Novel Markovian Change Detection Models in Computer Vision

A thesis submitted for the degree of
Doctor of Philosophy

Csaba Benedek

Scientific adviser:
Tamás Szirányi, D.Sc.

Faculty of Information Technology
Pázmány Péter Catholic University

Computer and Automation Research Institute
Hungarian Academy of Sciences

Budapest, 2008

Acknowledgements

First of all I would like to thank my supervisor Professor Tamás Szirányi for his consistent help and support during my studies.
The support of the Computer and Automation Research Institute of the Hungarian Academy of Sciences (MTA-SZTAKI) and Pázmány Péter Catholic University (PPCU), where I spent my Ph.D. years, is gratefully acknowledged. Thanks to Professor Tamás Roska, who provided me the opportunity to work and study here.
Thanks to my colleagues who have given – besides my supervisor – direct contribution to my scientific results: Josiane Zerubia, Xavier Descombes (both from INRIA Ariana), Zoltan Kato (University of Szeged). By the invitation of Prof. Zerubia, I could make three instructive visits to the INRIA Ariana research group (Sophia-Antipolis, France). As well, I enjoyed my time at the Ramon Llull University (Barcelona) when Xavier Vilasís-Cardona invited me to his group to give a seminar.
I say particular thanks to Tibor Vámos (MTA-SZTAKI), who employed me at SZTAKI during my M.Sc. studies and helped me a lot in the beginning of my scientific career.
I thank the reviewers of my thesis, for their work and valuable comments.
I thank my closest colleagues from the SZTAKI Distributed Events Analysis Research Group for their advices and with whom I could discuss all my ideas: Zoltán Szlávik, László Havasi, István Petrás and Levente Kovács.
Help related to my classes given at PPCU is acknowledged to Zsuzsa Vágó, Zsófia Ruttkay and Tamás Szirányi.

Thanks to Márton Péri for improving my English skills, and correcting linguistic mistakes in my manuscripts.
Thanks to my class mates in the doctoral school for professional and non professional helps: Barnabás Hegyi, Tamás Harczos, Éva Bankó, Gergely Soós, Gergely Gyimesi, Zsolt Szálka, Tamás Zeffer, Mária Magdolna Ercsey-Ravasz, Péter Horváth, Dániel Szolgay and Giovanni Pazienza. I am grateful to Kristóf Iván for providing technical help during the preparation of this document. Thanks to all the colleagues at PPCU, MTA-SZTAKI and INRIA.
For further financial supports, thanks to the Hungarian Scientific Research Fund (OTKA #49001), EU project MUSCLE (FP6-567752), Hungarian R&D Projects ALFA and GVOP (3.1.1.-2004-05-0388/3.0), and the MUSCLE Shape Modelling E-Team.
I am very grateful to my lovely Lívi, to my whole family and to all of my friends who always believed in me and supported me in all possible ways.

Abstract

In this thesis novel probabilistic models are proposed for three different change detection tasks of computer vision, primarily focusing on applications from video surveillance and aerial exploitation. The surveys are performed in a coherent Markov Random Field (MRF) segmentation framework, but the introduced models face different practical challenges such as shadow effects, image registration errors or presence of noisy and incomplete concept descriptors. Contributions are presented in efficient feature extraction, probabilistic modeling of natural processes and feature integration via local innovations in the model structures. We show by several experiments that the proposed novelties embedded into a strict mathematical toolkit can significantly improve the results in *real world* test images and videos.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Change detection is an important early vision task in several computer vision applications. Shape, size, number and position parameters of the relevant scene objects can be derived from an accurate change map and used among others in video surveillance [18][19], aerial exploitation [19], traffic monitoring [20], urban traffic control [21], forest fire detection [22], detection of changes in vegetations [23], urban change detection [24] or disaster protection [25].
As the large variety of applications shows, change detection is a wide concept: different classes of algorithms should be separated depending on the environmental conditions and the exact goals of the systems. This thesis attacks three selected tasks from the problem family. Although the abstract aim (indicating some kind of changes between consecutive images in an image sequence) and the applied mathematical tools (statistical modeling, feature differencing, Markov Random Fields) are similar for the introduced three problems, the further inspections will show that the solutions must be significantly different. We begin with a short introduction of the three tasks. (See also Fig 1.1.)

- **Task 1**: Separation of foreground, background and moving shadows in surveillance videos captured by static cameras. In this environment, video streams are available recorded from a fixed camera position, which enables building statistical background and shadow models based on temporal measurements. The goal is to extract the accurate shapes of the objects or object groups for further post processing.

In surveillance scenes, efficient shadow description and foreground modeling raises serious challenges, due to the presence of camera noise, various reflecting surfaces, low frame rate or background colored object parts. This thesis introduces a model, which considers such practical conditions, meanwhile it also exploits the advantages of robust Bayesian image segmentation techniques.

- **Task 2**: Moving object detection in airborne images captured by moving cameras. In this case, image pairs are only provided instead of videos. The task needs an efficient combination of image registration for camera motion compensation and frame differencing. However, using techniques from 3D geometry, perfect image registration cannot be generally performed. The proposed approach estimates the moving object regions through a statistical model optimization process.

- **Task 3**: Detecting built-in changes in registered airborne images captured with significant time difference. This task needs a more sophisticated approach than simple pixel value differencing, since due to seasonal changes or altered illumination, the appearance of the corresponding *unchanged* territories may be also significantly different. A new region based change detection model will be presented, which is robust against noise and incomplete description of the 'changed'/'unchanged' concepts.

Formally, the inputs of the above change detection tasks are digital images of the same size, and the aim is to generate a segmented image, where each pixel is assigned to a class (or cluster). In *Task 1*, we distinguish three classes: foreground, background and shadow. On the other hand, *Task 2* is a binary segmentation problem with classes: moving object and background, while *Task 3* uses also two clusters: built-in and natural areas.

In a practical point of view, the goal of the methods in this thesis is presenting general pre-processing steps for different families of high-level applications. Thus, the proposed models do not contain complex object shape features [26] or object descriptors [27] which can be highly specific for a given scene. Low level local features are extracted around each pixel, which are derived from the color values

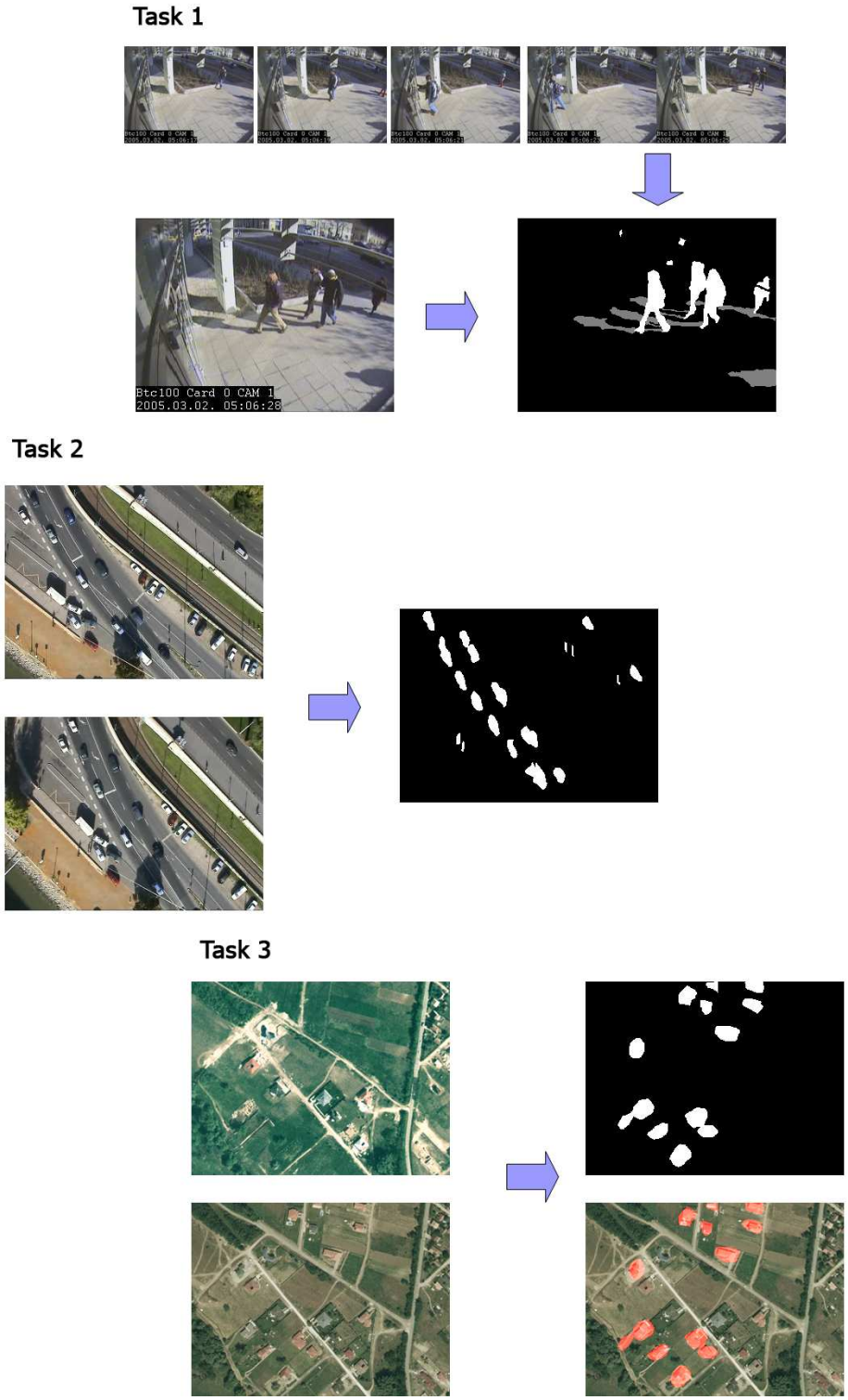


Figure 1.1: Demonstration of the expected results regarding the three tasks. In the change maps white pixels mark the foreground, while black ones the background regions. In task 1, we also have to indicate the moving shadows (with grey).

observed at the pixel or over its neighborhood. The segmentation is primarily based on these local measurements, which provide a posteriori (observation dependent) information for the process. To decrease the inaccuracies, a priori constraints are used as well: we prescribe that the pixels corresponding to the same class should form smooth connected regions in the cluster-map. Note that in most cases, the a priori information is also crucial, since the feature domains of the different clusters may be strongly overlapped, thus several pixels could be misclassified using only the per pixel descriptors.

For similar segmentation problems different solution schemas are proposed in the literature. Here, using the terminology of [28], we distinguish *deterministic* methods (e.g. [29]), which use on/off decision processes at each pixel, and *statistical* approaches (see [30]) which contain probability density functions to describe the class-memberships of a given image point. Note that per pixel decisions often can be interpreted by probabilistic functions as well, but a more important difference is observable in the sequence of the subtasks. *Deterministic* procedures consist of two consecutive levels: first, the algorithm compares the current pixel values to the class models, and classifies the individual nodes independently. After processing the whole image, morphology [31, pp. 449–490][1] can be used to ensure the a priori local connectivity constraint inside the different regions. For example, one can simply choose as the label of a given pixel the most frequent label in its 5×5 neighborhood. As a main drawback here, morphology only considers the current labels in the post processing phase and ignores the information, how 'sure' was the decision of the matching steps at the different pixel positions.

An alternative segmentation schema is a *statistical* Bayesian approach. The segmentation classes are considered to be stochastic processes which generate the observed pixel values according to locally specified distributions. The spatial interaction constraint of the neighboring pixels is also modelled in a probabilistic way by Markov Random Fields (MRF) [32]. Thus, a global probability term is assigned to all possible segmentations of a given input, which encapsulates both the a priori and a posteriori knowledge. Finally, an optimization process attempts to find the global labeling with the highest confidence.

[1]Chapter 15: Morphological Image Processing

On its positive points, Bayesian image segmentation approaches are robust and well established for many problems [33]. MRFs have been also widely used for different change detection tasks e.g. in [34][35][36][37][38][39][40][41]. However, as it will be explained in Chapter 2 in details, the MRF concept offers only a general framework, which has a high degree of freedom. Especially, two key issues should be appropriately chosen regarding a given task. The first one is extracting efficient features and building a proper probabilistic description of each class. The second key point is developing an appropriate model structure, which consists of simple interactive elements. The arrangement and dialogue of these units is responsible for smoothing the segmented image or integrating the effects of different features.

As for the contributions of this thesis, the novelties regarding *task 1* purely lies in how the a posteriori (data dependent) probabilistic terms are constructed. A traditional model structure is meanwhile used [42]. On the other hand, the main contribution regarding *task 2* and *3* is constructing a novel three-layer MRF structure, which integrates different but in themselves simple features.

This thesis uses the basic concepts and results of probability theory (e.g. random variables, probability density functions, Bayes rule etc.), which are supposed to be familiar for the Readers. An extensive introduction to this topic is given e.g. in [43] or in [44]. However, we have collected into Appendix A a few important mathematical definitions and consequences, which focus on some aspects of this work, while theorems referred in the text are presented as well.

The outline of the thesis is as follows. Chapter 2 offers a short introduction to image segmentation approaches via Markov Random Fields. The contributions of this thesis are presented in Chapters 3-6. Each of these chapters is dedicated to a separate problem to solve, which is introduced in the beginning of the section. As for details, in Chapter 3 a novel probabilistic approach is proposed for foreground and shadow detection in video surveillance. Chapter 4 deals with the problem of appropriate color space selection for cast shadow detection in the video frames (both issues correspond to *task 1*). In Chapter 5, focusing on the challenges of *task 2*, we introduce a Bayesian model for object motion detection in airborne image pairs attempting to remove registration and parallax error.

Finally, we propose a model framework in Chapter 6 for structural change detection and show its applicability to recognize newly appeared built-in areas (*task 3*). A short conclusion and a summary of scientific results is given at the end. The thesis contains two appendices as well. As mentioned earlier, Appendix A summarizes a few elementary results of probability theory which may help to understand some parts of the work. Appendix B offers a detailed overview on the used abbreviations and notations.

Chapter 2

Markov Random Fields in Image Segmentation

A digital image is defined over a two dimensional pixel lattice S having a finite size $\mathcal{W} \times \mathcal{H}$. Image segmentation can be formally considered as a labeling task where each pixel gets a label from a J-element label set (corresponding to J different segmentation classes), or, in other words, a J-colored image is generated for a given input.[1] As mentioned in the introduction, statistical methods will be used. Hence, based on the current observations, knowledge about the classes and a priori constraints, the segmentation model must assign a fitness (or probability) value to all the $J^{\mathcal{W}\cdot\mathcal{H}}$ possible segmentations, by the way that higher fitness values correspond to semantically more correct solutions.

The overcome the course of dimensionality, the fitness functions are usually modularly defined: they can be decomposed into individual subterms, and the domain of each subterm consists only of a few pixels. In this way, if we change a label of a single pixel, we should not re-calculate the whole fitness function, only those subterms, which are affected by the selected pixel. This property significantly decreases the computational complexity of iterative labeling optimization techniques [32][50].

An efficient segmentation approach can be based on a graph representation of the images, where each node of the graph corresponds to a pixel. We define edges between two nodes, if the corresponding pixel labels influence each other

[1] In our tasks we will use $J = 3$ or $J = 2$ classes.

directly, i.e. there is a subterm of the fitness function which depends on both pixels. For example, to ensure the spatial smoothness of the segmented images, one can prescribe that the neighboring pixels should have the same labels with high confidence [32][42].

Another important issue is creating interaction between different segmentation (sub)tasks. Multi-layer approaches have been proposed for such problems [45][46] [47][48], where each segmentation forms a 2D layer, which is considered as subgraph of the 3D multi-layer model. Besides the intra-layer connections (edges), which may have the same role as in the single-layer case, one can define inter-layer edges expressing direct links between nodes of different segmentations.

In this thesis, we will use both a conventional single-layer model (Chapters 3 and 4), and a novel multi-layer approach. Moreover, the proposed three-layer structure will be applied in two essentially different ways. In Chapter 5, we will perform fusion of interactive segmentations corresponding to the same input from different points of views. On the other hand, in Chapter 6 links will be created between segmentations of different images based on the same features.

Since the seminal work of Geman and Geman [32], Markov Random Fields (MRFs) offer powerful tools to ensure contextual classification. In the following part of this chapter we give the formal definitions and algorithmic steps regarding MRF based segmentation. To jointly handle the single- and multi-layer models, we will define MRF-s on graphs, following the terminology of [44]. A special case will be given at the end of this chapter.

2.1 Markov Random Fields and Gibbs Potentials

We begin with the formal definitions and notations used in MRF based image segmentation.

Let $\mathcal{G} = (Q, \varepsilon)$ be a graph where $Q = \{q_i | i = 1, \dots N\}$ is a set of nodes, and ε is the set of edges. Edges define the neighboring node pairs:

Definition 1 ***(Neighbors)*** *Two nodes q_i and q_k are neighbors, if there is an edge $e_{ik} \in \varepsilon$ connecting them. The set of points which are neighbors of a node q (i.e. the neighborhood of q) is denoted by $\mathcal{V}_q$.*

Considering all the neighbors in the graph we can talk about a neighborhood system.

Definition 2 ***(Neighborhood system)*** *$\mathcal{V} = \{\mathcal{V}_q | q \in Q\}$ is a neighborhood system for $\mathcal{G}$ if*

- $q \notin \mathcal{V}_q$,
- $q \in \mathcal{V}_r \Leftrightarrow r \in \mathcal{V}_q$.

The image segmentation problem is mapped to the graph as a labeling task over the nodes.

Definition 3 ***(Labeling)*** *To each node (q) of the graph, we assign a label ($\omega(q)$), from the finite label set $\Phi = \{\phi_1, \phi_2, \ldots, \phi_J\}$. Hence,*

$$\forall q \in Q: \quad \omega(q) \in \Phi. \tag{2.1}$$

The global labeling of the graph, $\underline{\omega}$, means the enumeration of the nodes with their corresponding labels:

$$\underline{\omega} = \{\ [q, \omega(q)] \mid \forall q \in Q\ \}. \tag{2.2}$$

Ω denotes the (finite) set of all the possible global labelings ($\underline{\omega} \in \Omega$)[1].

In some cases, instead of a global labeling, we need to deal with the labeling of a given subgraph:

Definition 4 ***(Subconfiguration)*** *The subconfiguration of a given global labeling $\underline{\omega}$ with respect a subset $Y \subseteq Q$ is:*

$$\underline{\omega}_Y = \{\ [q, \omega(q)] \mid \forall q \in Y\ \}. \tag{2.3}$$

Therefore, $\underline{\omega}_Y \subseteq \underline{\omega}$.

In the next step, we define Markov Random Fields (MRFs). As usual, Markov property means here that the labeling of a given node depends directly only on its neighbors.

[1]Since each node may have any of the J labels, the cardinality of Ω, $\#\Omega$ is $J^{\#Q}$.

Definition 5 ***(Markov Random Field)*** *$\mathcal{X}$ is a Markov Random Field (MRF), with respect to $\mathcal{V}$, if*

- *for all $\underline{\omega} \in \Omega$; $P(\mathcal{X} = \underline{\omega}) > 0$*
- *for every $q \in Q$ and $\underline{\omega} \in \Omega$:*

$$P(\omega(q) \mid \underline{\omega}_{Q\setminus\{q\}}) = P(\omega(q) \mid \underline{\omega}_{\mathcal{V}_q}). \tag{2.4}$$

Discussion about MRFs is most convenient by defining the neighborhood system $\mathcal{V}$ via the *cliques* of the graph.

Definition 6 ***(Clique)*** *A subset $C \subseteq Q$ is a clique if every pair of distinct nodes in C are neighbors. $\mathcal{C}$ denotes a set of cliques.*

Definition of $\mathcal{V}$ is equivalent to the enumeration of the cliques.
To characterize the goodness of the different global labelings, a so called Gibbs measure is defined on Ω. Let V be a potential function which assigns a real number $V_Y(\underline{\omega})$ to the subconfiguration $\underline{\omega}_Y$. V defines an energy $U(\underline{\omega})$ on Ω by

$$U(\underline{\omega}) = \sum_{Y \in 2^Q} V_Y(\underline{\omega}). \tag{2.5}$$

where 2^Q denotes the set of the subsets of Q.

Definition 7 ***(Gibbs distribution)*** *A Gibbs distribution is a probability measure π on Ω with the following representation:*

$$\pi(\underline{\omega}) = \frac{1}{Z} \exp\left(-U(\underline{\omega})\right) \tag{2.6}$$

where Z is a normalizing constant or partition function:

$$Z = \sum_{\underline{\omega} \in \Omega} \exp\left(-U(\underline{\omega})\right). \tag{2.7}$$

If $V_Y(\underline{\omega}) = 0$ whenever $Y \notin \mathcal{C}$, then V is called a nearest neighbor potential.
The following theorem is the principle of most MRF applications in computer vision [32]:

Theorem 1 ***(Hammersley-Clifford)*** *$\mathcal{X}$ is a MRF with respect to the neighborhood system $\mathcal{V}$ if and only if $\pi(\underline{\omega}) = P(\mathcal{X} = \underline{\omega})$ is a Gibbs distribution with nearest neighbor Gibbs potential V, that is*

$$\pi(\underline{\omega}) = \frac{1}{Z} \exp\left(- \sum_{C \in \mathfrak{C}} V_C(\underline{\omega}) \right) \tag{2.8}$$

2.2 Observation and A Posteriori Distribution

We mean by *observation* arbitrary measurements from real world processes (such as image sources) assigned to the nodes of the graph. In image processing, usually the pixels' color values or simple textural responses are used. However, later on we will also introduce different features. In all the considered cases, these features are locally obtained at the different pixels or over their neighborhoods. Formally, we only prescribe here that the observation process assigns a real valued vector to some (not necessarily to all) nodes of $\mathcal{G}$.

Definition 8 *Let be given a graph $\mathcal{G} = (Q, \varepsilon)$; a labeling process with domain Ω; and a subset of nodes $O \subseteq Q$. The observation process is defined in the following way:*

$$\mathcal{O} = \{ \ [q, o(q)] \mid \forall q \in O \ \}, \tag{2.9}$$

where

$$\forall q \in O : \ o(q) \in \mathbb{R}^D. \tag{2.10}$$

Two assumptions will be used:

1. There are J random processes corresponding the forthcoming labels ϕ_1, ϕ_2, ..., ϕ_J, which generate for each node $q \in O$ the observation $o(q)$ according to locally specified distributions.
   Consequently, regarding each $q \in O$ and $i = 1, \ldots, J$, we can define a probability density function *(pdf)* $p_{q,i}(x)$ by

   $$p_{q,i}(x) = P(o(q) = x | \omega(q) = \phi_i), \tag{2.11}$$

   which determines the probability (*pdf* value) that the ϕ_i random process generates the observed value $o(q)$ at node q.

2. The local observations are conditionally independent, given the global labeling:

$$P(\mathcal{O}|\underline{\omega}) = \prod_{q \in O} P\left(o(q)|\omega(q)\right). \tag{2.12}$$

2.3 Bayesian Labeling Model

Let $\mathcal{X}$ be a MRF on graph $\mathcal{G} = (Q, \varepsilon)$, with (a priori) clique potentials $\{V_C(\underline{\omega}) \mid C \in \mathcal{C}\}$. Consider an observation process $\mathcal{O}$ on $\mathcal{G}$. The goal is to find the labeling $\widehat{\underline{\omega}}$, which is the maximum a posteriori (MAP) estimate (see also Appendix A), i.e. the labeling with the highest probability given $\mathcal{O}$:

$$\widehat{\underline{\omega}} = \arg\max_{\underline{\omega} \in \Omega} P(\underline{\omega}|\mathcal{O}). \tag{2.13}$$

Following Bayes' rule and eq. 2.12,

$$P(\underline{\omega}|\mathcal{O}) = \frac{P(\mathcal{O}|\underline{\omega})P(\underline{\omega})}{P(\mathcal{O})} = \frac{1}{P(\mathcal{O})}\left[\prod_{q \in O} P\left(o(q)|\omega(q)\right)\right] P(\underline{\omega}) \tag{2.14}$$

Based on the Hammersley-Clifford theorem, $P(\underline{\omega})$ follows a Gibbs distribution:

$$P(\underline{\omega}) = \pi(\underline{\omega}) = \frac{1}{Z}\exp\left(-\sum_{C \in \mathcal{C}} V_C(\underline{\omega})\right) \tag{2.15}$$

while $P(\mathcal{O})$ and Z (in the Gibbs distribution) are independent of the current value of $\underline{\omega}$. Using also the monotonicity of the logarithm function and equations 2.13, 2.14, 2.15, the optimal global labeling can be written into the following form:

$$\widehat{\underline{\omega}} = \arg\min_{\underline{\omega} \in \Omega}\left\{\sum_{q \in O} -\log P\left(o(q)|\omega(q)\right) + \sum_{C \in \mathcal{C}} V_C(\underline{\omega})\right\}. \tag{2.16}$$

Note that some approaches in the literature use the concept of 'singleton clique', i.e. a clique, which consists of a single node [45]. Following this terminology, the joint *pdf* $P(\mathcal{O}, \underline{\omega})$ also derives from a MRF (see eq. 2.16). For the sake of convenience, we also consider later on the $-\log P\left(o(q)|\omega(q)\right)$ term as the *singleton potential* of clique $\{q\}$.

2.4 MRF Optimization

In applications using the MRF models, the quality of the segmentation depends both on the appropriate probabilistic model of the classes, and on the optimization technique which finds a good global labeling with respect to eq. (2.16). The latter factor is a key issue, since finding the global optimum is NP hard [49]. On the other hand, stochastic optimizers using simulated annealing (SA) [32][50] and graph cut techniques [49][51] have proved to be practically efficient offering a ground to validate different energy models.

The results shown in the following chapters have been partially generated by a SA algorithm which uses the Metropolis criteria [52] for accepting new states[1], while the cooling strategy changes the temperature after a fixed number of iterations. The relaxation parameters are set by trial and error taking aim at the maximal quality, and comparing the proposed model to reference MRF methods is done using the same parameter setting.

After verifying our models by the above stochastic optimizer, we have also tested some quicker techniques for practical purposes. We have found the deterministic Modified Metropolis (MMD) [53] relaxation algorithm similarly efficient but significantly faster for these tasks. We note that a coarse but quick MRF optimization method is the ICM algorithm [54], which usually converges after a few iterations, but the segmentation results are significantly worse. As for details, an algorithmic overview and an extensive experimental comparison of the optimization techniques can be found in [44]. For proof of convergence and some practical recommendations concerning the temperature schedule, see [32].

2.5 Image Segmentation with a Single Observation Vector

A simple 'single-layer' application of the Bayesian labeling framework introduced in Section 2.3 is the Potts model [42].

Let S be a 2-dimensional pixel lattice, while s denotes a single pixel of S. Assume that the problem is defined above S and we have a single measurement (a $\mathbb{R}^D$

[1] A state is a candidate for the optimal segmentation.

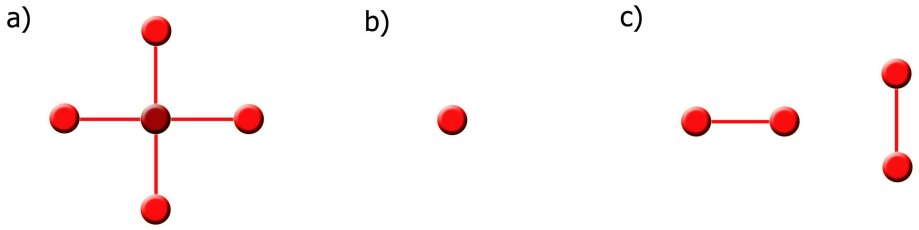


Figure 2.1: a) Illustration of the first ordered neighborhood of a selected node on the lattice, b) 'singleton' clique, c) doubleton cliques

vector) at each pixel s. The goal is to segment the input lattice with J pixel clusters corresponding to J random processes $(\phi_1, \ldots, \phi_J)$, where the segmentation fulfills the following requirements:

1. The clusters of the pixels are consistent with the local measurements.

2. The segmentation is smooth: pixels having the same cluster form connected regions.

2.5.1 Mapping the Potts Model to the Bayesian Labeling Problem

Several tasks can be mapped to a Bayesian labeling problem via the Potts model e.g. [37][38][40]. Here dealing still with an abstract task definition, we shortly introduce the modeling steps. Based on the previous notes, we must define $\mathcal{G} = (Q, \varepsilon)$, Ω, $\mathcal{O}$, $\pi(\underline{\omega})$ and $p_{q,i}(x) = P(o(q) = x|\omega(q) = \phi_i)$ for all $q \in Q$, and $i = 1, \ldots, J$.

1. **Definition of $\mathcal{G}$:** We assign to each pixel of the input lattice a unique node of the graph. First ordered neighborhood is used, i.e. each pixel has four neighbors. Therefore, the cliques of the graph are singletons or doubletons (see Fig. 2.1).

2. **Definition of Ω:** we use an application specific label-set $\Phi = \{\phi_1, \phi_2, \ldots, \phi_J\}$, which determines the set of the global labelings.

3. **Definition of the observation process**: In this model, observation vector is assigned to all nodes, hence $O = Q$. The exact $o(q)$ features ($\forall q \in Q$) should be fixed depending on the current task.

4. **Definition of the a priori distributions** $\pi(\underline{\omega}) = P(\underline{\omega})$ is defined by the doubleton clique potential functions. The a priori probability term is responsible for getting smooth connected components in the segmented images. Thus, we give penalty terms to each neighboring pair of nodes whose labels are different. For any $r, q \in Q$ node pairs, which fulfill $q \in \mathcal{V}_r$, $\{r, q\} \in \mathcal{C}$ is a clique of the graph, with potential:

$$V_{\{r,q\}}(\underline{\omega}) = \begin{cases} -\delta & \text{if } \omega(r) = \omega(q) \\ +\delta & \text{if } \omega(r) \neq \omega(q) \end{cases} \tag{2.17}$$

   Where $\delta \geq 0$ is a constant.

5. **Definition of the a posteriori distributions** Defining $p_{q,i}(x)$ for all $q \in Q$ and $i = 1 \ldots, J$ is a highly application specific task. Thereafter, singleton clique potentials are calculated by

$$V_{\{q\}} = -\log p_{q,\omega(q)}\big(o(q)\big). \tag{2.18}$$

   Note that in the above model, the a priori constraints are only responsible for smoothing the segmented image: the position, size and shape of the different clusters is mainly determined by the (a posteriori) probabilistic class models.

With the previous definitions, the Bayesian labeling problem is completely defined, and the optimal labeling can be determined by finding the optimum of eq. (2.16).

2.5.2 Demonstrating Example of the Potts Model Based Segmentation

For the sake of a quick demonstration, we introduce a simple segmentation problem in this section, and we give a solution using the above Potts-MRF based approach.

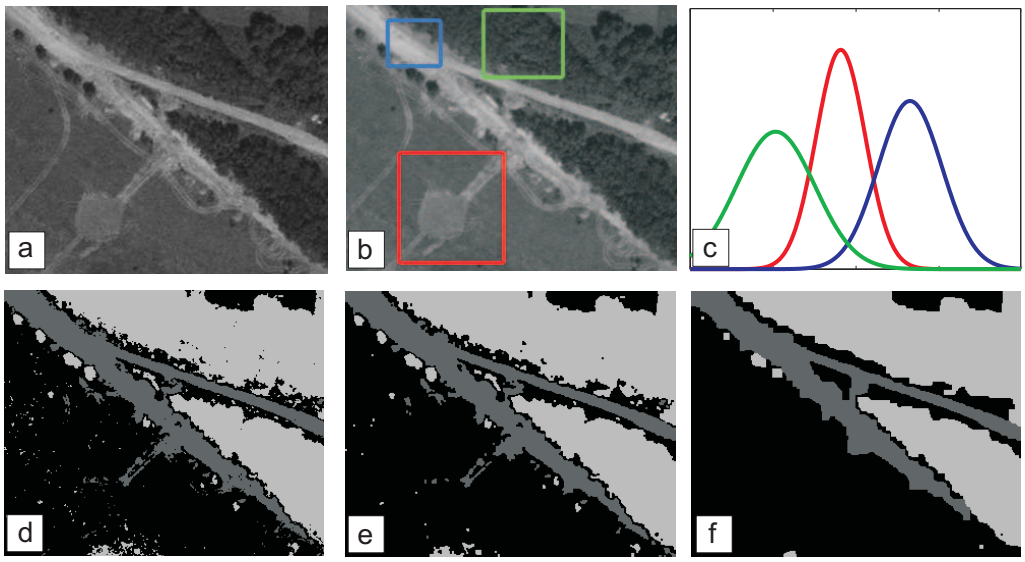


Figure 2.2: MRF segmentation example. Above: a) input image b) training regions c) Gaussian densities for the training regions. Below: segmentation results d) without neighborhood smoothing term ($\delta = 0$), e) ICM relaxation f) MMD optimization

Consider a grayscale aerial image shown in Fig. 2.2a. The goal is to segment this image using three classes: roads, plough-lands and forests. Assume that the user is allowed to assign a rectangular training region for each class by hand (Fig. 2.2b).

Following the model of Section 2.5, the observation and the a posteriori distributions should be defined depending on the current task, meanwhile the remaining model elements are fixed. Since significantly different pixel intensities correspond to the three regions in this case (e.g. the forests are dark), the observation will be the gray value of the pixels ($o(s)$ is the gray level of s). For each class, the a posteriori intensity distribution is modelled by a Gaussian density $p_{s,i}(o(s)) = \eta(o(s), \mu_i, \sigma_i)$, $i \in \{1, 2, 3\}$. The three Gaussian density functions are shown in Fig. 2.2c. The distribution parameters are estimated over the training regions of the classes (corresponding training regions in Fig. 2.2b and Gaussians in Fig. 2.2c have the same color).

In the next step, we estimate the optimal labeling (eq. 2.16) with different relax-

ation techniques. The MRF-segmentation results are shown below in Fig. 2.2[1]. The first image (Fig. 2.2d) is the output of the pixel by pixel maximum likelihood classification, or in other words, it is the output of a MRF, where the smoothing term eq. 2.17 is ignored by setting $\delta = 0$. This solution is notably noisy. In the other cases, we used $\delta = 2$, and applied the ICM (Fig. 2.2e), and the MMD (Fig. 2.2f) optimization strategies, respectively. One can observe, that using MMD results in smoother and more noiseless segmented regions.

We will use the Potts model in the first part of this thesis (in Chapters 3 and 4). However, the feature values and the probability distributions must be different from the above simple approach. We will also need to consider that usually the parameters cannot be set in a supervised manner, and in videos, they should be estimated both using temporal and spatial feature statistics. On the other hand, the single layer Potts structure will not be appropriate for *tasks 2* and *3*, therefore, a model extension will be given in Chapter 5.

[1]For the comparison, implementation of Csaba Gradwohl and Zoltan Kato was used [44].

Chapter 3

Bayesian Foreground Detection in Uncertain Frame Rate Surveillance Videos

---

In this section a new model will be proposed for foreground and shadow detection in surveillance videos captured by static cameras. The model works without detailed a priori object-shape information, and is also appropriate for low and unstable frame rate video sources.
Contribution is presented in three key issues:

- A novel adaptive shadow model is introduced, and improvements are shown versus previous approaches in scenes with difficult lighting and coloring effects.
- We give a novel description for the foreground based on spatial statistics of the neighboring pixel values, which enhances the detection of background or shadow-colored object parts.
- We show how microstructure analysis can be used in the proposed framework as additional feature components improving the results.

We validate our method on outdoor and indoor sequences including real surveillance videos and well-known benchmark test sets.

3.1 Introduction

Background subtraction is a key issue in automated video surveillance. Foreground areas usually contain the regions of interest, moreover, an accurate object-silhouette mask can directly provide useful information for several applications, for example people [55][56][57] or vehicle detection [34], tracking [58][59], biometrical identification through gait recognition [60][61] or activity analysis [62].
Although background removal is a well examined problem (see e.g. [38][40][39] [41][59][62][63][64][65]) it still raises challenging problems. Two of them is addressed in this chapter: *shadow detection* and *foreground modeling*. To enhance the results, a novel *microstructure* model is used as well.

3.1.1 Shadow Detection: an Overview

The presence of moving cast shadows on the background makes it difficult to estimate shape [66] or behavior [56] of moving objects, because they can be erroneously classified as part of the foreground mask. Since under some illumination conditions $40-50\%$ of the non-background points may belong to shadows, methods without shadow filtering [38][41][62] can be less efficient in scene analysis. Hence, we deal here with an image segmentation problem with three classes: *foreground* objects, *background* and *shadows* of the foreground objects being cast on the background. Note that we should not detect self shadows (i.e. shadows appearing on the foreground objects), which are part of the foreground, and static shadows (cast shadows of the static objects), because they correspond to the background.
In the literature, different approaches are available regarding shadow detection. Apart from a few geometry based techniques suited to specific conditions [67], [68], shadow detection is usually done by color filtering. Still image based methods [69][70] attempt to find and remove shadows in the single frames independently. However, these models have been evaluated only on high quality images where the background has a uniform color or texture pattern, while in video surveillance, we must expect images with poor quality and resolution. The authors in [70] note that their algorithm is robust when the shadow edges are clear, but artifacts

may appear in cases of images with complex shadows or diffuse shadows with poorly defined edges. For practical use, the computational complexity of these algorithms should be decreased [69].

Some other methods focus on the discrimination of the shadow edges, and edges due to objects boundaries [71][72]. However, it may be difficult to extract connected foreground regions from the resulting edge map, which is often ragged [71]. Complex scenarios containing several small objects or shadow-parts may be also disadvantageous for these methods.

For the above reasons, we focus on video (instead of still image) and region (instead of edge) based shadow modeling techniques in the following. Here, an important point of view regarding the categorization of the algorithms [28] is the discrimination of the *non parametric* and *parametric* cases. Non parametric, or 'shadow invariant' methods convert the pixel values into an illuminant invariant feature space: they remove shadows instead of detecting them. This task is often performed by a color space transformation. The normalized rgb [35][73] and $C_1C_2C_3$ spaces [74][1] are supposed to fulfill color constancy through using only chrominance color components. [75] exploits hue constancy under illumination changes to train a weak classifier as a key step of a more sophisticated shadow detector. We find an overview of the illumination invariant approaches in [74] indicating that several assumptions are needed regarding the reflecting surfaces and the lightings. Also [72] emphasizes the limits of these methods: outdoors, shadows will have a blue color cast (due to the sky), while lit regions have a yellow cast (sunlight), hence the chrominance color values corresponding to the same surface point may be significantly different in shadow and in sunlight. We have also found in our experiments that the shadow invariant methods fail outdoors several times, and they are rather usable indoors (Fig. 3.9). Moreover, since they ignore the luminance components of the color, these models become sensitive to noise.

