take into account that a person might change its speed as well as the direction of movement, we allow abrupt changes of the velocity vector $\dot{\boldsymbol{p}}_t = [\dot{x}_t, \dot{y}_t, \dot{z}_t]^T$ at the time steps t. Thus, we obtain the following linear motion model:

$$\boldsymbol{\psi}_{Motion,Vest}(\boldsymbol{s}_t) = D\boldsymbol{s}_t + \boldsymbol{\nu}_{system} \tag{2.53}$$

with

$$D = \begin{bmatrix} \boldsymbol{I_{3\times 3}} & f_a^{-1} \cdot \boldsymbol{I_{3\times 3}} \\ \boldsymbol{0_{3\times 3}} & \boldsymbol{I_{3\times 3}} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\nu}_{system} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mathcal{N}(0, \sigma_{\dot{x}}) \\ \mathcal{N}(0, \sigma_{\dot{y}}) \\ \mathcal{N}(0, \sigma_{\dot{z}}) \end{bmatrix} \tag{2.54}$$

where D denotes the system's dynamics matrix, f_a the image pair acquisition rate, $\boldsymbol{\nu}_{system}$ a white noise vector, referred to as the *system noise*, and $\mathcal{N}(0, \sigma)$ a random number drawn from a zero-mean normal distribution with standard deviation σ.

In our case, the system noise vector $\boldsymbol{\nu}_{system}$ models the uncertainty about the change in speed and change in direction of movement that the observed person may accomplish. It strongly influences the filter's ability to cope with abrupt movements and accelerations of the observed person. The choice of

$\sigma_{\dot{x}}$, $\sigma_{\dot{y}}$ and $\sigma_{\dot{z}}$ is a trade-off, as too low values lead to a high inertia of the particles while too high values result in a constant defocusing from the tracked object.

The second source of relative motion between the camera and an observed person is the movement of the camera itself. At the moment, no sensory input is provided to the system concerning the motion of the camera. However, information about its translational and rotational movement would be highly beneficial in order to predict the change of an observed person's position in the reference frame attached to the camera. Several hardware extensions that will be included in future versions of the camera system and that will provide the algorithm with motion-related information are discussed in Chap. 6. In the current implementation, we model the additional uncertainty arising from camera motion with increased values for the system noise included in Eq. 2.53.

Measurement Model

The measurement model relates the set of observations $\mathcal{P}_t$ to the state vector $\boldsymbol{s}_t$ by a function $\hat{\boldsymbol{p}}_t = \boldsymbol{\psi}_{Measurement}(\boldsymbol{s}_t)$. An equivalent probabilistic representation is given by the *measurement probability*, denoted $p(\hat{\boldsymbol{p}}_t|\boldsymbol{s}_t)$, which describes the likelihood to make a single observation $\hat{\boldsymbol{p}}_t$ assuming that the state of the system is $\boldsymbol{s}_t$. The measurement probability has to incorporate all the sources of uncertainty that exist in the formation process of a measurement $\hat{\boldsymbol{p}}_t$. Sources of errors include measurement noise due to noisy image material as well as erroneous distance estimation by the regressor. Here, we assume that the different errors are Gaussian distributed.

Figure 2.11 depicts the characteristic shape of the measurement probability $p(\hat{\boldsymbol{p}}_t|\boldsymbol{s}_t)$ in the x/z-plane for three different states $\boldsymbol{s}_t^{[0]}$, $\boldsymbol{s}_t^{[1]}$ and $\boldsymbol{s}_t^{[2]}$. The measurement probability of each state is represented with iso-lines at one and two standard deviations of a multivariate normal distribution. Due to the processing scheme employed to obtain a position estimate $\hat{\boldsymbol{p}}_t$, the measurement uncertainty is different in radial and tangential direction and represented respectively by the standard deviations σ_{rad} and σ_{tg}. Uncertainty in radial direction mainly originates from the estimation error committed by the distance regressor. In contrast, the variance in the detection of the tangential position arises from the fact that a reflective vest feature detected in the input images is not necessarily situated in the center of the reflective vest. Finally, measurement noise in the image material causes uncertainty in both directions as it influences the complete processing chain. Experimental results show that the values of σ_{rad} and σ_{tg} are relatively constant over the whole sensor range.

As it will be shown in Chap. 3, the distance predictions $\hat{d}$, estimated by the Random Forest regressor, are further prone to a systematic error, called *bias*, which is characterized by a constant overestimation of the distance at

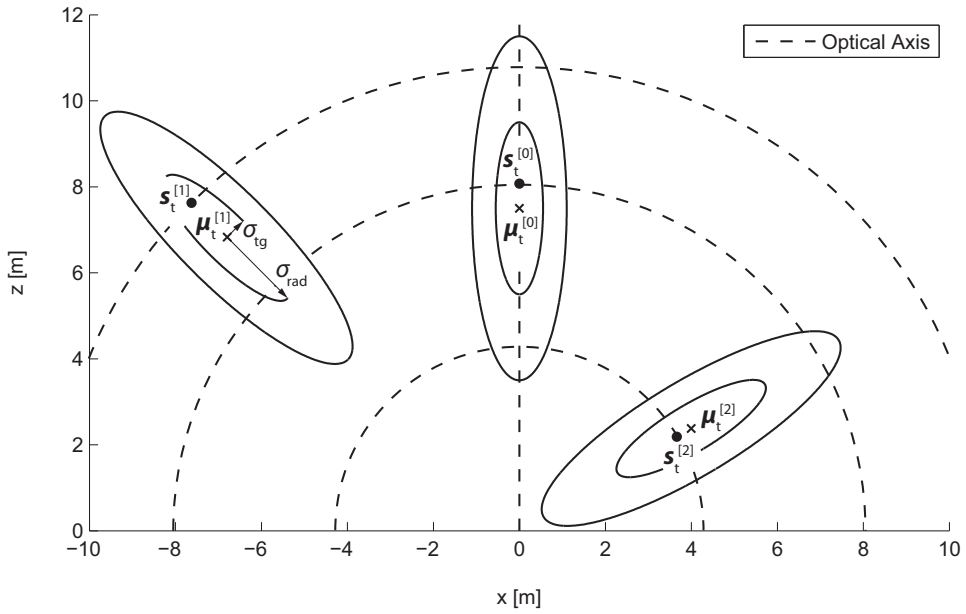


Figure 2.11: 2D representation of the characteristic measurement probability $p(\hat{\boldsymbol{p}}_t|\boldsymbol{s}_t)$ for three different example states $\boldsymbol{s}_t^{[0]}$, $\boldsymbol{s}_t^{[1]}$ and $\boldsymbol{s}_t^{[2]}$. The camera system is located at the origin. The measurement probability is modeled by a multivariate normal distribution with specific standard deviations σ_{rad} and σ_{tg} for the radial and tangential direction and with a mean value $\boldsymbol{\mu}_t$ that takes the bias of the distance estimator into account. The iso-lines show one and two standard-deviations.

short ranges and underestimation at higher ranges. In between, the bias evolves approximately linearly with the distance and thus we model it by a linear error function $\epsilon(d) = A_{bias} \cdot d + B_{bias}$.

To establish the measurement model, the parameters A_{bias}, B_{bias}, σ_{rad} and σ_{tg} are experimentally determined. A covariance matrix Σ_0 is defined that corresponds to the uncertainty of observations for states $\boldsymbol{s}_t$ situated on the camera's optical axis (cf. state $\boldsymbol{s}_t^{[0]}$ in Figure 2.11):

$$\Sigma_0 = \begin{bmatrix} \sigma_{tg}^2 & 0 & 0 \\ 0 & \sigma_{tg}^2 & 0 \\ 0 & 0 & \sigma_{rad}^2 \end{bmatrix} \tag{2.55}$$

The likelihood to make a single observation $\hat{\boldsymbol{p}}_t$, under the assumption of state $\boldsymbol{s}_t$, is then given by the multivariate Gaussian function

$$p(\hat{\boldsymbol{p}}_t|\boldsymbol{s}_t) = \frac{1}{(2\pi)^{3/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\hat{\boldsymbol{p}}_t - \boldsymbol{\mu}_t)^T \Sigma^{-1} (\hat{\boldsymbol{p}}_t - \boldsymbol{\mu}_t)\right) \tag{2.56}$$

where the covariance matrix Σ is obtained by rotation of Σ_0 so that the axis of Σ corresponding to σ_{rad} points in the radial direction. The center $\boldsymbol{\mu}_t$

of the Gaussian function is determined by making use of the distance error function $\epsilon(d)$, according to

$$\boldsymbol{\mu}_t = \left(1 + \frac{\epsilon(||\boldsymbol{p}_t||)}{||\boldsymbol{p}_t||}\right) \boldsymbol{p}_t \tag{2.57}$$

where $\boldsymbol{p}_t$ denotes the vector containing the first three elements of the state vector $\boldsymbol{s}_t$, according to Eq. 2.42.

Finally, the complete measurement model defines the likelihood to make the full set of observations $\mathcal{P}_t$, given the state $\boldsymbol{s}_t$. Under the assumption that the noise in the individual measurements $\hat{\boldsymbol{p}}_t^{[i]}$ is independent, it is obtained by the product of the individual measurement likelihoods $p(\hat{\boldsymbol{p}}_t|\boldsymbol{s}_t)$:

$$p(\mathcal{P}_t|\boldsymbol{s}_t) = \prod_{i=1}^{N_{\mathcal{P}_t}} p(\hat{\boldsymbol{p}}_t^{[i]}|\boldsymbol{s}_t) \tag{2.58}$$

Extensions to the basic Particle Filter

Two major extensions to the basic particle filter algorithm have proven to be effective for good tracking performance. They address two issues encountered during the evaluation of the system, namely the decrease in particle diversity and the data association problem.

As described in Sec. 2.11.2, particle resampling is performed according to the importance factors assigned to individual particles. Particles with high weights are probable to be resampled several times while particles with low weights might not appear at all in the generated particle set. After performing several resampling steps, this can lead to an effect referred to as *sample impoverishment* where diversity in the population of the state space is drastically reduced. The problem does usually not occur if the system noise included in the motion model (see Eq. 2.54) is large enough. If this is not the case, a certain degree of diversity can be introduced by artificial means. A simple method to do so is called *roughening* [14]. In roughening, the particles are perturbed after each resampling step by adding a random jitter drawn from a zero-mean normal distribution. The standard deviation σ_{jitter} of the m-th state component is proposed to be:

$$\sigma_{jitter,m} = (K_{jitter} \cdot (\boldsymbol{s}_{max,m} - \boldsymbol{s}_{min,m}) \cdot N_p)^{-\frac{1}{D}} \tag{2.59}$$

where K_j is a constant tuning parameter, N_p the number of particles, $\boldsymbol{s}_{min}$ and $\boldsymbol{s}_{max}$ the limits of the state space and D the state space dimension (here $D = 6$). Experiments have shown a slight increase in performance of the tracking algorithm when applying roughening.