Consequently, we develop a *parametric* model: first, we estimate the mean background values of the individual pixels trough a statistical background model [62], then we extract feature vectors from the actual and the estimated background

[1]We refer later to the normalized rgb as *rg* space, since the third color component is determined by the first and second.

values of the pixels and model the feature domain of shadows in a probabilistic way. Parametric shadow models may be *local* or *global*.

In a *local* shadow model [76] independent shadow processes are proposed for each pixel. The local shadow parameters are trained using a second mixture model similarly to the background in [62]. This way, the differences in the light absorption-reflection properties of the scene points can be notably considered. However, a single pixel should be shadowed several times till its estimated parameters converge, whilst the illumination conditions should stay unchanged. This hypothesis is often not satisfied in outdoor surveillance environments, therefore, this local process based approach is less effective in our case.

We follow the other approach: shadow is characterized with *global* parameters in an image (or in each subregion, in case of videos having separated scene areas with different lightings), and the model describes how the background values of the different pixels change, when shadow is projected on them. We consider the transformation between the shadowed and background values of the pixels as a random transformation, hence, we take several illumination artifacts into consideration. On the other hand, we derive the shadow parameters from global image statistics, therefore, the model performance is reasonable also on the pixel positions where motion is rare.

3.1.2 Modeling the Foreground

Another important issue is related to foreground modeling. Some approaches [62][65] consider background subtraction as a one class-classification problem, where foreground image points are purely recognized as non-matching pixels to the background model. Similarly, [30][39] build adaptive models for the background and shadow classes and detect foreground as outlier regions with respect to both models. This way, background and shadow colored object parts cannot be detected. To overcome this problem, foreground must be also modelled in a more sophisticated way.

Before going into the details, we make a remark on an important property of the examined video flows. For several video surveillance applications high-resolution

images are crucial. Due to the high bandwidth requirement, the sequences are often captured at low [77] or unsteady frame rate depending on the transmission conditions. These problems appear, especially, if the system is connected to the video sources through narrow band radio channels or oversaturated networks. For another example, quick off-line evaluation of the surveillance videos is necessary after a criminal incident. Since all the video streams corresponding to a given zone should be continuously recorded, these videos may have a frame rate lower than 1 fps to save up storage resources.

For these reasons, a large variety of temporal information, like pixel state transition probabilities [34][37][40], periodicity calculus [55][56], temporal foreground description [38], or tracking [58][78], are often hard to derive, since they usually need permanently high frame rate. Thus, we focus on using frame rate independent features to ensure graceful degradation if the frame rate is low or unbalanced. On the other hand, our model also exploits temporal information for background and shadow modeling.

For the above reasons, our model uses spatial color information instead of temporal statistics to describe the foreground. It assumes that foreground objects consist of spatially connected parts and these parts can be characterized by typical color distributions. Since these distributions can be multi-modal, the object-parts should not be homogenous in color or texture, while we exploit the spatial information without segmenting the foreground components.

Note that spatial object description has been already used both for interactive [79] and unsupervised image segmentation [45]. However, in the latter case, only large objects with typical color or texture are detected, since the model [45] penalizes the small segmentation classes. The authors in [38] have characterized the foreground by assuming temporal persistence of the color and smooth changes in the place of the objects. Nevertheless, in case of low frame rate, fast motion and overlaying objects, appropriate temporal information is often not available.

3.1.3 Further Issues

Besides the color values, we exploit microstructure information to enhance the accuracy of the segmentation. In some previous works [80][81] texture was used

as the only feature for background subtraction. That choice can be justified in case of strongly dynamic background (like a surging lake), but it gives lower performance than pixel value comparison in a stable environment. We find a solution for integrating intensity and texture differences for frame differencing in [82]. However, that is a slightly different task from foreground detection, since we should compare the image regions to background/shadow models. In aspect of the background class, our color-texture fusion process is similar to the joint segmentation approach of [40], which integrates gray level and local gradient features. We extend it by using different and adaptively chosen microstructural kernels, which suit the local scene properties better. Moreover, we show how this probabilistic approach can be used to improve our shadow model.

Color space choice is a key issue in several corresponding methods, as it will be intensively studied in Chapter 3. We have chosen the CIE L*u*v* space, for purposes which will be detailed there. Here, we only mark two well known properties of the CIE L*u*v* space: we can measure the perceptual distance between colors with the Euclidean distance [83], and the color components are approximately uncorrelated with respect to camera noise and changes in illumination [84]. Since we derive the model parameters in a statistical way, there is no need for accurate color calibration and we use the common CIE D65 standard. It is also not critical to consider exactly the physical meaning of the color components which is usually environment-dependent [74][85]; we use only an approximate interpretation of the L, u, v components and show the validity of the model via experiments.

For validation we use real surveillance video shots and also test sequences from a well-known benchmark set [28]. Table 3.1 summarizes the different goals and tools regarding some of the above mentioned state-of-the-art methods and the proposed model. For detailed comparison see also Section 3.7.

In summary, the main *contributions* of this chapter can be divided into three groups. We introduce a *statistical shadow model* which is robust regarding the forthcoming artifacts in real-world surveillance scenes (Section 3.3.2.), and a corresponding automatic parameter-update procedure, which is usually missing in previous similar methods (Section 3.5.2). We introduce a non-object based, spatial description of the *foreground* which enhances the segmentation result also in low frame rate videos (Section 3.4). Meanwhile, we show how *microstructure*

Table 3.1: Comparison of different corresponding methods and the proposed model. Notes:† high frame-rate video stream is needed ‡foreground estimation from the current frame * temporal foreground description, ** pixel state transitions

Method	**Needs high fps†**	**Shadow detection**	**Adaptive shadow**	**Spatial fg info‡**	**Scenes**	**Texture**
Mikic 2000 [30]	No	global, constant ratio	No	No	outdoor	No
Paragious 2001 [35]	No	illumination invariant	No	No	indoor	No
Salvador 2004 [74]	No	illumination invariant	No	No	both	No
Martel-Brisson 2005 [76]	No	local process	Yes	No	indoor	No
Sheikh 2005 [38]	Yes: tfd *	No	-	No	both	No
Wang 2006 [40]	Yes: pst **	global, constant ratio	No	No	indoor	first ordered edges
Proposed method	No	global, probabilistic	Yes	Yes	both	different micro-structures

analysis can improve the segmentation in this framework (Section 3.3.4).
We also use a few assumptions in the chapter. First, the camera stands in place and has no significant ego-motion. Secondly, we expect static background objects (e.g. there is no waving river in the background). The third assumption is related to the illumination: we deal with one emissive light source in the scene, however, we consider the presence of additional diffused and reflected light components.

3.2 Formal Model Description

The segmentation model follows the Bayesian labeling approach introduced in Section 2.3, more specifically the single layer model of Section 2.5. Denote by S

the two dimensional pixel grid and we use henceforward a first ordered neighborhood system on the lattice. As defined earlier, a unique node of the MRF-graph $\mathcal{G}$ is assigned to each pixel. Thus for simplicity, s will denote also a pixel of the image and the corresponding node of $\mathcal{G}$ in this chapter.

The procedure assigns a label $\omega(s)$ to each pixel $s \in S$ form the label-set: $\Phi = \{\text{fg,bg,sh}\}$ corresponding to three possible classes: foreground (fg), background (bg) and shadow (sh). As is typical, the segmentation is equivalent to a global labeling $\underline{\omega} = \{[s, \omega(s)] \mid s \in S\}$, and the probability of a given $\underline{\omega} \in \Omega$ in the label field follows Gibbs distribution.

The image data (observation) at pixel s is characterized by a 4 dimensional feature vector:

$$\overline{o}(s) = [o_L(s), o_u(s), o_v(s), o_\chi(s)]^{\mathsf{T}} \tag{3.1}$$

where the first three elements are the color components of the pixel in CIE L*u*v* space, and $o_\chi(s)$ is a texture term (more specifically a microstructural response) which we introduce in Section 3.3.4 in details. Set $\mathcal{O} = \{\overline{o}(s) |\ s \in S\}$ marks the global image data.

The key point in the model is to define the conditional density functions $p_\phi(s) = P(\overline{o}(s) |\ \omega(s) = \phi)$, for all $\phi \in \Phi$ and $s \in S$. For example, $p_{\text{bg}}(s)$ is the probability that the background process generates the observed feature value $\overline{o}(s)$ at pixel s. Later on $\overline{o}(s)$ in the background will be also featured as a random variable with probability density function $p_{\text{bg}}(s)$.

We define the conditional density functions in Section 3.3-3.5, and the segmentation procedure will be presented in Section 3.7 in details. Before continuing, note that we minimize the minus-logarithm of the global probability term (similarly to eq. 2.16) in fact. Therefore, in the following we use the $\epsilon_\phi(s) = -\log p_\phi(s)$ local energy terms, for easier notation.

3.3 Probabilistic Model of the Background and Shadow Processes

3.3.1 General Model

We model the distribution of feature values in the background and in the shadow by Gaussian density functions, like e.g. [28][37][40].
Considering the low correlation between the color components [84], we approximate the joint distribution of the features by a 4 dimensional Gaussian density function with diagonal covariance matrix:

$$\overline{\overline{\Sigma}}_{\phi}(s) = \mathrm{diag}\{\sigma^2_{\phi,L}(s), \sigma^2_{\phi,u}(s), \sigma^2_{\phi,v}(s), \sigma^2_{\phi,\chi}(s)\} \tag{3.2}$$

for $\phi \in \{\mathrm{bg}, \mathrm{sh}\}$.
Accordingly, the distribution parameters are $\overline{\mu}_{\phi}(s) = [\mu_{\phi,L}(s), \ldots, \mu_{\phi,\chi}(s)]^{\mathsf{T}}$ mean, and $\overline{\sigma}_{\phi}(s) = [\sigma_{\phi,L}(s), \ldots, \sigma_{\phi,\chi}(s)]^{\mathsf{T}}$ standard deviation vectors. With this 'diagonal' model we avoid matrix inversion and determinant recovering during the calculation of the probabilities, and the $\epsilon_{\phi}(s) = -\log p_{\phi}(s)$ terms can be derived directly from the one dimensional marginal probabilities:

$$\epsilon_{\phi}(s) = 2\log 2\pi + \sum_{i=\{L,u,v,\chi\}} \left[\log \sigma_{\phi,i}(s) + \frac{1}{2}\left(\frac{o_i(s) - \mu_{\phi,i}(s)}{\sigma_{\phi,i}(s)} \right)^2 \right] \tag{3.3}$$

According to eq. (3.3), each feature contributes with its own additional term to the energy calculus. Therefore, the model is modular: the one dimensional model parameters, $[\mu_{\phi,i}(s), \sigma^2_{\phi,i}(s)]$, can be estimated separately.

3.3.2 Color Features in the Background Model

The use of a Gaussian distribution to model the observed color of a single background pixel is well established in the literature, with the corresponding parameter estimation procedures such as in [62][86]. In our model, following one of the most popular approaches [62] we train the color components of the background parameters $[\overline{\mu}_{\mathrm{bg}}(s), \overline{\sigma}_{\mathrm{bg}}(s)]$ in a similar manner to the conventional online k-means algorithm. Although this algorithm is not our contribution, it is important to be

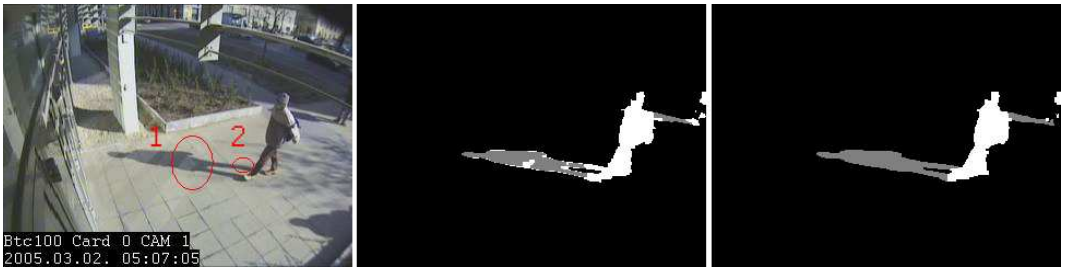


Figure 3.1: Illustration of two illumination artifacts (the frame has been chosen from the 'Entrance pm' test sequence). 1: light band caused by non-Lambertian reflecting surface (glass door) 2: dark shadow part between the legs (more object parts change the reflected light). The constant ratio model (in the middle) causes errors, while the proposed model (right image) is more robust.

understood in terms of the following parts of this section, thus we briefly introduce it.

We consider each pixel s as a separate process, which generates an observed pixel value sequence over time:

$$\{\overline{o}^{[1]}(s), \overline{o}^{[2]}(s), \ldots, \overline{o}^{[t]}(s)\}. \tag{3.4}$$

To model the recent history of the pixels, [62] suggested a mixture of K Gaussians distribution:

$$P(\overline{o}^{[t]}(s)) = \sum_{k=1}^{K} \kappa_k^{[t]}(s) \cdot \eta\left(\overline{o}^{[t]}(s), \overline{\mu}_k^{[t]}(s), \overline{\sigma}_k^{[t]}(s)\right), \tag{3.5}$$

where $\eta(.)$ is a Gaussian density function, with diagonal covariance matrix. We ignore here multi-modal background processes [62], and consider the background Gaussian term to be equivalent to the Gaussian component in the mixture, which has the highest weight. Thus, at time t:

$$\overline{\mu}_{\mathrm{bg}}(s) = \overline{\mu}_{k_{\max}}^{[t]}(s), \qquad \overline{\sigma}_{\mathrm{bg}}(s) = \overline{\sigma}_{k_{\max}}^{[t]}(s) \tag{3.6}$$

where

$$k_{\max} = \arg\max_k \kappa_k^{[t]}(s). \tag{3.7}$$

The parameters of the above distribution are estimated and updated without user interaction. First, we introduce a $\mathcal{D}$ matching operator between a pixel value and

a local Gaussian component as follows:

$$\mathcal{D}_s(\overline{o}(s),k)\begin{cases}1 & \text{if } \left[\overline{o}(s)-\overline{\mu}_k^{[t]}(s)\right]^{\mathsf{T}}\left(\Sigma_k^{[t]}\right)^{-1}\left[\overline{o}(s)-\overline{\mu}_k^{[t]}(s)\right]<\gamma \\ 0 & \text{otherwise}\end{cases} \tag{3.8}$$

where γ is a robust threshold parameter, [62] recommends $\gamma = 2.5$.
The weight parameters of the components are updated as

$$\kappa_k^{[t+1]}(s) = (1-\xi_1)\cdot\kappa_k^{[t]}(s) + \xi_1\cdot\mathcal{D}_s(\overline{o}^{[t]}(s),k) \tag{3.9}$$

which follows a normalization to ensure $\sum_{k=1}^{K}\kappa_k^{[t+1]} = 1$.
Usually, the mean and deviation parameters of the unmatched components do not change. However, if no match is found among all terms, the component with the lowest weight (indexed by $k_{\min}$) is replaced with a Gaussian with the current pixel value as its mean value, an initially high variance, and low prior weight:

$$\overline{\mu}_{k_{\min}}^{[t+1]}(s) = \overline{o}(s), \quad \overline{\sigma}_{k_{\min}}^{[t+1]}(s) = \overline{\sigma}_0, \quad \kappa_{k_{\min}}^{[t+1]}(s) = \kappa_0 \tag{3.10}$$

if $\mathcal{D}_s(\overline{o}(s),k) = 0$ for $k = 1\ldots K$.
Otherwise, if k_m is the matched component $[\mathcal{D}_s(\overline{o}(s),k_m) = 1]$, the following update process should be used:

$$\overline{\mu}_{k_m}^{[t+1]}(s) = (1-\xi_2)\cdot\overline{\mu}_{k_m}^{[t]}(s) + \xi_2\cdot\overline{o}^{[t]}(s) \tag{3.11}$$

$$\overline{\overline{\sigma}}_{k_m}^{[t+1]}(s) = (1-\xi_2)\cdot\overline{\overline{\sigma}}_{k_m}^{[t]}(s) + \xi_2\cdot\left[\overline{o}^{[t]}(s)-\overline{\mu}_{k_m}^{[t]}(s)\right]^{\circ} \tag{3.12}$$

where $\circ$ applied for a vector is the per element squaring operator.
In summary, $[\mu_{\mathrm{bg},L}(s),\mu_{\mathrm{bg},u}(s),\mu_{\mathrm{bg},v}(s)]^{\mathsf{T}}$ vector estimates the mean background color of pixel s measured over the recent frames, while $\overline{\sigma}_{\mathrm{bg}}(s)$ is an adaptive noise parameter.

3.3.3 Color Features in the Shadow Model

As we have stated in the introduction, we characterize shadows by describing the background-shadow color value transformation in the images. The shadow calculus is based on the illumination-reflection model [87], which has been originally

introduced for constant lighting, flat and Lambertian reflecting surfaces. Usually, our scene does not fulfill these requirements. The presented novelty is that we use a probabilistic approach to describe the deviation of the scene from the ideal surface assumptions, and get a more robust shadow detection.

3.3.3.1 Measurement of Color in the Lambertian Model

According to the illumination model [87] the response $g(s)$ of a given image sensor placed at pixel s can be written as

$$g(s) = \int e(\lambda, s)\rho(\lambda, s)\nu(\lambda)d\lambda \tag{3.13}$$

where $e(\lambda, s)$ is the illumination function at a given wavelength λ, $\rho(s)$ depends on the surface albedo and geometric, $\nu(\lambda)$ is the sensor sensitivity. Accordingly, the difference between the shadowed and illuminated background values of a given surface point is caused only by the different local value of $e(\lambda, s)$. For example, outdoors, the illumination function observed in sunlit is the composition of the direct component (sun), the Rayleigh scattering (sky), causing that the ambient light has a blue tingle [88], and residual light components reflected from other non-emissive objects. On the other hand, the effect of the direct component is missing in the shadow.

Although the validity of eq. (3.13) is already limited by several scene assumptions [87], in general, it is still too difficult to exploit appropriate information about the corresponding background-shadow values, since the components of the illumination function are unknown. Therefore, further strong simplifications are used in the applications. According to [70] the camera sensors must be exact Dirac delta functions: $\nu(\lambda) = q_0 \cdot \delta(\lambda - \lambda_0)$ and the illumination must be Planckian [89]. In this case, eq.(3.13) implies the well-known 'constant ratio' rule. Namely, the ratio of the shadowed $g_{\mathrm{sh}}(s)$ and illuminated value $g_{\mathrm{bg}}(s)$ of a given surface point is considered to be constant over the image: $\frac{g_{\mathrm{sh}}(s)}{g_{\mathrm{bg}}(s)} = A$.

The 'constant ratio' rule has been used in several applications [30][37][40]. Here the shadow and background Gaussian terms corresponding to the same pixel are related via a globally constant linear density transform. In this way, the results

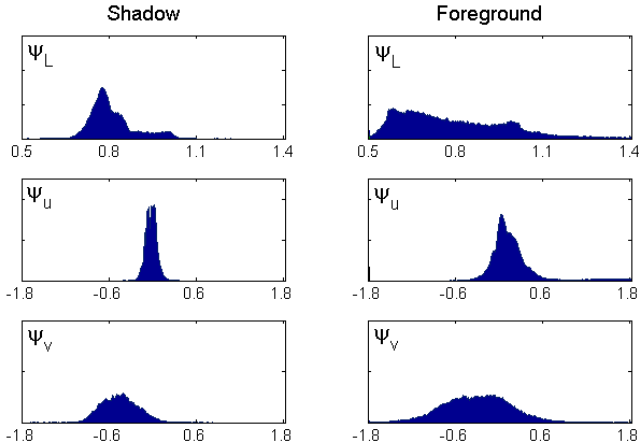


Figure 3.2: Histograms of the ψ_L, ψ_u and ψ_v values for shadowed and foreground points collected over a 100-frame period of the video sequence 'Entrance pm' (frame rate: 1 fps). Each row corresponds to a color component.

may be reasonable when all the direct, diffused and reflected light can be considered constant over the scene. However, the reflected light may vary over the image in case of several static or moving objects, and the reflecting properties of the surfaces may differ significantly from the Lambertian model (See Fig. 3.1). The efficiency of the constant ratio model is also restricted by several practical reasons, like quantification errors of the sensor values, saturation of the sensors, imprecise estimation of $g_{\mathrm{bg}}(s)$ and A, or video compression artifacts. Based on our experiments (Section 3.7), these inaccuracies cause poor detection rates in some outdoor scenes.

3.3.3.2 Proposed Model

The previous section suggests that the ratio of the shadowed and background luminance values of the pixels may be useful, but not powerful enough as a descriptor of the shadow process. Instead of constructing a more difficult illumination model, for example in 3D with two cameras, we overcome the problems with a statistical model. For each pixel s, we introduce the variable $\psi_L(s)$ by:

$$\psi_L(s) = \frac{o_L(s)}{\mu_{\mathrm{bg,L}}(s)} \tag{3.14}$$

where, as defined earlier, $o_L(s)$ is the observed luminance value at s, and $\mu_{\mathrm{bg},L}(s)$ is the mean value of the local Gaussian background term estimated over the previous frames [62].

Thus, if the $\psi_L(s)$ value is close to the estimated shadow darkening factor, s is more likely to be a shadowed point. More precisely, in a given video sequence, we can estimate the distribution of the shadowed ψ_L values globally in the video parts. Based on experiments with manually generated shadow masks, a Gaussian approximation seems to be reasonable regarding the distribution of shadowed ψ_L values (Fig. 3.2 shows the global ψ statistics regarding a 100-frame period of outdoor test sequence 'Entrance pm'). For comparison, we have also plotted the statistics for the foreground points, which follows significantly different, more uniform distribution.

Due to the spectral differences between the direct and ambient illumination, cast shadows may also change the u and v color components [72]. We have found an offset between the shadowed and background u values of the pixels, which can be efficiently modelled by a global Gaussian term in a given scene (similarly as for the v component). Hence, we define $\psi_u(s)$ (and $\psi_v(s)$) by

$$\psi_u(s) = o_u(s) - \mu_{\mathrm{bg,u}}(s) \tag{3.15}$$

As Fig. 3.2 shows, the shadowed $\psi_u(s)$ and $\psi_v(s)$ values follow approximately normal distributions.

Consequently, the shadow color process is characterized by a three dimensional Gaussian random variable:

$$\forall s \in S : \overline{\psi}(s) = [\psi_L(s), \psi_u(s), \psi_v(s)]^{\mathsf{T}} \sim N[\overline{\mu}_\psi, \overline{\sigma}_\psi] \tag{3.16}$$

Using the linear transform theorem (see Theorem 5 of page 135), eq. 3.14 and 3.15, the color values in the shadow at a given pixel position are also generated by Gaussian distribution,

$$[o_L(s), o_u(s), o_v(s)]^{\mathsf{T}} \ \sim \ N[\overline{\mu}_{\mathrm{sh}}(s), \overline{\sigma}_{\mathrm{sh}}(s)] \tag{3.17}$$

with the following parameters:

$$\mu_{\mathrm{sh},L}(s) = \mu_{\psi,L} \cdot \mu_{\mathrm{bg},L}(s) \tag{3.18}$$

$$\sigma^2_{\mathrm{sh},L}(s) = \sigma^2_{\psi,L} \cdot \mu^2_{\mathrm{bg},L}(s) \tag{3.19}$$

Regarding the u (and similarly to the v) component:

$$\mu_{\mathrm{sh},u}(s) = \mu_{\psi,u} + \mu_{\mathrm{bg},u}(s), \quad \sigma^2_{\mathrm{sh},u}(s) = \sigma^2_{\psi,u} \tag{3.20}$$

The estimation and the time dependence of parameters $[\overline{\mu}_\psi, \overline{\sigma}_\psi]$ are discussed in Section 3.5.2.

3.3.4 Microstructural Features

In this section, we define the 4^{th} dimension of the pixels' feature vectors (eq. (3.1)), which contains local microstructural responses.

3.3.4.1 Definition of the Used Microstructural Features

Pixels covered by a foreground object often have different local textural features from the background at the same location, moreover, texture features may identify foreground points with background or shadow like color. In our model, texture features are used together with color components and they enhance the segmentation results as an additional component in the feature vector. Therefore, we make restrictions regarding the texture features: we search for components that we can get by low additional computing time from the existing model elements, in exchange for some accuracy.

According to our model, the textural feature is retrieved from a color feature-channel by using microstructural kernels. For practical reasons, and following the fact that the human visual system mainly percepts textures as changes in intensity, we use texture features only for the 'L' color component. A novelty of the proposed model is (as being explained in Section 3.3.4.3) that we may use different kernels at different pixel locations. More specifically, there is a set of kernel coefficients for each pixel s: $\{a_s(r)|r \in \mathcal{K}_s\}$, where $\mathcal{K}_s$ is the set of pixels around s covered by the kernel. Feature $o_\chi(s)$ is defined by:

$$o_\chi(s) = \sum_{r \in \mathcal{K}_s} a_s(r) \cdot o_L(r) \tag{3.21}$$

3.3.4.2 Analytical Estimation of the Distribution Parameters

Here, we show that with some further reasonable assumptions the features defined by eq. (3.21) have also Gaussian distribution, and the distribution parameters $[\mu_{\phi,\chi}(s), \sigma_{\phi,\chi}(s)]$, $\phi \in \{\mathrm{bg}, \mathrm{sh}\}$ can be determined analytically.
As a simplification we use the fact that the neighboring pixels have usually the same labels, and calculate the probabilities by:

$$p_\phi(s) = P(\overline{o}(s)|\omega(s) = \phi) \approx P(\overline{o}(s)|\omega(r) = \phi, r \in \mathcal{K}_s) \tag{3.22}$$

This assumption is inaccurate near the border of the objects, but it is a reasonable approximation if the kernel size (and the size of set $\mathcal{K}_s$) is small enough. To ensure this condition, we use 3×3 kernels in the following.
Accordingly, with respect to eq. (3.21), $o_\chi(s)$ in the background (and similarly in the shadow) can be considered as a linear combination of Gaussian random variables from the following set Λ_s:

$$\Lambda_s = \{o_L(r)|\ r \in \mathcal{K}_s\} \tag{3.23}$$

where $o_L(r) \sim N[\mu_{\mathrm{bg},L}(r), \sigma_{\mathrm{bg},L}(r)]$. We assume that the $o_L(r)$ variables have joint normal distribution, therefore, $o_\chi(s)$ is also Gaussian with the mean and standard deviation parameters $[\mu_{\mathrm{bg},\chi}(s), \sigma_{\mathrm{bg},\chi}(s)]$. The mean value $\mu_{\mathrm{bg},\chi}(s)$ can be determined directly by

$$\mu_{\mathrm{bg},\chi}(s) = \sum_{r \in \mathcal{K}_s} a_s(r) \cdot \mu_{\mathrm{bg},L}(r) \tag{3.24}$$

as a consequence of widely used results of probability calculus (see Theorems 4 and 5 given in Appendix A page 135).
On the other hand, to estimate the $\sigma_{\mathrm{bg},\chi}(s)$ parameter, we should model the correlation between the elements of Λ_s.
In effect, the $o_L(r)$ variables in Λ_s are non-independent, since fine alterations in global illumination or camera white balance cause correlated changes of the neighboring pixel values. However, very high correlation is not usual, since strongly textured details or simply the camera noise result in some independence of the adjacent pixel levels. While previous methods have ignored this phenomenon e.g.

with considering the features to be uncorrelated [40], our goal is to give a more appropriate statistical model by estimating the order of correlation for a given scene.

We model the correlation factor between the 'adjacent' pixel values by a constant over the whole image. Let be q and r two pixels in the neighborhood of s ($q, r \in \mathcal{K}_s$), and denote by $c_{q,r}$ the correlation coefficient between q and r. Accordingly,

$$c_{q,r} = \begin{cases} 1 & \text{if } q = r \\ c & \text{if } q \neq r \end{cases} \tag{3.25}$$

where c is a global constant. To estimate c, we randomly choose some pairs of neighboring pixels. For each selected pixel pair (q, r), we make a set $I_{q,r}$ from time stamps corresponding to common background occurrences of pixels q and r. Thereafter, we calculate the normalized cross correlation $\hat{c}_{q,r}$ between time series $\{o_L^{[t]}(q)|t \in I_{q,r}\}$ and $\{o_L^{[t]}(r)|t \in I_{q,r}\}$, where t indices are time stamps of the o_L measurements. Finally, we approximate c by the average of the collected correlation coefficients $\hat{c}_{q,r}$ over all selected pixel pairs.

Thereafter, we can calculate $\sigma^2_{\mathrm{bg},\chi}(s)$ according to Theorems 4 and 5:

$$\sigma^2_{\mathrm{bg},\chi}(s) = \sum_{q,r \in \mathcal{K}_s} a_s(q) \cdot a_s(r) \cdot \sigma_{\mathrm{bg},L}(q) \cdot \sigma_{\mathrm{bg},L}(r) \cdot c_{q,r} \tag{3.26}$$

Similarly, the Gaussian shadow parameters regarding the microstructural components by using eq. (3.18), (3.19), (3.24):

$$\mu_{\mathrm{sh},\chi}(s) = \sum_{r \in \mathcal{K}_s} a_s(r) \cdot \mu_{\psi,L} \cdot \mu_{\mathrm{bg},L}(r) = \mu_{\psi,L} \cdot \mu_{\mathrm{bg},\chi}(s) \tag{3.27}$$

$$\sigma^2_{\mathrm{sh},\chi}(s) = \sigma^2_{\psi,L} \sum_{q,r \in \mathcal{K}_s} b_{q,r}(s) \tag{3.28}$$

where

$$b_{q,r}(s) = a_s(q) \cdot a_s(r) \cdot \mu_{\mathrm{bg},L}(q) \cdot \mu_{\mathrm{bg},L}(r) \cdot c_{q,r} \tag{3.29}$$

3.3.4.3 Strategies for Choosing Kernels

In the following we deal with zero-mean kernels ($\forall s : \sum_{r \in \mathcal{K}_s} a_s(r) = 0$) as a generalization of simple first-order edge features by [40]. Here we face an important problem from an experimental point of view. Each kernel has an adequate

pattern, for which it generates a significant nonzero response, while most of the pixel-neighborhoods in an image are 'untextured' with respect to it. Therefore, one single kernel is unable to discriminate an 'untextured' object point on an 'untextured' background. An evident enhancement uses several kernels which can recognize several patterns. However, increasing the number of the microstructural channels would intensify the noise, because at a given pixel position all the 'inadequate' kernels give irrelevant responses, which are accumulated in the energy term eq. (3.3). To overcome this problem we use one microstructural channel only (see eq. (3.1)), and we use the most appropriate kernel at each pixel. Our hypothesis is: if the kernel response at s is significant in the background, the kernel gives more information for the segmentation there. Therefore, after we have defined a kernel set for the scene, at each pixel position s the kernel having the highest absolute response in the background centered at s is used. According to our experiments, different kernel-sets, e.g. corresponding to the Laws-filters [31], or the Chebyshev polynomials [31][90], produce similar results. In the following sections we use the kernels shown in Fig. 3.3, which we have found reasonable for the scenes. Regarding the 'Entrance pm' sequence, each kernel of the set corresponds to a significant number of background points according to our choice strategy (distributed as 44-19-22-15%), showing that each kernel is valuable.

-1	0	1
0	0	0
1	0	-1

1	0	-1
-2	0	2
1	0	-1

-1	2	-1
0	0	0
1	-2	1

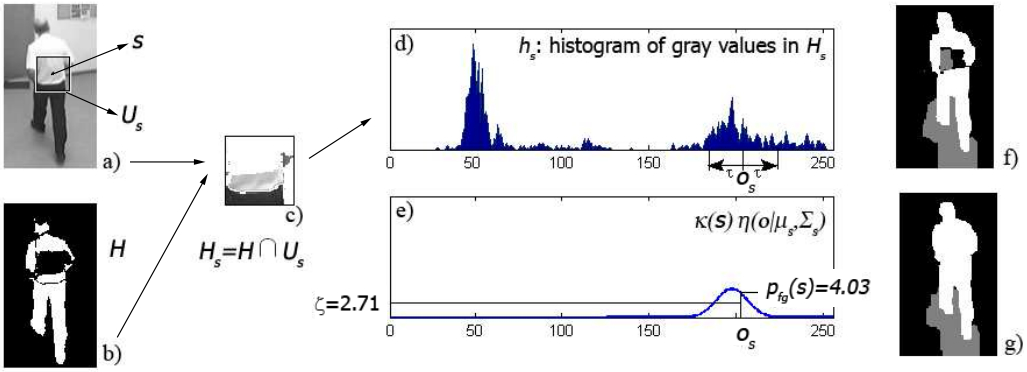
1	-2	1
-2	4	-2
1	-2	1

Figure 3.3: Kernel-set used in the experiments: 4 of the impulse response arrays corresponding to the 3×3 Chebyshev basis set proposed by [90]

3.4 Foreground Probabilities

The description of background and shadow characterizes the scene and illumination properties, consequently it has been possible to collect statistical information about them in time. In our case, the color distribution regarding the foreground

Foreground probability calculation



a) video image, with marking s and its neighborhood U_s (window side $z = 45$ is used.)

b) noisy preliminary foreground mask

c) Set H_s: preliminary detected foreground pixels in U_s. (Pixels of $U_s \backslash H_s$ are marked with white.)

d) Histogram of H_s, marking o_s, and its τ neighborhood.

e) Result of fitting a weighted Gaussian term for the $[o(s) - \tau, o(s) + \tau]$ part of the histogram. Here, $\zeta = 2.71$ is used (it would be the foreground probability density value for each pixel according to the 'uniform' model), but the procedure increases the foreground probability to 4.03.

f) Segmentation result of the model optimization with the uniform foreground calculus

g) Segmentation result by the proposed model

Figure 3.4: Determination of the foreground conditional probability term for a given pixel s (for simpler representation in grayscale).

areas is unpredictable in the same way. If the frame rate is very low and unbalanced, we must consider consecutive images containing different scenarios with different objects. Previous works [30][39] used uniform distribution to describe the foreground process which agrees with the long-term color statistics of the foreground pixels (Fig. 3.2), but it presents a weak description of the class. Since the observed feature values generated by the foreground, shadow and background processes overlap strongly in numerous real world scenes, many foreground pixels are misclassified that way.

Instead of temporal statistics we use spatial color information to overcome this problem by using the following assumption: whenever s is a foreground pixel, we should find foreground pixels with similar color in the neighborhood. Consequently, if we can estimate the color statistics of the nearby foreground pixels, we can decide if a pixel with a given color is likely part of the foreground or not. Unfortunately, when we want to assign a probability value to a given pixel describing its foreground membership, the positions of the nearby foreground pixels are also unknown. However, to estimate the local color distribution, we do not need to find all foreground pixels, just some samples in each neighborhood. The key point is that we identify some pixels which *certainly* correspond to the foreground: these are the pixels having significantly different levels from the locally estimated background and shadow values, thus they can be found by a simple thresholding:

$$\omega_s^0 = \begin{cases} \text{fg} & \text{if } (\epsilon_{\text{bg}}(s) > \zeta) \text{ AND } (\epsilon_{\text{sh}}(s) > \zeta) \\ \text{bg} & \text{otherwise} \end{cases} \tag{3.30}$$

where ζ is a threshold (which in analogous with the uniform value in previous models [39] choosing $\epsilon_{\text{fg}}(s) = \zeta$), and ω_s^0 is a 'preliminary' segmentation label of s.

Next, we estimate for each pixel s the local color distribution of the foreground, using the *certainly* foreground pixels in the neighborhood of s. The procedure is demonstrated in Fig. 3.4 (for easier visualization with 1D grayscale feature vectors). We use the following notations: H denotes the set of pixels marked as *certainly* foreground elements in the preliminary mask:

$$H = \{r \mid r \in S,\ w_r^0 = \text{fg}\} \tag{3.31}$$

Note that H may be a coarse estimation of the foreground (Fig. 3.4b).
Let be U_s the set of the neighboring pixels around s, considering rectangular neighborhood with window size $z \times z$ (Fig. 3.4a). Thereafter, H_s is defined with respect to s as the set of neighboring pixels determined as 'foreground' by the preprocessing step: $H_s = H \cap U_s$ (Fig. 3.4c).
The foreground color distribution around s can be characterized by a normalized histogram h_s over H_s (Fig. 3.4d). However, instead of using the noisy h_s directly, we approximate it by a 'smoothed' probability density function, $f_s(\overline{o})$, and determine the foreground probability term as $p_{\text{fg}}(s) = f_s\big(\overline{o}(s)\big)$.[1]
To deal with multi-colored or textured foreground components, the estimated $f_s(.)$ function should be multi-modal (see a bimodal case in Fig. 3.4d). Note that we use $f_s(.)$ only to calculate the foreground probability value of s as $f_s\big(\overline{o}(s)\big)$. Thus, it is enough to estimate the parameters of the mode of $f_s(.)$, which covers $\overline{o}(s)$ (see Fig. 3.4e). Therefore, we consider $f_s(.)$ as a mixture of a weighted Gaussian term $\eta(.)$ and a residual term $\vartheta_s(.)$, for which we only prescribe that $\vartheta_s(.)$ is a probability density function and $\vartheta_s(\overline{o}) = 0$ if $\|\overline{o}(s) - \overline{o}\| < \tau$. ($\kappa(s)$ is a weighting factor: $0 < \kappa(s) < 1$.) Hence,

$$f_s(\overline{o}) = \kappa(s) \cdot \eta(\overline{o}|\overline{\mu}_s, \overline{\overline{\Sigma}}_s) + \big(1 - \kappa(s)\big) \cdot \vartheta_s(\overline{o}) \tag{3.32}$$

Accordingly, the foreground probability value of pixel s is statistically characterized by the distribution of its neighborhood in the color domain:

$$\epsilon_{\text{fg}}(s) = -\log f_s\big(\overline{o}(s)\big) = -\log \kappa(s) - \log \eta\big(\overline{o}(s)|\overline{\mu}_s, \overline{\overline{\Sigma}}_s\big) \tag{3.33}$$

The steps of the foreground energy calculation are detailed in Fig. 3.5. We can speed up the algorithm, if we calculate the Gaussian parameters by considering only some randomly selected pixels in H_s [4]. We describe the parameter settings in Section 3.5.1.