A second extension concerns the measurement model introduced in Section 2.11.2. In its present form, the model assumes that every incoming measurement is the result of the observation of a reflective vest. For a state

hypothesis $\boldsymbol{s}_t^{[k]}$ to receive a high weight, all measurements provided in the set of observations $\mathcal{P}_t$ must obtain high individual likelihoods $p(\hat{\boldsymbol{p}}_t^{[i]}|\boldsymbol{s}_t)$, a fact which is expressed by the product rule applied in Eq. 2.58. However, despite classification, the set $\mathcal{F}_{vest}$ occasionally contains features that do not originate from a reflective vest. Typically, this occurs if a reflective object in the scene appears very similar in shape to the reflectors of a vest and therefore classification fails. We address this problem by introducing a data association mechanism. When calculating the weight $w_t^{[k]}$ of a particle $\boldsymbol{s}_t^{[k]}$, measurements are only considered if the Mahalanobis distance between the particle position $\boldsymbol{p}_t$ and the measurement $\hat{\boldsymbol{p}}_t$, defined by

$$d_M(\hat{\boldsymbol{p}}_t, \boldsymbol{p}_t) = \sqrt{(\hat{\boldsymbol{p}}_t - \boldsymbol{p}_t)^T \Sigma^{-1} (\hat{\boldsymbol{p}}_t - \boldsymbol{p}_t)} \tag{2.60}$$

is smaller than a threshold λ_M. Here, Σ denotes again the rotated covariance matrix introduced in Eq. 2.56. Good performance results have been achieved with $\lambda_M = 3$.

Chapter 3
Results

The reflective vest detection and tracking system is evaluated in four different test scenarios as listed in Table 3.1. The evaluation is carried out for distances up to 10 m as this represents the limit of ranges at which vests can be detected with the current hardware in use. A sensor unit consisting of the camera system and a 2D laser range scanner (SICK LMS-200), both fixed to a solid mechanical frame, is used for the data acquisition (cf. Figure 3.1a). An extrinsic calibration was carried out to obtain the position and orientation of the laser range scanner relative to the camera [16]. The sensor unit is mounted at a height of approximately 1.5 m on a mobile platform with four hard rubber wheels (cf. Figure 3.1b). The evaluation scenarios are all situated in even terrain in order to facilitate the extraction of ground-truth data. An evaluation of the system on uneven ground is left to future work.

3.1 Preprocessing

Several training and validation data sets are acquired for each scenario by simultaneously recording the raw camera images and the 2D laser readings. Figure 3.2 illustrates the characteristic appearance of the image material acquired in the different data sets. During the acquisition of all sets, a single person wearing a reflective vest is always in the field of view of the camera

Scenario	Environment
1	Indoors, warehouse-like environment
2	Outdoors, car parking area, clear weather conditions
3	Outdoors, car parking area, direct sunshine into the camera
4	Outdoors, storage yard, light snowfall

Table 3.1: Test scenarios featured in the evaluation of the system

a) b)

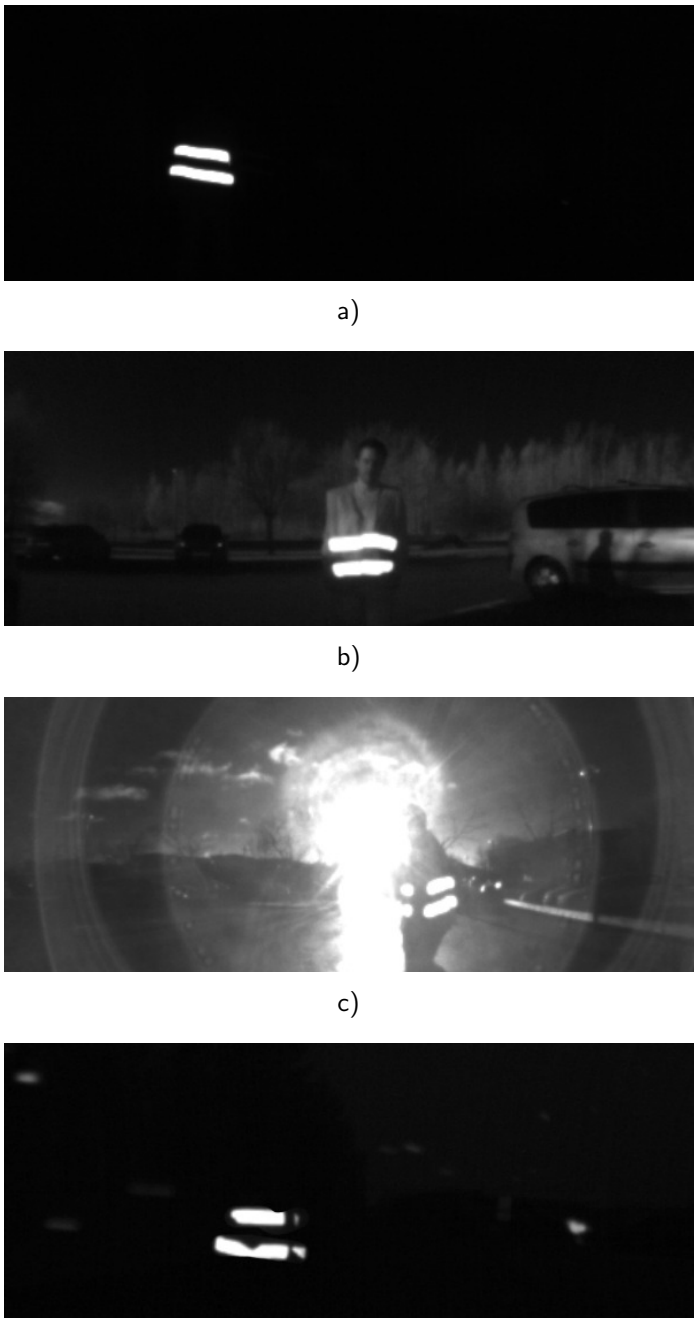
Figure 3.1: The figure shows the hardware setup used for data acquisition, consisting of **a)** the measurement unit with camera system and laser range scanner and **b)** the 4-wheeled mobile platform to which the measurement unit is attached.

and walking around in a distance range up to 10 m. The mobile platform is in constant motion at a speed of approximately 0.5 m/s. One data set per scenario is held back for evaluation purposes while the remaining sets served as training data. Table 3.2 summarizes the values of the different system parameters used in the evaluation setup.

All the acquired data sets are preprocessed to detect the set of raw features $\mathcal{F}_{raw}$ and to extract the corresponding local image descriptors $\boldsymbol{r}$. An upright SURF descriptor of 64 floating point variables, a BRIEF descriptor of 256 binary variables, and a BRISK descriptor of 512 binary variables are extracted for every feature. A ground-truth class label $\tilde{c}$ is manually assigned to each descriptor indicating whether it corresponds to a vest feature (label $\tilde{c} = 1$) or not (label $\tilde{c} = 0$). Furthermore, the ground-truth distance and position of the person wearing the reflective vest is extracted from the laser readings and assigned to the descriptors.

Supervised learning is applied to obtain the models of the feature classifier and the distance regressor. We train a Random Forest classifier on 50k extracted image descriptors and the corresponding labels to obtain the classifier model described in Section 2.8. Likewise, we train a Random Forest regressor on 30k image descriptors labeled as vest features and the corresponding ground-truth distance between the camera and the person to obtain the model of the regressor model described in Section 2.9.

The evaluation is then performed by processing the validation data set of each scenario and comparing the obtained results with the ground-truth labels assigned during preprocessing.



a)

b)

c)

d)

Figure 3.2: The figures illustrate the typical characteristics of the images I_f acquired in the different test scenarios. **a)** Scenario 1 with ideal dark background and bright reflectors **b)** Scenario 2 with average intensity values slightly increased **c)** Scenario 3 with heavily increased intensity values and various lens artifacts making the vest detection much more challenging **d)** Scenario 4 with several high intensity areas arising from the reflection of the IR flash on snowflakes near the camera unit.

Parameter	Description	Value
f_a	Image pair acquisition rate	∼15 Hz
t_a	Time delay between the acquisition of I_f and I_{nf}	∼35 ms
$W \times H$	Dimensions of the unwrapped input images I_f and I_{nf}	600x240 Pixel
b	Feature detection window border size	40 Pixel
ω_{LK}	Half window size of the LK feature tracker window	7 Pixel
ω_{ID}	Half window size for the intensity difference check	5 Pixel
λ_{ID}	Threshold for the intensity difference check	30.0
ω_{P_0}	Minimum patch size for descriptor extraction	8 Pixel
N_{tr}	Number of trees in the random forest classifier/regressor	20
λ_{vest}	Vest classification threshold	0.5
N_p	Number of particles in the particle filter	1000
$\sigma^2_{\dot{x}}, \sigma^2_{\dot{y}}, \sigma^2_{\dot{z}}$	Variance of the motion model uncertainty	0.25
σ^2_{rad}	Variance of the radial measurement uncertainty	0.5^2
σ^2_{tg}	Variance of the tangential measurement uncertainty	3.0^2
A_{bias}/B_{bias}	Measurement model bias correction parameters	-1.0/0.5
K_{jitter}	Particle roughening tuning factor	0.2
λ_M	Mahalanobis distance threshold for outlier elimination	3.0

Table 3.2: Values of the various system parameters used for the evaluation setup

Scenario	Average Features per Image I_f	Portion of Vest Features	Vest Detection Rate
1	2.35	100.00%	98.73%
2	5.26	53.85%	95.23%
3	56.72	4.03%	88.37%
4	2.83	96.48%	90.44%

Table 3.3: The table shows the result of the feature detection process for the different test scenarios. The *portion of vest features* indicates the percentage among all detected features that actually corresponds to a reflective vest. Finally, the *detection rate* represents the number of input images I_f in which a reflective vest is at least identified by one raw feature divided by the total number of input images.

3.2 Feature Detection

To evaluate its performance, the feature detector (Section 2.5) is applied on each image I_f in a validation data set, resulting in a set of raw features $\mathcal{F}_{raw}$. If a reflective vest is identified with at least one feature $f \in \mathcal{F}_{raw}$ the detection process for image I_f is declared successful. The *vest detection* rate is defined as the ratio between images in which the vest is successfully detected and the total number of images in the data set. Table 3.3 shows the results of the feature detection process for the different scenarios. The *average number of features per image* indicates the mean size of the feature set $\mathcal{F}_{raw}$ over the entire dataset while the *portion of vest features* is the ratio of features $f \in \mathcal{F}_{raw}$ that were manually labeled as vest features.

3.3 Feature Classification

In a second step, we evaluate the system's ability to correctly split the set of detected features $\mathcal{F}_{raw}$ into a set of vest features $\mathcal{F}_{vest}$ and a set of non-vest features $\mathcal{F}_{non-vest}$. The evaluation assesses the performance of several processing steps as a group (cf. Fig. 2.1, namely the feature tracking and intensity check (Section 2.6), the feature description (Section 2.7) and the feature classification (Section 2.8). Every set of raw features $\mathcal{F}_{raw}$ detected in the series of images I_f is processed to obtain a corresponding set of predicted vest features $\mathcal{F}_{vest}$. The set of predicted non-vest features is defined as $\mathcal{F}_{non-vest} = \mathcal{F}_{raw} \setminus \mathcal{F}_{vest}$. The result of the binary classification into vest and non-vest features is then compared to the ground-truth label manually assigned during preprocessing.

In order to assess the performance of the classification, we divide the classified features into four categories:

- **True Positives (TP):**
  Features $f \in \mathcal{F}_{raw}$ correctly assigned to $\mathcal{F}_{vest}$
- **True Negatives (TN):**
  Features $f \in \mathcal{F}_{raw}$ correctly assigned to $\mathcal{F}_{non-vest}$
- **False Positives (FP):**
  Features $f \in \mathcal{F}_{raw}$ incorrectly assigned to $\mathcal{F}_{vest}$
- **False Negatives (FN):**
  Features $f \in \mathcal{F}_{raw}$ incorrectly assigned to $\mathcal{F}_{non-vest}$

Using this terminology, we introduce the *precision*, a quantity that represents the fraction of features in $\mathcal{F}_{vest}$ that effectively corresponds to vest features:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3.1}$$

Additionally, we introduce the quantity named *recall* which is the fraction of effective vest features that is correctly classified:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3.2}$$

We finally define the classification *accuracy* which represents the overall fraction of correctly classified features:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{3.3}$$

The classification performance for the different scenarios is evaluated according to the above measures. Scenario 1 is situated in a perfect indoor environment with no other IR light source than the IR flash and no other reflective objects than the reflective vest. Therefore, the set of raw features

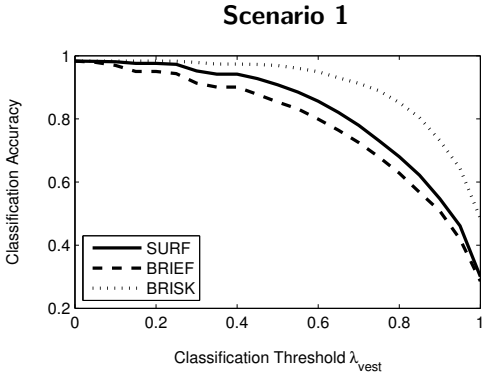


Figure 3.3: The figure shows the accuracy of the binary classification of raw features $f \in \mathcal{F}_{raw}$ into a set of vest features $\mathcal{F}_{vest}$ and a set of non-vest features $\mathcal{F}_{non-vest}$ for scenario 1. The curves show the results of classification based on the three different feature descriptors SURF, BRIEF and BRISK and for a varying classification threshold λ_{vest}. The case $\lambda_{vest} = 0$ corresponds to the situation where all features in $\mathcal{F}_{reflex}$ are considered as vest features ($\mathcal{F}_{vest} = \mathcal{F}_{reflex}$).

$\mathcal{F}_{raw}$ contains only items that truly correspond to vest features and consequently we have FP = 0, TN = 0, Precision = 1 and Accuracy = Recall. For this reason, only an accuracy graph with variable threshold λ_{vest} is shown for scenario 1 (see Figure 3.3).

Scenarios 2–4 feature image material acquired outdoors, with other reflective material than the reflective vest in the scene, including metallic surfaces, windows or even snowflakes. Furthermore, the IR irradiation of the sun produces images with higher average intensity values. Under these circumstances, the feature set $\mathcal{F}_{raw}$ contains both vest and non-vest features and the performance of the classification is most accurately assessed by precision-recall graphs, according to Figure 3.4.

3.4 Distance and Position Estimation

The trained model of the random forest regressor (Section 2.9) is applied to obtain a distance estimate for every predicted vest feature in $\mathcal{F}_{vest}$. The distance estimate combined with the feature coordinates $\boldsymbol{u_f} = (u_f, v_f)$ are then used with the intrinsic camera model to compute 3D position estimate according to Section 2.10. The resulting distance and position estimates per feature are compared to the ground-truth labels and the resulting estimation errors are shown in Figure 3.5–3.8.

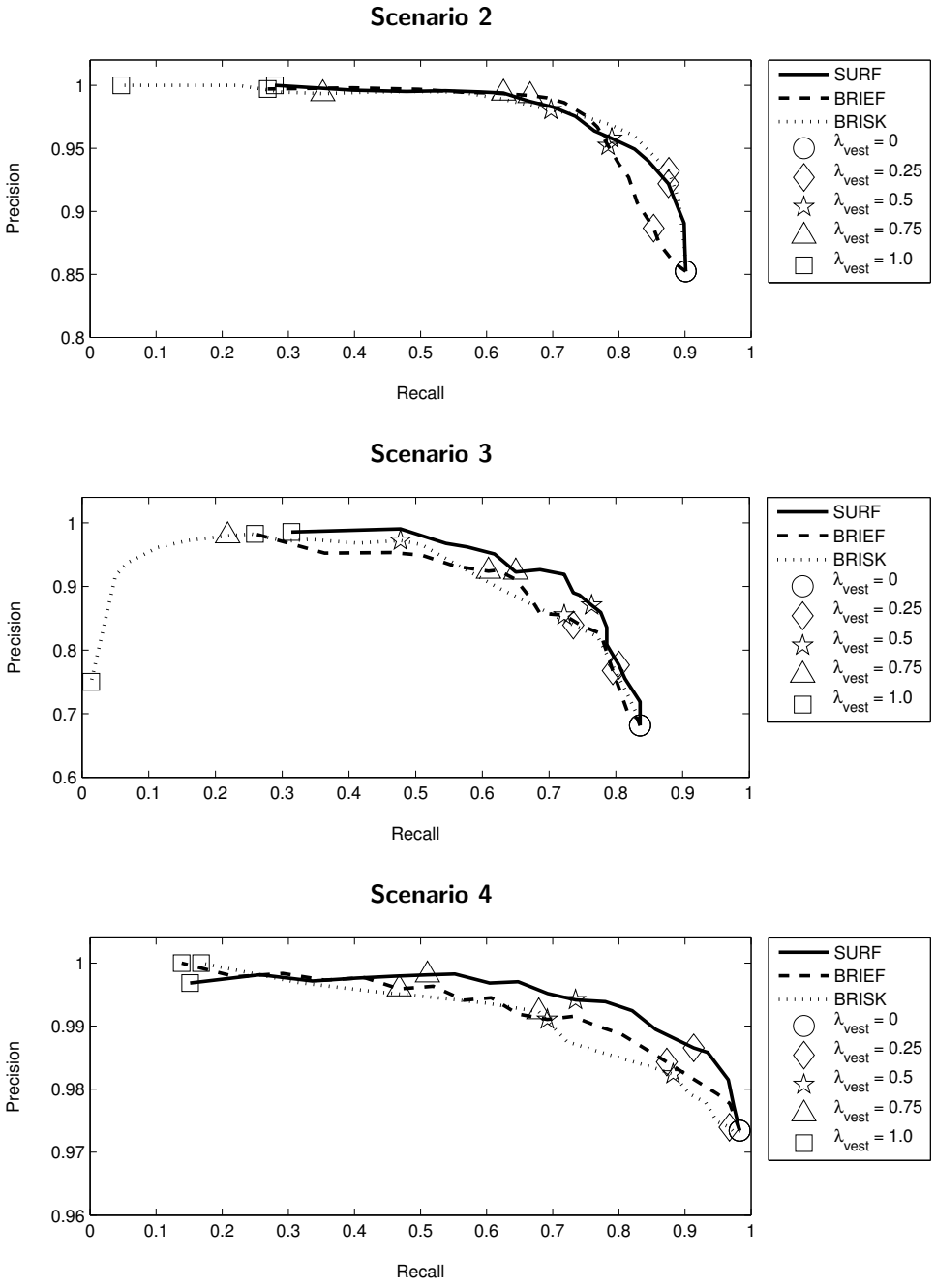


Figure 3.4: The figures show precision-recall curves obtained by classifying the raw features $f \in \mathcal{F}_{raw}$ into a set of vest features $\mathcal{F}_{vest}$ and a set of non-vest features $\mathcal{F}_{non-vest}$ based on the three different feature descriptors SURF, BRIEF and BRISK. The curves are obtained by varying the classification threshold λ_{vest} between 0 and 1. The point labeled with $\lambda_{vest} = 0$ (circle) represents the case where no classification with the classifier described in Section 2.8 is applied and $\mathcal{F}_{vest}$ is obtained by considering all the features $f \in \mathcal{F}_{reflex}$ as vest features.