[1] In the spatial foreground model, we must ignore the textural component of $\overline{o}$, since different kernels are used in different pixel locations, and the microstructural responses of the various pixels may be incomparable. Thus in this section, $\overline{o}$ is considered to be a three dimensional color vector, and h_s a three dimensional histogram.

Algorithm 1: foreground probability calculation

1. The pixels of H_s whose pixel values are close enough to $\overline{o}(s)$ are collected into a set:
$$H_s^D = \{r \mid r \in H_s,\ \|\overline{o}(s) - \overline{o}(r)\| < \tau\} \tag{3.34}$$

2. The empirical mean and deviation values are calculated regarding the color levels of set H_s^D: $\overline{\mu}_s^D$, $\overline{\sigma}_s^D$. These values estimate the mean and deviation parameters of the Gaussian component $\eta(.)$.

3. Denote by $\#H$ the number of the elements in set H. $\kappa^{(1)}(s) = \frac{\#H_s^D}{\#H_s}$ is introduced as the ratio of the number of pixels with similar color to s and all pixels, among the neighboring foreground initialized pixels.

4. An extra term is used to keep the probability low if there are not any or only a few foreground pixels in the neighborhood. Denote by $\kappa^{(2)}(s) = \frac{\#H_s}{z^2}$ the ratio of the number of pixels in H_s and the size of the neighborhood U_s. This term biases the weight through a sigmoid function:
$$\kappa(s) = \kappa^{(1)}(s) \cdot \frac{1}{1 + \exp\left[- \left(\kappa^{(2)}(s) - \kappa_{\min}/2\right)\right]} \tag{3.35}$$

5. Finally, the energy term is calculated as:
$$\epsilon_{\text{fg}}(s) = -\log \kappa(s) - \log \eta(\overline{o}(s), \overline{\mu}_s^D, \overline{\sigma}_s^D) \tag{3.36}$$

Figure 3.5: Algorithm for determination of the foreground probability term. Notations are defined in Section 3.4.

3.5 Parameter Settings

Our method works with scene-dependent and condition-dependent parameters. *Scene-dependent* parameters can be considered constant in a specific field, and are influenced by, e.g. camera settings, a priori knowledge about the appearing objects or reflection properties. We provide strategies on how to set these parameters if a surveillance environment is given. *Condition-dependent* parameters vary in time in a scene, therefore, we use adaptive algorithms to follow them.
We emphasize two properties of the presented model. Regarding the background and shadow processes, only the one dimensional marginal distribution parameters should be estimated (Section 3.3.1). On the other hand, we should estimate here the color-distribution parameters only, since the mean-deviation values corresponding to the microstructural component are determined analytically (see Section 3.3.4.2).

3.5.1 Background and Foreground Model Parameters

The *background* parameter estimation and update procedure is automated, based on the work in [62], which presents reasonable results, and it is computationally more effective than the standard EM algorithm.
The *foreground* model parameters (Section 3.4) correspond to a priori knowledge about the scene, e.g. the expected size of the appearing objects and the contrast. These features exploit basically low-level information and are quite general, therefore the method is able to consider a large variety of moving objects in a scene. In our experiments, we set these parameters empirically using the following strategies:

- z: the size of the neighborhood window U_s in pixels considered in the process. It depends on the expected size of the objects in the scene, used $z = 1/3\sqrt{T_B}$, where T_B is the approximate average territory of the objects' bounding boxes.

- $\kappa_{\min}$: control parameter for the minimum required number of pre-classified foreground pixels in the neighborhood. If the ratio of the pixels and the size of the neighborhood is smaller than $\kappa_{\min}$, the foreground probability

Figure 3.6: Different parts of the day on 'Entrance' sequence, segmentation results. Above left: in the morning ('am'), right: at noon, below left: in the afternoon ('pm'), right: wet weather

will be low there, due to the sigmoid function of eq. (3.35). Small $\kappa_{\min}$ increases the number of detected foreground pixels and can be used if the objects are of compact shape like in the sequence 'Highway'. Otherwise small $\kappa_{\min}$ causes high false foreground detection rate. Applying $\kappa_{\min} = 0.1$ for vehicle monitoring and $\kappa_{\min} = 0.25$ for pedestrians (including cyclists, baby carriages etc.) proved to be good.

- τ: the threshold parameter which defines the maximum distance in the feature space between pixels generated by one Gaussian process. We use outdoors in high contrast, $\tau = 0.2 \cdot d_{\max}$, indoors $\tau = 0.1 \cdot d_{\max}$, where $d_{\max}$ is the maximum occurring distance in the feature space.

Notes on parameter ζ are given in Section 3.7 and in Fig. 3.11.

3.5.2 Shadow Parameters

The changes in the global illumination significantly alter the shadow properties (Fig. 3.6). Moreover, changes can be performed rapidly: indoors due to switch on/off different light sources, while outdoors due to the appearance of clouds.
Regarding the shadow parameter settings, we discriminate parameter initialization and re-estimation. From a practical point of view, initialization may be supervised with marking shadowed regions in a few video frames by hand, once after switching on the system. Based on the training data, we can calculate

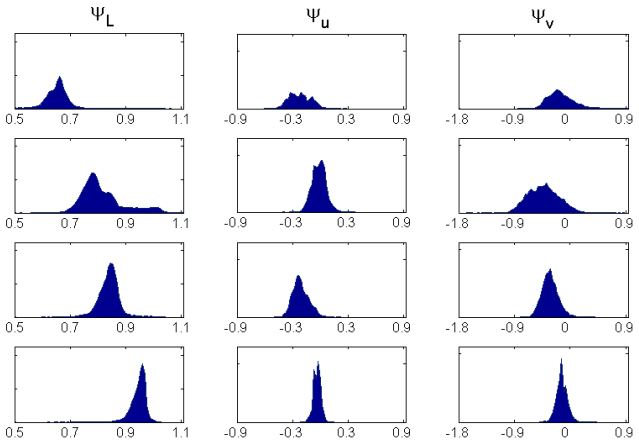


Figure 3.7: Shadow $\overline{\psi}$ statistics on four sequences recorded by the 'Entrance' camera of our University campus. Histograms of the occurring ψ_L, ψ_u and ψ_v values of shadowed points. Rows correspond to video shots from different parts of the day. We can observe that the peak of the ψ_L histogram strongly depends on the illumination conditions, while the change in the other two shadow parameters is much smaller.

maximum likelihood estimates of the shadow parameters. On the other hand, there is usually no opportunity for continuous user interaction in an automated surveillance environment, thus the system must adopt the illumination changes raising a claim to an automatic re-estimation procedure.

For the above reasons, we use supervised initialization, and focus on the parameter adaption process in the following. The presented method is built in a 24-hour surveillance system of our university campus. We validate our algorithm via four manually evaluated ground truth sequences captured by the same camera under different illumination conditions (Fig. 3.6).

According to section 3.3.2, the shadow parameters are 6 scalars: 3-3 components of $\overline{\mu}_\psi$ respectively $\overline{\sigma}_\psi$ vectors. Fig. 3.7 shows the one-dimensional histograms for the occurring ψ_L, ψ_u and ψ_v values of shadowed points for each video shot. We can observe that while the variation of parameters $\overline{\sigma}_\psi$, $\mu_{\psi,u}$ and $\mu_{\psi,v}$ are low, $\mu_{\psi,L}$ varies in time significantly. Therefore, we update the parameters in two different ways.

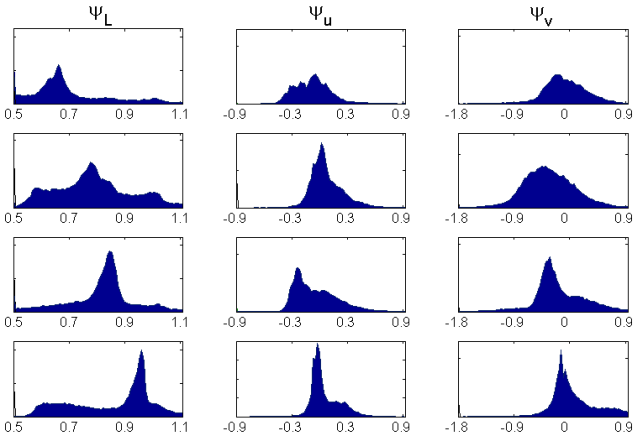


Figure 3.8: $\overline{\psi}$ statistics for all non-background pixels Histograms of the occurring ψ_L, ψ_u and ψ_v values of all the non-background pixels in the same sequences as in Figure 3.7.

3.5.2.1 Re-estimation of the Chrominance Parameters

The update procedure of parameters $[\mu_{\psi,u}, \sigma_{\psi,u}]$ and $[\mu_{\psi,v}, \sigma_{\psi,v}]$ is similar to that was used in [39]. We show it regarding the u component only, since the v component is updated in the same way.

We re-estimate the parameters at fixed time-intervals $\mathcal{T}$. Denote $\mu_{\psi,u}[t], \sigma_{\psi,u}[t]$ the parameters at time t. W_{t_2} is the set containing the observed ψ_u values collected over the pixels detected as shadow between time $t_1 = t_2 - \mathcal{T}$ and t_2:

$$W_{t_2} = \{\psi_u^{[t]}(s) | t = t_1, \ldots, t_2 - 1,\ \omega^{[t]}(s) = \text{sh},\ s \in S\} \tag{3.37}$$

where upper index $[t]$ refers to time, $\#W_{t_2}$ is the number of the elements, M_{t_2} and D_{t_2} are the empirical mean and the standard deviation values of W_{t_2}. We update the parameters:

$$\mu_{\psi,u}[t_2] = (1 - \xi^{[t_2]}) \cdot \mu_{\psi,u}[t_1] + \xi^{[t_2]} \cdot M_{t_2} \tag{3.38}$$

$$\sigma^2_{\psi,u}[t_2] = (1 - \xi^{[t_2]}) \cdot \sigma^2_{\psi,u}[t_1] + \xi^{[t_2]} \cdot D^2_{t_2} \tag{3.39}$$

Parameter $\xi^{[t]}$ is a weighting term ($0 \leq \xi^{[t]} \leq 1$) depending on $\#W_t$, namely greater number of detected shadow points increase $\xi^{[t]}$ and the influence of the M_t respectively D_t^2 term. We use $\mathcal{T} = 60$ sec.

3.5.2.2 Re-estimation of the Luminance Parameters

Parameter $\mu_{\psi,L}$ corresponds to the average background luminance darkening factor of the shadow. Except from window-less rooms with constant lightning, $\mu_{\psi,L}$ is strongly condition dependent. Outdoors, it can vary between 0.6 in direct sunlight and 0.95 in overcast weather. The simple re-estimation from the previous section does not work in this case, since the illumination properties between time t and $t+\mathcal{T}$ may rapidly change a lot, which would result in absolutely false detected shadow values in set W_t presenting false M_t and D_t parameters for the re-estimation procedure.

For this reason, we gain the actual $\mu_{\psi,L}$ from the statistics of all non-background ψ_L-s (where the background filtering should be done by a good approximation only, we use the Stauffer-Grimson algorithm). In Fig. 3.8 we can observe that the peaks of the 'non-background' ψ_L-histograms are approximately in the same location as they were in Fig. 3.7. The video shots corresponding to the first and second rows were recorded around noon where the shadows were relatively small, however, the peak is still in the right place in the histogram.

These experiments encourage us to identify $\mu_{\psi,L}$ with the location of the peak on the 'non-background' ψ_L-histograms for the scene.

The description of the update-algorithm of $\mu_{\psi,L}$ is as follows. We define a data structure which contains a ψ_L value with its timestamp: $[\psi_L, t]$. We store the 'latest' occurring $[\psi_L, t]$ pairs of the non-background points in a set $\mathcal{Q}$, and update the histogram h_L of the ψ_L values in $\mathcal{Q}$ continuously. The key point is the management of set $\mathcal{Q}$. We define MAX and MIN parameters which controls the size of $\mathcal{Q}$. The queue management algorithm has the following steps:

1. For each frame t we determine:

$$\Psi_t = \{\ [\psi_L^{[t]}(s), t] \mid s \in S,\ \omega^{[t]}(s) \neq \mathrm{bg}\} \tag{3.40}$$

2. We append Ψ_t to $\mathcal{Q}$.

3. We may remove elements from $\mathcal{Q}$:

   - if $\#\mathcal{Q} <$ MIN, we keep all the elements.

- if $\#\mathcal{Q} \geq$ MIN we find the eldest timestamp t_e in $\mathcal{Q}$ and remove all the elements from $\mathcal{Q}$ with time stamp t_e.

4. If $\#\mathcal{Q} >$ MAX after step 3: in order of their timestamp we remove further ('old') elements from $\#\mathcal{Q}$ till we reach $\#\mathcal{Q} \leq$ MAX.

5. We update the histogram h_L regarding $\mathcal{Q}$ and apply:

$$\mu_{\psi,L}^{[t+1]} = \text{argmax}\{h_L\} \tag{3.41}$$

Hence, the algorithm, follows four intentions:

- $\mathcal{Q}$ contains always the latest available ψ_L values.
- The algorithm keeps the size of $\mathcal{Q}$ between prescribed bounds MAX and MIN ensuring the topicality and relevancy of the data contained.
- The actual size of $\mathcal{Q}$ is around MAX in case of cluttered scenarios.
- In the case of few or no motion in the scene, the size of $\mathcal{Q}$ decreases until MIN. This fact increases the influence of the forthcoming elements, and causes quicker adaptation, since it is faster to modify the shape of a smaller histogram.

Parameter $\sigma_{\psi,L}$ is updated similarly to $\sigma_{\psi,u}$ but only in the time periods when $\mu_{\psi,L}$ does not change significantly.
Note that the above update process may fail in shadow free scenarios. However, that case occurs mostly under artificial illumination conditions, where the shadow detector module can be switched off using a priori knowledge.

3.6 MRF Optimization

The MAP estimator (eq. 2.16) is realized by combining a conditional independent random field of signals and an unconditional Potts model [42]. The optimal segmentation corresponds to the global labeling, $\widehat{\underline{\omega}}$, defined by

$$\widehat{\underline{\omega}} = \arg\min_{\underline{\omega}\in\Omega} \left\{ \sum_{s\in S} \underbrace{-\log P\big(\overline{o}(s) \mid \omega(s)\big)}_{\epsilon_{\omega(s)}(s)} + \sum_{r,s\in S} \Theta\left(\omega(r), \omega(s)\right) \right\} \tag{3.42}$$

Table 3.2: Comparing the processing speed of our proposed model to three latest reference methods (using the published frame-rates). Note that [76] does not use any spatial smoothing (like MRF), and [38] performs only a two-class separation.

	M-Brisson05 [76]	Sheikh05 [38]	Wang06 [40]	Proposed
Classes	3	2	3	3
MRF Opt	-	Graph cut	ICM	ICM
Frame-rate	10 fps	11 fps	1-2 fps	3 fps

where the minimum is searched over all the possible segmentations (Ω) of a given input frame. The first part of eq. (3.42) contains the sum of the local class-energy terms regarding the pixels of the image (see eq. (3.3) and eq. (3.36)). The second part is responsible for the smooth segmentation: $\Theta\left(\omega(r), \omega(s)\right) = 0$ if s and r are not neighboring pixels, otherwise:

$$\Theta\left(\omega(r), \omega(s)\right) = \begin{cases} -\delta & \text{if } \omega(r) = \omega(s) \\ +\delta & \text{if } \omega(r) \neq \omega(s) \end{cases} \tag{3.43}$$

As for optimization, we have found the deterministic Modified Metropolis (MMD) [53] relaxation method similarly efficient but significantly faster for this task than the original stochastic SA algorithm mentioned in Section 2.4: processing 320×240 images runs with 1 fps using it. If we use ICM with our model, the running speed is 3 fps, in exchange for some degradation in the segmentation results. For comparison, frame-rates of three latest reference methods are shown in Table 3.2. We can observe that our model has approximately the same complexity as [40]. Although the speed of [38] and [76] is notably higher, one should consider that [76] does not use any spatial smoothing (like MRF), thus a separate noise filter must be applied there in the post-processing phase. On the other hand [38] performs only a two-class segmentation (background and foreground). That simplification enables using the quick graph cut based MRF optimization techniques, unlike in the three-class cases [49].

3.7 Results

The goal of this section is to demonstrate the benefit of using the introduced contributions of this chapter: the novel foreground calculus, shadow model and the benefit of the textural features. The demonstration is done in two ways: in Fig. 3.12–3.11 we show segmented images by the proposed and previous methods, while regarding three sequences we perform numerical evaluation.

3.7.1 Test Sequences

We have validated our method on several test sequences, here, we show results regarding the following 7 videos:

- 'Laboratory' test sequence from the ATON benchmark set [28] (available at `http://cvrr.ucsd.edu/aton/shadow/`) This shot contains a simple environment where previous methods [40] have produced already accurate results.

- 'Highway' video (ATON benchmark set). This sequence contains dark shadows but homogenous background without illumination artifacts. Contrast to [30] our method reaches the appropriate results without post processing, which is strongly environment-dependent.

- 'Corridor' indoor surveillance video. Although, it is on the face of a simple office environment the bright objects and background elements often saturate the image sensors and it is hard to accurately separate the white shirts of the people from the white walls in the background.

- 4 surveillance video sequences captured by the 'Entrance' (outdoor) camera of our university campus in different lightning condition. (See Fig 3.6: 'Entrance am', 'Entrance noon', 'Entrance pm' and 'Entrance overcast'). These sequences contain difficult illumination and reflection effects and suffer from sensor saturation (dark objects and shadows). Here, the presented model improves the segmentation results significantly versus previous methods.

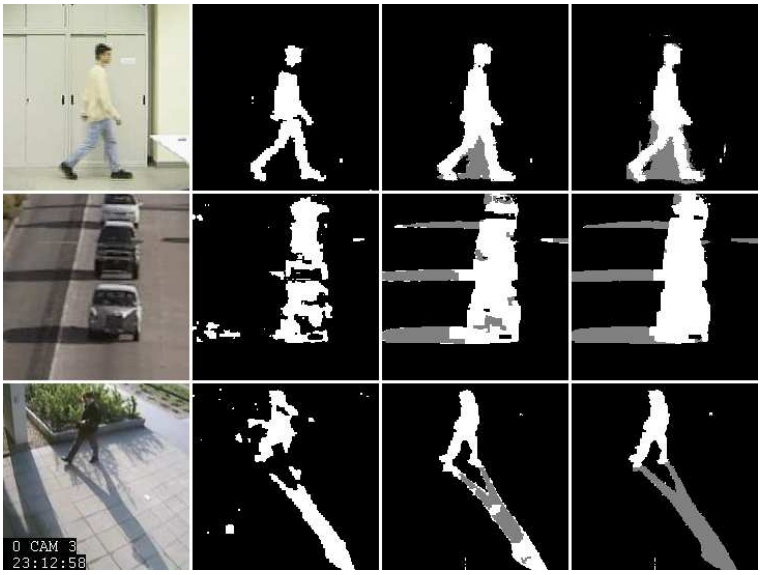


Figure 3.9: *Shadow model validation:* Comparison of different shadow models in 3 video sequences (From above: 'Laboratory','Highway','Entrance am') . Col. 1: video image, Col. 2: $C_1C_2C_3$ space based illumination invariants [74]. Col. 3: 'constant ratio model' by [30] (without object-based postprocessing) Col 4: Proposed model

3.7.2 Demonstration of the Improvements via Segmented Images

In the introduction we gave an overview on the state-of-the art methods (Table 3.1) indicating their way of (i) shadow detection (ii) foreground modeling (iii) textural analysis.

3.7.2.1 Comparison of Shadow Models

Results of different shadow detectors are demonstrated in Fig. 3.9. For the sake of comparison we have implemented in the same framework an illumination invariant ('II') method based on [74], and a constant ratio model ('CR'), similarly to [30]. We have observed that the results of the previous and the proposed methods are similar in simple environments, but our improvements become significant in the surveillance scenes:

- In the *'Laboratory'* sequence, the 'II' approach is reasonable, while the 'CR' and the proposed method are similarly accurate.

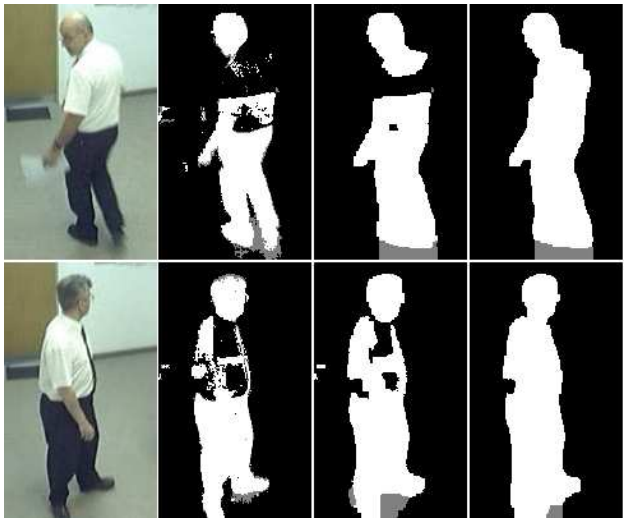


Figure 3.10: *Foreground model validation* regarding the 'Corridor' sequence. Col. 1: video image, Col. 2: Result of the preliminary detector. Col. 3: Result with uniform foreground calculus Col 4: Proposed foreground model

- Regarding the *'Highway'* video, although the 'II' and 'CR' find the objects without shadows approximately, but the results are much noisier than it is with our model.

- On the *'Entrance am'* surveillance video, the 'II' method fails completely: shadows are not removed, while the foreground component is also noisy due to the lack of using luminance features in the model. The 'CR' model produces poor results also: due to the long shadows and various field objects the constant ratio model becomes inaccurate. Our model handles these artifacts robustly.

The improvements of the proposed method versus the 'CR' model can be also observed in Fig. 3.14 (2^{nd} and 5^{th} row).

3.7.2.2 Comparison of Foreground Models

In this chapter we have proposed a basically new approach regarding foreground modeling, which needs neither high frame rate contrasted to [37][38][40], nor high level object descriptors [78]. Other previous models [30][39] have used the uniform

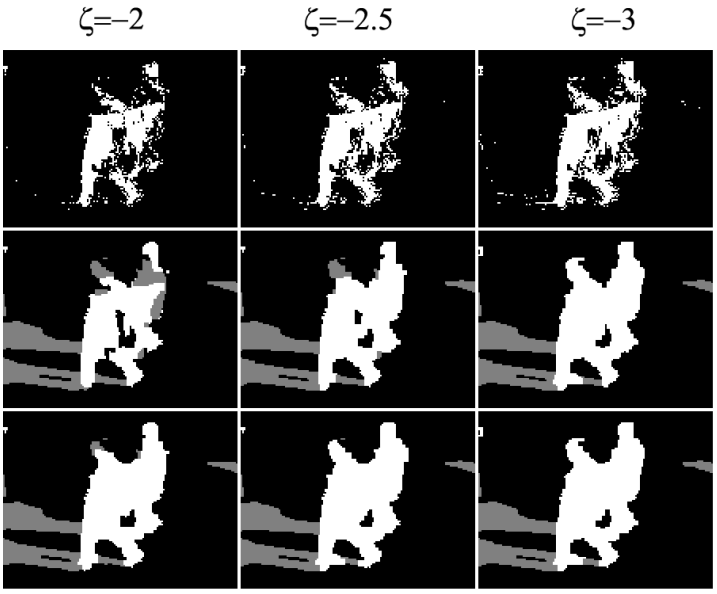


Figure 3.11: *Effect of changing the ζ foreground threshold parameter.* Row 1: preliminary masks (H), Row 2: results with uniform foreground calculus using $\epsilon_{\text{fg}}(s) = \zeta$, Row 3. results with the proposed model. Note: for the uniform model, $\zeta = -2.5$ is the optimal value with respect to the whole video sequence.

calculus expressing foreground may generate any colors in a given domain with the same probability. As it is shown in Fig. 3.13, 3.10 and 3.14 (3$^{\text{rd}}$ and 5$^{\text{th}}$ rows), the uniform model is often a coarse approximation, and our method is able to improve the results significantly. Moreover, we have observed that our model is robust with respect to fine changes in the threshold parameter ζ (an example is shown in Fig. 3.11, 3$^{\text{rd}}$ row). On the other hand, the uniform model is highly sensitive to set ζ appropriately, even in scenarios which can be segmented properly with an adequate uniform value (Fig. 3.11, 2$^{\text{nd}}$ row).

3.7.2.3 Microstructural Features

Completing the pixel-level feature vector with the microstructural component enhances the segmentation result if the background or the foreground is textured. To demonstrate the additional information, Fig. 3.12 shows a synthetic example. Consider Fig. 3.12a as a frame of a sequence where the bright rectangle in the middle corresponds to the foreground (image v. shows an enlarged part of it).

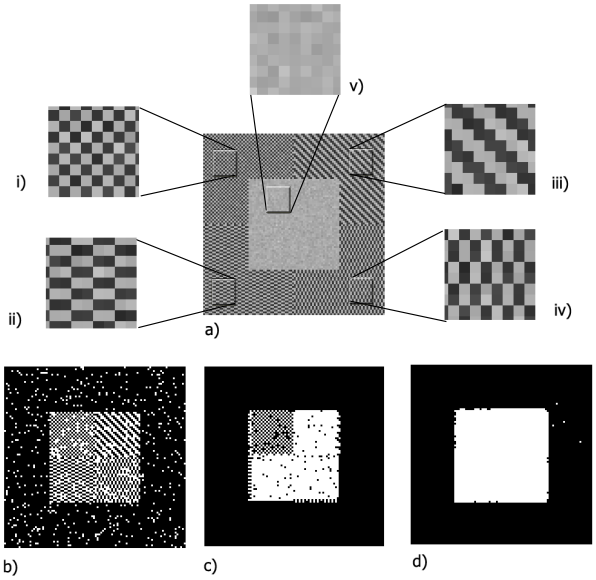


Figure 3.12: Synthetic example to demonstrate the benefits of the microstructural features. a) input frame, i-v) enlarged parts of the input, b-d) result of foreground detection based on: (b) gray levels (c) gray levels with vertical and horizontal edge features [40] (d) proposed model with adaptive kernel

The background consists of four equal rectangular regions, each of them has a particular texture, which are enlarged in i-iv. images. Similarly to the real-world case, the observed pixel values are affected by Gaussian noise. Below, we can see results of background subtraction. First (image b), the feature vector only consists of the gray value of the pixel. Secondly (image c), we complete it with horizontal and vertical edge detectors similarly to [40]. Finally (image d), we use the kernel set of Fig. 3.3, with the proposed kernel selection strategy, providing the best results.
In Fig 3.14, the 4^{th} and 5^{th} rows show the segmentation results without and with the textural components, improvements are observable in fine details, especially near the legs of the people in the magnified regions.

3.7.3 Numerical Evaluation

The evaluations are done through manually generated ground truth sequences. Since the goal is foreground detection, the crossover between shadow and back-

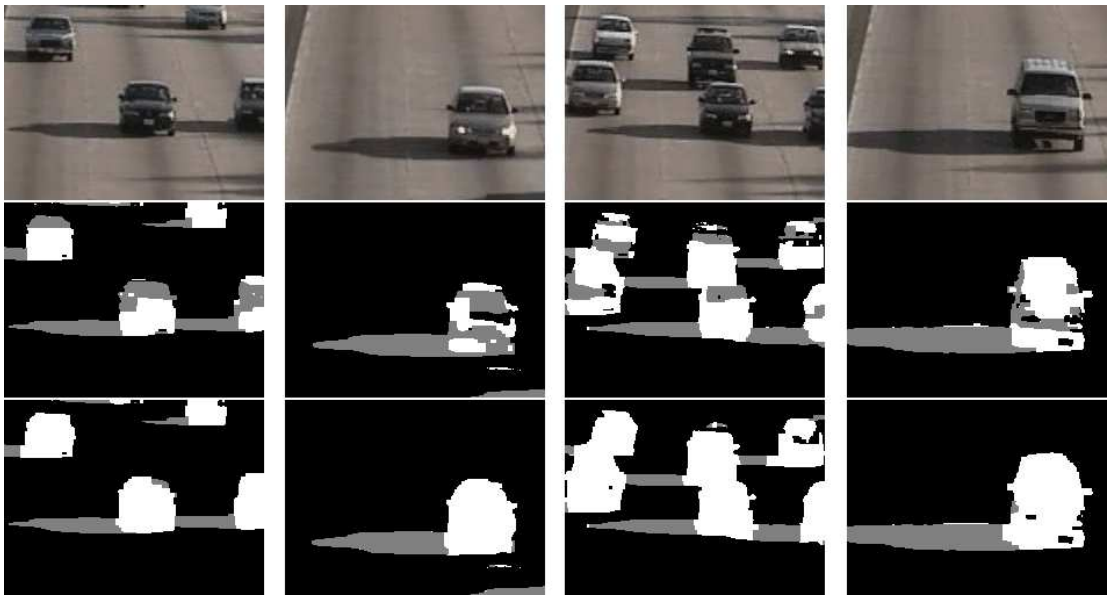


Figure 3.13: *Foreground model validation:* Segmentation results on the 'Highway' sequence. Row 1: video image; Row 2: results by uniform foreground model; Row 3: Results by the proposed model

ground does not count for errors.

Denote the number of correctly identified foreground pixels of the evaluation images by TP (*true positive*). Similarly, we introduce FP for misclassified background points, and FN for misclassified foreground points.

The evaluation metrics consists of the *Recall* rate and the *Precision* of the detection.

$$\text{Recall} = \frac{TP}{TP + FN} \qquad \text{Precision} = \frac{TP}{TP + FP} \tag{3.44}$$

Later on, we will use the F-measure (FM) [91] which combines *Recall* (Rc) and *Precision* (Pr) in a single efficiency measure (it is the harmonic mean of Rc and Pr):

$$FM = \frac{2 \cdot Rc \cdot Pr}{Rc + Pr}. \tag{3.45}$$

Note that while Rc and Pr characterize a given algorithm only together[1], FM is in itself an efficient evaluation metrics.

[1]Consider an algorithm which classifies each pixel as foreground. It is obviously a weak segmenter, but its Rc is equal to 1. However, Pr is low in that case.

Figure 3.14: *Validation of all improvements* in the segmentation regarding 'Entrance pm' video sequence Row 1. Video frames, Row 2. Ground truth Row 3. Segmentation with the 'constant ratio' shadow model [30], Row 4. Our shadow model with 'uniform foreground' calculus [39] Row 5. The proposed model without microstructural features Row 6. Segmentation results with our final model.

Table 3.3: Overview on the evaluation parameters regarding the five sequences. Notes: *number of frames in the ground truth set. ** *Frame rate of evaluation* (fre): number of frames with ground truth within one second of the video. ***Length of the evaluated video part. †fre was higher in 'busy' scenarios.

Video	Frames*	fre**	Duration (min) ***
Laboratory	205	2-4 fre†	1:28
Entrance am	160	2 fre	1:20
Entrance pm	75	1 fre	1:15
Entrance noon	251	1 fre	4:21
Highway	170	5-8 fre†	0:29

For numerical validation, we used in the aggregate 861 frames chosen from the 'Laboratory', 'Highway', 'Entrance am', 'Entrance noon' and 'Entrance pm' sequences. Details about the test sets are given in Table 3.3.

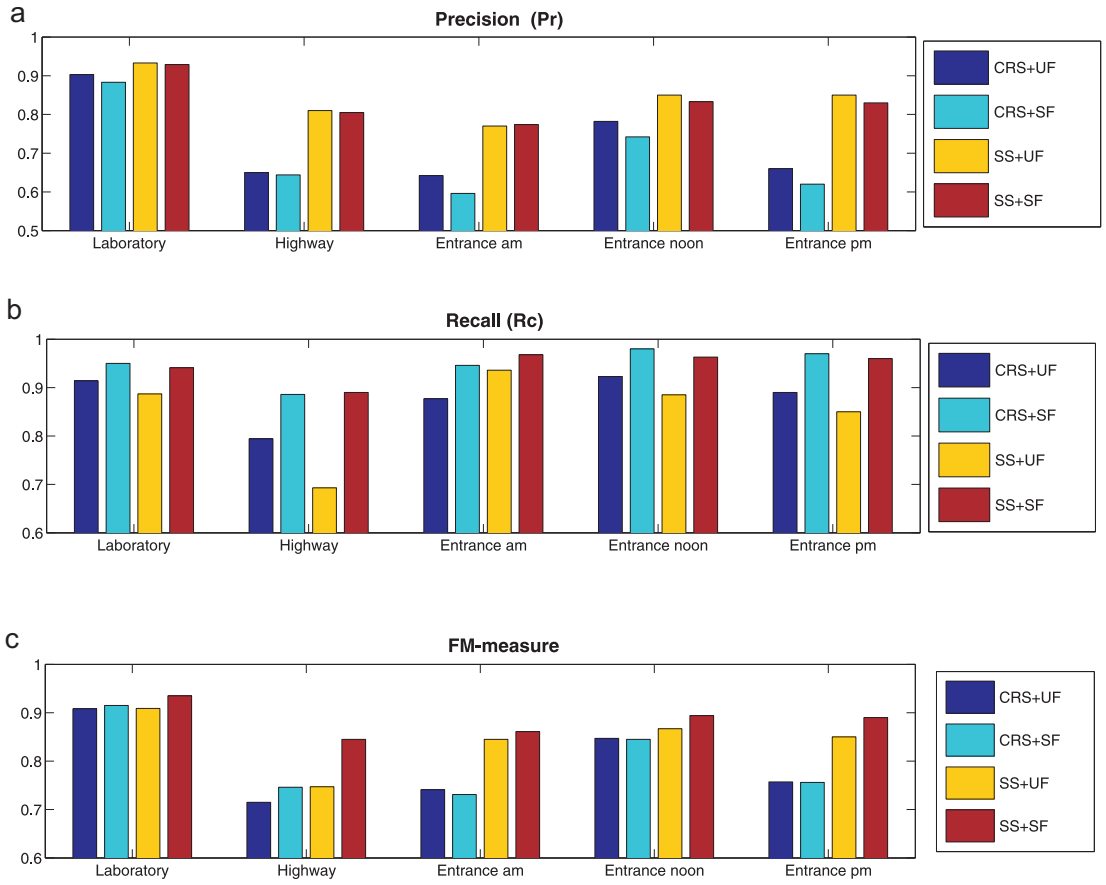
As for competitor methods used in the verification procedure, we focus on the state-of-the-art MRF models, since advantages of using Markov Random Fields versus morphology based approaches were examined previously [40]. The evaluation of the improvements is done by exchanging our new model elements one by one for the latest similar solutions in the literature, and we compare the segmentation results.

Regarding shadow detection, the 'CR' model is the reference, and we compare the foreground model to the 'uniform' calculus again.

In Fig 3.15, we compare the shadow and foreground model to the reference methods. The results confirm that our shadow calculus improves the precision rate, since it decreases the number of false negative detected shadow pixels significantly. Due to the foreground model, the recall rate increases through detecting several background/shadow colored foreground parts. If we ignore both improvements both Rc and Pr decrease. Fig 3.15c shows that regarding the FM-measure the proposed model outperforms the former ones in all cases.

3.7.4 Influence of CCD Selection on the Shadow Domain

We have introduced a statistical shadow model without any knowledge about the technical details and embedded control/LUT of the different cameras. Regarding



- CRS: constant ratio shadow model
- SS: proposed statistical shadow model
- UF: uniform foreground model
- SF: proposed spatial foreground model

Figure 3.15: Comparing the proposed model (red columns) to previous approaches. The total gain due to the introduced improvements can be got by comparing the corresponding CRS+UF and SS+SF columns: regarding the FM measure, the benefit is more than 12% for three out of the five sequences, $3-5\%$ for the remaining two ones.

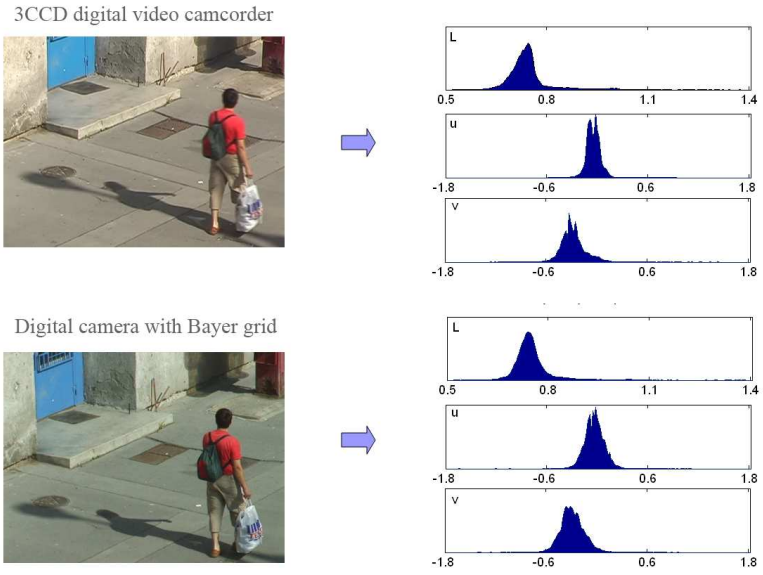


Figure 3.16: Distribution of the shadowed $\overline{\psi}$ values in simultaneous sequences from a street scenario recorded by different CCD cameras. Note: the camera with Bayer grid has higher noise, hence the corresponding u/v components have higher variance parameters.

the test sequences, we have had 3 different sources, partly from the Internet. The paper is dealing with the appropriate color-space, considering that any camera sensor type can be transformed into such space.

We have performed an additional experiment regarding this issue. We have recorded simultaneous videos from a street scenario with two different cameras: a 3CCD digital video camcorder and a conventional digital camera, which uses a Bayer grid. By examining the corresponding shadow domains (see Fig. 3.16), we can observe that the distributions of the shadowed $\overline{\psi}$ values are very similar. However, the higher noise of the Bayer grid camera results in higher variance parameters regarding the u and v components.

3.8 Conclusion of the Chapter

The present chapter has introduced a general model for foreground segmentation without any restrictions on a priori probabilities, image quality, objects' shapes and speed. The frame rate of the source videos might also be low or unstable,

while the method is able to adapt to the changes in lighting conditions. We have contributed to the state-of-the-art in three areas: (1) we have introduced a more accurate, adaptive shadow model; (2) we have developed a novel description for the foreground based on spatial statistics of the neighboring pixel values; (3) We have shown how different microstructure responses can be used in the proposed framework as additional feature components improving the results.

We have compared each contribution of our model to previous solutions in the literature, and observed its superiority. The proposed method now works in a real-life surveillance system (see Fig. 3.6) and its efficiency has been validated.