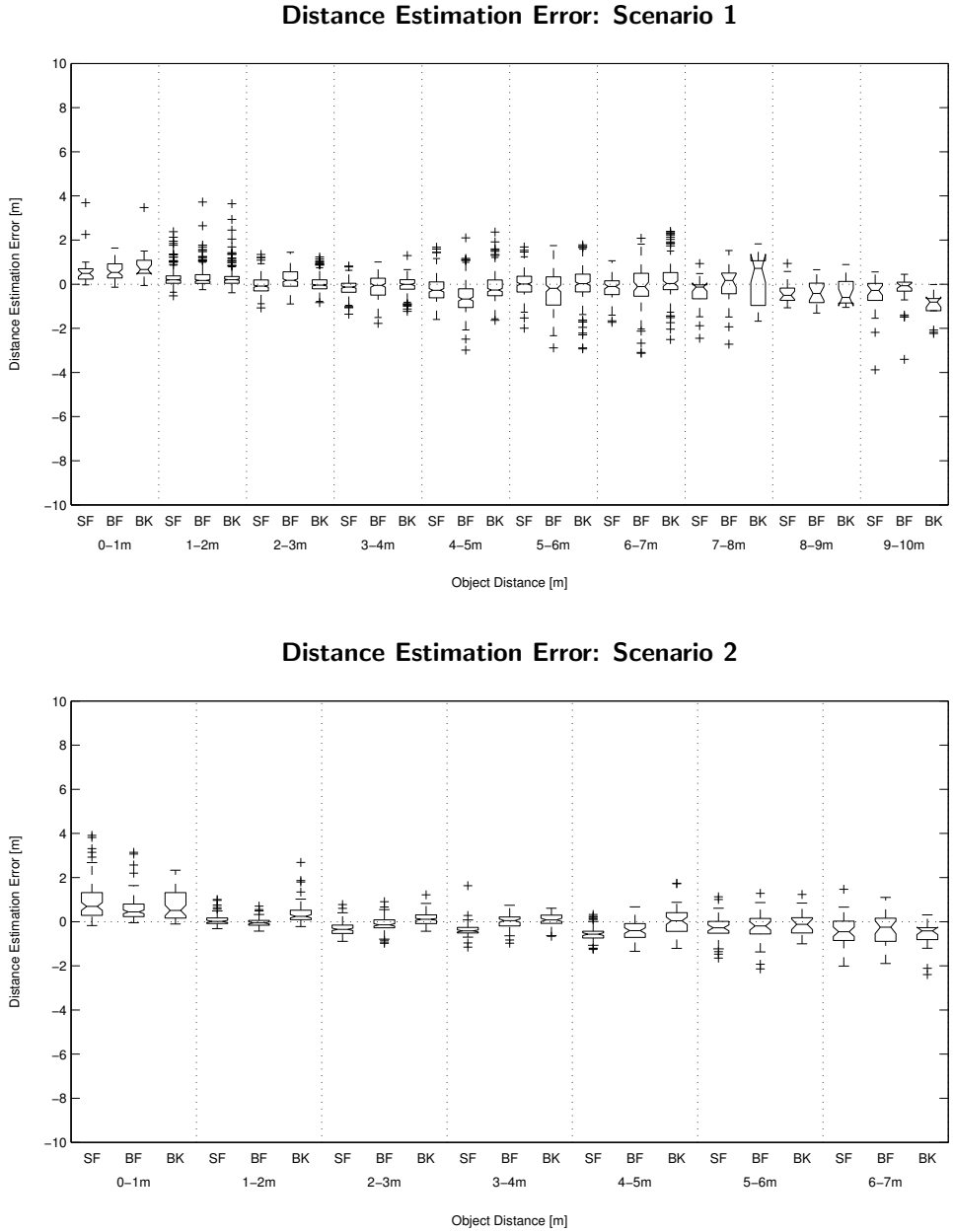


Figure 3.5: Distance estimation error for the scenarios 1 and 2 at different distances ranges. The indications SF (SURF), BF (BRIEF) and BK (BRISK) specify the image descriptor on which the estimation is based.

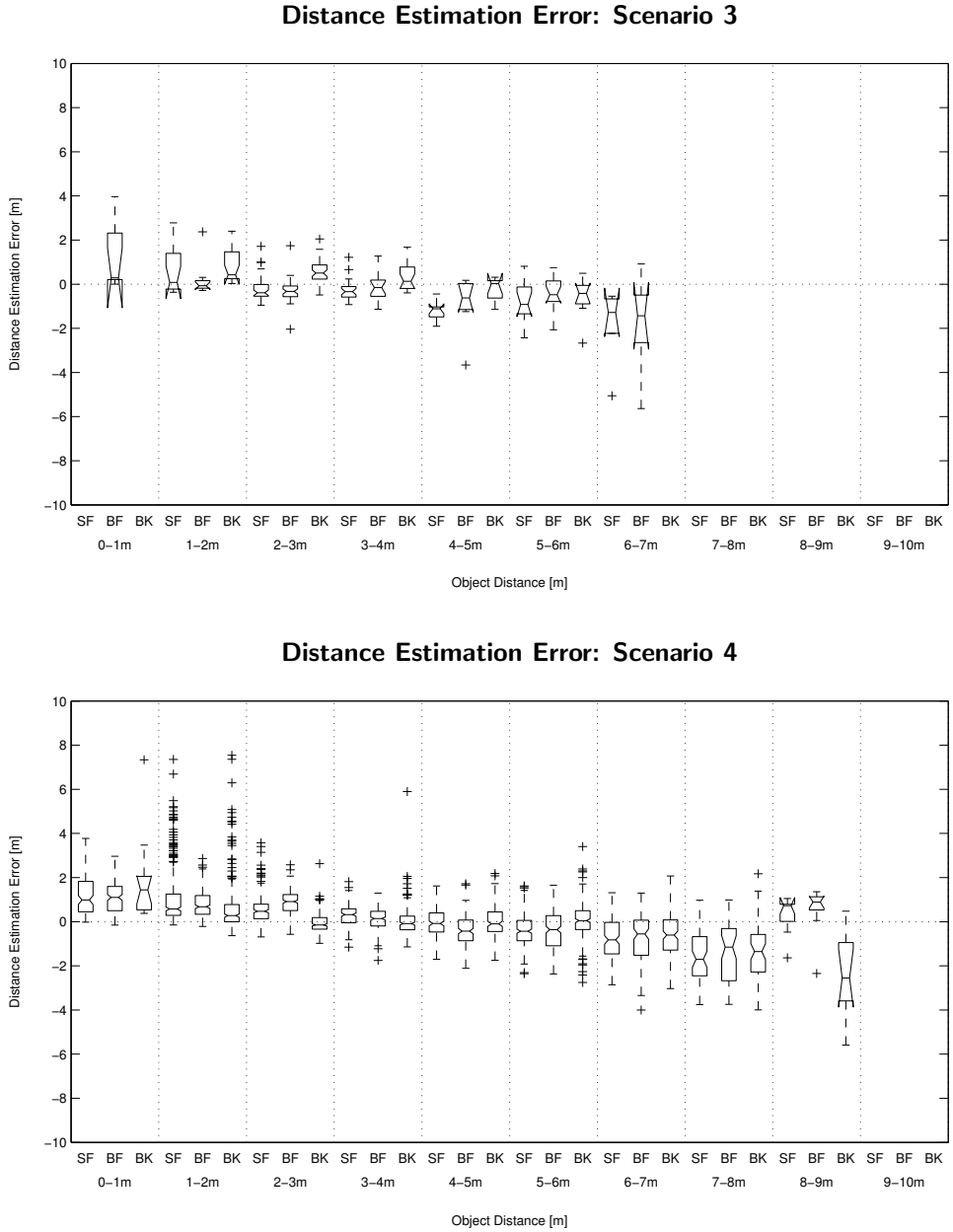


Figure 3.6: Distance estimation error for the scenarios 3 and 4 at different distances ranges. The indications SF (SURF), BF (BRIEF) and BK (BRISK) specify the image descriptor on which the estimation is based. Missing plots indicate that the vest detection failed and no distance estimation could be performed.

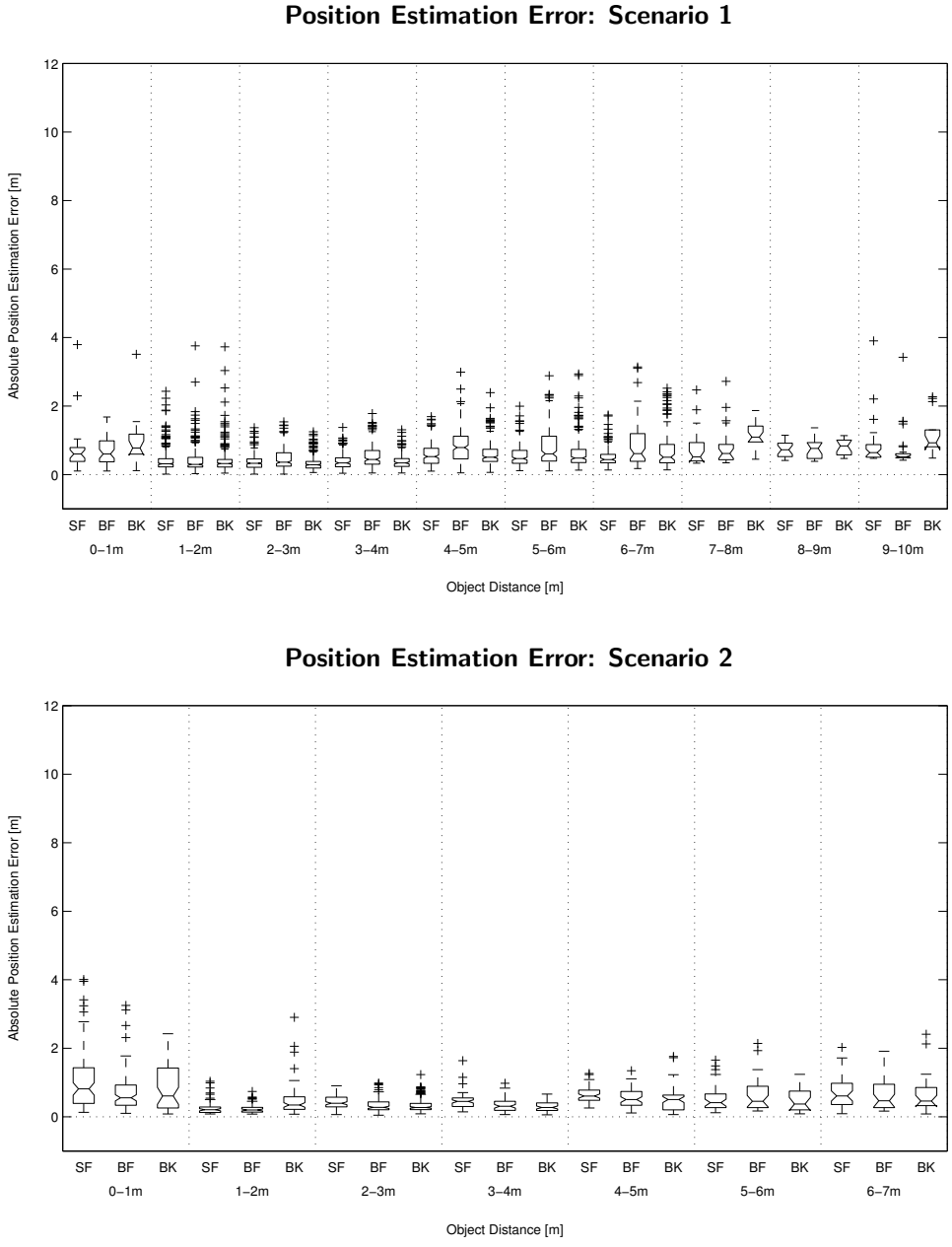


Figure 3.7: Absolute position estimation error for the scenarios 1 and 2 at different distances ranges. The indications SF (SURF), BF (BRIEF) and BK (BRISK) specify the image descriptor on which the estimation is based.

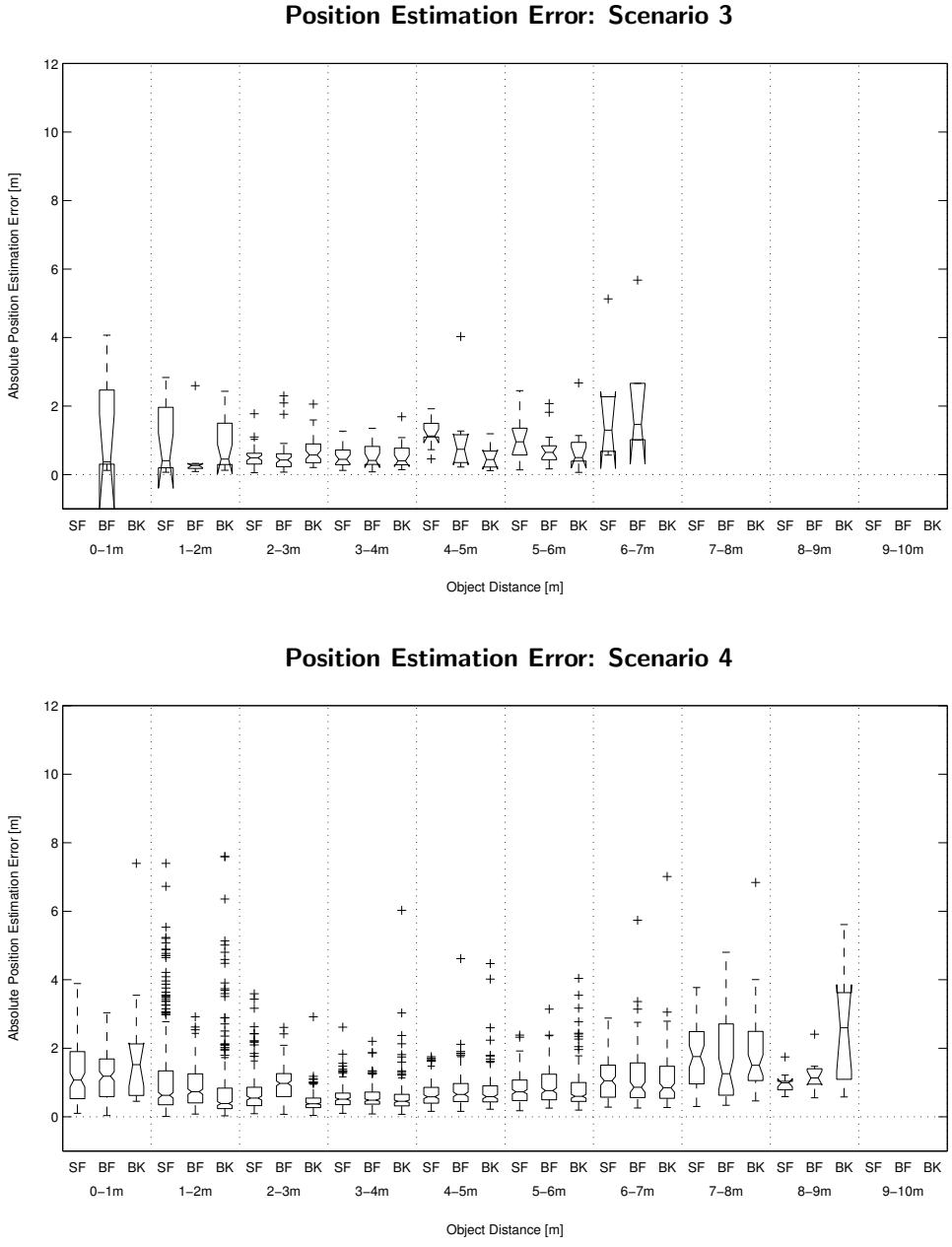


Figure 3.8: Absolute position estimation error for the scenarios 3 and 4 at different distances ranges. The indications SF (SURF), BF (BRIEF) and BK (BRISK) specify the image descriptor on which the estimation is based. Missing plots indicate that the vest detection failed and no distance estimation could be performed.

3.5 Vest Tracking

Finally, an evaluation of the particle filter based vest tracking algorithm is carried out. Two quantities are observed in the evaluation, the ability of the algorithm to consistently keep track of the reflective vest in the scene as well as the ability to accurately estimate the position of the vest in cases where it is considered as successfully tracked.

To decide whether a vest is tracked, a characteristic measure for the spread of the particles in the state space is elaborated. At any time step t, we perform a *Principal Component Analysis (PCA)* by eigenvalue decomposition of the 6×6 sample covariance matrix computed from all the N_p individual particle states $\boldsymbol{s}_{\boldsymbol{t}}^{[i]}$. The highest eigenvalue, representing the variance on the principal axis of the state space, is used as a measure for the spread of the particles. We refer to this measure as the particle spread $\Lambda(\boldsymbol{s}_t)$ and consider a vest as tracked if $\Lambda(\boldsymbol{s}_t) < \lambda_{spread} = 5\,m^2$. We define the *Tracking Rate (TR)* as the ratio between the sum of time intervals in which the reflective vest is successfully tracked and the total length of the sequence featured in a given scenario.

Only if at a given time t a vest is considered as tracked, a final state estimate $\hat{\boldsymbol{s}}_t$ is computed using the weighted mean of the particle states according to Eq. 2.51. We define the *Mean Absolute Error (MAE)* as the average value of the absolute error that is committed in estimating the position of the reflective vest over the ensemble of images featured in a scenario.

Table 3.4 shows the results of the feature tracking process for the different test scenarios. Figure 3.9–3.10 illustrate the temporal evolution of the distance and position estimation error as well as for the particle spread. The results shown in the figures correspond to reflective vest tracking based on the SURF descriptor, as it showed the best average performance over the different data sets. The respective results for the BRIEF and BRISK descriptors are presented in Figure A.1–A.4 in the Appendices.

Descriptor	**Scenario 1**		**Scenario 2**		**Scenario 3**		**Scenario 4**	
	TR [%]	MAE [m]	TR [%]	MAE [m]	TR [%]	MAE [m]	TR [%]	MAE [m]
SURF	93.54	0.43	89.55	0.62	86.10	0.71	80.48	0.73
BRIEF	94.68	0.49	89.86	0.57	91.64	0.89	79.17	0.92
BRISK	93.57	0.45	67.91	0.53	49.52	1.01	84.07	0.67

Table 3.4: The table summarizes the results of the reflective vest tracking for the different test scenarios. The *Tracking Rate (TR)* is the percentage of time at which the vest is considered as successfully tracked. The *Mean Absolute Error (MAE)* represents the average position estimation error committed by the tracking algorithm.

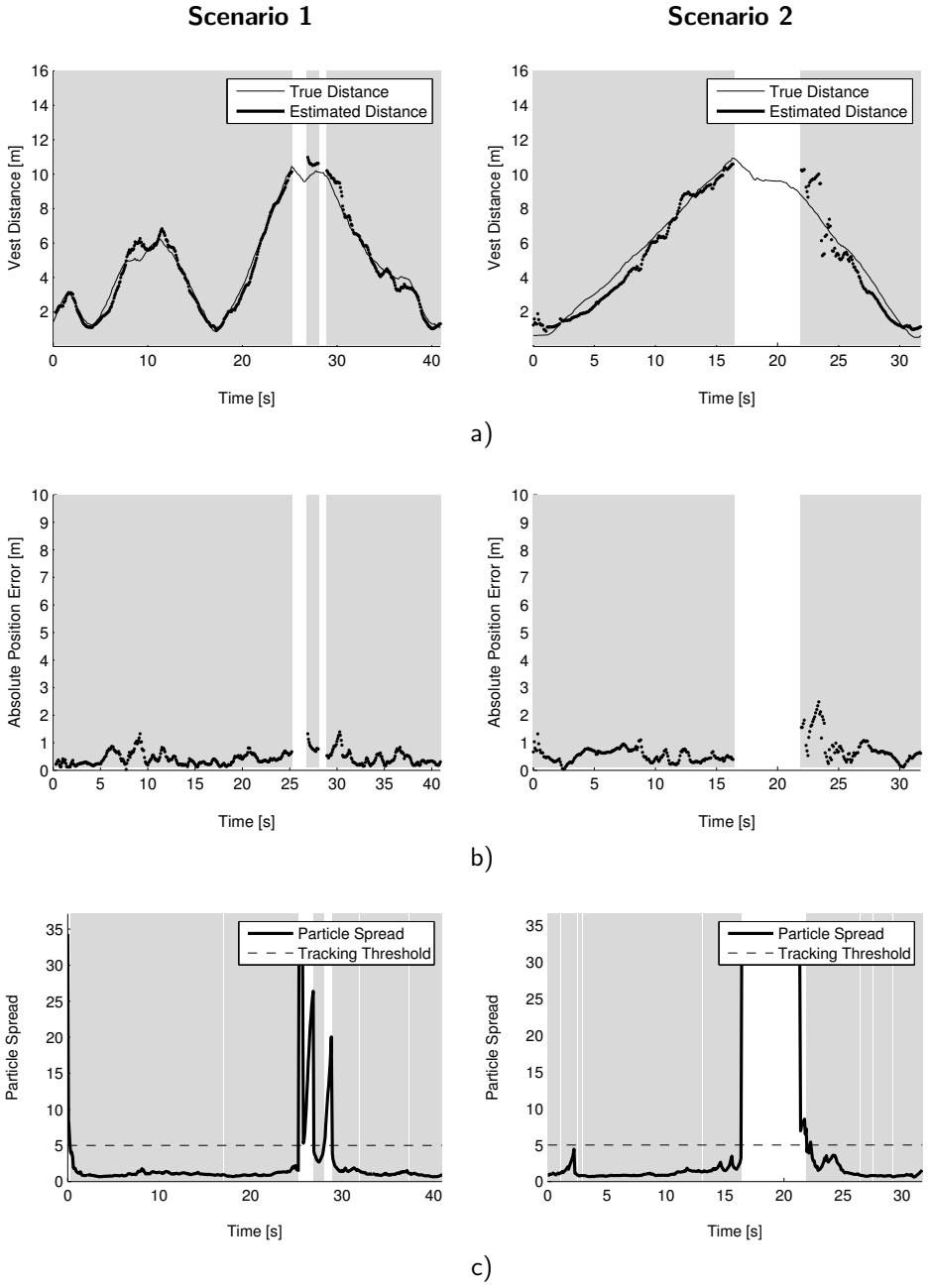


Figure 3.9: Temporal evolution of the reflective vest tracking for scenarios 1 and 2 in case of the SURF descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