Chapter 4

Color Space Selection in Cast Shadow Detection

In this chapter we focus on a particular aspect of shadow detection: we illustrate that the performance of segmentation can be significantly improved through appropriate color space selection, if for practical purposes, we should keep the number of free parameters of the method low. We show experimental results regarding the following questions:

- What is the gain of using color images instead of grayscale ones?
- What is the gain of using uncorrelated spaces instead of the standard RGB?
- Are chrominance (illumination invariant), luminance, or 'mixed' spaces more efficient?
- In which scenes are the differences significant?

We qualify the metrics both in color based clustering of the individual pixels and in the case of Bayesian foreground-background-shadow segmentation through generalizing the model introduced in Chapter 3. Experimental results on the test videos show that CIE L*u*v* color space is the most efficient in both cases.

4.1 Introduction

Appropriate color space selection is a crucial step for many image processing problems [72][92][93]. Since the shadow model proposed in Chapter 3 is primarily based on describing the shadow's color domain, issues on color spaces should also be investigated in this case. Although shadow detection is a very well examined problem and some comparative works [28][94] have also been published in this topic, previous reviews classify and compare the existing methods based on their *model structures*. The authors [28] note that the methods work in different color spaces, like RGB [30] and HSV [29]. However, it remains open-ended, how important is the appropriate color space selection, and which color space is the most effective regarding shadow detection. Moreover, we find also further examples: [37] used only gray levels for shadow segmentation, other approaches were dealing with the CIE L*u*v* [76] and CIE L*a*b* [95] spaces, respectively (an overview is in Table 4.1). Note that an experimental evaluation of color spaces has been already done for shadow edge classification in [72], but in the current thesis, we address the detection of the shadowed and foreground regions, which is a fairly different problem.

For the above reasons, the main issue of this chapter is to give an experimental comparison of different color models regarding cast shadow detection on the video frames. Of course, the validity of such experiments is limited to the examined model structures, thus it is important to make the comparison in a relevant framework. Taking a general approach, we consider the task as a classification problem in the space of the extracted features, describing the different cluster domains with relatively few free parameters.[1]

As mentioned in the introduction models in the literature use usually *deterministic* (per pixel, e.g. [29])) or *statistical* (probabilistic, see [30]) approaches. Up to now, we have only dealt with statistical models, since they are more advantageous considering the whole segmentation process. On the contrary, here we introduce a deterministic method first, where the pixels are classified independently before the rate of the correct pixel-classification is investigated. That way, we can perform

[1]Most models in Table 4.1 also contain 2 parameters for each color channel, drawbacks of methods using less parameters have been emphasized in Chapter 3.

a relevant quantitative comparison of the different color spaces, since the decision for each pixel depends only on the corresponding local color-feature value; post processing and a priory effects whose efficiency may be environment-dependent do not take account here. Thereafter, we give a probabilistic interpretation to this model and we insert it to the adaptive MRF framework which was introduced in Chapter 3. We also compare the results after MRF optimization qualitatively and quantitatively.

Consequently, this chapter can be considered both as premise and generalization of Chapter 3. Our previous choice for using the CIE L*u*v* space will be justified here, but on the other hand, experiments will refer to the previously introduced model elements, extending their validity to various color models. Reasons for dedicating an independent chapter to this issue is that statistical feature modeling and color space analysis are two different and in themselves composite aspects of shadow detection. Although interaction between the two approaches will be emphasized several times, the separate discussion helps the clarity of presentation. Note as well that due to the various experiments the consequences of this chapter may be more generally usable than in the context of the proposed MRF model.

4.2 Feature Vector

Feature extraction is done similarly to Chapter 3, but here we give a generalization of the $\overline{\psi}$ shadow features to handle different color spaces.

We remind the Reader of the constant ratio model introduced in Section 3.3.3.1, where the ratio of the shadowed and illuminated sensor values have been considered near constant over the images. To handle the different artifacts, one can prescribe a *domain* [96] or a *distribution* (see Section 3.3.3.2 of this thesis) instead of a single value for the ratios, which results in a powerful detector.

Next, we should examine, how one can use this approach in different color systems. We begin the description with some notes. We assume that the camera presents the frames in the RGB space, and for the different color space conversions, we use the formulas of [97]. The ITU D65 standard is used again for the calibration of the CIE L*u*v* and L*a*b* spaces.

Table 4.1: Overview on state-of-the-art methods. † In cases of parametric methods, the (average) number of shadow parameters for one color channel. ‡Proportional to the number of support vectors after supervised training.

Method	**Color space**	**Number of param./ color channels**†
Cavallaro 2004 [73]	*rg*	invariant
Salvador 2004 [74]	$C_1C_2C_3$	invariant
Paragios 2001 [35]	*rg*	invariant
Mikic 2000 [30]	RGB	1
Rittscher 2002[37]	grayscale	2
Wang 2006 [40]	grayscale	2
Cucchiara 2001 [29]	HSV	1.33
Martel-Brisson 2005 [76]	CIE L*u*v*	2
Rautiainen 2001 [95]	CIE L*a*b*/HSV	N.a.
Siala 2004 [96]	RGB	N.a.‡
Proposed	All from above	2

Table 4.2: Luminance-related and chrominance channels in different color spaces

Color space	gray	rg	$C_1C_2C_3$	HSV	RGB	L*a*b*	L*u*v*
luminance ch.	g	-	-	H	R,G,B	L*	L*
chrominance ch.	-	r,g	C_1,C_2,C_3	S,V	-	a*,b*	u*,v*

As we did in the CIE L*u*v* based model (page 31) we will separately handle the color components which are directly related to the brightness of the pixels (we refer to them later as 'luminance' components), and the remaining ones which correspond to 'chrominances' of the observed colors. Classification of channels regarding different color spaces can be found in Table 4.2. In this way, we can also classify the color spaces: since the normalized *rg* and $C_1C_2C_3$ spaces contain only chrominance components we will call them 'chrominance spaces', while grayscale and RGB are purely 'luminance spaces'. In this terminology, HSV, CIE L*u*v* and L*a*b* are 'mixed spaces'.

The shadow descriptor is derived in an analogous manner to the approach of Section 3.3.3.2: the 'probabilistic ratio' method is used for the 'luminance'

components, while the offsets between the shadowed and illuminated 'chrominance' values of the pixels are modelled by a Gaussian additive term. In summary, if the current value of a given pixel in a given color space is $[o_0, o_1, o_2]$ (indices $0, 1, 2$ correspond to the different color components), the estimated (illuminated) background value is there $[\mu_{\mathrm{bg},0}, \mu_{\mathrm{bg},1}, \mu_{\mathrm{bg},2}]$, we define the shadow descriptor $\overline{\psi} = [\psi_0, \psi_1, \psi_2]$ by the following, for $i = \{0, 1, 2\}$:

- If i is the index of a 'luminance' component:

$$\psi_i(s) = \frac{o_i(s)}{\mu_{\mathrm{bg},i}(s)}. \tag{4.1}$$

- If i is the index of a 'chrominance' component:

$$\psi_i(s) = o_i(s) - \mu_{\mathrm{bg},i}(s). \tag{4.2}$$

We define the descriptor in grayscale and in the rg space similarly to eq. (4.1) and (4.2) considering that $\overline{\psi}$ will be a one and a two dimensional vector, respectively. The efficiency of the proposed feature selection regarding three color spaces can be observed in Fig. 4.1, where we plot the one dimensional marginal histograms of the occurring ψ_0, ψ_1 and ψ_2 values for manually marked shadowed and foreground points of a 75-frames long outdoor surveillance video sequence ('Entrance pm'). Apart from some outliers, the shadowed ψ_i values lie for each color space and each color component in a 'short' interval, while the difference of the upper and lower bounds of the foreground values is usually greater.

4.3 Deterministic Classifier

In this section, we temporarily put aside the MRF concept, and taking a deterministic approach, we consider the shadow detection problem as a simple classification task in the $\overline{\psi}$-feature space. Considering Fig. 4.1, an important note should be taken here. While the $\overline{\psi}$ statistics characterizes the scene and illumination conditions, the foreground $\overline{\psi}$ histograms only correspond to the occurring foreground objects in the evaluated sequence. On the other hand, an efficient shadow model is expected to work with differently colored objects as well. Therefore, the upcoming discrimination process will follow a one-class-classification approach: pixel

s will by classified as a shadowed point, if its $\overline{\psi}(s)$ value lies in the estimated *shadow domain*, and the *outlier points* will be labeled as foreground. As usual, the shadow domain is defined by a manifold having a prescribed number of free parameters, which fit the model to a given scene/situation. For grayscale images shadowed ψ features should be included by an interval [40], while regarding color scenes different domain models are used in the literature, like a three dimensional rectangular bin [29] (ratio/difference values for each channel lie between defined threshold), an ellipsoid [30], or the domain may have general shape [96]. In the latter case a Support Vector Domain description is proposed in the RGB color ratios' space.

By each domain-selection we must consider overlap between the classes, e.g. foreground points may appear whose feature values lie in the shadow domain. Therefore, the optimal domain should be as narrow as possible meanwhile containing 'almost all' the feature values corresponding to the occurring shadowed points. Accordingly, if we 'only' prescribe that a shadow descriptor should be accurate, the most general domain shape seems to be the most appropriate. However, in practise, we also have to consider issues of parameter estimation and adaption (see Section 3.5.2). Therefore, we prefer the domains with relatively few free parameters, for which we can construct an automatic update strategy.

Observe that according to Fig. 4.1, the shadowed ψ_0, ψ_1 and ψ_2 values follow approximately normal distributions, therefore, a 3D joint normal representation of the $\overline{\psi}$ features in shadows is straightforward (similarly to Chapter 3). Since the equipotential surfaces of the 3D Gaussian density functions are ellipsoids, a natural choice is using an elliptical shadow domain boundary. We will use the equation of a standard ellipsoid body having *parallel axes* with the coordinate axes in the $\psi_0 - \psi_1 - \psi_2$ Cartesian coordinate system:

$$\text{Pixel } s \text{ is shadowed} \Leftrightarrow \sum_{i=0}^{2} \left(\frac{\psi_i(s) - a_i}{b_i} \right)^2 \leq 1, \tag{4.3}$$

where $[a_0, a_1, a_2]$ is the coordinate of the ellipsoid center and (b_0, b_1, b_2) are the semi-axis lengths. In other words, $[a_0, a_1, a_2]$ is equivalent to the mean $\overline{\psi}(s)$ value of shadowed pixels in a given scene, while b_0, b_1 and b_2 depend on the spatiotemporal variance of the $\overline{\psi}(s)$ measurements under shadows. Later on we will show

that the similarity to the $\overline{\mu}_\psi$ and $\overline{\sigma}_\psi$ parameters from Chapter 3 is not by chance, thus, parameter adaption can also be done in a similar manner.
Note that with the SVM method [96], the number of free parameters is related to the number of the support vectors, which can be much greater than the six scalars of our model. Moreover, for each situation, a novel SVM should be trained. Note as well that one could use an arbitrarily oriented ellipsoid, but compared to eq. 4.3, it is also more difficult to define, since it needs the accurate estimation of 9 parameters.
For the sake of completeness, we note that the domain defined by eq. (4.3) becomes an interval if we work with grayscale images, and a two dimensional ellipse in the rg space.
Fig. 4.2 shows the two dimensional scatter plots about the occurring foreground and shadow $\overline{\psi}$ values. We can observe here that the components of vector $\overline{\psi}$ are strongly correlated in the RGB space (and also in $C_1C_2C_3$), and the previously defined ellipse cannot present a narrow boundary. In the HSV space, the shadowed values are not within a convex hull, even if we considered that the hue component is actually periodical (hue $= k * 2\pi$ means the same color for each $k = 0, 1, \ldots$). Based on the above facts, the CIE L*u*v* space seems to be a good choice. In the following, we support this statement by numerical results.

4.3.1 Evaluation of the Deterministic Model

The evaluations were done through manually generated ground truth sequences regarding five of the previously introduced test videos: namely the 'Laboratory', 'Highway', 'Entrance am', 'Entrance noon' and 'Entrance pm' sequences, with the same test parameters as before (see Table 3.3 in page 55 for details).
In this section, we show the tentative limits of the elliptical shadow domain defined by eq. (4.3). The goal of these experiments is to compare the foreground-shadow discriminating ability of the different color spaces purely based on the extracted per pixel $\overline{\psi}$ features. Therefore, we set here the parameters manually, and do not take into consideration local connectivity or post processing.
In the upcoming experiments, we collect for each test sequence two sets of $\overline{\psi}$ values

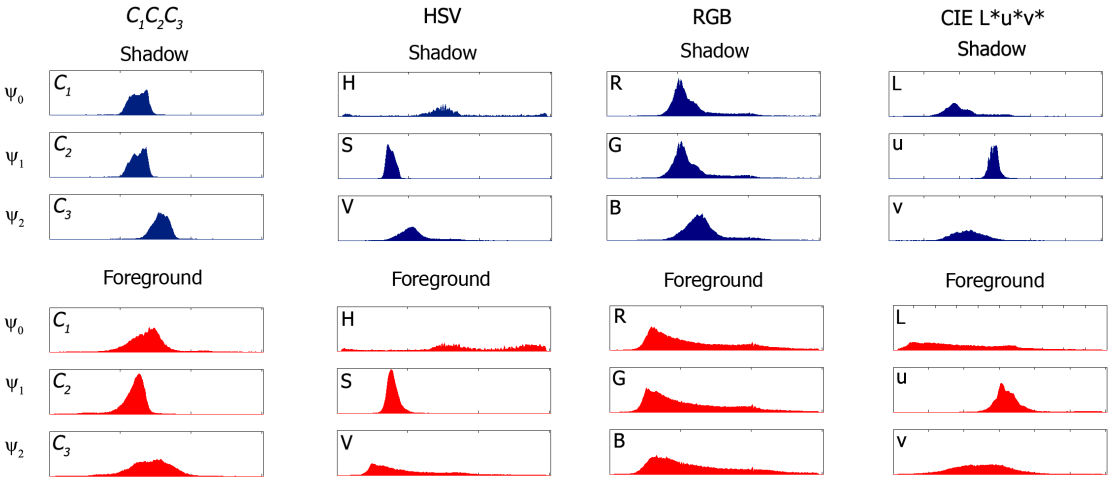


Figure 4.1: One dimensional projection of histograms of shadow (above) and foreground (below) $\overline{\psi}$ values in the 'Entrance pm' test sequence.

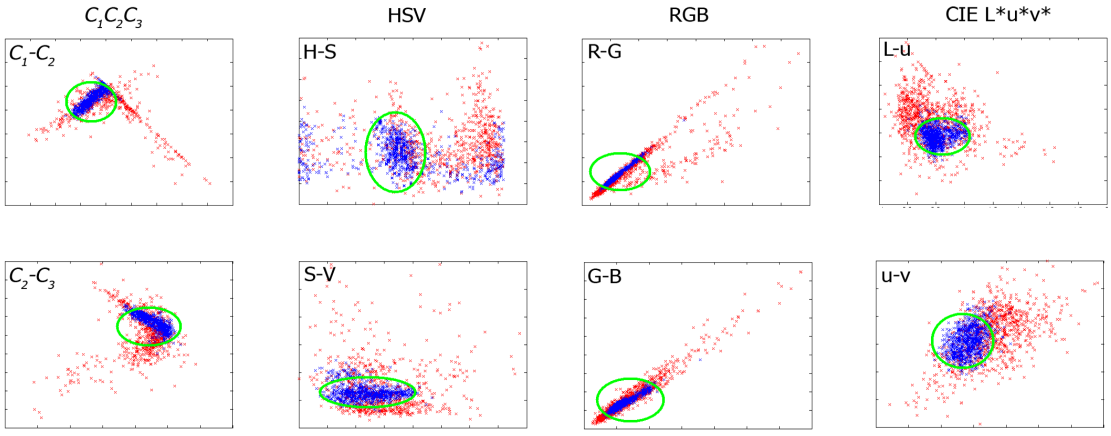


Figure 4.2: Two dimensional projection of foreground (red) and shadow (blue) $\overline{\psi}$ values in the 'Entrance pm' test sequence. Green ellipse is the projection of the optimized shadow boundary.

corresponding to manually marked foreground and shadowed pixels, respectively. We investigate on the correct-classification-rates of the pixels by using the ellipse model (eq. 4.3) with different color spaces. We henceforward use the *Recall* (Rc), *Precision* (Pr) measures, which were introduced in Section 3.7.3.
For some optimized ellipse parameters, we plot the corresponding *Precision* and *Recall* values regarding the 'Laboratory' and 'Entrance pm' test sequences in Fig. 4.3. We can observe that the CIE L*a*b* and L*u*v* spaces produce the best results in both cases (the corresponding Pr/Rc curves are the highest). However, the relative performance of the other color systems is strongly different regarding the two videos. In the indoor scene, the grayscale and RGB segmentations are less efficient than the other ones, while regarding the 'Entrance pm' sequence, the performance of the chrominance spaces is prominently poor.
In the further tests, we will use the FM-measure (eq. 3.45). We summarized the FM rates in Fig. 4.4, regarding the test sequences. Also here, we can see that the CIE L*a*b* and L*u*v* spaces are the most efficient. As for the other color systems, in sequences containing dark shadows ('Entrance pm', 'Highway'), the 'chrominance spaces' produce poor results, while the gray, RGB and L*a*b*/L*u*v* results are similarly effective. If the shadow is brighter ('Entrance am', 'Laboratory'), the performance of the 'chrominance spaces' becomes reasonable, but the 'luminance spaces' are relatively poor. In the latter case, the color constancy of the chrominance channels seems to be more relevant than the luminance-darkening domain. We have also observed that the hue coordinate in HSV is very sensitive to the illumination artifacts (see also Fig. 4.1), thus the HSV space is more efficient in case of light-shadows. We give a summary about the relationship between the darkness of shadow and the performance of color spaces in Table 4.3, where 'darkness' is characterized by the mean of the grayscale-ψ_0 values of shadowed points.

4.4 MRF Segmentation with Different Color Spaces

The results in the previous section confirm that using the elliptical shadow domain defined by eq. 4.3, the CIE L*u*v* color space is the most efficient regarding the separation of shadowed and foreground pixels. However, those experiments

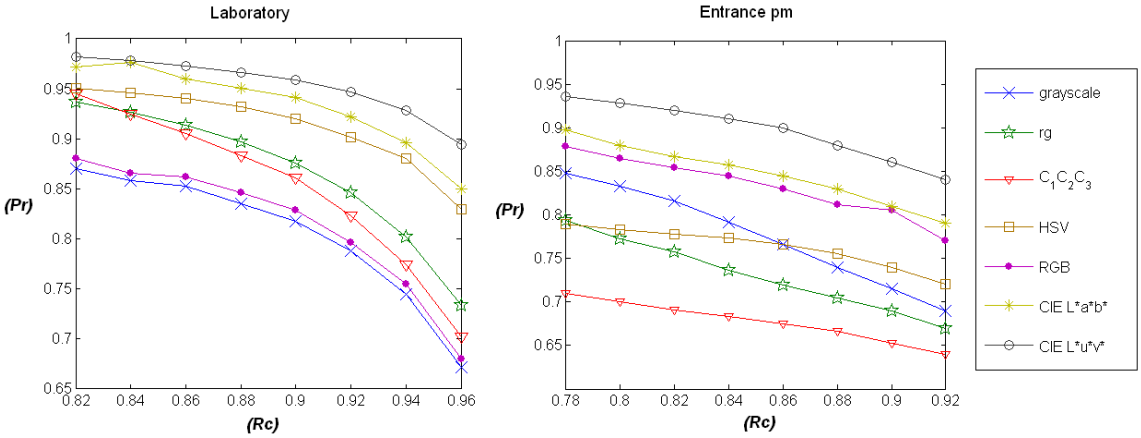


Figure 4.3: Evaluation of the deterministic model. Recall-precision curves corresponding to different parameter-settings on the 'Laboratory' and 'Entrance pm' sequences.

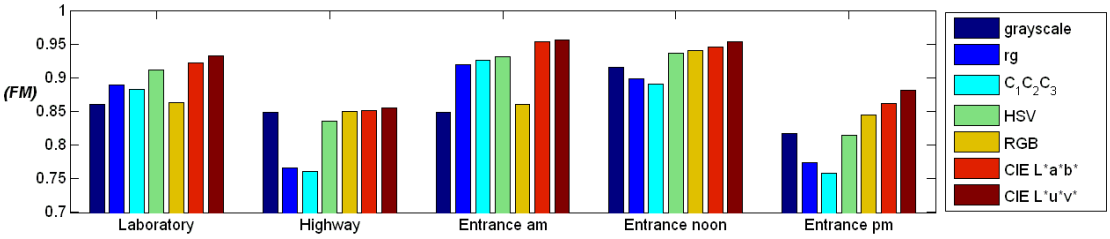


Figure 4.4: Evaluation of the deterministic model. *FM* coefficient (eq. 3.45) regarding different sequences

Table 4.3: Indicating the two most successful and the two less efficient color spaces regarding each test sequence, based on the experiments of Section 4.3.1 (For numerical evaluation see Fig. 4.3 and 4.4). To compare the scenarios, we also denote †the mean darkening factor of shadows in grayscale.

Video	Scene	Dark†	Worst	Best
Laboratory	indoor	0.73	gray, RGB	Luv, Lab
Entrance am	outdoor	0.50	gray, RGB	Luv, Lab
Entrance pm	outdoor	0.39	$C_1C_2C_3$, rg	Luv, Lab
Entrance noon	outdoor	0.35	$C_1C_2C_3$, rg	Luv, Lab
Highway	outdoor	0.23	$C_1C_2C_3$, rg	Luv, Lab

needed manually evaluated training data to set the parameters. In the following, we suit the above model to the adaptive Bayesian model-framework of Chapter 3, and show that the advantage of using the appropriate color space can be also measured directly in the applications.

Here, the optimal segmentation corresponds to the labeling defined by:

$$\widehat{\underline{\omega}} = \arg\min_{\underline{\omega}\in\Omega} \left\{ \sum_{s\in S} -\log p_{\omega(s)}(s) + \sum_{r,s\in S} \Theta\left(\omega(r),\omega(s)\right) \right\}. \tag{4.4}$$

The definition of the density functions $p_{\mathrm{bg}}(s)$ and $p_{\mathrm{fg}}(s)$ $s \in S$ is the same, as we defined in Chapter 3.

Before inserting our model into the MRF framework, we give to the shadow-classification step defined in Section 4.3, a probabilistic interpretation. We rewrite eq. (4.3): we match the current $\overline{\psi}(s)$ value of pixel s to a probability density function $f\left(\overline{\psi}(s)\right)$, and decide its class by:

$$\text{pixel } s \text{ is shadowed} \Leftrightarrow f\left(\overline{\psi}(s)\right) \geq t. \tag{4.5}$$

Based on the one dimensional marginal histograms in Fig. 4.1, we model $f\left(\overline{\psi}(s)\right)$ by a multi variate Gaussian density function, similarly to the CIE L*u*v* case introduced in Chapter 3. To keep the six-parameter shadow model, a diagonal covariance matrix will be used (i.e. the three element-mean value vector, and the three diagonal components of the covariance matrix should be defined). In this way, we model the variety of the $\overline{\psi}$ values observed in shadows, which variety is caused by camera noise, fine alterations in illumination, and differences in albedo and geometry of the different surface points. However, the changes in the different color components are considered to be independent exploiting that many color spaces (like CIE L*u*v*, CIE L*a*b*, HSV) have approximately uncorrelated basis [84]. As for the RGB space, this 'diagonal' approach is less accurate. However, we show later on that for most of the sequences the performance of this oversimplified RGB-model is also reasonable.

Based on Theorem 2 in Appendix A (page 132), the domains defined by eq. (4.3) and eq. (4.5) are equivalent, if f is a Gaussian density function (η):

$$f\left(\overline{\psi}(s)\right) = \eta(\overline{\psi}(s), \overline{\mu}_{\psi}, \overline{\overline{\Sigma}}_{\psi}) = \tag{4.6}$$

$$= \frac{1}{(2\pi)^{\frac{3}{2}}\sqrt{\det \overline{\overline{\Sigma}}_\psi}} \exp\left[-\frac{1}{2}(\overline{\psi}(s)-\overline{\mu}_\psi)^T \overline{\overline{\Sigma}}_\psi^{-1}(\overline{\psi}(s)-\overline{\mu}_\psi)\right] \tag{4.7}$$

with the following parameters:

$$\overline{\mu}_\psi = [a_0, a_1, a_2]^T, \quad \overline{\overline{\Sigma}}_\psi = \mathrm{diag}\{b_0^2, b_1^2, b_2^2\}, \tag{4.8}$$

while

$$t = (2\pi)^{-\frac{3}{2}}(b_0 b_1 b_2)^{-1} e^{-\frac{1}{2}}. \tag{4.9}$$

In the following, we use the previously defined probability density functions in the MRF model in a straightforward way:

$$p_{\mathrm{sh}}(s) = f\left(\overline{\psi}(s)\right). \tag{4.10}$$

The flexibility of this MRF model comes from the fact that we defined $\overline{\psi}(s)$ shadow descriptors for different color spaces differently in Section 4.2.

4.4.1 MRF Test Results

Fig. 4.5 shows the MRF-segmentation results of two frames from each test sequence using five color spaces: grayscale, $C_1C_2C_3$, HSV, RGB and CIE L*u*v*. (Note that in the experiments, the results of the CIE L*a*b* space have been very similar to the L*u*v* outputs, while the *rg* has worked similarly to $C_1C_2C_3$, thus we skip them in this comparison). We can observe that the CIE L*u*v* space outperforms the other ones significantly, while we get the largest errors with $C_1C_2C_3$, especially in the cases of sharp shadows. We find a typical problem regarding the HSV and RGB spaces: foreground 'glories' may appear around some dark shadowed parts due to the penumbra of cast shadow [74] and video compression. These erroneous areas correspond to shadows, but they are lighter than the central areas, thus they get out of the shadow domain in the feature space. On the other hand, the proposed probabilistic model removes these artifacts with the other color spaces.

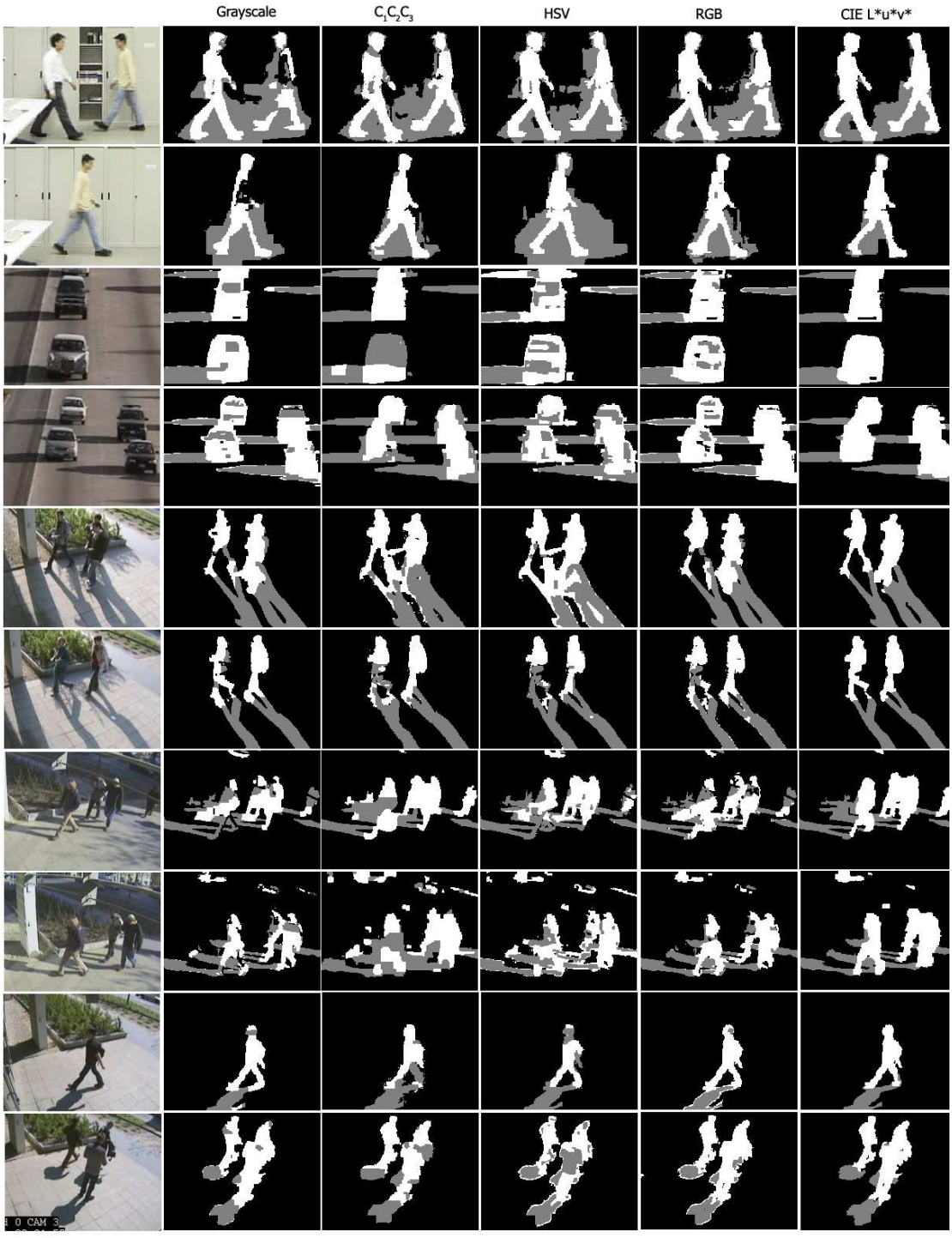


Figure 4.5: MRF segmentation results with different color models. Test sequences (up to down): rows 1-2 'Laboratory', rows 3-4: 'Highway', rows 5-6: 'Entrance am', rows 7-8: 'Entrance pm', rows 9-10: 'Entrance noon'.

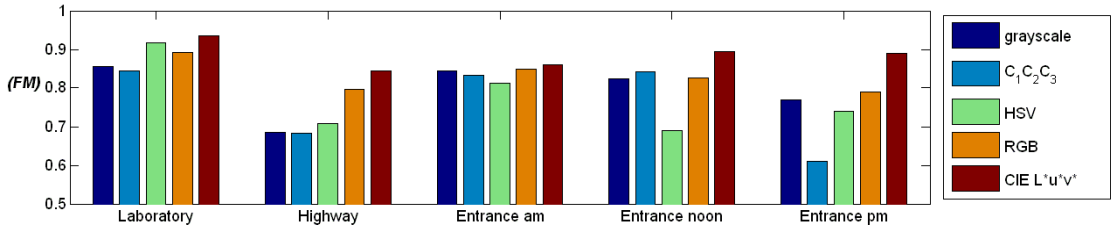


Figure 4.6: Evaluation of the MRF model. F^* coefficient regarding different sequences

Hereinafter, we perform quantitative evaluations using the MRF model. In Section 4.3.1, we measured purely the ability to discriminate foreground and shadowed pixels. Since the present model uses three classes and the goal is accurate foreground detection, we should also consider the confusion rate between foreground and background. However, similarly to Section 3.7.3 the crossover between shadow and background does not count for errors (both of them are non-foreground areas).

We observe in Fig. 4.6 the clear superiority of the CIE L*u*v* space. However, the relative performance of the color spaces does not show exactly the same tendencies as we have measured in Section 4.3.1. The differences between Fig. 4.4 and 4.6 are caused by effects of the composite foreground model, MRF neighborhood conditions and errors in parameter estimation, since the artifacts may appear differently in the different sequences. Therefore, we consider the numerical results from Section 4.3.1 to be more relevant to characterize the capabilities of the color spaces for shadow separation. However, the experiments of this section confirm that appropriate color space selection is also crucial in the applications, and the CIE L*u*v* space is preferred for this task.

4.5 Conclusion of the Chapter

This chapter has examined the color modeling problem of shadow detection. We have generalized the model framework of Chapter 3 for this task, which can work with different color spaces. With this model, we compared several well known color spaces, and observed that the appropriate color space selection is an

important issue regarding the segmentation results. We validated our method on five video shots, including well-known benchmark videos and real-life surveillance sequences, indoor and outdoor shots, which contain both dark and light shadows. Experimental results show that CIE L*u*v* color space is the most efficient both in the color based clustering of the individual pixels and in the case of Bayesian foreground-background-shadow segmentation.

Chapter 5

A Three-Layer MRF Model for Object Motion Detection in Airborne Images

In this chapter, a probabilistic model is proposed for automatic change detection on airborne images captured by moving cameras. To ensure robustness, an unsupervised coarse matching is used instead of a precise image registration. The challenge of the proposed model is to eliminate the registration errors, noise and the parallax artifacts caused by the static objects having considerable height (buildings, trees, walls etc.) from the difference image. The background membership of a given image point is described through two different features, and a novel three-layer Markov Random Field (MRF) model is introduced to ensure connected homogenous regions in the segmented image.

5.1 Introduction

Object motion detection is a key issue in aerial surveillance and exploitation [19]. An important preprocessing task is identifying the accurate silhouettes of moving objects or object-groups in urban roads carrying a great deal of traffic. In this chapter, we focus on this problem dealing with image pairs taken by moving airborne vehicles in consecutive moment. The task needs significantly different approaches in scene modeling and regarding the integration of different measurements from solutions used in the previous chapters. Thus, we begin with short overviews on these issues. The following notations will be used: G_1 and G_2 are two consecutive frames of the image sequence above the same pixel lattice S. The gray value of a given pixel $s \in S$ is $g_1(s)$ in the first image and $g_2(s)$ in the second one. A pixel is defined by a two dimensional vector containing its x-y coordinates: $s = [s_x, s_y]$, $s_x = 1...\mathcal{W}$, $s_y = 1...\mathcal{H}$.

5.1.1 Effects of Camera Motion in 3D Geometry

In Chapters 3 and 4, we have considered change detection as a purely 2D image segmentation problem marking the pixels with foreground, background or shadow labels. In fact, classifying a pixel s to 'background' means that a 3D scene point, which is projected to the s pixel of the image plane, corresponds to the background in the 3D environment. Using a static camera, it is not needed to model the relationship between the 2D image plane and the 3D world, since in a given pixel position, the same background surface point (with the same color) is permanently observable, unless it is occluded by a foreground object. On the other hand, in case of camera motion the static 'voxels' of the scene are projected to different pixel positions in the consecutive frames (see Fig. 5.1a). Finding the corresponding pixels in the images which represent the same 3D scene points is called *image registration*.

Although registration is one of the fundamental problems of image processing, we still find challenges in the context of the current application. Here, we try to demonstrate a few of them. An important approach is based on feature correspondence, where the goal is looking for corresponding pixels or other primitives such as edges, corners, contours, shape etc. in the images which are compared

[98][99][100][101][102]. Unfortunately, these procedures may fail at occlusion boundaries and within regions where the chosen primitives or features cannot be reliably detected. Although we can find methods focusing on the reduction of errors at object boundaries caused by occlusion [103][104], these approaches work with slightly different images used in stereo vision. On the other hand, taking the photos from a rapidly moving airborne vehicle may cause i.a. significant global offset and rotation between the consecutive frames. As for the synthesis of wide-baseline composite views, [105] presented a motion-based method for automatic registration of images in multi-camera systems. However, the latter method needs video flows recorded by static cameras, while in the present application we have only one image in each camera position.

In summary, using the existing techniques we must expect that feature matching presents correct pixel correspondences only for sparsely distributed feature points instead of matching the two frames completely. A possible way to handle this problem is searching for a global projective transform $\mathfrak{T}$ between the images. Thus, for a given pixel $r = [r_x, r_y]$ of the second frame, the corresponding pixel position $s = [s_x, s_y]$ in the first frame is approximated as: $s \approx \tilde{r} = \mathfrak{T}(r)$. Using that an arbitrary projective transform can be represented by a linear transform of homogeneous coordinates [111, p. 3], $\mathfrak{T}$ can be written in matrix form:

$$[\mathfrak{p}_x, \mathfrak{p}_y, \mathfrak{p}_w]^\mathsf{T} = \mathbb{T} \cdot [r_x, r_y, 1]^\mathsf{T} \tag{5.1}$$

where $\tilde{r}_x = \mathfrak{p}_x/\mathfrak{p}_w$, $\tilde{r}_y = \mathfrak{p}_y/\mathfrak{p}_w$, $\tilde{r} = [\tilde{r}_x, \tilde{r}_y]$ and $\mathbb{T}$ is the 3×3 homography matrix of transform $\mathfrak{T}$. Here $e_r = s - \tilde{r}$ is the error of approximation at pixel r. In the following, we denote by $\widetilde{G}_2$ the warped second image, which is obtained by applying $\mathfrak{T}$ for G_2, thus its pixel values are $\tilde{g}_2(r) = g_2\left(\mathfrak{T}^{-1}(r)\right)$.

The above defined procedure is called 2D image matching [106], and two main approaches are available for unsupervised estimation of $\mathfrak{T}$. Pixel correspondence based techniques estimate the optimal coordinate transform (e.g. homography) which maps the extracted feature points of the first image to the corresponding pixels identified by the feature tracker module in the second frame [102]. In global correlation methods, the goal is to find the parameters of a similarity [107] or affine transform [108] for which the correlation between the original first and

transformed second image is maximal. For computational purposes, global correlation methods work in the Fourier domain.

Although we find sophisticated ways to enhance the accuracy of the linear 2D mappings [109] (up to subpixel accuracy: [110]), these approaches only result in reasonable registration if the scene can be approximated by a flat surface [111, p. 8], the camera is very far from the ground plane or the camera motion is slight [109]. Otherwise, scene points out of the dominant plane (e.g. the plane of the roadway in a street scene) cause significantly different 2D displacements than calculated by the global projective transform. This effect is called *parallax distortion* (see Fig. 5.2).