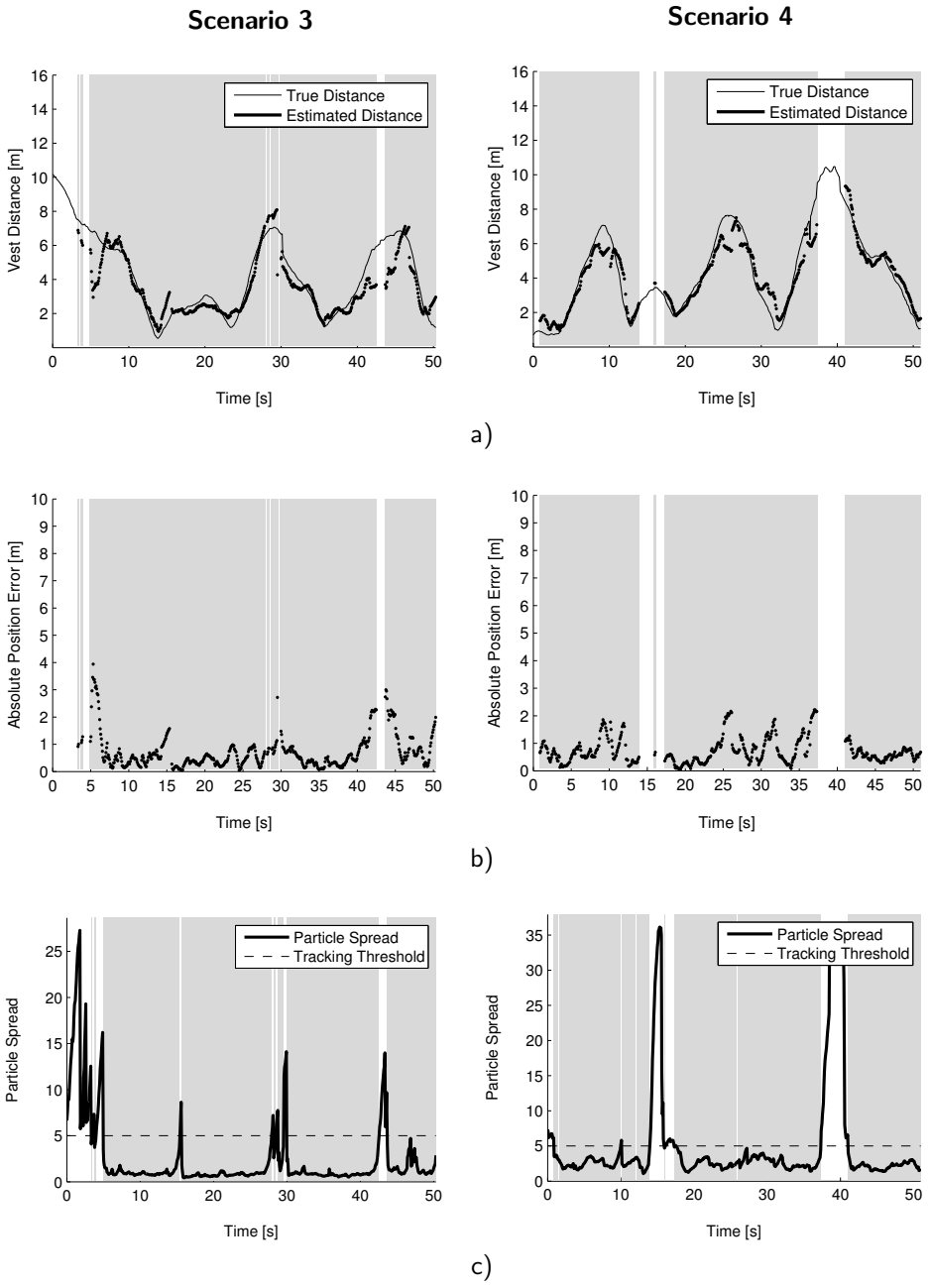


Figure 3.10: Temporal evolution of the reflective vest tracking for scenarios 3 and 4 in case of the SURF descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

Chapter 4
Discussion

In Chapter 3, the vest detection and tracking system was evaluated in different test scenarios. This chapter discusses the important aspects of the results obtained for the different sections of the algorithm.

Feature Detection
The evaluation of the feature detection process as presented in Table 3.3 reveals three important aspects. First, the number of raw image features extracted from the input images I_f heavily depends on the presence of external infrared light sources. The more ambient light is present with IR wavelengths corresponding to the center wavelength of the camera's bandpass filter, the higher the overall brightness of the acquired image material (cf. Figure 3.2), and consequently, the higher the number of detected high intensity blobs. This drastically influences the ratio among all detected features that truly correspond to a reflective vest. Secondly, this same ratio is also affected by the presence of reflective materials that do not correspond to a reflective vest. While in scenario 1, the only reflective material are the vest reflectors, scenarios 2 and 3 contain reflective metallic surfaces on cars and in scenario 4, perturbing reflections are caused on snowflakes. The reflection of the IR flash emitted by these objects lead to additional high intensity regions in the image and entail features detected in the corresponding areas. Finally, the results show that the vest detection rate is decreased by external disturbances that limit the range of distances at which the vest reflectors reliably produce features detected in the image.

Scenario 1 represents the optimal case for successful vest detection, as it is situated indoors in an environment with no disturbing external IR light source. Furthermore, no reflective object but the vest appears in the images. In consequence, feature detections exclusively originate from vest reflectors. The vest detection rate is close to 100 % as visibility is not limited by any disturbing factors. Scenario 2 is situated outdoors in clear weather

conditions and the acquired images thus appear slightly brighter due to the influence of the IR wavelengths contained in the sunlight. Nonetheless, the intensity of the additional infrared light is modest compared to the reflected IR flash, and the vest detection rate is not seriously affected. The number of detected features is doubled as a result of the increased image brightness and the presence of reflective metallic surfaces, reducing the portion of true vest features to roughly 50 %. In Scenario 3 and 4, visibility is seriously restricted either by the direct sunshine into the camera which produces numerous lens artifacts (Scenario 3) or by snowfall (Scenario 4). Reliable generation of vest features is only provided up to 6-7 m distance and consequently, the vest detection rate is decreased by approximately 10 %. Furthermore, the much higher average intensity of the images in Scenario 3 lead to the fact that the predominant part of detected features does not originate from a reflective vest.

The evaluated quantities play an important role on two different levels. On the one hand, the portion of vest features indicates the feature ratio that the subsequent processing steps need to extract while discarding all other items. This task is considerably simplified if the portion is high. On the other hand, the vest detection rate plays a more important role when it comes to tracking a vest over time. The higher the vest detection rate, the better is the chance that a vest can be consistently tracked over the entire image sequence.

Feature Classification

The extraction of vest features from the initial raw feature set is accomplished by an ensemble of four processing steps, namely feature tracking, intensity check, feature description and feature classification. The results of this feature elimination process are shown in Figure 3.3 and 3.4. In Scenario 1, no features are detected that do not correspond to a reflective vest, due the perfect conditions that were already discussed. In this ideal case, the concerned processing steps, that in other conditions serve to eliminate non-vest features, can only be counterproductive because features are removed that are erroneously classified as non-vest features. This is illustrated in Figure 3.3 where the accuracy represents the fraction of raw features that is still preserved in the vest feature set, depending on the choice of the classification threshold λ_{vest}. The figure shows that for Scenario 1, the least amount of classification errors is committed if the Random Forest classifier is trained on the BRISK descriptor. The SURF descriptor ranks second with a small advantage over BRIEF. The graph further indicates that the accuracy decreases relatively moderate for classification thresholds lower than 0.5. To conclude, we shall take note of the fact that the algorithm's negative impact on the number of extracted vest features should not mistakenly lead to the conclusion that the evaluated processing steps are dispensable or that the best choice for λ_{vest} is 0, as the perfect conditions encountered

in this scenario are only of illustrative value and do only seldom correspond to a real-world industrial environment.

Figure 3.4 shows the results of the feature elimination process for the scenarios 2, 3 and 4. The interest of the detection algorithm now becomes apparent. The elimination of non-vest features by means of the first two processing steps, namely feature tracking and intensity difference check, results in a feature set $\mathcal{F}_{reflex}$ whose content is described by the circular marker in the precision-recall curves. The precision represents the ratio of effective vest features among all features in $\mathcal{F}_{reflex}$. In the ideal case it equals 1. To assess the benefit of applying the first two processing steps, the precision at the location of the circular marker can be compared to the initial portion of vest features in the respective data set, given in Table 3.3. A comparison reveals that the ratio of true vest features in the set $\mathcal{F}_{reflex}$ is significantly increased when compared to the initial set of detected features. In scenario 2 the value increases from about 54 % to 85 % and in scenario 3 from 4 % to roughly 70 %. In scenario 4 the initial ratio is already high and is increased by approximately 1 % to reach 97 %. While increasing precision, recall is kept at a high level with values around 90 %, 83 % and 99 % for the scenarios 2, 3 and 4. This means that the number of false negatives caused by misclassification is very low and, thus, only little vest features are mistakenly eliminated by performing the feature tracking and intensity check.

The application of the feature description and classification process then helps to further improve precision, primarily by eliminating features that correspond to objects that are reflective but other than reflective vests. The higher the classification threshold λ_{vest} is chosen, the more restrictive the classifier acts in selecting the features to place in $\mathcal{F}_{vest}$. This not only reduces the number of false positives but also increasingly results in an important amount of false negatives, that is, vest features are mistakenly eliminated. The choice of λ_{vest} is therefore a trade-off and should account for both high precision and recall. After a comparison of all four scenarios and all three different feature descriptors we suggest the use of a SURF descriptor together with $\lambda_{vest} = 0.5$.

Distance and Position Estimation

Figure 3.5 and 3.6 depict the estimation error resulting from the individual distance predictions for features $f \in \mathcal{F}_{vest}$ with the Random Forest regressor. For scenarios 1 and 2 the precision of the distance estimation is relatively stable over the entire distance range considered in the evaluation and accuracy is within a decimeter range. A slight tendency to overestimate the distance at short ranges and to underestimate it at higher ranges can be observed. This effect is mainly due to the fact that the distance has a lower bound of zero and no training data was provided with distances higher than 10 m. The plots also report sporadic but large outliers indicating a dis-

tance estimation error of several meters. Further investigation revealed that most of the outliers originate from misclassification errors, namely cases where non-vest features are classified as vest features (false positives). Under these circumstances, the distance regressor encounters a pattern that does not correspond to a reflective vest and that has not been trained during learning. Consequently, the estimated distance value is meaningless but will undesirably be included in the set of measurements.

Under difficult conditions as it is the case in scenario 3 and 4, accuracy and precision of the distance estimation are negatively affected and detections are restricted to ranges of 7 m and 9 m for the respective scenario. However, the system still provides reliable measurements. The resulting absolute position estimation errors are reported in Figure 3.7 and 3.8 and show the same tendencies. This indicates that the final accuracy of the vest position estimation primarily depends on the distance estimator and that the accuracy of the 3D projection of features by means of the camera model is much higher.

The three evaluated feature descriptors yield all fairly similar results, with small differences in individual scenarios and at individual distance ranges. The rotation invariance of the BRISK descriptor seems not to lead to a clear advantage over SURF and BRIEF. This can be justified by the fact that the observed patterns themselves show already a high degree of rotational symmetry and rotational invariance of the image descriptor is therefore superfluous.

Vest Tracking

The evaluation of the vest tracking algorithm assesses the ability of the particle filter to consistently keep track of the observed reflective vest over time. The results in Table 3.4 and Fig. 3.9–3.10 show that consistent tracking is possible over a large part of scenarios 1 and 2 and over considerable parts of scenarios 3 and 4. The filtering effect becomes very clear, especially in the first two scenarios, where from position estimates with considerable outliers in the meter range, a position estimate is obtained whose error lies in the centimeter range for big parts of the image sequence. Tracking based on the use of the SURF and BRIEF descriptors leads to similar performance results, with SURF offering lower estimation error and BRIEF a slightly higher tracking rate. The BRISK descriptor seems to be the most sensitive to changing external conditions among all descriptors.