To overcome this problem 'plane+parallax' (P+P) models have become widely used: we also follow this way in this chapter. Here, the images are registered up to a global 2D projective transform, thereafter the parallax is locally handled. As it is pointed out in [106], different environmental conditions and circumstances may raise essentially different challenges, thus 'P+P' methods can be onward divided into subcategories. An example for 'sparse parallax' is given in [112], which deals with very low altitude aerial videos captured from sparsely cultural scenes, where shape constancy constraints can be used together with global motion estimation. In that case, the '3Dness' of the scene is sparsely distributed containing a few moving objects, while the algorithm needs at least three frames from a video sequence. On the other hand, for scenarios being investigated in the current application a 'dense parallax' method should be developed, since both the 3D static objects and object motions may occur densely in the scene. Here, compared to [112], the frames are captured from higher altitude, and the parallax distortions after 2D registration usually cause errors of a few pixels. Consequently, if s and r are the corresponding pixels in G_1 and G_2, respectively, we assume that the magnitude of the 2D estimation error, $||e_r|| = ||s - \tilde{r}||$ is lower than a threshold parameter. In other words, for a given r, the corresponding pixel s should be searched in a given neighborhood of $\tilde{r}$ denoted by $H_{\tilde{r}}$. We will use rectangular neighborhoods with a fixed size (see Fig. 5.1b). Note that using $H_{\tilde{r}}$ is symmetric: for a given s in G_1, the corresponding pixel in the $\widetilde{G}_2$ transformed image, $\tilde{r}$, is in the rectangular neighborhood of s, H_s.

Since the length and orientation of the parallax error vectors e_r are different at

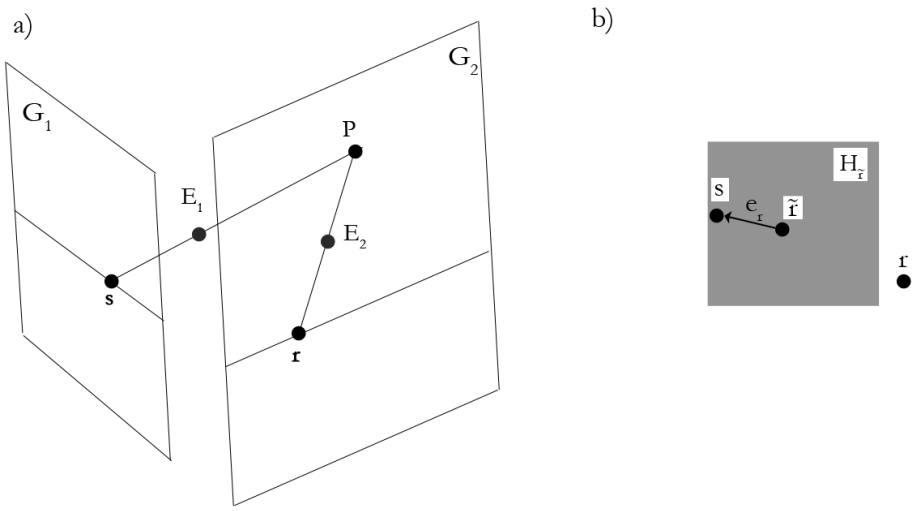


Figure 5.1: a) Illustrating the stereo problem in 3D. E_1 and E_2 are the optical centers of the cameras taking G_1 and G_2 respectively. P is a point in the 3D scene, s and r are its projections in the image planes. b) A possible arrangement of pixels r, $\tilde{r}$ and s; the 2D search region, $H_{\tilde{r}}$. e_r is the error of the projective estimation, $\tilde{r}$ for s.

the different r pixel positions, the above approach does not solve the exact pixel matching problem, which may still remain difficult. It can only be stated that s lies in the *search region* $H_{\tilde{r}}$ assigned to r, unless it corresponds to an object displacement. A key point in our approach is that the proposed model will not aim at finding the corresponding point pairs. We get around this problem in a statistical way, via a probabilistic description of the local *search regions*.
Note that the model does not exploit the well known epipolar constraint [111, p. 240]. As emphasized in [106], the performance of such approaches is very sensitive to find the accurate epipoles, which may fail if, besides camera motion, many independent object displacements are present in the scene.

As for further corresponding issues, in this chapter we search for object displacement in image pairs taken with approximately 1-2 second time difference. It should be emphasized here that this is a different task from processing high frame rate aerial videos [115][116], where the camera motion can be predicted based on previously processed frames.
As being noted earlier, we will introduce a two stage algorithm which consists of a coarse 2D image registration for camera motion compensation, and a parallax

error-eliminating step. From this point of view, this approach is similar to [117], where the authors assume that 2D registration errors mainly appear near sharp edges. Therefore, at locations where the magnitude of the gradient is large in both images, they consider that the differences of the corresponding pixel-values are caused with higher probability by registration errors than by object displacements. However, this method is less effective, if there are several small objects (containing several edges) in the scene, because the post processing may also remove some real objects, but it leaves errors in smoothed textured areas (e.g. group of trees, corresponding test results are shown in Section 5.9).

5.1.2 Approaches on Observation Fusion

Another important issue is related to feature selection. Scalar valued features may be weak to model complex classes appropriately, therefore integration of multiple observations has been intensively examined recently for different problems [40][45]-[48][82][118][119][120][121][122][123]. For observation fusion, we have already given an example in Chapter 3, where different color components and microstructural responses have been integrated in an $n(=4)$ dimensional feature vector, and for each class, the distribution of the features has been approximated by an n dimensional multinomial density function (for another similar fusion example, see [120]). However, this straightforward approach may fail regarding several practical problems: although the feature vector's one dimensional marginal distributions can be often modelled well with well-known densities (e.g. Gaussian, Beta, uniform, or a finite mixture of them), the joint distribution may be hard to express. As shown later this problem raises such challenge, since the first feature-dimension will be modelled by a Gaussian term while the second one follows a Beta distribution. Moreover, efficient methods for probability calculation and parameter estimation are only available for certain distributions. The correspondence between the feature components may be also difficult to model, or, at least, increases the number of free parameters (e.g. the Gaussian correlation matrix must be non-diagonal).

For the above reasons, multi-layer models have became popular nowadays [45] [46][47][48]. In this case, individual layers are assigned to the different feature

components (or to a group of components). Each layer's segmentation is directly influenced by its corresponding measurement component(s) and indirectly by features of the other layers. The inter-layer connections may achieve data interaction [45][46][48] (the inter-layer interactions also use the features' data and the segmentation labels directly) or label fusion [122][124] (the interactions use only the labels in the different layers). Usually, the right choice between these two approaches depends on the domain which we model. We show later that regarding the problem, which we investigate in this chapter, the label fusion is a more natural model.

In this chapter, we follow a Bayesian approach to tackle the above change detection problem. We derive features describing the background membership of a given image point in two independent ways, and develop a three-layer Bayesian labeling model to integrate the effects of the different features. We use a similar model structure to [45]-[48], which has two layers corresponding to the different observations, and a third one presenting the final foreground-background segmentation result. However, there are two essential differences: while in [45]-[48], the segmentation classes in the combined layer were constructed as the cross product of the classes at the observation layers, we use the same classes in each layer: foreground and background. On the other hand, we define the inter-layer connections also differently: in [45]-[48], the observation layers were directly connected with the segmentation layer via doubleton cliques, while we define connections between all three layers via cliques of node-triples.

5.2 2D Image Registration

In this section, we introduce briefly two approaches on coarse 2D image registration. Thereafter, we compare the methods on the images of our datasets, and we choose the most appropriate one to be the preprocessing step of our Bayesian labeling model.

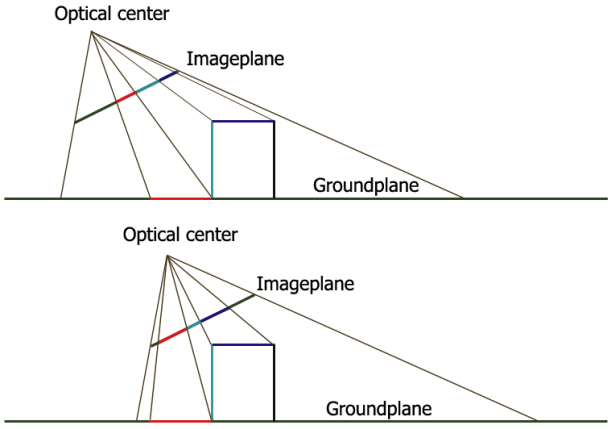


Figure 5.2: Illustration of the parallax effect, if a rectangular high object appears on the ground plane. We mark different sections with different colors on the ground and on the object, and plot their projection on the image plane with the same color. We can observe that the appearance of the corresponding sections is significantly different.

5.2.1 Pixel-Correspondence Based Homography Matching (PCH)

This approach consists of two consecutive steps. First, corresponding pixels are collected in the images (for an example see [113][114]), thereafter, the optimal coordinate transform is estimated between the elements of the extracted point pairs [102]. Therefore, only the first step is influenced directly by the observed image data, and the method may fail if the feature tracker produces poor result. On the other hand, we can obtain an arbitrary projective transform in this way. The set of the resulting point pairs contains several outliers, which are filtered out by the RANSAC algorithm [111, p. 290], while the optimal homography is estimated so that the back-projection error is minimized [125].

5.2.2 FFT-Correlation Based Similarity Transform (FCS)

Reddy and Chatterji [107] proposed an automatic and robust method for registering images, which are related via a similarity transform (translation, rotation and scaling). In this approach, the goal is to find the parameters of the similarity transform $\mathfrak{T}$ for which the correlation between G_1 and $\widetilde{G}_2 = \mathfrak{T}(G_2)$ is maximal. The method is based on the Fourier shift theorem. In the first step, we assume

that G_1 and G_2 images differ only in displacement, namely there exists an offset vector d^*, for which $g_1(s) = g_2(s+d^*) : \forall s, s+d^* \in S$. Let us denote with G_2^d the image we get by shifting G_2 with offset d. In this case, $d^* = \text{argmax}_d R(d)$, where R is the correlation map: $R(d) = \text{Corr}\{G_1, G_2^d\}$. R can be determined efficiently in the Fourier domain. Let $\mathcal{F}_1$ and $\mathcal{F}_2$ be the Fourier transforms of the images G_1 and G_2. We define the Cross Power Spectrum (CPS) by:

$$\text{CPS(i, k)} = \frac{\mathcal{F}_1(i,k) \cdot \overline{\mathcal{F}}_2(i,k)}{|\mathcal{F}_1(i,k) \cdot \overline{\mathcal{F}}_2(i,k)|} = e^{j2\pi(d_x i + d_y k)}, \tag{5.2}$$

where $\overline{\mathcal{F}}_2$ means the complex conjugate of $\mathcal{F}_2$. Finally, the inverse Fourier transform of the CPS is equal with the correlation map R [107].

The Fourier shift theorem also offers a way to determine the angle of the rotation. Assume that G_2 is a translated and rotated replica of G_1, where the translation vector is o and the angle of rotation is α_0. It can be shown that considering $|\mathcal{F}_1|$ and $|\mathcal{F}_2|$ as images, $|\mathcal{F}_2|$ is the purely rotated replica of $|\mathcal{F}_1|$ with angle α_0. On the other hand, rotation in the Cartesian coordinate system is equivalent to a translational displacement in the polar representation [107], which can be calculated similarly to the determination of d^*.

The scaling factor of the optimal similarity transform may be retrieved in an analogous way [107].

In summary, we can determine the optimal similarity transform $\mathfrak{T}$ between the two images based on [107], and derive the (coarsely) registered second image, $\widetilde{G}_2$.

5.2.3 Experimental Comparison of PCH and FCS

The PCH and FCS algorithms have been tested on our test image pairs. Obviously, both gives only a coarse registration, which is inaccurate and is disturbed by parallax artifacts. In fact, FCS is less effective if the projective distortion between the images is significant. The weak point of PCH appears if the object motion is dense, thus a lot of point pairs may be in moving objects, and the automatic outlier filtering may fail, or at least, the homography estimation becomes inaccurate.

In our test database, the latter artifacts are more significant, since the corners of the several moving cars present dominant features for the Lucas-Kanade tracker.

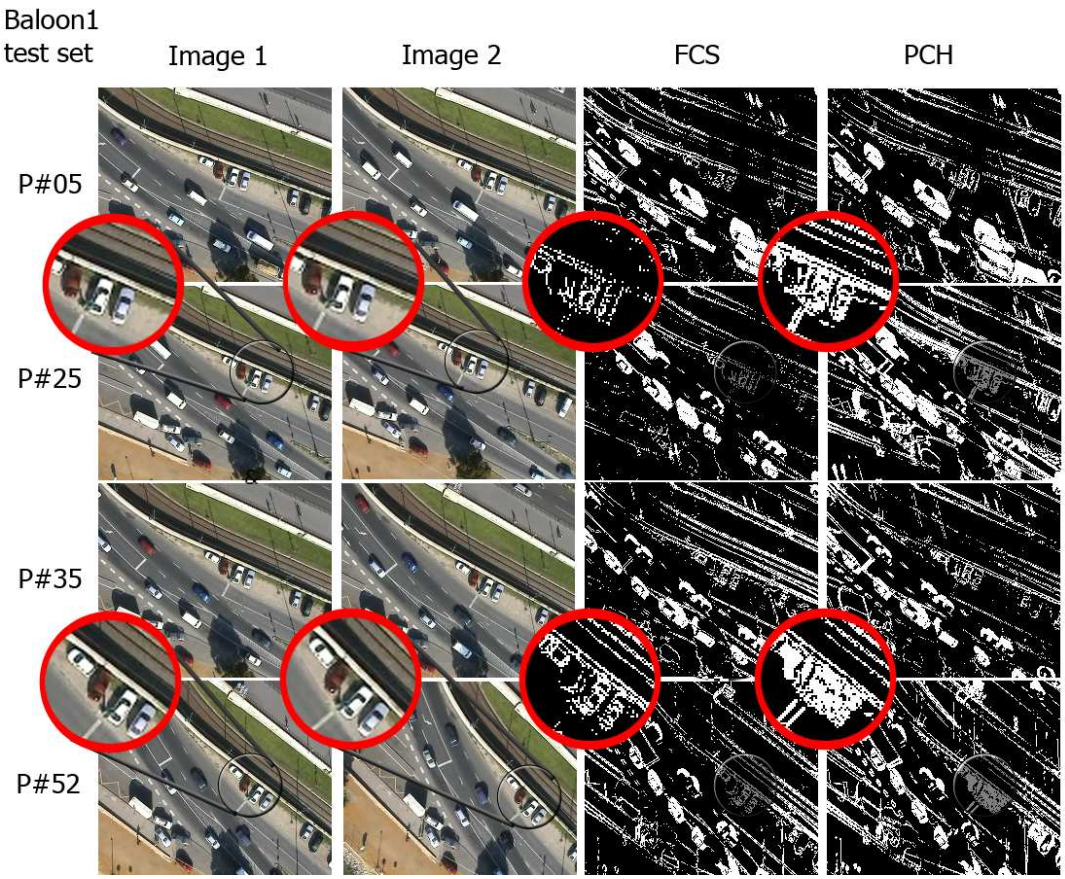


Figure 5.3: Qualitative illustration of the coarse registration results presented by the FFT-Correlation based similarity transform (FCS), and the pixel-correspondence based homography matching (PCH). In col 3 and 4, we find the thresholded difference of the registered images. Both results are quite noisy, but using FCS, the errors are limited to the static object boundaries, while regarding P#25 and P#52 the PCH registration is erroneous. Our Bayesian post processing is able to remove the FCS errors, but it cannot deal with the demonstrated PCH gaps.

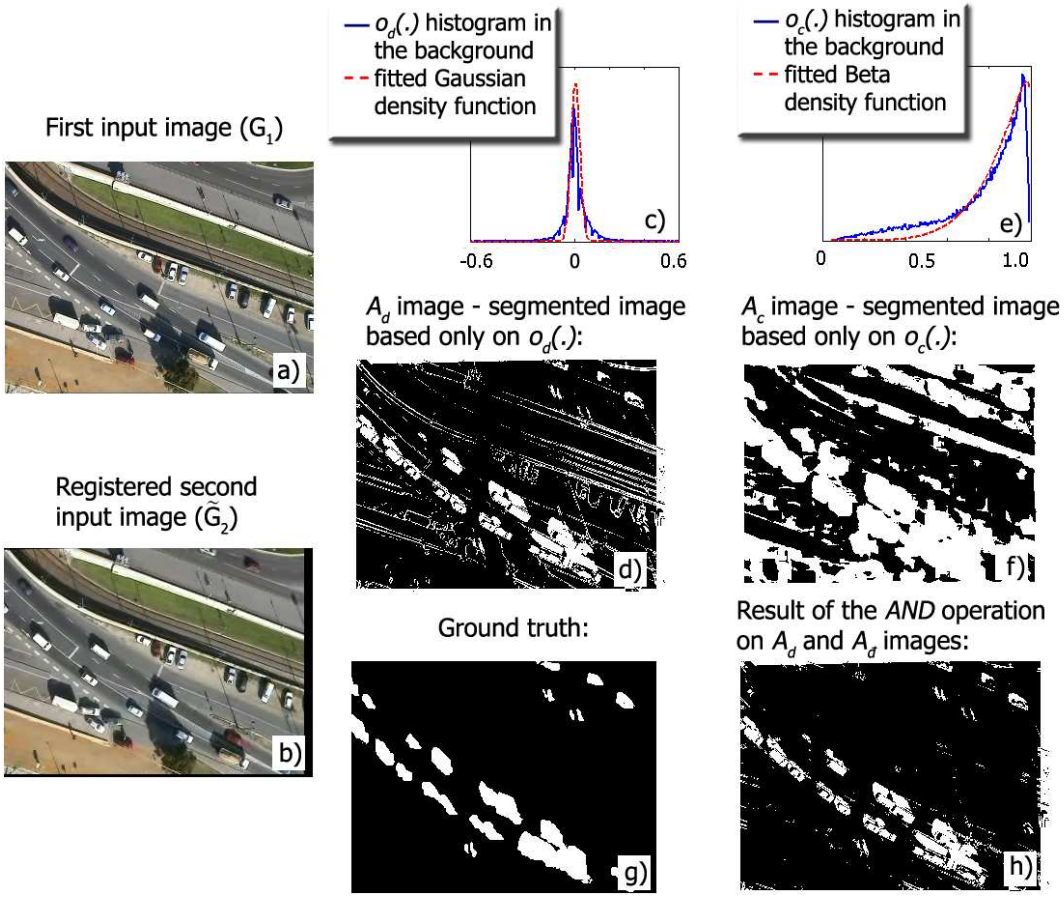


Figure 5.4: Feature selection. Notations are in the text of Section 5.4.

Some corresponding results are presented in Fig. 5.3. We can observe that using FCS, the error-appearances are limited to the static objects boundaries, while regarding two out of the four frames, the PCH registration is highly erroneous. We note that the Bayesian post processing, which will be proposed later in this chapter, can remove the FCS errors, but it is unable to deal with the large PCH gaps.

For the above mentioned reasons, we will use the FCS method for preliminary registration in the following part of this chapter, however, in other test scenes it can be replaced with PCH in a straightforward way.

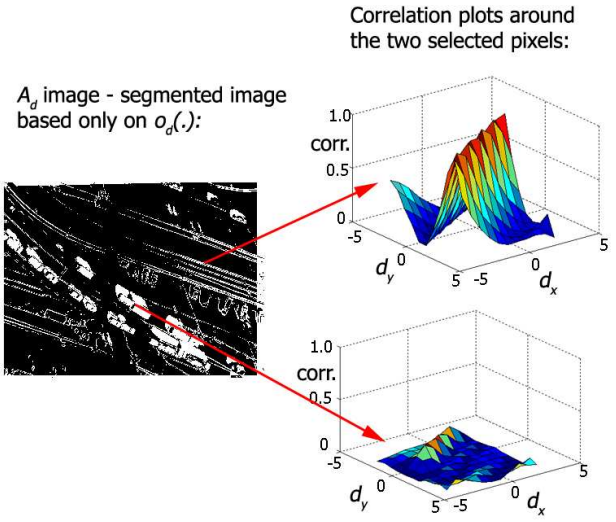


Figure 5.5: Plot of the correlation values over the search window around two given pixels. The upper pixel corresponds to a parallax error in the background, while the lower pixel is part of a real object displacement.

5.3 Change Detection with 3D Approach

The desired output of the method is a *dense* change map, i.e. a label should be assigned to each pixel $s \in S$ from the binary label-set: $\Phi = \{\text{fg}, \text{bg}\}$, corresponding to the two classes: foreground (fg) and background (bg), where foreground means object displacement.

To interpret the segmentation task, we must consider the problem in 3D (see Fig. 5.1a). We define the labeling in the following way:

Definition 9 *(**Foreground in case of moving camera**) Pixel s belongs to foreground (*fg*), if the 3D scene point P, which is projected to pixel s in the first frame (G_1), changes its position in the scene's (3D) world coordinate system or is covered by a moving object by the time taking the second image (G_2). Otherwise, pixel s belongs to the background (*bg*).*

5.4 Feature Selection

In this section, we introduce the feature selection using an airborne photo pair.[1] Taking a probabilistic approach, first we extract features, and then consider the class labels to be random processes generating the features according to different distributions.

The first feature is the gray level difference of the corresponding pixels in the registered images:

$$o_d(s) = \tilde{g}_2(s) - g_1(s). \tag{5.3}$$

Although due to the imperfect registration, $g_1(s)$ and $\tilde{g}_2(s)$ usually do not represent exactly the same scene point, we can use the spatial redundancy in the images. Since the pixel levels in a homogenous surface are similar, the occurring $o_d(.)$ feature values in the background can be statistically characterized by a random variable with a given mean value μ (i.e. global intensity offset between the images) and deviation σ (uncertainty due to camera noise and registration errors). We validate this feature through experiments (Fig. 5.4c): if we plot the histogram of $o_d(s)$ values corresponding to manually marked background points, then we can observe that a Gaussian approximation is reasonable:

$$P(o_d(s)|\text{bg}) = N(o_d(s), \mu, \sigma) =$$

$$= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(o_d(s)-\mu)^2}{2\sigma^2}\right). \tag{5.4}$$

On the other hand, any $o_d(s)$ value may occur in the foreground, hence the foreground class is modeled by a uniform density:

$$P(o_d(s)|\text{fg}) = \begin{cases} \frac{1}{b_d - a_d}, & \text{if } \; o_d(s) \in [a_d, b_d] \\ 0 & \text{otherwise.} \end{cases} \tag{5.5}$$

Next, we demonstrate the limitations of this feature. After supervised estimation of the distribution parameters, we derive the A_d image in Fig. 5.4d as the maximum likelihood estimate: the label of s is

$$\text{argmax}_{\phi \in \{\text{fg,bg}\}} P(o_d(s)|\phi). \tag{5.6}$$

[1] We have also observed similar tendencies regarding the other test images, provided by the ALFA project.

We can observe that several false positive foreground points are detected, however, these artifacts are mainly limited to textured 'background' areas and to the surface boundaries. In these cases, the $g_1(s)$ and $\tilde{g}_2(s)$ values correspond to different surfaces in the 3D scene, so $o_d(s)$ may have an arbitrary value, which appears as an outlier with respect to the previously defined Gaussian distribution.
For the above reasons, we introduce a second feature. Denote the rectangular neighborhood of s, with a fixed window size, by $\Lambda_1(s)$ in G_1, and by $\Lambda_2(s)$ in $\widetilde{G}_2$. Assuming the presence of errors of a few pixels, if s is in the background, we can usually find an $d_s = [d_x, d_y]$ offset vector, for which $\Lambda_1(s)$ and $\Lambda_2(s + d_s)$ are strongly correlated. Here, we use the normalized cross correlation as similarity measure: namely, correlation of two image parts $\mathcal{A} = \{\alpha_1, \alpha_2, \dots \alpha_n\}$ and $\mathcal{B} = \{\gamma_1, \gamma_2, \dots \gamma_n\}$, where (α_i, γ_i) are the values of the corresponding pixels, $\overline{\alpha}$ and $\overline{\gamma}$ being the mean values in the images, is computed by:

$$\mathrm{Corr}(\mathcal{A}, \mathcal{B}) = \frac{\sum_{i=1}^{n} (\alpha_i - \overline{\alpha})(\gamma_i - \overline{\gamma})}{\sqrt{\sum_{i=1}^{n} (\alpha_i - \overline{\alpha})^2 \sum_{i=1}^{n} (\gamma_i - \overline{\gamma})^2}}. \tag{5.7}$$

In Fig 5.5, we plot the correlation values between $\Lambda_1(s)$ and $\Lambda_2(s + d_s)$ for different values of the offset d_s around two given pixels, which are marked with the beginning of the arrows. The upper pixel corresponds to a parallax error in the background, while the lower one is part of a real object displacement. The correlation plot has high peak only in the upper case. We use $o_c(s)$, the maxima in the local correlation function around pixel s as second feature:

$$o_c(s) = \max_{(s+d_s) \in H_s} \mathrm{Corr}\{\Lambda_1(s), \Lambda_2(s + d_s)\}, \tag{5.8}$$

where the search window of the offset d_s, is equal to rectangle H_s defined in Section 5.1.1.
By examining the histogram of $o_c(s)$ values in the background (Fig 5.4e), we find that it can be approximated by a beta density function:

$$P\left(o_c(s)|\mathrm{bg}\right) = B\left(o_c(s), \beta_1, \beta_2\right), \tag{5.9}$$

where

$$B(c, \beta_1, \beta_2) = \begin{cases} \frac{\Gamma(\beta_1+\beta_2)}{\Gamma(\beta_1)\Gamma(\beta_2)} c^{\beta_1 - 1}(1-c)^{\beta_2 - 1}, & \text{if } \; c \in (0,1) \\ 0 & \text{otherwise} \end{cases} \tag{5.10}$$

$$\Gamma(\alpha) = \int_0^\infty \lambda^{\alpha-1} e^{-\lambda} d\lambda. \tag{5.11}$$

As for the foreground class we will use a uniform probability $P(o_c(s)|\text{fg})$ with a_c and b_c parameters, similarly to eq. 5.5.

We see in Fig. 5.4f (A_c image) that the $o_c(.)$ descriptor alone also causes poor result: similarly to the gray level difference, a lot of false alarms are present. However, the errors appear at different locations compared to the previous case. First of all, due to the block matching, the spatial resolution of the segmented map decreases, and the blobs of object displacements became erroneously large. Secondly in homogenous areas, the variance of the pixel values in the blocks to compare may be very low, thus the normalized correlation coefficient will be highly sensitive to noise. In summary, the $o_d(.)$ and $o_c(.)$ features may cause quite a lot of false positive foreground points, however the rate of false negative detection is low in both cases: they appear only at location of background-colored object parts, and they can be partially eliminated by spatial smoothing constraints discussed later in this chapter. Moreover, examining the gray level difference, $o_d(s)$, results usually in a false positive decision if the neighborhood of s is textured, but in that case the decision based on the correlation peak value, $o_c(s)$, is usually correct. Similarly, if $o_c(s)$ votes erroneously, we can usually trust in the hint of $o_d(s)$.

Consequently, if we consider A_d and A_c as a Boolean lattice, where 'true' corresponds to the foreground label, the logical AND operation on A_d and A_c improves the results significantly (Fig. 5.4h). We note that this classification is still quite noisy, although in the segmented image, we expect connected regions representing the motion silhouettes. Here again, Markov Random Fields (MRFs) will be used to ensure the contextual classification. However, our case is particular: we have two weak features, which present two different (poor) segmentations, while the final foreground-background clustering depends directly on the labels of the weak segmentations. To decrease noise, we must prescribe, that both the weak and the final segmentations must be 'smooth'. For the above reasons, we introduce a novel segmentation model in Section 5.5.

Note that the limitation of the $o_c(.)$ descriptor is caused by the denominator

term in the normalized correlation expression (eq. 5.7). Here, we offer as alternative descriptor a non-normalized similarity factor, namely, the simple squared difference. For $\mathcal{A} = \{\alpha_1, \alpha_2, \dots \alpha_n\}$ and $\mathcal{B} = \{\gamma_1, \gamma_2, \dots \gamma_n\}$:

$$\text{Sqdiff}(\mathcal{A}, \mathcal{B}) = \sum_{i=1}^{n} (\alpha_i - \gamma_i)^2, \tag{5.12}$$

and denote by $o_c^*(s)$ the minimal Sqdiff value around s, while A_c^* is the segmented image based on $o_c^*(.)$. We show some comparative experimental results for features A_c and A_c^* in Fig. 5.6. We can observe that in itself, A_c^* has significantly better quality than A_c, but $o_c(.)$ is a better complementary feature of $o_d(.)$, and the $A_d - A_c$ joint segmentation is better than the clustering based on $A_d - A_c^*$.

5.5 Multi-Layer Segmentation Model

In the proposed approach, we construct a Markov Random Field (MRF) model on a graph $\mathcal{G}$ whose structure is shown in Fig. 5.7. In the previous section, we segmented the images in two independent ways, and derived the final result by a label fusion using the two segmentations. Therefore, we arrange the nodes of $\mathcal{G}$ into three layers S^d, S^c and S^*, each layer has the same size as the image lattice S. We assign to each pixel $s \in S$ a unique node in each layer: e.g. s^d is the node corresponding to pixel s on the layer S^d. We denote $s^c \in S^c$ and $s^* \in S^*$ similarly.

We introduce a labeling process, which assigns a label $\omega(.)$ to all nodes of $\mathcal{G}$ from the label-set: $\Phi = \{\text{fg}, \text{bg}\}$. The labeling of S^d/S^c corresponds to the segmentation based on the $o_d(.)/o_c(.)$ feature, respectively; while the labels at the S^* layer present the final change mask. A global labeling of $\mathcal{G}$ is

$$\underline{\omega} = \left\{\omega(s^i) | s \in S, i \in \{d, c, *\}\right\}. \tag{5.13}$$

In the proposed model, the labeling of an arbitrary node depends directly on the labels of its neighbors (MRF property). For this reason, we must define the neighborhoods (i.e. the connections) in $\mathcal{G}$ (see Fig. 5.7). To ensure the smoothness of the segmentations, we put connections within each layer between

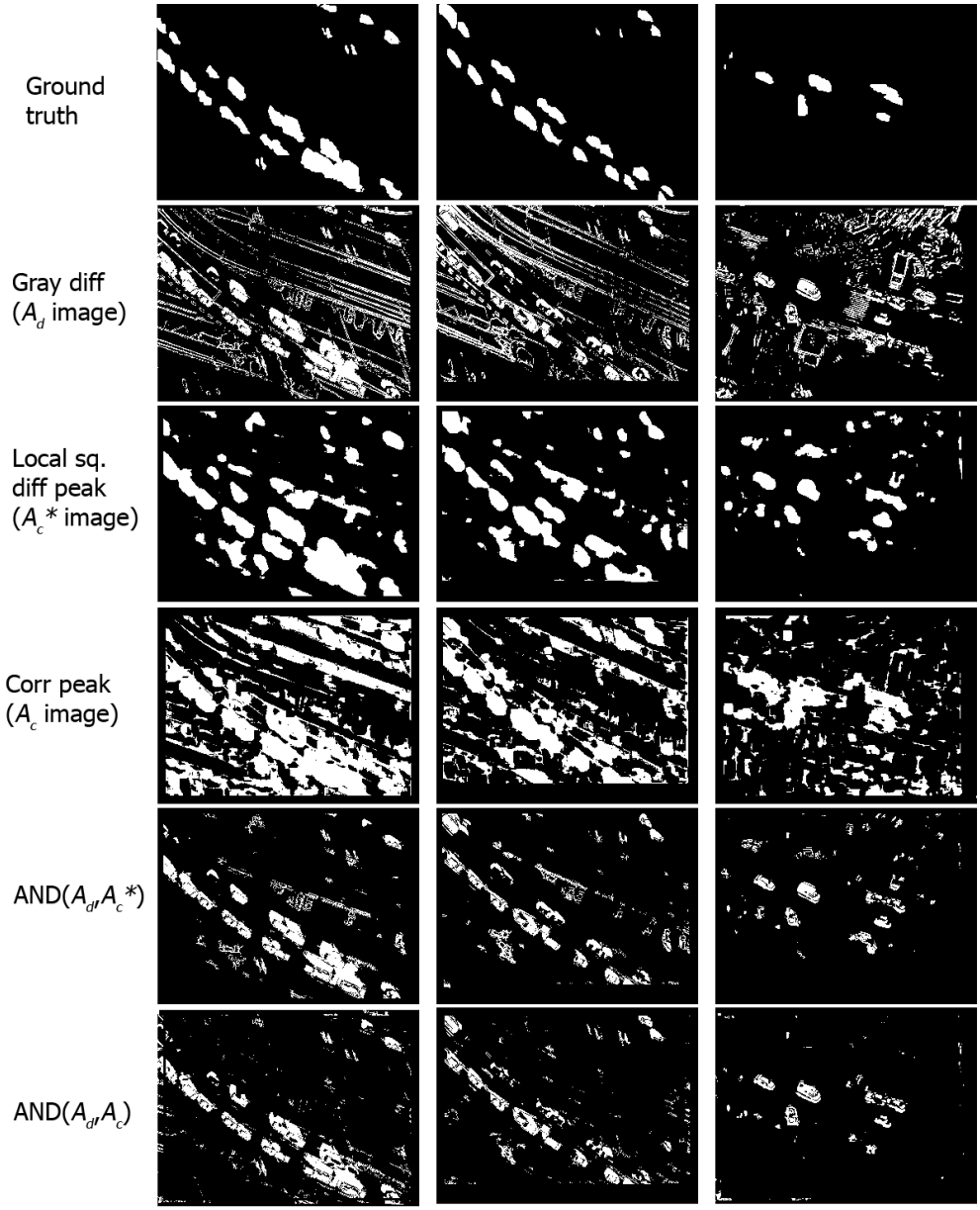


Figure 5.6: Qualitative comparison of the 'sum of local squared differences' (A_c^*) and the 'normalized cross correlation' (A_c) similarity measures with our label fusion model. In itself, the segmentation A_c^* is significantly better than A_c, but after fusion with A_d, the normalized cross correlation outperforms the squared difference.

node pairs corresponding to neighboring pixels of the image lattice S.[1] On the other hand, the nodes at different layers corresponding to the same pixel must interact in order to produce the fusion of the two different segmentations labels in the S^* layer. Hence, we introduce 'inter-layer' connections between nodes s^i and s^j: $\forall s \in S;\ i,j \in \{d,c,*\},\ i \neq j$. Therefore, the graph has doubleton 'intra-layer' cliques (their set is $\mathcal{C}_2$) which contain pairs of nodes, and 'inter-layer' cliques ($\mathcal{C}_3$) consisting of node-triples. We also use singleton cliques ($\mathcal{C}_1$), which are one-element sets containing the individual nodes: they will link the model and the local observations. Hence, the set of cliques is $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3$.
Denote the observation process by

$$\mathcal{O} = \left\{ o(s^i) |\ s \in S, i \in \{d,c\} \right\}, \tag{5.14}$$

where $o(s^d) = o_d(s)$, $o(s^c) = o_c(s)$, $O = S^d \cup S^c$.
Our goal is to find the optimal labeling $\underline{\widehat{\omega}}$, which maximizes the a posterior probability $P(\underline{\omega}|\mathcal{O})$ that is a maximum a posteriori estimate (MAP) defined by eq. 2.13:

$$\underline{\widehat{\omega}} = \text{argmax}_{\underline{\omega} \in \Omega} P(\underline{\omega}|\mathcal{O}). \tag{5.15}$$

where Ω denotes the set of all the possible global labelings. Based on the Hammersley-Clifford Theorem (theorem 1) the a posterior probability of a given labeling follows a Gibbs distribution:

$$P(\underline{\omega}|\mathcal{O}) = \frac{1}{Z} \exp\left(-\sum_{C \in \mathcal{C}} V_C(\underline{\omega}_C) \right), \tag{5.16}$$

where V_C is the *clique potential* of $C \in \mathcal{C}$, which is 'low' if $\underline{\omega}_C$ (the label- sub-configuration corresponding to C) is semantically correct, 'high', if not. Z is a normalizing constant, which does not depend on $\underline{\omega}$.
In the following part of this section, we define the clique potentials. We refer to a given clique as the set of its nodes (in fact, each clique is a subgraph of $\mathcal{G}$), e.g. we denote the doubleton clique containing nodes s^d and r^d with $\{s^d, r^d\}$.
The observations affect the model through the singleton potentials. As we stated

[1] We use first order neighborhoods in S, where each pixel has 4 neighbors.

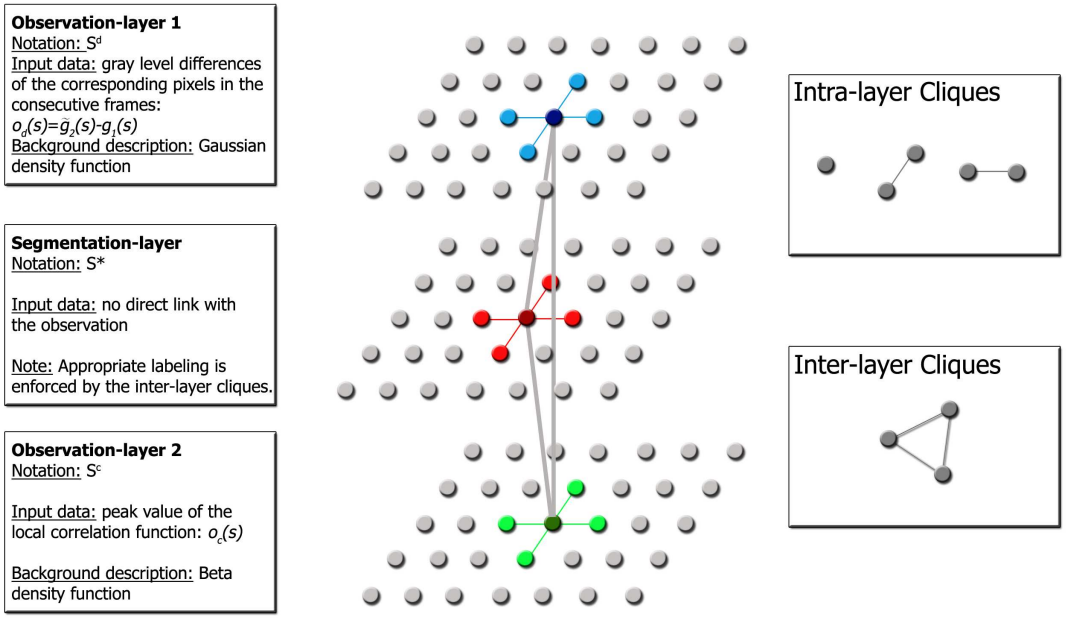


Figure 5.7: Summary of the proposed three layer MRF model

previously, the labels in the S^d and S^c layers are directly influenced by the $o_d(.)$ and $o_c(.)$ values, respectively, $\forall s \in S$:

$$V_{\{s^d\}}\left(\omega(s^d)\right) = -\log P(o_d(s)|\omega(s^d)), \tag{5.17}$$

$$V_{\{s^c\}}\left(\omega(s^c)\right) = -\log P(o_c(s)|\omega(s^c)), \tag{5.18}$$

where the probabilities that the given foreground or background classes generate the $o_d(s)$ or $o_c(s)$ observation, were already defined in Section 5.4 by eq. 5.4, 5.5 and 5.9.

On the other hand, the labels at S^* have no direct links with these measurements:

$$V_{\{s^*\}}\left(\omega(s^*)\right) = 0. \tag{5.19}$$

In order to get a smooth segmentation in each layer, the potential of an intra-layer clique $C_2 = \{s^i, r^i\} \in \mathcal{C}_2$, $i \in \{d, c, *\}$ has the following form [42]:

$$V_{C_2} = \Theta\left(\omega(s^i), \omega(r^i)\right) = \begin{cases} -\delta^i & \text{if } \omega(s^i) = \omega(r^i) \\ +\delta^i & \text{if } \omega(s^i) \neq \omega(r^i) \end{cases} \tag{5.20}$$

with a constant $\delta^i > 0$.

As we concluded from the experiments in Section 5.4, a pixel is likely generated by the background process, if and only if in the S^d and S^c layers, at least one corresponding node has the label 'bg'. We introduce the following indicator function:

$$I_{\text{bg}} : S^d \cup S^c \cup S^* \to \{0,\ 1\}, \tag{5.21}$$

where

$$I_{\text{bg}}(q) = \begin{cases} 1 & \text{if } \omega(q) = \text{bg} \\ 0 & \text{if } \omega(q) \neq \text{bg}. \end{cases} \tag{5.22}$$

With this notation the potential of an inter-layer clique $C_3 = \{s^d, s^c, s^*\}$ is:

$$V_{C_3}(\underline{\omega}_{C_3}) = \varsigma(\omega(s^d), \omega(s^c), \omega(s^*)) = \begin{cases} -\varrho & \text{if } I_{\text{bg}}(s^*) = \max\left(I_{\text{bg}}(s^d), I_{\text{bg}}(s^c)\right) \\ +\varrho & \text{otherwise.} \end{cases} \tag{5.23}$$

with $\varrho > 0$.

Therefore, the optimal MAP labeling $\widehat{\underline{\omega}}$, which maximizes $P(\widehat{\underline{\omega}}|\mathcal{O})$ (hence minimizes $-\log P(\widehat{\underline{\omega}}|\mathcal{O})$) can be calculated as:

$$\widehat{\underline{\omega}} = \arg\min_{\underline{\omega}\in\Omega} \Bigg\{ -\sum_{s\in S} \log P(o_d(s)|\omega(s^d)) - \sum_{s\in S} \log P(o_c(s)|\omega(s^c)) \\ + \sum_{C_2\in\mathcal{C}_2} V_{C_2}\left(\underline{\omega}_{C_2}\right) + \sum_{C_3\in\mathcal{C}_3} V_{C_3}\left(\underline{\omega}_{C_3}\right) \Bigg\}. \tag{5.24}$$

The above energy minimization is performed with simulated annealing. (See Section 5.7 for details.) The final segmentation is taken as the labeling of the S^* layer.

5.6 Parameter Settings

In the following we define a possible grouping of the free parameters in the process: the first group is related to the correlation calculation and the second one to the potential functions.

5.6.1 Parameters Related to the Correlation Window

The correlation window defined in Section 5.4 should not be significantly larger than the expected objects to ensure low correlation between an image part which contains an object and one from the same 'empty' area. We use a 9×9 pixel window in our experiments for images of size 320×240.

The *maximal offset* of the search window determines maximal parallax error, which can be compensated by the method. We note that in homogenous background, object motions with less than the offset parameter can be falsely detected as parallax errors. Therefore, at the given resolution, we use ± 3 pixels for the maximal offset, and detect the moving objects whose displacement is larger.

5.6.2 Parameters of the Potential Functions

The singleton potentials are values of conditional density functions as it was defined in Section 5.4 by eq. 5.4, 5.5 and 5.9.

The Gaussian mean parameter (μ) corresponds to the average gray value difference between the images caused by quick changes in the lighting conditions or in the camera white balance, the deviation (σ) depends on the noise. These parameters can be estimated by creating a histogram for A_d difference image, and estimating the parameters of the area close to the main peak of this histogram.

The Beta distribution parameters and the uniform values are determined from one image to another one by trial and error. We use $\beta_1 = 4.5$, $\beta_2 = 1$ and $a_c = 0, b_c = 1$ for all image pairs (with the assumption that the gray values of the images are normalized between 0 and 1), while the optimal value of a_d and b_d shows significant differences in the different image sets. Using the '2σ-rule' proved to be a good initial approximation, namely $\frac{1}{b_d - a_d} = N(\mu + 2\sigma, \mu, \sigma)$. Here, following the Chebyshev inequality [43]:

$$P(|o_d(s) - \mu| > 2\sigma \mid \omega(s) = \mathrm{bg}) < \frac{1}{4}. \tag{5.25}$$

The parameters of the intra-layer potential functions, δ^d, δ^c and δ^* influence the size of the connected blobs in the segmented images. Higher δ^i ($i \in \{d, c, *\}$) values result in more compact foreground regions, however, fine details of the silhouettes may be distorted that way. We have used in each layer $\delta^i = 0.7$ for

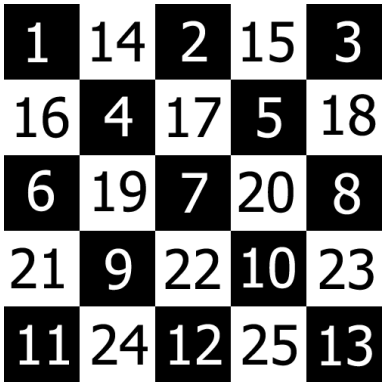


Figure 5.8: Ordinal numbers of the nodes in a 5×5 layer according to the 'checkerboard' scanning strategy

test images with relatively small objects (e.g. 'balloon1' and 'Budapest' sets, introduced in Section 5.9.1), while $\delta^i = 1.0$ have been proved to be appropriate regarding images captured from lower altitude ('balloon2').

Parameter ϱ of the inter-layer potentials determines the strength of the relationship between the segmentation of the different layers. We have used $\varrho = \delta^*$: this choice gives the same importance to the intra-layer smoothness and the inter-layer label fusion constraints.

5.7 MRF Optimization

We have used the Modified Metropolis (MMD) [53] algorithm in this chapter, since we have found it is nearly as efficient but significantly quicker than the original Metropolis [52] in this application. We give the detailed pseudo code of the MMD adapted to the three layer segmentation model in Fig. 5.9. If we use ICM with our model [54], its processing time is negligible compared to the other parts of the algorithm, in exchange for some degradation in the segmentation results.

1. Pick up randomly an initial configuration $\underline{\omega}$, with $k := 0$ and $T := T_0$.
2. Denote by $|Q|$ the number of nodes in the three-layer model. Assign to each node a unique ordinal number between 1 and $|Q|$, applying the 'checkerboard' scanning strategy (Fig. 5.8) for the consecutive layers. Let $j := 1$.
3. Let q the j^{th} node, $i \in \{d, c, *\}$ is the layer which contains q, while $s \in S$ is the corresponding pixel in the image lattice: $q = s^i$.
4. Denote the label of q in $\underline{\omega}$ by $\omega(q)$. Flip the label of q and denote it by $\breve{\omega}(q)$.
5. Compute ΔU as follows:

$$\Delta U := \Delta U_1 + \Delta U_2 + \Delta U_3, \quad \text{where}$$

a. Calculate ΔU_1 as:

$$\Delta U_1 := \begin{cases} \log P\left(d(s)|\omega(q)\right) - \log P\left(d(s)|\breve{\omega}(q)\right) & \text{if } \; i = d, \\ \log P\left(c(s)|\omega(q)\right) - \log P\left(c(s)|\breve{\omega}(q)\right) & \text{if } \; i = c, \\ 0 & \text{if } \; i = * \end{cases}$$

b. Using eq. 5.20, calculate ΔU_2 as:

$$\Delta U_2 := \sum_{r \in \Phi_s} \Theta\left(\breve{\omega}(s^i), \omega(r^i)\right) - \Theta\left(\omega(s^i), \omega(r^i)\right).$$

c. Denote by $\varsigma_0 = \varsigma\left(\omega(s^d), \omega(s^c), \omega(s^*)\right)$ (eq. 5.23). Calculate ΔU_3 as:

$$\Delta U_3 := \begin{cases} \varsigma\left(\breve{\omega}(q), \omega(s^c), \omega(s^*)\right) - \varsigma_0 & \text{if } \; i = d, \\ \varsigma\left(\omega(s^d), \breve{\omega}(q), \omega(s^*)\right) - \varsigma_0 & \text{if } \; i = c, \\ \varsigma\left(\omega(s^d), \omega(s^c), \breve{\omega}(q)\right) - \varsigma_0 & \text{if } \; i = * \end{cases}$$

9. Update the label of q:

$$\omega(q) := \begin{cases} \breve{\omega}(q) & \text{if } \; \log \tau \leq -\frac{\Delta U}{T}, \\ \omega(q) & \text{otherwise.} \end{cases}$$

where τ is a constant threshold ($\tau \in (0, 1)$).

10. If $j < |Q|$: $\{j := j + 1$ and goto step 3.$\}$
11. Set $T := T_{k+1}$, $k := k + 1$, $j := 1$ and goto step 3, until convergence (i.e. the number of the changed labels between the k^{th} and $(k+1)^{\text{th}}$ iteration is lower than a threshold.)

Figure 5.9: Pseudo-code of the Modified Metropolis algorithm used for the current task. Corresponding notations are given in Sections 5.2, 5.4, 5.5 and 5.7. In the tests, we used $\tau = 0.3$, $T_0 = 4$, and an exponential heating strategy: $T_{k+1} = 0.96 \cdot T_k$

Table 5.1: Processing time of the correlation calculator algorithm as a function of the search window sizes, using 320×240 images, C++ implementation and a Pentium desktop computer (Intel(R) Core(TM)2 CPU, 2GHz)

Window size (W)	3×3	5×5	7×7	9×9	11×11
Time (sec)	0.5	1.1	2.4	4.2	6.3

5.8 Implementation Issues

A key point from a practical point of view is using an effective algorithm to calculate the correlation map used by the $o_c(.)$ feature (Section 5.4, eq. 5.7). The proposed algorithm, introduced in [13] in details, uses box filtering technique with the integral image trick [126] similarly to [127]. However, since our method does not assume accurate epipolar matching, the region where we search for pixel correspondences is a rectangle instead of a line, like in [127], which works with epipolar rectified images [128]. On the other hand, exploiting that due to the preliminary registration and the expected low parallax distortion the corresponding pixels are relatively close to each other, we can also extend the box matching technique to search in the moving window. Here, we need a 4D representation of the local correlation map, instead of 3D [127].

5.8.1 Running Speed

We tested the implemented correlation calculating algorithm with different sized *search windows* (H_s). Some results about the corresponding processing time are in Table 5.1. In the tests of Section 5.9, we use 7×7 pixel search windows. If larger window is necessary, we can speed up the method with multi-resolution techniques [129].

With C++ implementation and a Pentium desktop computer (Intel(R) Core(TM)2 CPU, 2GHz), processing 320×240 images takes $5-6$ seconds. For the main parts of the algorithm, the measured processing times are shown in Table 5.2.

Table 5.2: Running time of the main parts of the algorithm

Procedure	FCS	PCH	Corr. map	MRF opt.
Time (sec)	0.15	0.04	2.4	2.9

5.9 Results

In this section, we validate our method via image pairs from different test sets. We compare the results of the three layer model with three reference methods first qualitatively, then using different quantitative measures. Thereafter, we test the significance of the inter-layer connections in the joint segmentation model. Finally, we comment on the complexity of the algorithm.

5.9.1 Test Sets

The evaluations are conducted using manually generated ground truth masks regarding different aerial images. We use three test sets which contain 83 (=52+22+9) image pairs. The time difference between the frames to compare is about 1.5-2 seconds. The 'balloon1' and 'balloon2' test sets contain image pairs from a video-sequence captured by a flying balloon, while in the set 'Budapest', we find different image pairs taken from a plane. For each test set, the model parameters are estimated over 2-5 training pairs and we examine the quality of the segmentation on the remaining test pairs.

5.9.2 Reference Methods and Qualitative Comparison

We have compared the results of the proposed three-layer model to three other solutions. The first reference method (Layer1) is constructed from our model by ignoring the segmentation and the second observation layers. This comparison emphasizes the importance of using the correlation-peak features, since only the gray level differences are used here. The second reference is the method of Farin and With [117]. The third comparison is related to the limits of [109]: the optimal affine transform between the frames (which was automatically estimated in [109]) is determined in our comparative experiments in a supervised way, through

manually marked matching points. Thereafter, we create the change map based on the gray level difference of the registered images with using a similar spatial smoothing energy term to eq. 5.20.
Fig. 5.10 shows the image pairs, ground truth and the segmented images with the different methods. For numerical evaluation, we perform first a pixel based, then an object based comparison.

5.9.3 Pixel Based Evaluation

For pixel based evaluation, we use the Rc, Pr and FM measures again, which were introduced in Section 3.7.3. The results are presented in Table 5.3 for each image-set independently.
Regarding the 'balloon1'/'balloon2'/'Budapest' test sets, the gain of using our method considering the FM is 26/35/16% in contrast to the Layer1 segmentation and 12/19/13% compared to Farin's method. The results of the frame global affine matching, even with manually determined control points, is 5/10/11% worse than what we get with the proposed model.

5.9.4 Object Based Evaluation

Although our method does not segment the individual objects, the presented change mask can be the input of an object detector module. It is important to know, how many object-motions are correctly detected, and what is the false alarm rate.
If an object changes its location, two blobs appear in the binary motion image, corresponding to its first and second positions. Of course, these blobs can overlap, or one of them may be missing, if an object just appears in the second frame, or if it leaves the area of the image between the two shots. In the following, we call one such blob an 'object displacement', which will be the unit in the object based comparison.
Given a binary segmented image, denote by M_o (missing objects) the number of object displacements, which are not included in the motion silhouettes, while

Table 5.3: Numerical comparison of the proposed method (3-layer MRF) with the results that we get without the correlation layer (Layer1) and Farin's method [117] and the supervised affine matching. Rows correspond to the three different test image-sets with notation of their cardinality (e.g. number of image-pairs included in the sets).

Set		Recall				Precision			
Name	Cardi-nality	Layer1	Farin's	Sup. affine	**3layer MRF**	Layer1	Farin's	Sup. affine	**3layer MRF**
balloon1	52	0.83	0.76	0.85	**0.92**	0.48	0.74	0.79	**0.85**
balloon2	22	0.86	0.68	0.89	**0.88**	0.35	0.64	0.65	**0.83**
Budapest	9	0.87	0.80	0.85	**0.89**	0.56	0.65	0.65	**0.79**

Table 5.4: Numerical comparison of the proposed and reference methods via the *FM*-rate. Notations are the same as in Table 5.3.

Set		*FM*			
Name	Cardi-nality	Layer1	Farin's	Sup. affine	**3layer MRF**
balloon1	52	0.61	0.75	0.82	**0.87**
balloon2	22	0.50	0.66	0.75	**0.85**
Budapest	9	0.68	0.71	0.73	**0.84**

F_o (false objects) is the number of the connected blobs in the silhouette images, which do not contain real object displacements, but their size is at least as large as one expected object. For the selected image pairs of Fig. 5.10, the numerical comparison to Farin's and the supervised affine method is given in Table 5.3. A limitation of our method can be observed in the 'Budapest' #2 image pair: the parallax distortion of a standing lamp is higher than the length of the correlation search window side, which results in two false objects in the motion mask. However, the number of missing and false objects is much lower than with the reference methods.

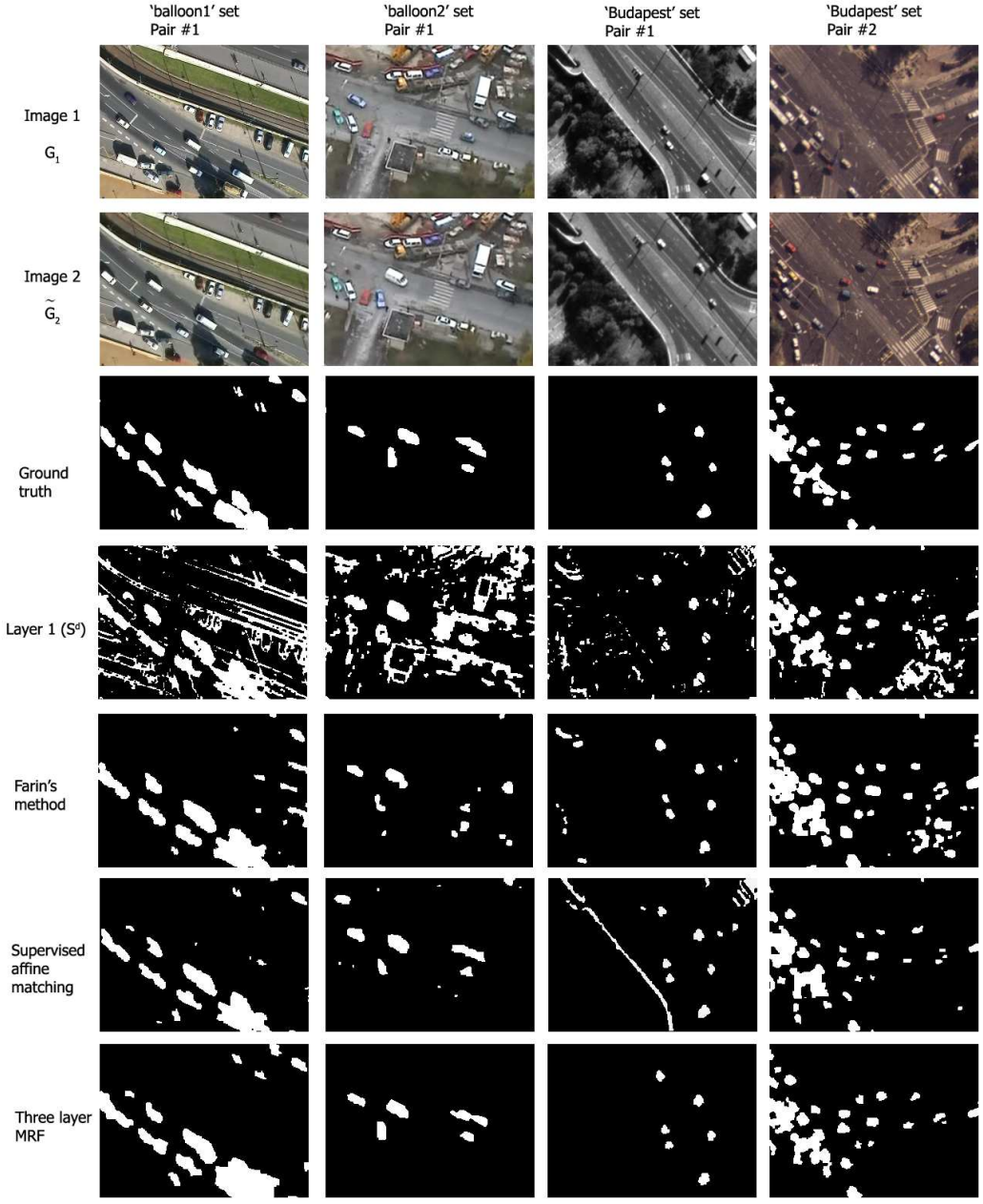


Figure 5.10: Test image pairs and segmentation results with different methods.

Table 5.5: Object-based comparison of the proposed and the reference methods. A_o means the number of all object displacements in the images, while the number of missing and false objects is respectively M_o and F_o.

Test pair		A_0	M_o			F_o		
Set	No.		Far.	Sup. aff.	**3lay. MRF**	Far.	Sup. aff.	**3lay. MRF**
balloon1	#1	19	0	0	**0**	6	1	**1**
balloon2	#1	6	0	0	**0**	3	2	**0**
Budapest	#1	6	1	0	**0**	7	7	**0**
Budapest	#2	32	0	1	**1**	10	6	**3**
	All	63	3	1	**1**	26	16	**4**

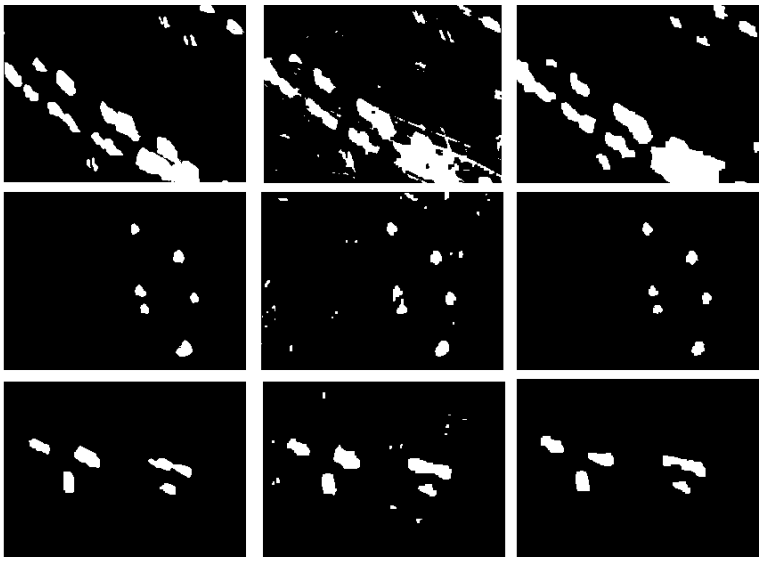


Figure 5.11: Illustration of the benefit of the inter-layer connections in the joint segmentation. Col 1: ground truth, Col 2: results after separate MRF segmentation of the S^d and S^c layers, and deriving the final result with a per pixel AND relationship. Col 3. Result of the proposed joint segmentation model

5.9.5 Significance of the Joint Segmentation Model

In the proposed model, the segmentations based on the $o_d(.)$ and $o_c(.)$ features are not performed independently: they interact through the inter-layer cliques. Although similar approaches have been already used for different image segmentation problems [45]-[48], the significance of intra-layer connections should be justified with respect to the current task. Note, that increasing the number of connections in the MRF results in a more complex energy model (eq. 5.24), which increases the computational complexity of the method.

We demonstrate the role of the inter-layer cliques by comparing the proposed scheme with a sequential model, where first, we perform two independent segmentations based on $o_d(.)$ and $o_c(.)$ (i.e. we segment the S^d and S^c layers ignoring the inter-layer cliques), thereafter, we get the segmentation of S^* by a per pixel AND operation on the A_d and A_c segmented images. In Fig. 5.11, we can observe that the separate segmentation gives noisy results, since in this case, the intra-layer smoothing terms do not take into account in the S^* layer. Consequently, the proposed label fusion process enhances the quality of segmentation versus the sequential model.

5.10 Conclusion of the Chapter

This chapter has addressed the problem of exploiting accurate change masks from image pairs taken by a moving camera. A novel three-layer MRF model has been proposed, which integrates the information from two different observations. The efficiency of the method has been validated through real-world aerial images, and its behavior versus three reference methods has been quantitatively and qualitatively evaluated.

Chapter 6

Markovian Framework for Structural Change Detection with Application on Detecting Built-in Changes in Airborne Images

In this chapter we address the problem of change detection in airborne image pairs taken with significant time difference. In reconnaissance and exploration tasks, finding the slowly changing areas through a long tract of time is disturbed by the temporal parameter changes of the considered clusters. We introduce a new joint segmentation model, containing two layers corresponding to the same area of different far times and the detected change map. We tested this co-segmentation model considering two clusters on the photos: built-in and natural/cultivated areas. We propose a Bayesian segmentation framework which exploits not only the noisy class-descriptors in the independent images, but also creates links between the segmentation of the two pictures, ensuring to get smooth connected regions in the segmented images, and also in the change mask. The domain dependent part of the model is separated, therefore the proposed structure can be used for significantly different descriptors and problems also.

6.1 Introduction

Tasks of Chapters 3-5 compare images taken with at most a few seconds time difference, based on comparing the gray or color values of the corresponding pixels. (In Chapter 5 we use a probabilistic interpretation for the pixel correspondency.) It is more difficult to define changes in situations, where the images, which we compare, were taken with significant time difference (several months or years). Due to the illumination changes and altering shadow effects the appearance of corresponding territories may be much different. In these cases, we have to carefully define what kind of differences we are looking for, while irrelevant changes should be ignored.

Automatic evaluation of aerial photo repositories is an important field of research, since periodically repeated manual processing is time-consuming and cumbersome in cases of high number of images and dynamically changing content. Although most of the corresponding state-of-the art models deal with multispectral [130][131][132] or SAR [133][134] imagery, the significance of handling optical photos is also increasing. In this chapter, we focus on built-in change detection in graycale/RGB photos provided by the Hungarian Institute of Geodesy, Cartography and Remote Sensing (FÖMI).

One of the few previous methods which can be also applied for optical images, is the PCA-based model [131]. Its main assumption is that the 'unimportant' color differences are caused by alteration of illumination and camera settings. Since these effects influence the observed sensor values in a multiplicative or additive fashion, they modelled the relationship of the corresponding pixel levels within the unchanged regions by a globally constant linear transform. Similar approaches can be also found in [135]. However, experiments show that this technique is not efficient enough regarding real images, because of strong noise effects.

In the following, we introduce a region based approach which is significantly different from the above pixel levels techniques. We show the applicability of the proposed model using aerial images.

6.2 Basic Goals and Notes

In the presented model we search for changes in image pairs from the same areas with respect of given properties. In aspect of these properties, we segment the images using J pixel-clusters: $(\phi_1, \phi_2, \dots \phi_J)$, and mark the connected image regions whose clusters have changed. For example, in the demonstrating application, a binary segmentation ($J = 2$) is achieved: built-in (ϕ_1) and unpopulated natural/cultivated (ϕ_2) areas are discriminated in airborne photos. The test-database contains a huge number of preliminary registered images whose manual checking would be cumbersome and time-consuming.

In the resulting segmented images and change-masks, we expect smooth connected regions corresponding to the different clusters, which can be ensured via MRFs. However, we must expect noisy cluster descriptors, which may alter by time, moreover, the exact borders of the clusters in the images may be ambiguous, similarly to the case of built-in and unpopulated areas. For this reason, if we apply two independent segmentation algorithms for the two images, the segmented regions may have slightly different shapes and sizes, even if the image parts have not changed in fact. Therefore, in this case, the result of simple local identity checking on the segmented images is corrupted by several artifacts corresponding to the different segmentations instead of real structural changes[1]. To solve this problem, during the segmentation procedure of the first image we must consider the second one and vice versa. Hence, we segment the images 'together' forcing the corresponding regions to have the same segmentation-masks regarding the two images.

In this chapter, we give a Bayesian approach on the above problem. Here, we derived features describing the different class-memberships of a given image point through a simple textural feature and we have developed a MRF model to perform the common segmentation. We emphasize that our model framework may work together with more sophisticated features [136] and for significantly different problems [e.g. trees, rivers]. However, the improved segmentation versus earlier methods segmenting the images separately can be already observed with

[1]We show some corresponding experimental results in Section 6.6.

this problem and feature selection. For simpler notation, we use only two clusters ($J = 2$) in the following descriptions, since it is appropriate for the selected problem, and the generalization for arbitrary number of segmentation-classes is straightforward.

The sketch of our method is as follows: first, we map the change detection problem to the same 3-layer MRF structure, which was introduced in Chapter 5. We assign a label to each node of the model, and a field energy corresponds to each global labeling. Next, we find the optimal (or at least, a good suboptimal) global labeling on the above model with respect of the previous energy term. Finally, we map the resulting labeling back to the segmentation problem. The appropriate construction of the field energy operator is responsible for getting appropriate segmentation with respect of the above mentioned notes. The key point in our model is that we assign three different nodes to each pixel having three different labels. The first and second components indicates whether the given pixel corresponds to the ϕ_1 (built-in) or ϕ_2 (unpopulated) cluster in the first and second images, respectively. The third component gives the 'changed'/'unchanged' result.

6.3 Image Model and Feature Extraction

Denote henceforward by G_1 and G_2 the two frames to compare above the same pixel lattice S. Built-in areas usually contain several sharp edges near the borders of houses and roads, while in the fields and forests the density of edges is lower. In the experiments, we found the texture descriptor of Rosenfeld and Troy [137] as a good indicator for discriminating these areas. Namely, if $\mathcal{E}(s)$ is the element corresponding to pixel s in the binary (Prewitt) edge image of G, the edge density descriptor χ is defined by:

$$\chi(s) = \frac{1}{(2\gamma+1)^2} \sum_{||s-r||\leq\gamma}^{r\in S} \mathcal{E}(r). \tag{6.1}$$

Let χ_1 and χ_2 be the edge density images of G_1 and G_2, respectively.

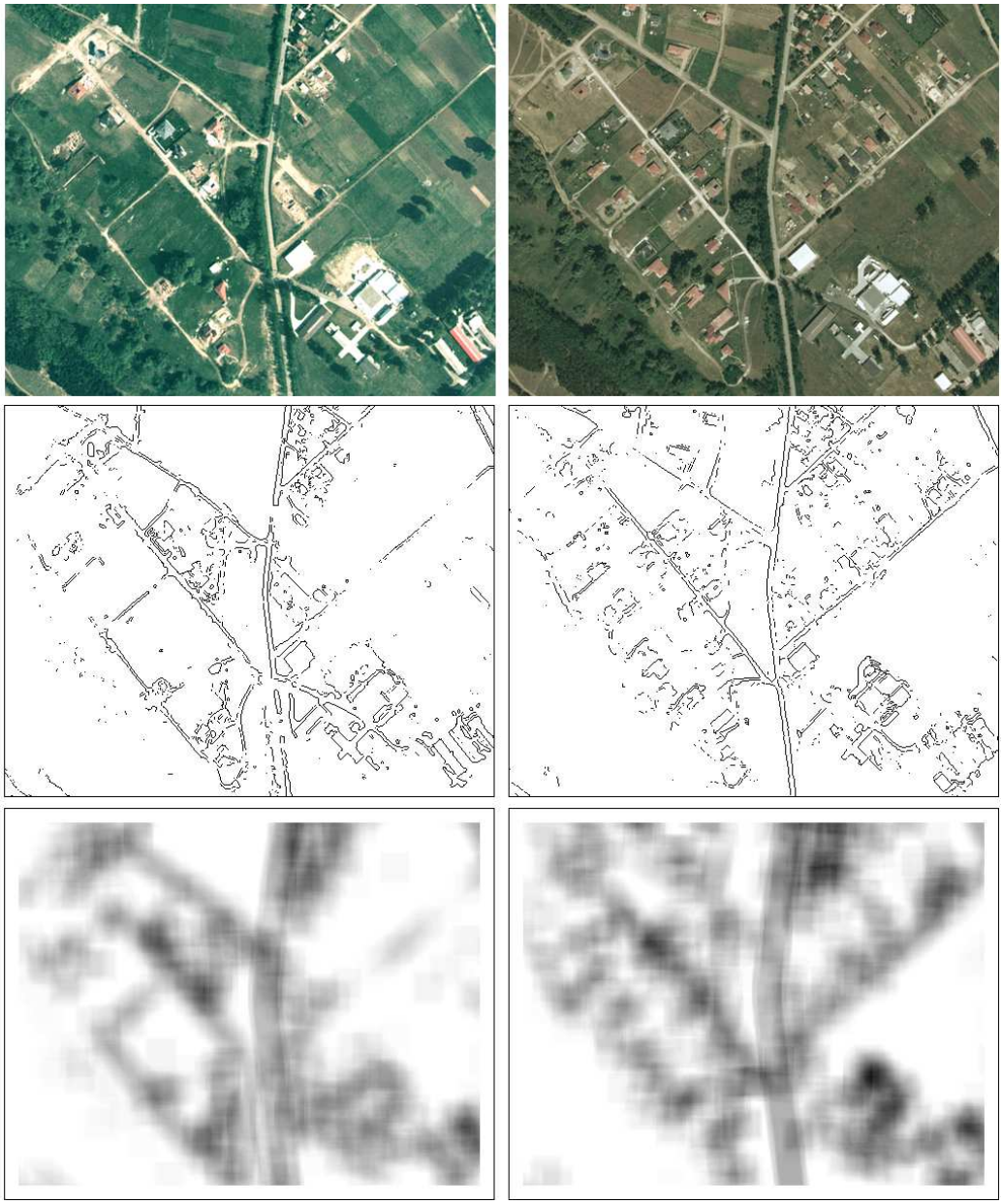


Figure 6.1: Feature extraction. Row 1: images (G_1 and G_2), Row 2: Prewitt edges ($\mathcal{E}_1$ and $\mathcal{E}_2$), Row 3: edge density images (χ_1 and chi_2; dark pixel correspond to higher edge densities)

6.4 MRF Segmentation Model

In the present task, the same 3-layer MRF model is used as introduced in Chapter 5, however the interpretation is different. In this case, two layers of the graph $\mathcal{G}$, S^1 and S^2, correspond to the built-in/unpopulated segmentations of the input images G_1 and G_2 respectively, while the S^* layer represents the final change mask. In other words, the two observation-dependent layers are responsible to segment two different images based on the same feature, while in Chapter 5 we segmented the same data at both layers (an image pair) based on different features.

First, we define two label sets: $\Phi_b \triangleq \{\phi_1, \phi_2\}$ are used in the S^1 and S^2 layers, while $\Phi_c \triangleq \{+, -\}$ are labels (changed, unchanged) for the S^* layer:

$$\omega := \begin{cases} S^1 \cup S^2 \to \Phi_b \\ S^* \to \Phi_c \end{cases} \tag{6.2}$$

The global labeling has the following form:

$$\underline{\omega} = \left\{ [s^i, \omega(s^i)] \mid s \in S, i \in \{1, 2, *\} \right\}, \tag{6.3}$$

where $\omega(s^1)$ and $\omega(s^2)$ labels define the ϕ_1/ϕ_2 segmentation classes of pixel s in the first and second images, respectively[1]. Change label $\omega(s^*)$ indicates whether there was built-in change (+), or not (-) at s pixel.

Note that this model is slightly different from the labeling procedure defined in page 9, where at each node the same label set is used. A possible way to convert this model to the abstract framework of Chapter 2, is that we use the united $\Phi_b \cup \Phi_c$ label set at each node, but to all global labelings negligible probability is assigned, which use a label from Φ_b at layer S^*, or a label from Φ_c at layers S^1 or S^2. Since the practical result of both interpretations are the same, we will use the simpler one defined by eq. 6.2.

The output of the change detector consists of the change labels of the different pixels. However, we show in the following that during the optimizing procedure, the segmentation labels also play important roles to get smooth and consistent

[1] As it was defined earlier that ϕ_1 means 'built-in', ϕ_2 indicates unpopulated regions.

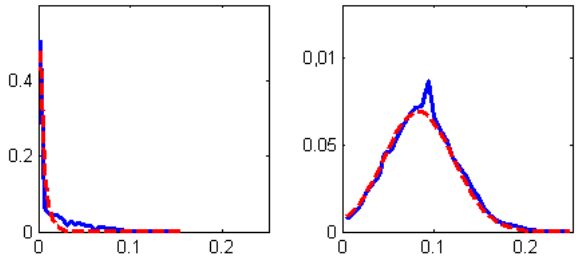


Figure 6.2: Left: Histogram (blue continuous line) of the occurring $\chi(.)$ values regarding manually marked 'unpopulated' (ϕ_2) pixels and the fitted Beta density function (with red dashed line). Right: Histogram for 'built-in' (ϕ_1) pixels and the fitted Gaussian density.

solution.

We define the observation process by the following: $O = S^1 \cup S^2$

$$\mathcal{O} = \left\{ [s^i, o(s^i)] \mid s \in S, i \in \{1, 2\} \right\}, \tag{6.4}$$

where for all $s \in S$:

$$o(s^1) = \chi_1(s) \tag{6.5}$$

$$o(s^2) = \chi_2(s) \tag{6.6}$$

Again, based on the Hammersley-Clifford theorem, the probabilities of the different global labelings follow a Gibbs distribution, and the optimal labeling $\underline{\widehat{\omega}}$ can be determined as:

$$\underline{\widehat{\omega}} = \arg\min_{\underline{\omega} \in \Omega} \sum_{C \in \mathcal{C}} V_C(\underline{\omega}_C) \tag{6.7}$$

We define singleton, doubleton and inter-layer cliques in the same way as in Chapter 5. Finally, the potential functions should be given.

To make the outline of the model simpler, we visualized the structure in Fig. 6.6, where we gave examples how the different clique potentials can be calculated considering the given labelings at two neighboring node-triples.

6.4.1 Singletons

The set of singleton cliques is defined by

$$\mathcal{C}_1 = \left\{ \{s^i\} \mid s \in S, i \in \{1, 2, *\} \right\} . \tag{6.8}$$

The potential of the singleton cliques expresses that the $\omega(s^1)$, $\omega(s^2)$ labels should be consistent with the $\chi_1(s)$ and $\chi_2(s)$ observation values:

$$V_{\{s^i\}}\left(\omega(s^i)\right) = -\log P\left(o(s^i)|\omega(s^i)\right) = -\log P\left(\chi_i(s)|\omega(s^i)\right) \quad i \in \{1,2\} \quad (6.9)$$

For example $P(\chi_1(s)|\omega(s^1) = \phi_2)$ is the probability of the fact that the ϕ_2 class process generates the observation $\chi_1(s)$ at pixel s.
Meanwhile the labels at the S^* layers do not depend directly on the observations:

$$V_{\{s^*\}}\big(\omega(s^*)\big) \equiv 0. \quad (6.10)$$

Our next task is to define an appropriate probabilistic description of the occurring observation values generated by the ϕ_1/ϕ_2 classes. First, we performed experiments: regarding different image pairs, we plot the histograms of the occurring $\chi_1(s)$ and $\chi_2(s)$ values corresponding to manually marked 'built-in' and 'unpopulated' region points in the input images. Fig. 6.2 contains the histograms generated for the second image from Fig 6.1. We observed, that regarding the distribution of the ϕ_2-classed $\chi(s)$ values, a Beta density function, $B(.,\beta_1,\beta_2)$, was an appropriate approximation, while the values in 'built-in' areas followed Gaussian distribution $N(.,\mu,\sigma)$. With these notations:

$$P(\chi_1(s)|\omega(s^1) = \phi_2) = B(\chi_1(s),\beta_{11},\beta_{12}), \quad (6.11)$$

$$P(\chi_2(s)|\omega(s^2) = \phi_2) = B(\chi_2(s),\beta_{21},\beta_{22}), \quad (6.12)$$

$$P(\chi_1(s)|\omega(s^1) = \phi_1) = N(\chi_1(s),\mu_1,\sigma_1), \quad (6.13)$$

$$P(\chi_2(s)|\omega(s^2) = \phi_1) = N(\chi_2(s),\mu_2,\sigma_2). \quad (6.14)$$

Here we note that the only application-dependent part of the segmentation model is defining the above a posteriori probabilities. Other features and distributions may be used for other problems.