It has been discussed above that the maximum range for vest detections is at roughly 10 m in optimal conditions but can be reduced by external influences as in scenarios 3 and 4. Consequently, the observed person gets out of focus if its distance approaches the limit, as the particle filter is provided with less and less measurements. To refocus on a person entering the sensor range, the particle filter needs to be provided with a certain amount of measurements until its state belief distribution is able to focus.

Chapter 5
Conclusion

In this Master thesis report we presented an approach for the detection and tracking of a person wearing a reflective vest. The system has been evaluated in an indoor warehouse-like environment as well as outdoors in different weather conditions. The experiments show that with a single camera setup we are able to detect a person wearing a reflective vest and produce accurate position estimates for distances ranging up to 10 m in good conditions, based on the processing of image features with an approach that includes machine learning. Yet, the results also indicate that tracking a person's position over time, based on single position estimates obtained per input image pair, is difficult, as sporadic but considerably large outliers occur. It has further been shown that the sensor range depends on the environment and in particular on the weather conditions, as factors such as snow or direct exposure to sunlight limit the visibility or create lens artifacts in the input images.

A tracking algorithm based on a particle filter, featuring a motion and a measurement model, has proved to be a valuable tool for combining the individual measurements as they are obtained over time. It has been illustrated how a probabilistic measurement model can account for both random and systematic errors when incorporating the individual measurements in order to obtain a filtered position estimate, which is expressed in form of an approximated probability density over the entire position search space.

Chapter 6
Further Work

The current version of the camera system allows reflective vest detection up to 10 m distance in good conditions. The limitation is mainly due to the decrease in intensity of the reflected portion of the IR flash and the decrease in spatial image resolution for increasing distances. A flash system equipped with more powerful IR LEDs and an imaging sensor with higher resolution will thus provide the hardware base for an improved version of the camera system with a sensor range extended to 20 m or above.

Moreover, the classical version of the particle filter, as employed in the underlying application, represents a single-target particle filter that only deals with the estimation of a single state. In order to simultaneously track several persons in the field of view of the camera, the problem of *multiple target tracking* has to be addressed. The extended problem consists in estimating multiple state processes while taking into account that even the number of estimated states evolves over time.

A future version of the camera system hardware will further include a combined accelerometer and gyroscope unit that will allow to substantially improve the motion model employed in the particle filter. Changes of an observed person's relative position due to rotation and acceleration of the camera system will be estimated by the additional sensory input. This will allow to reduce the considerable amount of uncertainty that is currently included in the motion model.

Future work also includes an extensive long-term evaluation of the system performance in a real-world industrial environment where the variety of encountered situations is much higher than in the evaluation carried out during this project. Scenarios that have not been evaluated so far, such as persons that are partly occluded or lying on the floor, need to be examined. Furthermore, the influence of various degrees of image motion blur caused by strong angular motion and vehicle vibration in uneven terrain has to be evaluated.

Appendix A
Additional Tracking Results

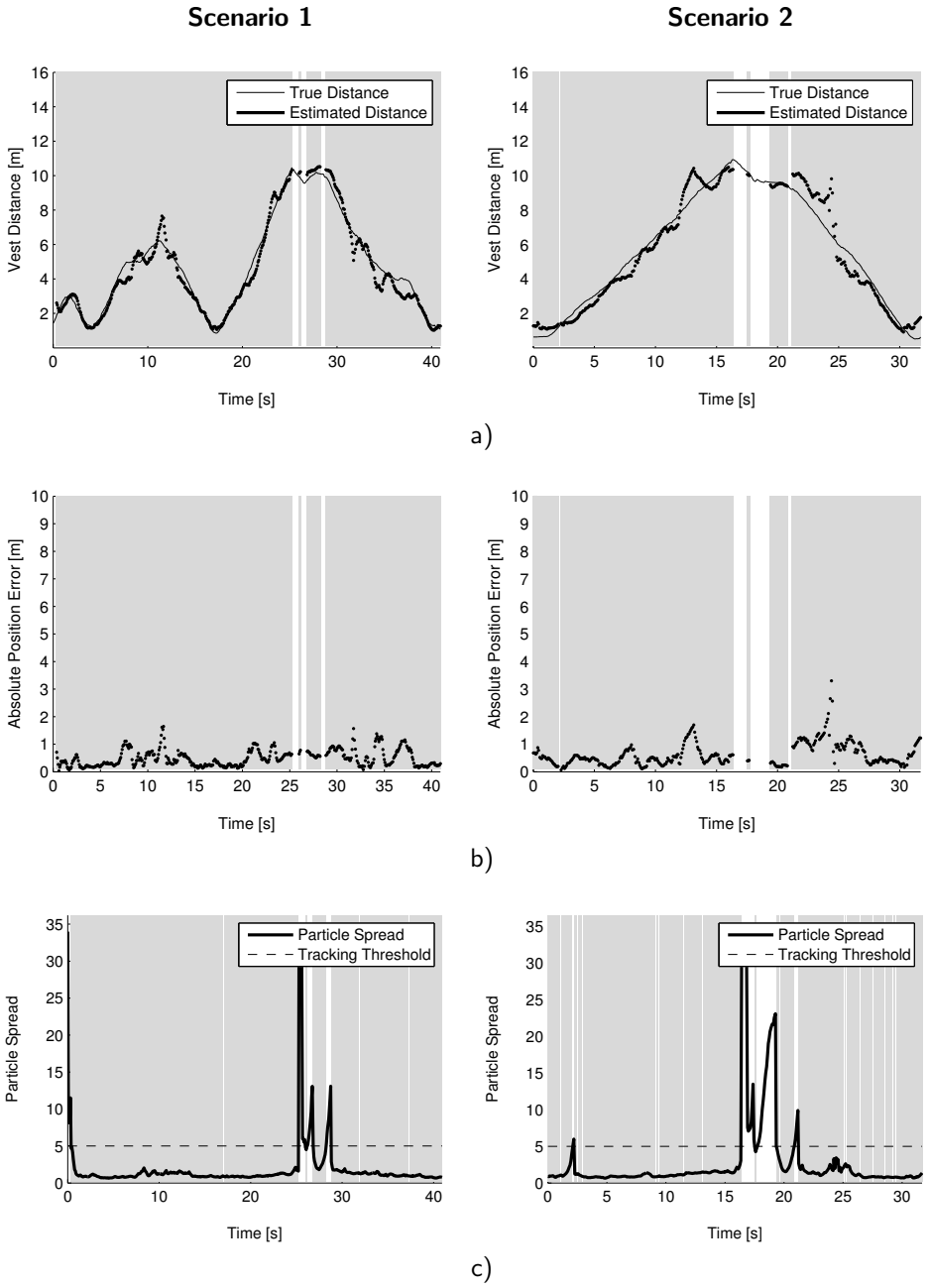


Figure A.1: Temporal evolution of the reflective vest tracking for Scenarios 1 and 2 in case of the **BRIEF** descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

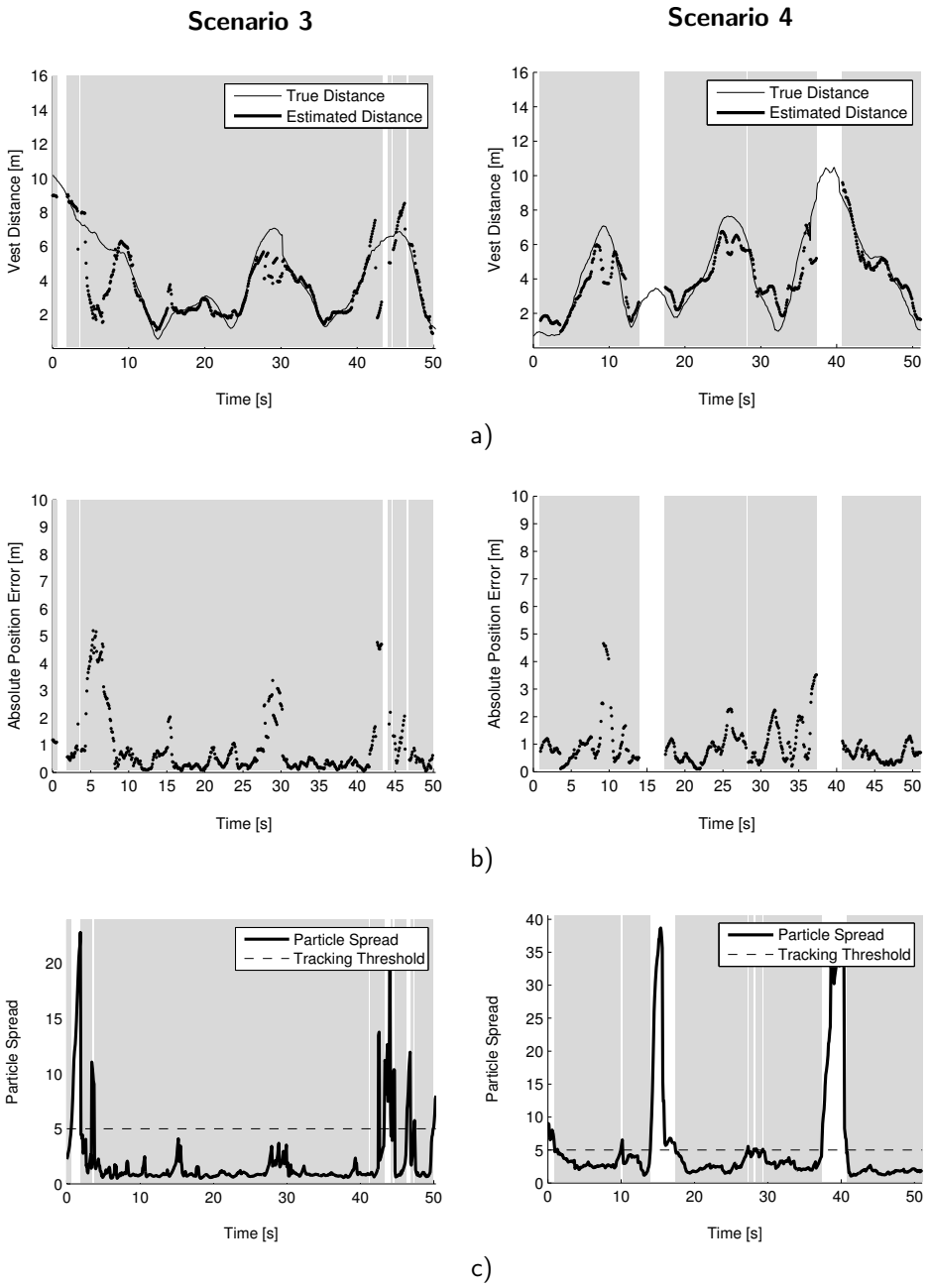


Figure A.2: Temporal evolution of the reflective vest tracking for Scenarios 3 and 4 in case of the **BRIEF** descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