6.4.2 Doubleton (Intra-Layer) Cliques

Doubleton cliques are responsible for getting smooth connected regions of nodes with the same label both during the built-in/unpopulated segmentation of the inputs and also in the change mask. As in all cases in this dissertation, the smoothness is ensured by forcing the neighboring nodes to have usually the same labels, by the Potts energy term. Therefore, doubleton cliques are defined:

$$\mathcal{C}_2 = \big\{\ \{s^i, r^i\} \mid i \in \{1, 2, *\};\ r \in \mathcal{V}_s;\ r, s \in S\big\}. \tag{6.15}$$

The potential of an intra-layer clique $C_2 = \{s^i, r^i\} \in \mathcal{C}_2,\ i \in \{1, 2, *\}$ has the following form [42]:

$$V_{C_2} = \Theta\left(\omega(s^i), \omega(r^i)\right) = \begin{cases} -\delta^i & \text{if } \ \omega(s^i) = \omega(r^i) \\ +\delta^i & \text{if } \ \omega(s^i) \neq \omega(r^i) \end{cases} \tag{6.16}$$

with a constant $\delta^i > 0$.

6.4.3 Inter-Layer Cliques

Finally, we introduce the inter-layer cliques, which are responsible for forcing the desired relationship between the corresponding segmentation and change labels. Usually, the change label of a given pixel is '+' (change), if and only if its segmentation labels are different. However, we consider that noise or segmentation artifacts may also cause erroneous different segmentation labels. Therefore, we give only penalty if the label triple is not consistent, but do not exclude these cases.

The set of inter-layer cliques, $\mathcal{C}_3$ has henceforward the following form:

$$\mathcal{C}_3 = \big\{\ \{s^1, s^2, s^*\} \mid s \in S\big\}. \tag{6.17}$$

We introduce the following indicator function for $i \in \{1, 2, *\}$:

$$I_i : S \to \{0,\ 1\}, \tag{6.18}$$

where

$$I_i(s) = \begin{cases} 1 & \text{if } \ \omega(s^i) \in \{\phi_1, +\} \\ 0 & \text{if } \ \omega(s^i) \in \{\phi_2, -\} \end{cases} \tag{6.19}$$

With this notation, for $C_3 = \{s^1, s^2, s^*\}$:

$$V_{C_3} = \varsigma(\omega(s^1), \omega(s^2), \omega(s^*)) = \begin{cases} -\varrho & \text{if } I_*(s) = I_1(s) \oplus I_2(s) \\ +\varrho & \text{otherwise.} \end{cases} \tag{6.20}$$

where $\oplus$ means modulo 2 addition.

6.5 Parameter Settings

The free parameters of the method can be classified into different groups. W determines the size of the window, where the edge density texture is collected. We used $\gamma = 5$ for images of size 320×256.

6.5.1 Parameters of the Observation Dependent Term

We determined the 'built-in' class' Gaussian parameters μ_1, σ_1, μ_2, σ_2 and the unpopulated areas' Beta parameters β_{11}, β_{12}, β_{21}, β_{22} with supervision, using manually marked training images.

6.5.2 Parameters of the Clique Regularization Terms

The parameters of the intra-layer clique potential functions, δ^1, δ^2 and δ^* influence the size of the connected blobs in the segmented images, while ϱ relates to the strength of the constraint between the segmentation labels and the 'change label' corresponding to a given node. We set these parameters to 1.

6.6 Results

We tested our method on registered airborne image pairs captured with 5-20 years time differences. The primary goal of the test was the validation of the proposed co-segmentation framework, not the appropriateness of the edge density feature as built-in area detector. However, to demonstrate the difficulty of the problem, we also compared our approach to the PCA-based model [131] introduced in the beginning of this chapter. Therefore, we generated the results for comparison in the following ways:

1. *PCA-based:* Implementation of the method [131].

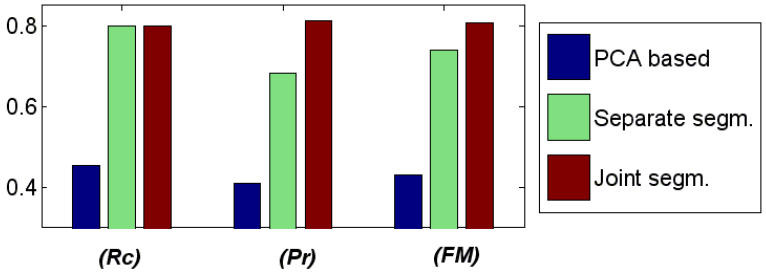


Figure 6.3: Comparison of the *Recall*, the *Precision*, and the FM rates regarding the PCA-based approach [131], and the introduced region based model, using 'separate segmentation' and the proposed 'joint segmentation' methods, respectively.

2. *Joint segm:* We jointly segmented the images and derived the change mask by the proposed model.

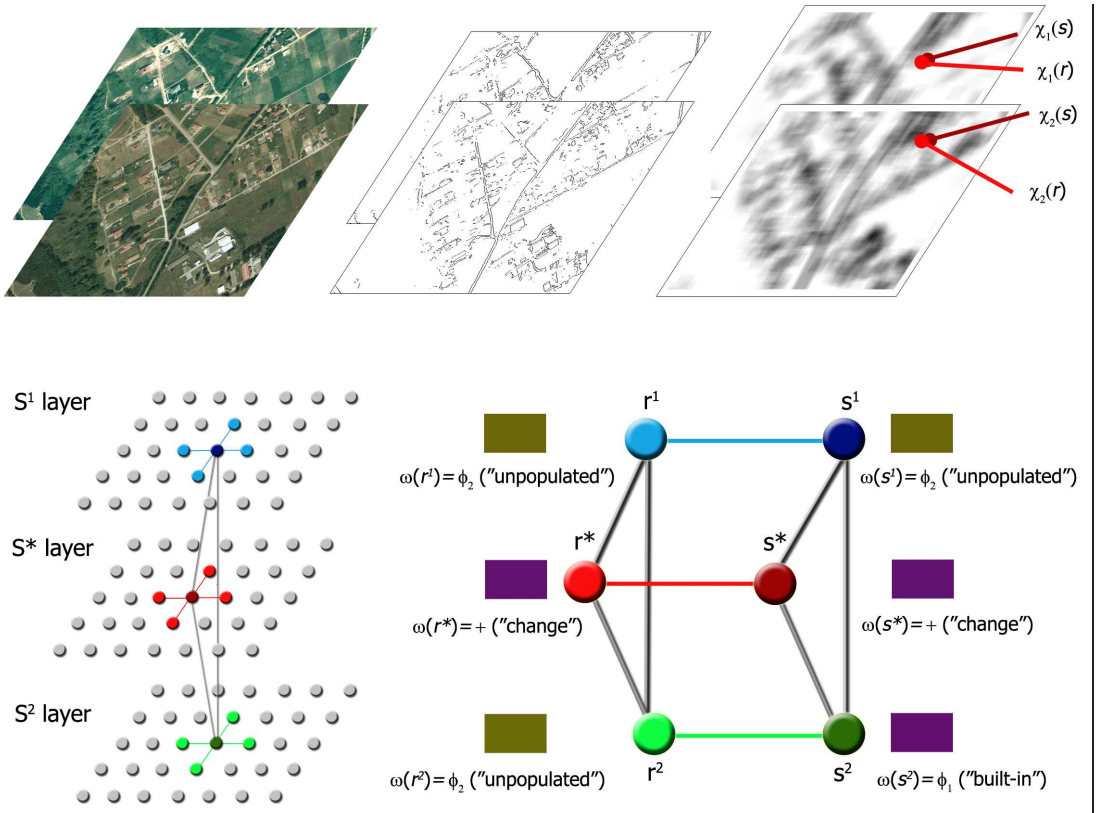
3. *Separate segm:* We used a two-step process. First, we segmented the images individually, and secondly, we used a simple xor operation to derive the change mask. More precisely, in the proposed framework, we ignored the $\varsigma(\omega^*(s))$ change mask regularization term ($\varrho = 0$), otherwise we optimized the MRF model with the same parameters as before. Finally, we set the change term to fulfill

$$I_*(s) = I_1(s) \oplus I_2(s). \tag{6.21}$$

The evaluations were done through manually generated ground truth masks. Segmentation results with the three methods for three different image pairs are in Fig. 6.5. The PCA-based method only gives reasonable performance regarding the third image pair, while working with 'separate segmentation' presents several false positive change-regions.

The results regarding the numerical evaluation are in the diagram of Fig. 6.3. We can observe that for PCA, each evaluation metrics gives poor results. Although the *Recall* rates with the separate and joint segmentation methods are very similar, the *Precision* of the joint segmentation is significantly better, since the proposed model is able to eliminate the slightly different segmentations' artifacts.

Finally, we note that the proposed model presents also the 'built-in'/'unpopulated'



The clique potentials concerning the selected nodes:
Singletons:

$$V_{\{s^1\}} = -\log P\big(\chi_1(s)|\omega(s^1)=\phi_2\big) = -\log B\big(\chi_1(s),\beta_{11},\beta_{12}\big)$$
$$V_{\{r^1\}} = -\log P\big(\chi_1(r)|\omega(r^1)=\phi_2\big) = -\log B\big(\chi_1(r),\beta_{11},\beta_{12}\big)$$
$$V_{\{s^2\}} = -\log P\big(\chi_2(s)|\omega(s^2)=\phi_1\big) = -\log N\big(\chi_2(s),\mu_2,\sigma_2\big)$$
$$V_{\{r^2\}} = -\log P\big(\chi_2(r)|\omega(r^2)=\phi_2\big) = -\log B\big(\chi_2(r),\beta_{21},\beta_{22}\big)$$

Doubletons: $V_{\{s^1,r^1\}} = -\delta^1$, $V_{\{s^*,r^*\}} = -\delta^*$, $V_{\{s^2,r^2\}} = +\delta^2$

Inter-layer potentials: $V_{\{s^1,s^2,s^*\}} = -\varrho$, $V_{\{r^1,r^2,r^*\}} = +\varrho$

Figure 6.4: Summary of the proposed model structure and examples how different clique-potentials are defined there. Assumptions: r and s are two selected neighboring pixels, while $\omega(r^1) = \omega(s^1) = \omega(r^2) = \phi_2$, $\omega(s^2) = \phi_1$ and $\omega(r^*) = \omega(s^*) = +$. In this case, the clique potentials have the calculated values.

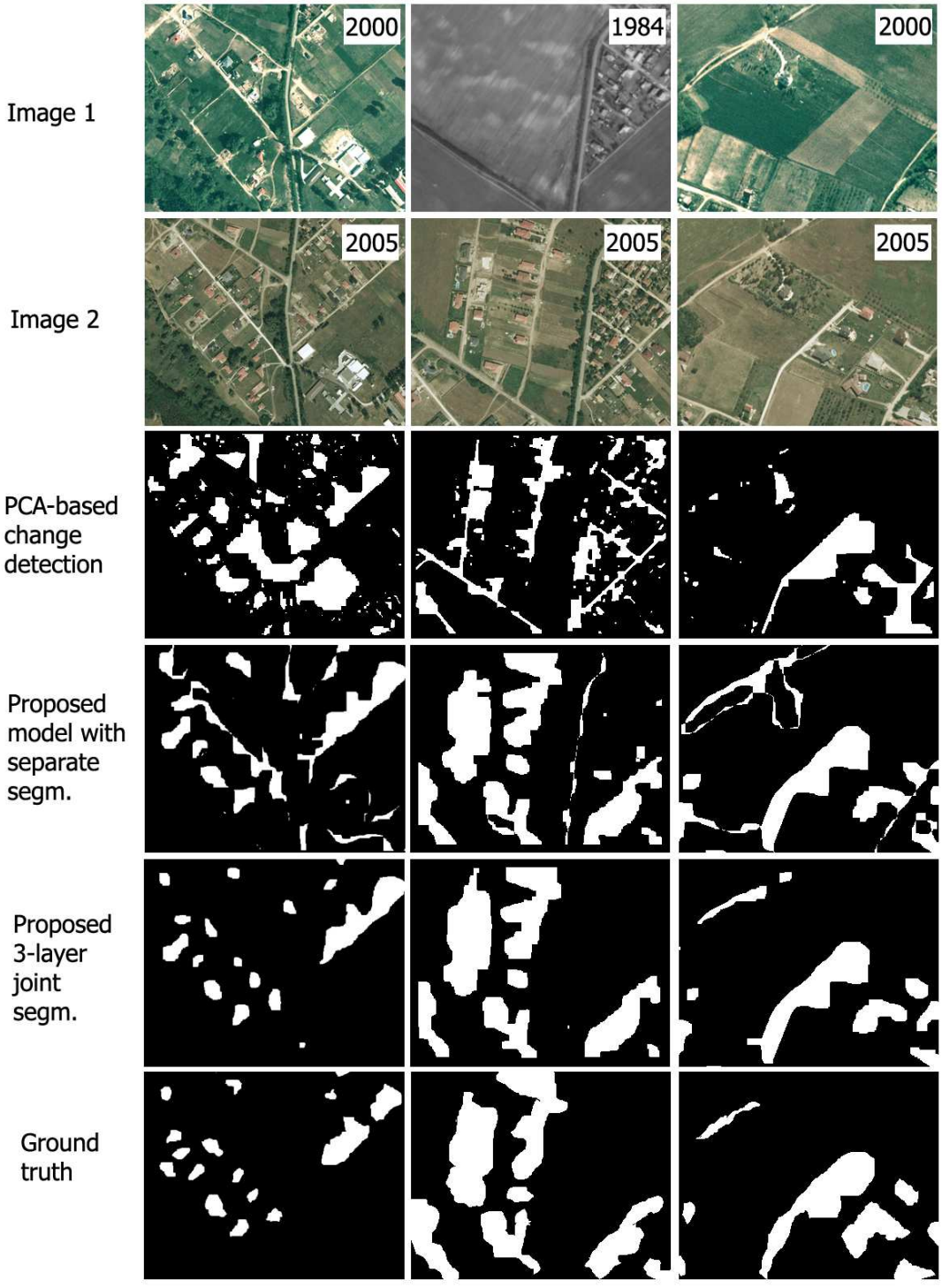


Figure 6.5: Validation. Rows 1 and 2: inputs (with the year of the photos), Row 3. Detected changes with the PCA-based method [131] Row 4. Change-result with 'separate segmentation'. Row 5. Change-result with the proposed 'joint segmentation' model, Row 5: Ground truth for built-in change detection.

Figure 6.6: Illustration of the segmentation results after optimization of the proposed MRF model. Left and middle: marking built-in areas in the first and second input images, respectively. Right: marking the built-in changes in the second photo.

segmentation of the input images by considering the $\omega(s^1)$ and $\omega(s^2)$ labels, respectively (Fig. 6.6).

6.7 Conclusion of the Chapter

In this chapter, we addressed the problem of change detection in image pairs taken with significant time difference. We introduced a general co-segmentation model and illustrated its advantages versus segmenting the images separately via a selected application: detecting built-in area changes in airborne photos.

Chapter 7

Conclusions of the thesis

This thesis has been dealing with three different change detection problems which still raise important challenges to experts in computer vision. Markov Random Fields have been chosen as framework for the surveys and improvements versus previous approaches have been proposed both in class modeling and in building different model structures. It has been shown that with appropriate feature selection and an efficient probabilistic description of the segmentation classes (such as foreground, background and shadow), the performance of processing *real scenes* can be significantly enhanced. As being shown by an example, we cannot always jointly extract temporal and spatial features for all the considered processes, however, in the same coherent model temporal and spatial descriptions of the different classes can be integrated. On the other hand, we have shown that in some cases of change detection composite multi dimensional probability distributions may hardly describe the desired feature interaction, but instead of this, simple structural innovations can also express complex probabilistic contexts.

The proposed models have been validated in real data sets, comparison to the state-of-the-art has been given. Mathematical descriptions of the methods have followed Bayesian approaches, and practical improvements versus previous models for the same problems have been demonstrated.

Summary

7.1 Methods Used in the Experiments

In the course of my work, theorems and assertions from the field of mathematical statistics, probability theory, optimization and reported results of image and video processing were explored.The proposed models are different implementations of *Markov Random Field*s (MRF, [32]). The output is a segmented image (e.g. a change mask), which is obtained by a global energy optimization:

$$\arg\max_{\underline{\omega}} P(\underline{\omega}|\mathcal{O}) = \arg\min_{\underline{\omega}} \left(-\log P(\mathcal{O}|\underline{\omega}) + \sum_{C\in\mathcal{C}} V_C(\underline{\omega}) \right), \qquad (7.1)$$

where $\mathcal{O}$ denotes observed image features, $\underline{\omega}$ is a possible segmentation, $\mathcal{C}$ is the set of *cliques*, i.e. pixel groups containing pairwise *neighboring* nodes, P is probability, and V_C denotes a clique potential function. Markovian property means here that only the neighboring nodes interact directly.

The test environment for *task 1* is the PPKEyes which is a digital video surveillance system developed at the Pázmány Péter Catholic University (PPCU) which is operating in the university campus. Evaluation of the proposed algorithms has also been performed on publicly available video databases. The aerial images used in the test regarding *task 2* and *task 3* were provided by the ALFA project, the photos were partially bought from the Hungarian Institute of Geodesy, Cartography and Remote Sensing (FÖMI).

For the design and testing of algorithms I have used Matlab and Visual Studio .Net environments. Implementing image processing routines in C/C++have been highly facilitated by the OpenCV software toolbox [125] provided by Intel. This thesis and the corresponding publications of the author have been prepared in LaTeX.

7.2 New Scientific Results

1. Thesis: I have worked out a novel spatio-temporal probabilistic model based on MRF for foreground - background separation and cast shadow detection in video frames. I have experimentally shown that the proposed method outperforms the recently published models with the same goals and scene assumptions.

Published in [1][2][4][5][16]

Co-author publications from the writer of this thesis, where the proposed model has been applied: [8][9][10][11][12]

The introduced model aims to efficiently separate foreground, background and cast shadows in videos provided by *real* surveillance applications. The method assumes that the sequences have been captured by static cameras, however, they may have low quality and low/uncertain frame rate. The model considers camera noise, temporal changes in illumination and presence of reflecting scene surfaces with inhomogeneous albedo and geometry. The energy term defined by eq. 7.1 has the following form:

$$\sum_{s \in S} -\log P\left(o(s)|\omega(s)\right) + \sum_{\{r,s\} \in \mathcal{C}} \Theta(\omega(s), \omega(r)), \tag{7.2}$$

where $o(s)$ is the feature value measured at pixel s, while $\omega(s)$ denotes the label of s indicating its segmentation class: foreground, background or shadow. $P\left(o(s)|\omega(s)\right)$ is the probability that $o(s)$ is

generated by the class featured by $\omega(s)$. The proposed model focuses on efficient feature extraction, and appropriate probabilistic modeling of the different classes. The $\Theta(.,.)$ term is responsible for the spatial smoothness of the segmentation, penalizing neighboring node pairs with different labels.

1.1. I have proposed a novel statistical and adaptive color model for detecting cast shadows. I have shown that the procedure is more efficient than using previous approaches if the scene reflection properties are not ideally Lambertian.

The most significant drawback of previously published shadow models is that their validity is limited to very specific environments, e.g. they expect presence of purely Lambertian reflecting surfaces. The performance of these methods notably decreases in lack of satisfying the scene assumptions.
I have introduced a novel parametric shadow model. My method can be used under variant illumination conditions, and it stochastically models the differences of real scenes from an ideal Lambertian environment. Local feature vectors are derived at the individual pixels, and the shadow's domain is represented by a global probability density function in that feature space. The parameter adaption algorithm is based on following the changes in the shadow's feature domain. Test results confirm that in real scenes the number of correctly detected shadowed pixels is significantly higher than using the purely Lambertian model.

1.2. A novel foreground description has been given based on spatial statistics of the nearby pixel values. I have shown that the introduced approach enhances the detection of background or shadow-colored object parts, even in low and/or unsteady frame rate videos.

Most of the previous methods identified foreground areas purely by recognizing the image regions which match neither to the background nor to the shadow models.That approach may result in erroneous classification of background/shadow colored object parts. In some other

cases frame rate sensitive features have been used which may not be available in several real applications.
I have proposed a novel multi-modal color model for foreground. My method exploits spatial color statistics instead of high frame rate temporal information to describe the regions of moving objects. Using the assumption that any object consists of spatially connected parts which have typical color/texture patterns, the distribution of the likely foreground colors have been locally estimated in each pixel neighborhood. Based on the test, several objects' parts were correctly detected in this way, which were erroneously ignored by models using a uniform foreground color distribution.

1.3. I have given a probabilistic model of the microstructural responses in the background and in the shadow. Thereafter, I have completed the MRF segmentation model with microstructure analysis. The proposed adaptive kernel selection strategy considers the local background properties. I have shown via synthetic and real-world examples, that the improved framework outperforms the purely color based model, and methods using a single kernel.

Although integration of simple color and texture features are widely used in image segmentation, textural components only have favourable contribution to the results if the local texture of the scene or the objects matches the selected features. Usually in a real world environment, we cannot find one proper textural feature for the whole scene. On the other hand, an irrelevant descriptor may increase the noise instead of enhancing the quality of segmentation.
I have developed a probabilistic description of microstructural responses observed in the background and in shadows. The features can be defined by arbitrary 3×3 kernels. At different pixel positions different kernels can be used, and an adaptive kernel selection strategy is proposed considering the local textural properties of the background regions. I have shown that the improved shadow model can also collaborate with the microstructural descriptors, and the distribution parameters can be analytically estimated. I have experimentally

shown that the proposed solution outperforms both the purely color based segmentation model, and the single kernel based color-texture fusion technique.

*1.4. I have experimentally shown that among the widespread color spaces, the CIE L*u*v* model is the best for cast shadow detection, both using an elliptical separation in the space of the introduced pixel-level descriptors and regarding a color space independent extension of the proposed MRF-segmentation model.*

Finding the most appropriate color space for cast shadow detection is still an open question. I have shown that color space selection is a key issue in shadow detection, if for practical purposes, shadow models with less free parameters are preferred.
I have developed a foreground/shadow pixel by pixel classifier which can work with different color spaces. Since at pixel-level, the statistics of the expected foreground colors is hard to estimate, I described the shadow domain in the feature space following a one-class-classification approach with elliptical border surface. I have supported the general relevancy of the proposed schema by an extensive study. Using this model, I have performed a detailed experimental comparison of seven widely used color systems. A color space independent extension of the proposed MRF-segmentation model has also been given with corresponding comparative experiments. Both evaluations showed the clear superiority of the CIE L*u*v* color space.
Since the first experiment series did not exploit any accessory information beyond the pixel by pixel shadow descriptors, the obtained results are more objective and general regarding the direct effects of color space selection. On the other hand, the comparison using the composite Markovian model – which also integrates neighborhood connection, spatial color statistics and texture information – shows that the advantage of using the appropriate color space can be also measured directly in the applications.

2. Thesis: I have developed novel three layer MRF models for object motion detection in unregistered aerial image pairs

and built-in change detection in aerial photos captured with several years time difference. I have experimentally validated the proposed models.

Published in [3][6][13]

2.1. I have developed a novel statistical model for object motion detection in image pairs captured by moving airborne vehicles. I have experimentally shown that the proposed approach outperforms previous models which use purely linear image registration techniques or local parallax removal.

This model deals with object motion detection in aerial image pairs taken from a moving platform. We assume that the photos contain 'dense' parallax, but after projective registration, the resulting local distortion has a bounded magnitude. For the above case, I have shown that gray level differencing (d) and local block correlation with a moving window (c) provide efficient complementary features to remove registration errors from the motion mask. Thereafter, I have developed a novel three layer MRF model structure for this change detection task. The segmentations of the first and third layers are based on the two different features, while the second layer represents the final change mask without direct links to the observations. Intra-layer connections ensure smoothness of the segmentation within each layer, while inter-layer links provide semantically correct labeling in the middle (second) layer. The Markovian energy term (eq. 7.1) is calculated as follows:

$$\sum_s -\log P_s^d + \sum_s -\log P_s^c + \sum_{i,\{r,s\}} \Theta(\omega(r^i), \omega(s^i)) + \sum_s \varsigma_s,$$

where P_s^d and P_s^c characterize the consistency of the extracted features and the corresponding segmentation labels, similarly to eq. 7.2. The $\Theta(.,.)$ function ensures smoothed segmentation within each layer (indexed by i). The value of ς_s is $\pm\rho$, depending whether the labels assigned to pixel s in the three layers agree with the prescribed label fusion rules or not.

Validation shows the superiority of the proposed model versus previous approaches for the same problem.

2.2. I have developed a Markovian framework for structural change detection in aerial photos captured with significant time difference. I have shown through an application on built-in change detection that connecting the segmentations of the different images via pixel-level links results in an efficient region based change detection method, which is robust against the noise and incompleteness of the class descriptors.

I have proposed a MRF framework for structural change detection based on the three layer model introduced in Thesis 2.1. In this case, two layers correspond to photos from the same area taken with large time differences and one for the detected change map. I tested this co-segmentation model considering two clusters on the photos: built-in and natural/cultivated areas. The proposed Bayesian segmentation framework exploits not only the extracted noisy class-descriptors, but also creates links between the segmentations of the two images, ensuring to get smooth connected regions in the change mask. I have shown that this joint segmentation model enhances the detection of changes versus the conventional composition of two independent single-layer MRF processes.

7.3 Examples for Application

All the developed algorithms can be used as preprocessing steps of high level computer vision applications, especially in video surveillance, traffic monitoring and aerial exploitation.
The proposed methods directly correspond to ongoing research projects with the participation of the Pázmány Péter Catholic University or the MTA-SZTAKI. Particularly, the *Shape Modeling E-Team of the EU Project MUSCLE* is interested in learning and recognizing shapes as a central part of image database indexing strategies. Its scope includes shape analysis and learning, prior-based segmentation and

shape-based retrieval. In shape modeling, however, accurate silhouette extraction is a crucial preprocessing task.

The primary aim of the *Hungarian R&D Project ALFA* (NKFP 2/046 /04 project funded by NKTH) is to create a compact vision system that may be used as autonomous visual recognition and navigation system of unmanned aerial vehicles. In order to make long term navigational decisions, the system has to evaluate the captured visual information without any external assistance. The civil use of the system includes large area security surveillance and traffic monitoring, since effective and economic solution to these problems is not possible using current technologies. The *Hungarian GVOP (3.1.1.-2004-05-0388/3.0)* tackles the problem of semantic interpretation, categorizing and indexing the video frames automatically. For all these applications, object motion detection provides significant information.

Appendix A

Some Relevant Issues of Probability Calculus

This appendix summarizes some basic concepts and theorems of probability calculus, which are used in the thesis. We will refer to elementary definitions of probability theory, such as random variables, probability density functions or Bayes decision, for which a detailed introduction can be found in [44] or [43].

A.1 MAP and ML Decision Strategies

Let be X_1, X_2, ..., X_n random variables representing the outcomes of different hidden stochastic processes. Let be o an observation value (measurement), which is generated by one of the processes. The task is to find the 'source process' of o. Following the Bayesian decision strategy, the optimal class is the maximum a posteriori (MAP) estimate, which is defined by:

$$i_{\mathrm{MAP}} = \arg\max_{i=1\ldots n} P(i|\ o). \tag{A.1}$$

On the other hand, the maximum likelihood (ML) estimate can be determined by:

$$i_{\mathrm{ML}} = \arg\max_{i} P(o|\ i) = \arg\max_{i} P(X_i = o). \tag{A.2}$$

$P(o|\ i)$ and $P(i)$ are called *a posteriori* and *a priori* probabilities, respectively. Using the Bayes theorem:

$$P(i|\ o) = \frac{P(o|\ i) \cdot P(i)}{P(o)}. \tag{A.3}$$

Since $P(o)$ is independent of i:

$$i_{\text{MAP}} = \arg\max_i P(o|\ i) \cdot P(i). \tag{A.4}$$

Consequently, the outcomes of the MAP and ML decisions are equivalent if $P(i) = \frac{1}{n}$ for all $i = 1, \ldots, n$.

A.2 Particular Probability Distributions

Real-world stochastic processes are often modeled by elementary probability density functions (*pdf*). In this thesis uniform, normal (or Gaussian), Beta distributions and finite mixtures are used, which will be introduced in the following. Henceforward, $f_X(.)$ denotes the *pdf* of random variable X.

A.2.1 Uniform Distribution

The *pdf* of a uniform variable X_U with parameters (a, b), where $b > a$, has the following form:

$$f_{X_U}(o, a, b) = \begin{cases} \frac{1}{a-b}, & \text{if } \ o \in (a, b) \\ 0 & \text{otherwise} \end{cases} \tag{A.5}$$

Modeling a process with uniform density usually results in a weak probabilistic description, since all the occurring observation values in a given domain are generated with the same probability, which may give less information for a MAP or ML classifier. Taking a binary clustering, a uniform process has simply a 'threshold'-like role: between a uniform *pdf* $f_{X_U}(.)$ and an arbitrary *pdf* $f_X(.)$, ML decision over the (a, b) domain is equivalent with thresholding $f_X(.)$ by $\frac{1}{b-a}$ (see Section 5.4).

A.2.2 Normal Distribution

Normal (or Gaussian) distributions are very widespread in signal processing, since they can efficiently model noise and inaccuracies in the measurements. X_N is normal variable if

$$f_{X_N}(o, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(o-\mu)^2}{\sigma^2}}. \tag{A.6}$$

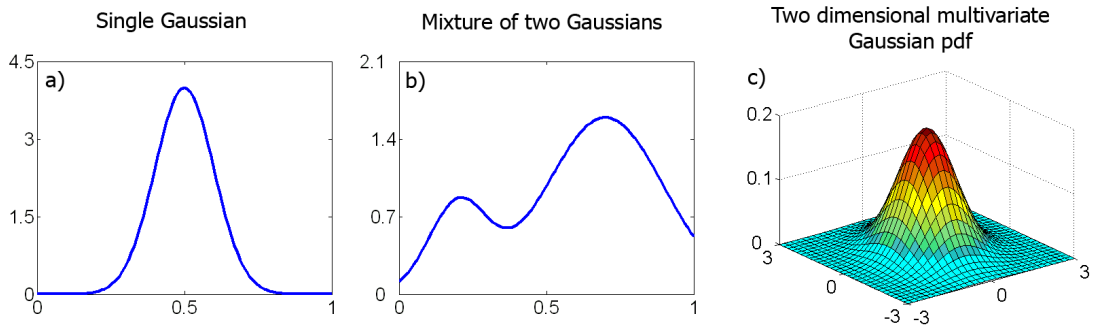


Figure A.1: Probability density function of a) a single Gaussian, b) a mixture of two Gaussians and c) a two dimensional multivariate Gaussian random variable

In this case, we use the notation

$$X_N \sim N[\mu, \sigma^2]. \tag{A.7}$$

The Gaussian *pdf* has a typical bell shape as it can be seen in Fig A.1a. To enhance its usability, multimodal (mixture) and multivariate extensions of normal distribution are widely used as well.

A.2.3 Mixtures

Let be X_1, X_2, ..., X_K random variables, and κ_1, κ_2, ..., κ_n positive constants for which $\sum_{k=1}^{K} \kappa_k = 1$ holds. Random variable X is a mixture, if

$$f_X(o) = \sum_{k=1}^{K} \kappa_k \cdot f_{X_k}(o). \tag{A.8}$$

If X_1, X_2, ..., X_K are normal variables X is called mixture of Gaussians (see Fig A.1b for a $K = 2$ case).

A.2.4 n-Dimensional Multivariate Normal Distribution

Let $\overline{x} = [x_1, \ldots, x_n]^T$, $\overline{\mu} = [\mu_1, \ldots, \mu_n]^T$ and Σ an $n \times n$ positive definite matrix. $\overline{X}$ is a multi variate normal variable, if its *pdf* is as follows:

$$f_{\overline{X}}(x_1, \ldots, x_n) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2}(\overline{x} - \overline{\mu})\Sigma^{-1}(\overline{x} - \overline{\mu})^T\right) \tag{A.9}$$

A two dimensional ($n = 2$) Gaussian *pdf* is shown in Fig A.1c.

A.2.5 Multivariate Normal Distribution with Uncorrelated Components

An important special case of multivariate normal distributions is if the covariance matrix is diagonal:

$$\Sigma = \begin{pmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_n^2 \end{pmatrix} \tag{A.10}$$

here, substituting formula (A.10) to (A.9) results in:

$$f_{\overline{X}}(x_1, \dots, x_n) = \exp\left\{ -\frac{n}{2}\log 2\pi - \sum_{i=1}^{n} \log \sigma_i - \frac{1}{2}\left(\frac{x_i - \mu_i}{\sigma_i}\right)^2 \right\} \tag{A.11}$$

The following theorem is used in Chapter 4.

Theorem 2 *Let $\overline{X}$ be a 3 dimensional multivariate normal variable with uncorrelated components. The equipotential surfaces of the $f_{\overline{X}}(.)$ density function are ellipsoids, having parallel axes with the x, y, z coordinate axes.*

Proof: the equation of a standard ellipsoid body in an x-y-z Cartesian coordinate system has the following form for $\overline{o} = [o_1, o_2, o_3] \in \mathbb{R}^3$:

$$\sum_{i=0}^{2} \left(\frac{o - a_i}{b_i}\right)^2 = 1, \tag{A.12}$$

where $[a_0, a_1, a_2]$ is the coordinate of the ellipsoid center and (b_0, b_1, b_2) are the semi-axis lengths.
For a given z, the set of $\overline{o}$ can be expressed, where $f_{\overline{X}}(\overline{o}) = z$

$$f_{\overline{X}}(\overline{o}) = \exp\left\{ -\frac{3}{2}\log 2\pi - \sum_{i=1}^{3} \log \sigma_i - \frac{1}{2}\left(\frac{o_i - \mu_i}{\sigma_i}\right)^2 \right\} = f_0 \tag{A.13}$$

With realigning this equation:

$$\frac{3}{2}\log 2\pi + \sum_{i=1}^{2} \log \sigma_i + \frac{1}{2}\left(\frac{o_i - \mu_i}{\sigma_i}\right)^2 = -\log f_0 \tag{A.14}$$

$$\sum_{i=1}^{3} \left(\frac{o_i - \mu_i}{\sigma_i}\right)^2 = 2\left(-\log f_0 - \frac{3}{2}\log 2\pi - \sum_{i=1}^{3} \log \sigma_i \right) \tag{A.15}$$

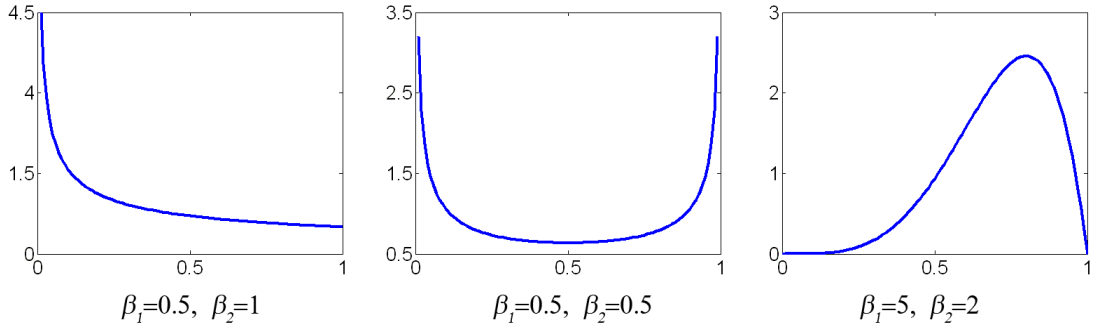


Figure A.2: Shapes of a Beta density function in cases of three different parameter settings

Eq. A.15 is equivalent to eq. A.12, if $\mu_i = a_i$, $\sigma_i = b_i$ for $i = 0, 1, 2$, and

$$2\left(-\log f_0 - \frac{3}{2}\log 2\pi - \sum_{i=1}^{3}\log b_i\right) = 1 \tag{A.16}$$

Expressing f_0:

$$f_0 = \frac{e^{-\frac{1}{2}}}{(2\pi)^{\frac{3}{2}} \cdot b_1 b_2 b_3} \tag{A.17}$$

A.2.6 Beta Distribution

The Gaussian density function is symmetric about its mean value, while it has a typical bell shape which is not favourable for some occurring processes. An alternative model is using a Beta random variable, whose *pdf* is defined as follows:

$$f_{X_b}(o, \beta_1, \beta_2) = \begin{cases} \frac{\Gamma(\beta_1+\beta_2)}{\Gamma(\beta_1)\Gamma(\beta_2)} o^{\beta_1-1}(1-o)^{\beta_2-1}, & \text{if } \; o \in (0,1) \\ 0 & \text{otherwise} \end{cases} \tag{A.18}$$

where the $\Gamma(.)$ function is defined by:

$$\Gamma(\beta) = \int_0^\infty \lambda^{\beta-1} e^{-\lambda} d\lambda. \tag{A.19}$$

As Fig. A.2 shows, depending on β_1 and β_2, $f_{X_b}(.)$ may have various shapes. Note that with $\beta_1 = 1$ and $\beta_2 = 1$, the Beta distribution is equivalent to a uniform density.

A.3 Estimation of the Distribution Parameters

Let us model a stochastic process X by a given density function (e.g. uniform or mixture of two Gaussians etc.) with a set of parameters θ. Let be o_1, o_2, ..., o_n observations generated by X, where o_t corresponds to time t. Using a ML estimation strategy, the goal is to find the optimal θ_{ML} parameters defined by:

$$\theta_{\mathrm{ML}} = \arg\max_{\theta} P(o_1, \ldots, o_n|\ \theta) \tag{A.20}$$

A standard tool for ML parameter estimation is the Expectation Maximization algorithm (EM). However, using EM is practically inefficient in some cases. The ML estimate of the Gaussian parameters for given samples can be obtained by simply getting the empirical mean and variance values: $\mu_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n} o_i$, $\sigma^2_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n}(o_i - \mu_{\mathrm{ML}})^2$.