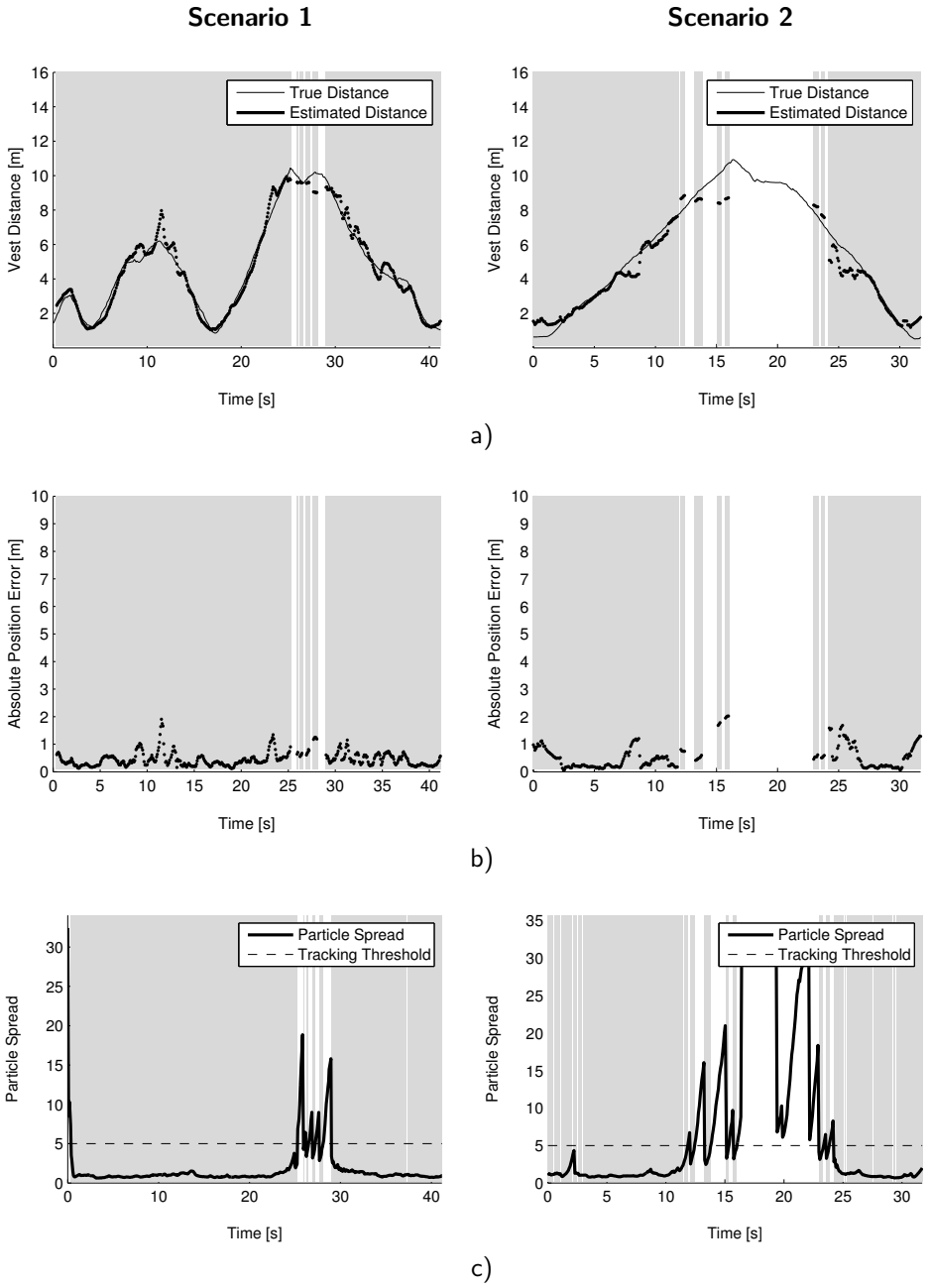


Figure A.3: Temporal evolution of the reflective vest tracking for Scenarios 1 and 2 in case of the **BRISK** descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

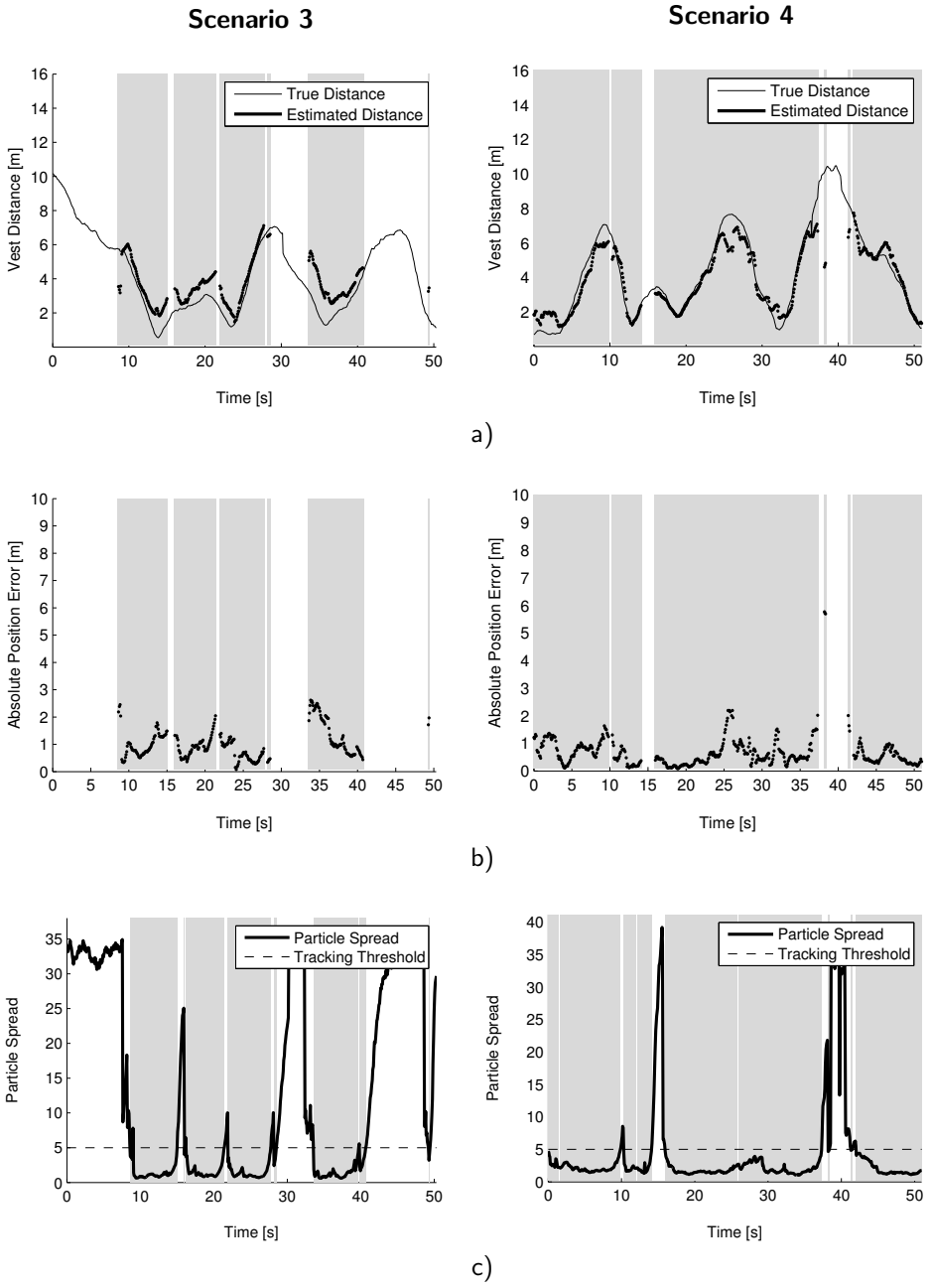


Figure A.4: Temporal evolution of the reflective vest tracking for Scenarios 3 and 4 in case of the **BRISK** descriptor. Regions marked with gray background indicate the time periods during which the vest is considered as tracked. **a)** Ground-truth and estimated distance between the camera the reflective vest **b)** Absolute estimation error of the vest position **c)** Spread of the particle set

Bibliography

[1] H. Andreasson, A. Bouguerra, T. Stoyanov, M. Magnusson, and A. Lilienthal, "Vision-based people detection utilizing reflective vests for autonomous transportation applications.," *IROS Workshop on Metrics and Methodologies for Autonomous Robot Teams in Logistics (MMART-LOG)*, 2011.

[2] G. Gate, A. Breheret, and F. Nashashibi, "Fast pedestrian detection in dense environment with a laser scanner and a camera," in *VTC Spring*, 2009.

[3] B. Mičušík and T. Pajdla, "Estimation of omnidirectional camera model from epipolar geometry," 2003.

[4] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *Proc. of The IEEE International Conference on Computer Vision Systems (ICVS)*, 2006.

[5] M. Agrawal, K. Konolige, and M. R. Blas, "Censure: Center surround extremas for realtime feature detection and matching," in *ECCV (4)* (D. A. Forsyth, P. H. S. Torr, and A. Zisserman, eds.), vol. 5305 of *Lecture Notes in Computer Science*, pp. 102–115, Springer, 2008.

[6] J.-Y. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker description of the algorithm," 2000.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Surf: Speeded up robust features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, pp. 346–359, 2008.

[8] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features.," in *ECCV (4)* (K. Daniilidis, P. Maragos, and N. Paragios, eds.), vol. 6314 of *Lecture Notes in Computer Science*, pp. 778–792, Springer, 2010.

[9] S. Leutenegger, M. Chli, and R. Siegwart, "BRISK: Binary Robust Invariant Scalable Keypoints," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011.

[10] L. Breiman, "Random forests.," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[11] A. Bosch, A. Zisserman, and X. Munoz, "Image classification using random forests and ferns," in *Proceedings of the 11th International Conference on Computer Vision, Rio de Janeiro, Brazil*, 2007.

[12] V. Lepetit and P. Fua, "Keypoint Recognition using Randomized Trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1465–1479, 2006.

[13] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Wadsworth and Brooks, 1984.

[14] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *Radar and Signal Processing, IEEE Proceedings F*, vol. 140, no. 2, pp. 107–113, 1993.

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

[16] Q. Zhang, "Extrinsic calibration of a camera and laser range finder," in *In IEEE International Conference on Intelligent Robots and Systems (IROS)*, pp. 2301–2306, 2004.