On the other hand, EM is an offline algorithm: at time t the input of the parameter estimation process is the whole $[o_1, o_2, \ldots, o_t]$ observation vector. This approach is computationally expensive and the storage of the past o_i measurements is needed.

Online estimators calculate the parameters for time $t+1$, θ_{t+1}, using only the observation and parameters at t: o_t and θ_t. The empirical mean and variance values can be easily determined in an online way based on the following theorem, which is a straightforward consequence of the previous definitions of μ_{ML} and σ^2_{ML}:

Theorem 3 *If $\{o_i | i = 1 \ldots t\}$ is a set of real numbers, $S_t = \sum_{i=1}^{t} o_i$, $Q_t = \sum_{i=1}^{t} (o_i)^2$, the empirical mean and variance values are in the following form:*

$$\mu_{\mathrm{ML}}[t+1] = \frac{S_t}{t} \qquad \sigma^2_{\mathrm{ML}}[t+1] = \frac{Q_t}{t} - \left(\frac{S_t}{t}\right)^2. \tag{A.21}$$

S_t and Q_t can be online updated by $S_{t+1} = S_t + o_{t+1}$, $Q_{t+1} = Q_t + (o_{t+1})^2$

An online parameter estimator for a mixture of Gaussians distribution is given by [62]. Their proposed *on-line k-means* algorithm is introduced in Section 3.3.2 in details. Note that online k-means does not guarantee to find the ML estimate, but it is efficient for some practical problems like background modeling.

A.4 Transformation of Random Variables

The following theorems about random variable transformations are used in Chapter 3.

Theorem 4 *If $X_1, \ldots, X_n$ are random variables with mean values $\mu_1, \ldots, \mu_n$, and finite variances $\sigma_1^2, \ldots, \sigma_n^2$, and $Y = X_1 + \ldots + X_n$, with μ_Y expected value and σ_Y^2 variance, while finite correlation factor, $\rho_{j,k}$ exists for $j, k \in \{1, \ldots, n\}$, then*

$$\mu_Y = \sum_{i=1}^{n} \mu_i, \quad \sigma_Y^2 = \sum_{i=1}^{n} \sigma_i^2 + 2 \sum_{j<k} \sigma_j \sigma_k \cdot \rho_{j,k} \tag{A.22}$$

Proof: Consequence of the theorem about the sum of random variables in [43, p. 216], with notation of $\mathrm{Cov}\{X_j, X_k\}$ for the covariance of X_j and X_k, substituting $\rho_{j,k} = \frac{\mathrm{Cov}\{X_j, X_k\}}{\sigma_j \cdot \sigma_k}$.

Theorem 5 ***(Linear transform of a normal variable)*** *If there are given $a \neq 0$ and b scalars, and $X \sim N[\mu, \sigma^2]$ normal variable, then $Y = aX + b$ is also a normal variable:*

$$Y \sim N[a\mu + b, a^2\sigma^2]. \tag{A.23}$$

Proof: Based on the assumption that X is Gaussian,

$$P(X < x) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{\sigma^2}} dt. \tag{A.24}$$

Assume that $a > 0$. In this case, the distribution function of Y has the following form:

$$P(Y < y) = P(aX + b < y) = P\left(X < \frac{y-b}{a}\right) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{\frac{y-b}{a}} e^{-\frac{(t-\mu)^2}{\sigma^2}} dt \tag{A.25}$$

substituting $\lambda = at + b$, and $\frac{dt}{d\lambda} = \frac{1}{a}$ implies:

$$P(Y < y) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{y} \frac{1}{a}\, e^{-\frac{\left(\frac{\lambda-b}{a}-\mu\right)^2}{\sigma^2}} d\lambda = \frac{1}{\sqrt{2\pi}a\sigma} \int_{-\infty}^{y} e^{-\frac{\left(\lambda-(a\mu+b)\right)^2}{a^2\sigma^2}} d\lambda \tag{A.26}$$

which means that $Y \sim N[a\mu + b, a^2\sigma^2]$. Case of $a < 0$ can be handled in a similar manner.

Appendix B

Summary of Abbreviations and Notations

Abbreviation	**Concept**
MRF	Markov Random Field
MAP	Maximum a posteriori
ML	maximum likelihood
SA	Simulated Annealing (optimization method)
pdf	probability density function
ICM	Iterated Conditional Modes (MRF optimization technique)
MD	Metropolis Dynamic (SA relaxation technique)
MMD	Modified Metropolis Dynamic (SA relaxation technique)
CR	content ratio (shadow model)
II	Illumination invariant (shadow model)
TP, TN	true positive, true negative (evaluation parameters)
FP, FN	false positive, false negative
e.g.	for example (in *latin:* 'exempli gratia')
i.a.	inter alia
'P+P'	plane+parallax (approach on image registration)

Variable	**Definition**
j	imaginary unit
i, k, m	arbitrary index (number or enumeration)
n	dimension parameter, index
$\mathcal{G}(Q, E)$	MRF graph with set of nodes Q and edges E.
q	abstract node of a graph $\mathcal{G}$, $q \in Q$ (without emphasizing which is the corresponding pixel in the input image)

Variable	Definition
ε	edge of a graph $\varepsilon \in E$
S	pixel lattice
s, r	pixel ($s, r \in S$), or its corresponding node in $\mathcal{G}$, in case of a single layer MRF model
s^i	node at the i layer, which corresponds to pixel $s \in S$ (in case of multi-layer MRF models)
Φ	label set ($\#\Phi = J$)
ϕ, ϕ_i	abstract label or class identifier
bg, fg, sh	labels used in foreground/object motion detection ('sh' only in Chapters 3, 4
+, -	labels used in structural change detection
$\mathcal{V}$	neighborhood system of $\mathcal{G}$
$\mathcal{V}_q$	neighborhood of node q in $\mathcal{G}$ ($\mathcal{V}_q \in \mathcal{V}$)
$\omega(q)$	label of node q in $\mathcal{G}$ ($\omega(q) \in \Phi$).
$\underline{\omega}$	global labeling: $\{[q, \omega(q)] \vert q \in Q\}$
Ω	set of all the possible global labelings ($\underline{\omega} \in \Omega$)
$\underline{\omega}_Y$	label subconfiguration corresponding to set $Y \subseteq Q$ ($\underline{\omega}_Y \subseteq \underline{\omega}$)
$o(s)$, $\overline{o}(s)$	observation vector ($\in \mathbb{R}^n$) at pixel s (or at node s in single-layer case)
$o(q)$, $\overline{o}(q)$	observation vector ($\in \mathbb{R}^n$) assigned to node $q \in Q$
$o_i(q)$ $\big(o_i(s)\big)$	ith component of vector $o(q)$ ($o(s)$)
$\mathcal{O}$	global observation on $\mathcal{G}$: $\{o(q) \mid q \in Q\}$
C	clique of $\mathcal{G}$
$\mathcal{C}$	set of cliques in $\mathcal{G}$
V_C	potential of clique C
$V_{\{q_1,\ldots,q_n\}}$	potential of a clique containing nodes $q_1, \ldots, q_n$
L, u, v	color components in the CIE L*u*v* space
$\underline{\chi}$	texture/microstructural feature, or feature component index
$\underline{\psi}$, ψ_i	shadow descriptor vector and its ith component
$\Theta(\omega_1, \omega_2)$	Potts smoothing term
δ, δ^i	parameter of the smoothing term (in the ith layer)

Variable	Definition
$\varsigma(.)$	inter-layer potential function
ϱ	parameter of the inter-layer potential term
λ	wavelength or other integrand variable
$G(\leftarrow S)$, G_i	image (over S lattice), the ith image
$g(s)$, $g_i(s)$	gray value/image sensor value at pixel s (in the ith image)
$\nu(\lambda)$	sensor sensitivity at wavelength λ
$e(\lambda, s)$	illumination function
e_r	local error vector of 2D registration at pixel r
$\mathbf{E}_i$	ith camera center
$\rho(\lambda, s)$	descriptor of surface albedo-geometry
$N(\mu, \sigma)$	normal distribution with mean value μ and standard deviation σ
$N(\overline{\mu}, \overline{\overline{\Sigma}})$	n dimensional normal distribution with mean value vector $\overline{\mu}$ and covariance matrix $\overline{\overline{\Sigma}}$
$N(\overline{\mu}, \overline{\sigma})$	n dimensional normal distribution with diagonal covariance matrix, where $\overline{\sigma}$ vector denotes the root of the diagonal elements.
$f(x)$	arbitrary probability density function (pdf)
$\eta(x, \mu, \sigma)$	Gaussian (normal) pdf with parameters μ, σ.
$B(x, \beta_1, \beta_2)$	beta pdf with parameters β_1, β_2.
$p_\phi(s)$	pdf value corresponding to pixel s and class ϕ.
$\epsilon_\phi(s)$	$-\log p_\phi(s)$.
$\vartheta(x)$	residual pdf term in the foreground model
h	histogram
H_s	A given rectangular neighborhood of pixel s (different roles in Chapters 3 and 5).
$\kappa(s)$, $\kappa_i(s)$	weight (of the ith term) in a mixture pdf corresponding to pixel s
$o_d(s)$	gray level difference feature
$o_c(s)$	correlation peak value feature
ξ, ξ_j, ξ_t	control parameter of learning speed in background/shadow modeling

Variable	Definition
t, $.^{[t]}$	time (upper) index (for any quantities)
T	transpose
T	temperature (for simulated annealing)
$\mathcal{T}$	time constant (e.g. period of parameter update)
$\mathfrak{T}$	transform
τ	foreground threshold parameter (Gaussian term)
ζ	foreground threshold parameter (preliminary filtering)
$\mathcal{D}$	matching operator for a Gaussian component
R	correlation map
$\mathcal{F}$	2D Fourier transform

References

The author's journal publications

[1] **Cs. Benedek** and T. Szirányi, "Bayesian foreground and shadow detection in uncertain frame rate surveillance videos," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 608–621, 2008.

[2] **Cs. Benedek** and T. Szirányi, "Study on color space selection for detecting cast shadows in video surveillance," *International Journal of Imaging Systems and Technology*, vol. 17, no. 3, pp. 190–201, 2007.

The author's international conference publications

[3] **Cs. Benedek**, T. Szirányi, Z. Kato, and J. Zerubia, "A multi-layer MRF model for object-motion detection in unregistered airborne image-pairs," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. VI, (San Antonio, Texas, USA), pp. 141–144, IEEE, Sept. 2007.

[4] **Cs. Benedek** and T. Szirányi, "Markovian framework for foreground-background-shadow separation of real world video scenes," in *Proc. Asian Conference on Computer Vision (ACCV), Lecture Notes in Computer Science (LNCS) 3851*, (Hyderabad, India), pp. 898–907, Springer, Jan. 2006.

[5] **Cs. Benedek** and T. Szirányi, "Color models of shadow detection in video scenes," in *Proc. International Conference on Computer Vision Theory and Applications (VISAPP)*, vol. IFP/IA, (Barcelona, Spain), pp. 225–232, INSTICC, March 2007.

[6] **Cs. Benedek** and T. Szirányi, "Markovian framework for structural change detection with application on detecting built-in changes in airborne images," in *Proc. IASTED International Conference on Signal Processing, Pattern Recognition and Applications (SPPRA)*, (Innsbruck, Austria), pp. 68–73, ACTA, February 2007.

[7] D. Szolgay, **Cs. Benedek**, and T. Szirányi, "Fast template matching for measuring visit frequencies of dynamic web advertisements," in *Proc. International Conference on Computer Vision Theory and Applications (VISAPP)*, (Funchal, Madeira, Portugal), pp. 228–233, INSTICC, January 2008.

[8] Z. Szlávik, L. Havasi, **Cs. Benedek**, and T. Szirányi, "Motion-based flexible camera registration," in *Proc. IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, (Como, Italy), pp. 439–444, Sept. 2005.

[9] Z. Szlávik, T. Szirányi, L. Havasi, and **Cs. Benedek**, "Optimizing of searching co-motion point-pairs for statistical camera calibration," in *Proc. IEEE International Conference on Image Processing*, vol. II, (Genoa, Italy), pp. 1178–1181, Sept. 2005.

[10] Z. Szlávik, T. Szirányi, L. Havasi, and **Cs. Benedek**, "Random motion for camera calibration," in *European Signal Processing Conference (EUSIPCO)*, (Antalya, Turkey), Sept. 2005.

[11] L. Havasi, Z. Szlávik, **Cs. Benedek**, and T. Szirányi, "Learning human motion patterns from symmetries," in *Proc. ICML Workshop on Machine Learning for Multimedia*, (Bonn, Germany), pp. 32–37, Aug. 2005.

[12] L. Havasi, **Cs. Benedek**, Z. Szlávik, and T. Szirányi, "Extracting structural fragments from images showing overlapping pedestrians," in *Proc. IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP)*, (Marbella, Spain), pp. 943–948, Sept. 2004.

The author's other publications

[13] **Cs. Benedek**, T. Szirányi, Z. Kato, and J. Zerubia, "A three-layer MRF model for object motion detection in airborne images," Research Report 6208, INRIA Sophia Antipolis, France, June 2007.

[14] **Cs. Benedek** and T. Szirányi, "Detecting built-in changes in airborne images," in *Conference of Hungarian Association for Image Analysis and Pattern Recognititon*, (Debrecen, Hungary), January 2007.

[15] D. Szolgay, **Cs. Benedek**, T. Szirányi, and Z. Vidnyánszky, "On-line statistic generation for visit-frequencies of web advertisements," in *Conference of Hungarian Association for Image Analysis and Pattern Recognititon*, (Debrecen, Hungary), January 2007. In Hungarian.

[16] **Cs. Benedek** and T. Szirányi, "A Markov random field model for foreground-background separation," in *Proc. Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition (HACIPPR)*, (Veszprém, Hungary), May 2005.

[17] **Cs. Benedek** and T. Szirányi, "Tracking pedestrians using gait pattern analysis," in *Conference of Hungarian Association for Image Analysis and Pattern Recognititon*, (Miskolc-Tapolca, Hungary), January 2004. In Hungarian.

Publications connected to the dissertation

[18] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa, "A system for video surveillance and monitoring," Tech. Rep. CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 2000. 1

[19] R. Kumar, R. H. Sawhney, S. Samarasekera, S. Hsu, H. Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hirvonen, M. Hansen, and P. Burt, "Aerial video surveillance and exploitation," in *Proceeding of the IEEE*, vol. 8, pp. 1518–1539, 2001. 1, 5.1

[20] C. Setchell, *Applications of Computer Vision to Road-traffic Monitoring*. PhD thesis, Department of Computer Science, University of Bristol, September 1997. 1

[21] M. Esteve and C. E. Palau, "A flexible video streaming system for urban traffic control," *IEEE MultiMedia*, vol. 13, no. 1, pp. 78–83, 2006. 1

[22] B. C. Arrue, A. Ollero, and J. R. M. de Dios, "An intelligent system for false alarm reduction in infrared forest-fire detection," *IEEE Intelligent Systems*, vol. 15, no. 3, pp. 64–73, 2000. 1

[23] Y. Jia, Y. Liu, H. Yu, and D. Li, "Vegetation change detection based on image fusion technique," in *Proceedings of the SPIE Image Analysis Techniques* (D. Li and H. Ma, eds.), vol. 6044 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pp. 388–395, Nov. 2005. 1

[24] X. Liu and R. Lathrop, "Urban change detection based on an artificial neural network," *International Journal of Remote Sensing*, vol. 23, pp. 2513–2518, June 2002. 1

[25] Y. Kosugi, M. Sakamoto, M. Fukunishi, L. Wei, T. Doihara, and S. Kakumoto, "Robust real-time periodic motion detection, analysis, and applications," *IEEE Geoscience and Remote Sensing Letters*, vol. 1, no. 3, pp. 152 – 156, 2003. 1

[26] E. G. M. Petrakis, A. Diplaros, and E. Milios, "Matching and retrieval of distorted and occluded shapes using dynamic programming," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1501–1516, 2002. 1

[27] F. Lafarge, X. Descombes, J. Zerubia, and M. Pierrot-Deseilligny, "An automatic building reconstruction method : A structural approach using high resolution images," in *Proc. IEEE International Conference on Image Processing (ICIP)*, (Atlanta, USA), October 2006. Copyright IEEE. 1

[28] A. Prati, I. Mikic, M. M. Trivedi, and R. Cucchiara, "Detecting moving shadows: algorithms and evaluation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 7, pp. 918–923, 2003. 1, 3.1.1, 3.1.3, 3.3.1, 3.7.1, 4.1

[29] R. Cucchiara, C. Grana, G. Neri, M. Piccardi, and A. Prati, "The Sakbot system for moving object detection and tracking," in *Video-Based Surveillance Systems-Computer Vision and Distributed Processing*, pp. 145–157, 2001. 1, 4.1, 4.1, 4.3

[30] I. Mikic, P. Cosman, G. Kogut, and M. M. Trivedi, "Moving shadow and object detection in traffic scenes," in *Proc. International Conference on Pattern Recognition*, 2000. (document), 1, 3.1.2, 3.1, 3.3.3.1, 3.4, 3.7.1, 3.9, 3.7.2.1, 3.7.2.2, 3.14, 4.1, 4.1, 4.3

[31] W. K. Pratt, *Digital Image Processing.* No. ISBN 0-471-85766-1, John Wiley & Sons, second ed., 1991. 1, 3.3.4.3

[32] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, 1984. 1, 2, 2.1, 2.4, 7.1

[33] S. Z. Li, *Markov random field modeling in computer vision.* London, UK: Springer-Verlag, 1995. 1

[34] J. Kato, T. Watanabe, S. Joga, L. Ying, and H. Hase, "An HMM/MRF-based stochastic framework for robust vehicle tracking," *IEEE Trans. on Intelligent Transportation Systems*, vol. 5, no. 3, pp. 142–154, 2004. 1, 3.1, 3.1.2

[35] N. Paragios and V. Ramesh, "A MRF-based real-time approach for subway monitoring," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1034–1040, 2001. 1, 3.1.1, 3.1, 4.1

[36] J. Rittscher, J. Kato, S. Joga, and A. Blake, "A probabilistic background model for tracking," in *Proc. European Conf. on Computer Vision*, 2000. 1

[37] J. Rittscher, J. Kato, S. Joga, and A. Blake, "An HMM-based segmentation method for traffic monitoring," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1291–1296, 2002. 1, 2.5.1, 3.1.2, 3.3.1, 3.3.3.1, 3.7.2.2, 4.1, 4.1

[38] Y. Sheikh and M. Shah, "Bayesian modeling of dynamic scenes for object detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1778–1792, 2005. (document), 1, 2.5.1, 3.1, 3.1.1, 3.1.2, 3.1, 3.2, 3.6, 3.7.2.2

[39] Y. Wang and T. Tan, "Adaptive foreground and shadow detection in image sequences," in *Proc. International Conference on Pattern Recognition*, pp. 983–986, 2002. (document), 1, 3.1, 3.1.2, 3.4, 3.4, 3.5.2.1, 3.7.2.2, 3.14

[40] Y. Wang, K.-F. Loe, and J.-K. Wu, "A dynamic conditional random field model for foreground and shadow segmentation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 2, pp. 279–289, 2006. (document), 1, 2.5.1, 3.1, 3.1.2, 3.1.3, 3.1, 3.3.1, 3.3.3.1, 3.3.4.2, 3.3.4.3, 3.2, 3.6, 3.7.1, 3.7.2.2, 3.12, 3.7.2.3, 3.7.3, 4.1, 4.3, 5.1.2

[41] Y. Zhou, Y. Gong, and H. Tao, "Background segmentation using spatial-temporal multi-resolution MRF," in *Workshop on Motion and Video Computing*, pp. 8–13, IEEE, 2005. 1, 3.1, 3.1.1

[42] R. Potts, "Some generalized order-disorder transformation," in *Proceedings of the Cambridge Philosophical Society*, no. 48, p. 106, 1952. 1, 2, 2.5, 3.6, 5.5, 6.4.2

[43] W. Feller, *An introduction to probability theory and its applications*, vol. 1. John Wiley & Sons, second ed., 1966. 1, 5.6.2, A, A.4

[44] Z. Kato, *Modélisations markoviennes multirésolutions en vision par ordinateur. Application ŕ la segmentation d'images SPOT (Multiresolution Markovian models in computer vision. Application on segmentation of SPOT images)*. PhD thesis, INRIA, Sophia Antipolis, France, 1994. Available in French and English. 1, 2, 2.4, 1, A

[45] Z. Kato, T. C. Pong, and G. Q. Song, "Multicue MRF image segmentation: Combining texture and color," in *Proc. of International Conference on Pattern Recognition*, (Quebec, Canada), pp. 660–663, Aug. 2002. 2, 2.3, 3.1.2, 5.1.2, 5.9.5

[46] Z. Kato, T. C. Pong, and G. Q. Song, "Unsupervised segmentation of color textured images using a multi-layer MRF model," in *Proc. of International Conference on Image Processing*, vol. 1, (Barcelona, Spain), pp. 961–964, Sept. 2003. 2, 5.1.2

[47] Z. Kato and T. C. Pong, "Video object segmentation using a multicue Markovian model," in *Proc. Joint Hungarian-Austrian Conference on Image Processing and Pattern Recognition*, (Veszprém, Hungary), pp. 111–118, May 2005. 2, 5.1.2

[48] Z. Kato and T. C. Pong, "A multi-layer MRF model for video object segmentation," in *Proc. Asian Conference on Computer Vision (ACCV), Lecture Notes in Computer Science (LNCS) 3851*, (Hyderabad, India), pp. 953–962, Springer, Jan. 2006. 2, 5.1.2, 5.9.5

[49] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001. 2.4, 3.6

[50] E. Aarts and J. Korst, *Simulated Annealing and Boltzman Machines*. New York: John Wiley & Sons, 1990. 2, 2.4

[51] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, 2004. 2.4

[52] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *Journal of Chem. Physics*, vol. 21, pp. 1087–1092, 1953. 2.4, 5.7

[53] Z. Kato, J. Zerubia, and M. Berthod, "Satellite image classification using a modified Metropolis dynamics," in *Proc. International Conference on Acoustics, Speech and Signal Processing*, pp. 573–576, March 1992. 2.4, 3.6, 5.7

[54] J. Besag, "On the statistical analysis of dirty images," *Journal of Royal Statistics Society*, vol. 48, pp. 259–302, 1986. 2.4, 5.7

[55] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 781–796, 2000. 3.1, 3.1.2

[56] L. Havasi, Z. Szlávik, and T. Szirányi, "Higher order symmetry for non-linear classification of human walk detection," *Pattern Recognition Letters*, vol. 27, pp. 822–829, 2006. 3.1, 3.1.1, 3.1.2

[57] L. Havasi, Z. Szlávik, and T. Szirányi, "Detection of gait characteristics for scene registration in video surveillance system," *IEEE Trans. Image Processing*, vol. 16, no. 2, pp. 503–510, 2007. 3.1

[58] L. Czúni and T. Szirányi, "Motion segmentation and tracking with edge relaxation and optimization using fully parallel methods in the cellular non-linear network architecture," *Real-Time Imaging*, vol. 7, no. 1, pp. 77–95, 2001. 3.1, 3.1.2

[59] Z.Zivkovic, *Motion Detection and Object Tracking in Image Sequences*. PhD thesis, PhD thesis, University of Twente, 2003. 3.1

[60] J. B. Hayfron-Acquah, M. S. Nixon, and J. N. Carter, "Human identification by spatio-temporal symmetry," in *Proc. International Conference on Pattern Recognition*, vol. 1, (Washington, DC, USA), p. 10632, IEEE Computer Society, 2002. 3.1

[61] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1505–1518, 2003. 3.1

[62] C. Stauffer and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747–757, 2000. 3.1, 3.1.1, 3.1.2, 3.3.2, 3.3.2, 3.3.2, 3.3.2, 3.3.3.2, 3.5.1, A.3

[63] N. Friedman and S. Russell, "Image segmentation in video sequences: A probabilistic approach," in *Proc. Conf. on Uncertainty in Artificial Intelligence*, pp. 175–181, 1997. 3.1

[64] M. Harville, G. G. Gordon, and J. Woodfill, "Foreground segmentation using adaptive mixture models in color and depth," in *IEEE Workshop on Detection and Recognition of Events in Video*, pp. 3–11, 2001. 3.1

[65] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh, "Background modeling and subtraction of dynamic scenes," in *Proc. IEEE International Conference on Computer Vision*, vol. 2, (Washington, DC, USA), pp. 1305–1312, IEEE Computer Society, 2003. 3.1, 3.1.2

[66] S. C. Zhu and A. L. Yuille, "A flexible object recognition and modeling system," *Int'l Journal of Computer Vision*, vol. 20, no. 3, 1996. 3.1.1

[67] L. Havasi and T. Szirányi, "Estimation of vanishing point in camera-mirror scenes using video," *Optics Letters*, vol. 31, no. 10, pp. 1411–1413, 2006. 3.1.1

[68] A. Yoneyama, C. H. Yeh, and C.-C. J. Kuo, "Moving cast shadow elimination for robust vehicle extraction based on 2D joint vehicle/shadow models," in *IEEE Conference on Advanced Video and Signal Based Surveillance*, IEEE, 2003. 3.1.1

[69] C. Fredembach and G. D. Finlayson, "Hamiltonian path based shadow removal," in *Proc. British Machine Vision Conference*, pp. 970–980, 2005. 3.1.1

[70] D. Finlayson, S. D. Hordley, C. Lu, and M. S. Drew, "On the removal of shadows from images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 59–68, 2006. 3.1.1, 3.3.3.1

[71] T. Gevers and H. Stokman, "Classifying color edges in video into shadow-geometry, highlight, or material transitions," *IEEE Trans. on Multimedia*, vol. 5, no. 2, pp. 237–243, 2003. 3.1.1

[72] E. A. Khan and E. Reinhard, "Evaluation of color spaces for edge classification in outdoor scenes," in *Proc. of International Conference on Image Processing*, vol. 3, (Genoa, Italy), pp. 952–955, IEEE, Sept. 2005. 3.1.1, 3.3.3.2, 4.1

[73] A. Cavallaro, E. Salvador, and T. Ebrahimi, "Detecting shadows in image sequences," in *Proc. of European Conference on Visual Media Production*, pp. 167–174, March 2004. 3.1.1, 4.1

[74] E. Salvador, A. Cavallaro, and T. Ebrahimi, "Cast shadow segmentation using invariant color features," *Computer Vision and Image Understanding*, no. 2, pp. 238–259, 2004. (document), 3.1.1, 3.1.3, 3.1, 3.9, 3.7.2.1, 4.1, 4.4.1

[75] F. Porikli and J. Thornton, "Shadow flow: a recursive method to learn moving cast shadows," in *Proc. IEEE International Conference on Computer Vision*, vol. 1, pp. 891–898, 2005. 3.1.1

[76] N. Martel-Brisson and A. Zaccarin, "Moving cast shadow detection from a gaussian mixture shadow model," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 643–648, June 2005. (document), 3.1.1, 3.1, 3.2, 3.6, 4.1, 4.1

[77] S. Chaudhuri and D. Taur, "High-resolution slow-motion sequencing: How to generate a slow-motion sequence from a bit stream," *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 16–24, 2005. 3.1.2

[78] A. Yilmaz, X. Li, and M. Shah, "Object contour tracking using level sets," in *Proc. Asian Conference on Computer Vision, (ACCV 2004)*, (Jaju Islands, Korea), 2004. 3.1.2, 3.7.2.2

[79] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr, "Interactive image segmentation using an adaptive GMMRF model," in *Proc. European Conference on Computer Vision*, pp. 456–468, Springer, 2004. 3.1.2

[80] M. Heikkila and M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 657–662, 2006. 3.1.3

[81] J. Zhong and S. Sclaroff, "Segmenting foreground objects from a dynamic textured background via a robust Kalman filter," in *Proc. IEEE International Conference on Computer Vision*, pp. 44–50, 2003. 3.1.3

[82] L. Li and M. Leung, "Integrating intensity and texture differences for robust change detection," *IEEE Trans. on Image Processing*, vol. 11, no. 2, pp. 105–112, 2002. 3.1.3, 5.1.2

[83] Y. Haeghen, J. Naeyaert, I. Lemahieu, and W. Philips, "An imaging system with calibrated color image acquisition for use in dermatology," *IEEE Transactions on Medical Imaging*, vol. 19, no. 7, pp. 722–730, 2000. 3.1.3

[84] M. G. A. Thomson, R. J. Paltridge, T. Yates, and S. Westland, "Color spaces for discrimination and categorization in natural scenes," in *Proc. Congress of the International Colour Association*, pp. 877–880, June 2002. 3.1.3, 3.3.1, 4.4

[85] T. Gevers and A. W. Smeulders, "Color based object recognition," *Pattern Recognition (PR)*, vol. 32, pp. 453–464, 1999. 3.1.3

[86] D. S. Lee, "Effective gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, pp. 827–832, 2005. 3.3.2

[87] D. A. Forsyth, "A novel algorithm for color constancy," *International Journal of Computer Vision*, vol. 5, no. 1, pp. 5–36, 1990. 3.3.3, 3.3.3.1, 3.3.3.1

[88] D. K. Lynch and W. Livingstone, *Color and Light in Nature*. Cambridge University Press, 1955. 3.3.3.1

[89] G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulas*. John Wiley & Sons, second ed., 1982. 3.3.3.1

[90] R. Haralick, "Digital step edges from zero crossing of second directional derivatives," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 58–68, 1984. (document), 3.3.4.3, 3.3

[91] C. J. V. Rijsbergen, *Information Retrieval*. London: Butterworths, second ed., 1979. 3.7.3

[92] V. Meas-Yedid, E. Glory, E. Morelon, C. Pinset, G. Stamon, and J.-C. Olivo-Marin, "Automatic color space selection for biological image segmentation," in *Proc. International Conference on Pattern Recognition*, vol. 3, (Washington, DC, USA), pp. 514–517, IEEE Computer Society, 2004. 4.1

[93] P. Guo and M. R. Lyu, "A study on color space selection for determining image segmentation region number," in *Proc. International Conference on Artificial Intelligence*, vol. III, (Las Vegas, Nevada, USA), pp. 1127–1132, June 2000. 4.1

[94] A. Prati, I. Mikic, C. Grana, and M. Trivedi, "Shadow detection algorithms for traffic flow analysis: A comparative study," in *Proc. IEEE Intelligent Transportation Systems Conference*, (Oakland, CA, USA), pp. 340–345, 2001. 4.1

[95] M. Rautiainen, T. Ojala, and H. Kauniskangas, "Detecting perceptual color changes from sequential images for scene surveillance," *IEICE Transactions on Information and Systems*, pp. 1676 – 1683, 2001. 4.1, 4.1

[96] K. Siala, M. Chakchouk, F. Chaieb, and O. Besbes, "Moving shadow detection with support vector domain description in the color ratios space," in *Proc. International Conference on Pattern Recognition*, vol. 4, pp. 384–387, 2004. 4.2, 4.1, 4.3, 4.3

[97] M. Tkalcic and J. Tasic, "Colour spaces - perceptual, historical and applicational background," in *Proc. Eurocon*, 2003. 4.2

[98] S. T. Barnard and W. B. Thompson, "Disparity analysis of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 2, pp. 333–340, 1980. 5.1.1

[99] J. K. Cheng and T. S. Huang, "Image registration by matching relational structures," *Pattern Recognition*, vol. 17, pp. 149–159, 1984. 5.1.1

[100] I. Miyagawa and K. Arakawa, "Motion and shape recovery based on iterative stabilization for modest deviation from planar motion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1176–1181, 2006. 5.1.1

[101] J. Weng, N. Ahuja, and T. S. Huang, "Matching two perspective views," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, pp. 806–825, 1992. 5.1.1

[102] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry," *Artificial Intelligence Journal*, vol. 78, pp. 87–119, 1995. 5.1.1, 5.1.1, 5.2.1

[103] H. Hirschmüller, P. R. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *International Journal of Computer Vision*, vol. 47, no. 1/2/3, pp. 229–246, 2002. 5.1.1

[104] H. Hirschmüller, F.Scholten, and G. Hirzinger, "Stereo vision based reconstruction of huge urban areas from an airborne pushbroom camera (HRSC)," in *Proc. 27th DAGM Symposium*, (Vienna, Austria), pp. 58–66, LNCS 3663, Sept 2005. 5.1.1

[105] Z. Szlávik, T. Szirányi, and L. Havasi, "Stochastic view registration of overlapping cameras based on arbitrary motion," *IEEE Trans. Image Processing*, vol. 16, no. 3, pp. 710–720, 2007. 5.1.1

[106] M. Irani and P. Anandan, "A unified approach to moving object detection in 2D and 3D scenes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no. 6, pp. 577–589, 1998. 5.1.1

[107] B. Reddy and B. Chatterji, "An FFT-based technique for translation, rotation and scale-invariant image registration," *IEEE Trans. on Image Processing*, vol. 5, no. 8, pp. 1266–1271, 1996. 5.1.1, 5.2.2, 5.2.2

[108] L. Lucchese, "Estimating affine transformations in the frequency domain," in *Proc. Int. Conf. On Image Processing*, vol. II, (Thessaloniki, Greece), pp. 909–912, Sept. 2001. 5.1.1

[109] S. Kumar, M. Biswas, and T. Nguyen, "Global motion estimation in spatial and frequency domain," in *Proc. IEEE International Conference on Acoustics,Speech,and Signal Processing*, (Montreal, Canada), pp. 333–336, May 2004. 5.1.1, 5.9.2

[110] H. Shekarforoush, M. Berthod, and J. Zerubia, "Subpixel image registration by estimating the polyphase decomposition of cross power spectrum," in *Proceedings of the 1996 Conference on Computer Vision and Pattern Recognition*, (Washington, DC, USA), p. 532, IEEE Computer Society, 1996. 5.1.1

[111] R. I. Hartley and A. Zissermann, *Multiple View Geometry in Computer Vision*. Cambridge: Cambridge University Press, 2000. 5.1.1, 5.1.1, 5.2.1

[112] H. Sawhney, Y. Guo, and R. Kumar, "Independent motion detection in 3D scenes," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1191–1199, 2000. 5.1.1

[113] J.-Y. Bouguet, "Pyramidal implementation of the Lucas Kanade feature tracker: Description of the algorithm," tech. rep., Intel Corporation, 1999. 5.2.1

[114] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of International Joint Conference on Artificial Intelligence*, (Vancouver, BC, Canada), pp. 674–679, August 1981. 5.2.1

[115] J. M. Odobez and P. Bouthemy, "Detection of multiple moving objects using multiscale MRF with camera motion compensation," in *Proc. Int. Conf. On Image Processing*, vol. II, (Austin, Texas, USA), pp. 257–261, 1994. 5.1.1

[116] R. Pless, T. Brodsky, and Y. Aloimonos, "Detecting independent motion: The statistics of temporal continuity," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 68–73, 2000. 5.1.1

[117] D. Farin and P. With, "Misregistration errors in change detection algorithms and how to avoid them," in *Proc. International Conference on Image Processing (ICIP)*, (Genova, Italy), pp. 438–441, Sept. 2005. (document), 5.1.1, 5.9.2, 5.3

[118] H. El-Askary, A. Agarwal, T. El-Ghazawi, M. Kafatos, and J. L. Moigne, "Enhancing dust storm detection using PCA based data fusion," in *Proc. Geoscience and Remote Sensing Symposium*, vol. 2, pp. 1424–1427, July 2005. 5.1.2

[119] Q. Iqbal and J. K. Aggarwaly, "Feature integration, multi-image queries and relevance feedback in image retrieval," in *Proc. International Conference on Visual Information Systems*, (Miami, Florida), pp. 467–474, Sept. 2003. 5.1.2

[120] Z. Kato and T. C. Pong, "A Markov random field image segmentation model for color textured images," *Image and Vision Computing*, vol. 24, no. 10, pp. 1103–1114, 2006. 5.1.2

[121] S. Khan and M. Shah, "Object based segmentation of video using color, motion and spatial information," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, (Hawaii, USA), pp. 746–751, May 2001. 5.1.2

[122] A. Kushki, P. Androutsos, K. Plataniotis, and A. Venetsanopoulos, "Retrieval of images from artistic repositories using a decision fusion framework," *IEEE Trans. on Image Processing*, vol. 13, no. 3, pp. 277–292, 2004. 5.1.2

[123] E. Saber and A. Tekalp, "Integration of color, edge, shape, and texture features for automatic region-based image annotation and retrieval," *Journal of Electronic Imaging*, vol. 7, no. 3, pp. 684–700, 1998. 5.1.2

[124] P.-M. Jodoin and M. Mignotte, "Motion segmentation using a K-nearest-neighbor-based fusion procedure, of spatial and temporal label cues," in *Proc. of International Conference on Image Analysis and Recognition*, (Toronto, Canada), pp. 778–788, 2005. 5.1.2

[125] *OpenCV documentation*. 5.2.1, 7.1

[126] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, vol. 1, (Hawaii, USA), pp. 511–518, 2001. 5.8

[127] C. Sun, "Fast stereo matching using rectangular subregioning and 3D maximum-surface techniques," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 99–117, 2002. 5.8

[128] D. Oram, "Rectification for any epipolar geometry," in *Proc. British Machine Vision Conference*, (London, UK), pp. 653–662, 2001. 5.8

[129] S. Kumar and U. Desai, "New algorithms for 3D surface description from binocular stereo using integration," *Journal of the Franklin Institute*, vol. 331B, no. 5, pp. 531–554, 1994. 5.8.1

[130] S. Ghosh, L. Bruzzone, S. Patra, F. Bovolo, and A. Ghosh, "A context-sensitive technique for unsupervised change detection based on hopfield-type neural networks," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 45, pp. 778–789, March 2007. 6.1

[131] R. Wiemker, "An iterative spectral-spatial bayesian labeling approach for unsupervised robust change detection on remotely sensed multispectral imagery," in *Proc. Int. Conf. on Computer Analysis of Images and Patterns*, vol. LNCS 1296, (Kiel, Germany), pp. 263–270, 1997. (document), 6.1, 6.6, 1, 6.3, 6.5

[132] L. Bruzzone and D. F. Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Trans. on Image Processing*, vol. 11, no. 4, pp. 452–466, 2002. 6.1

[133] Y. Bazi, L. Bruzzone, and F. Melgani, "An unsupervised approach based on the generalized gaussian model to automatic change detection in multitemporal SAR images," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 43, pp. 874–887, April 2005. 6.1

[134] P. Gamba, F. Dell'Acqua, and G. Lisini, "Change detection of multitemporal SAR data in urban areas combining feature-based and pixel-based techniques," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 44, no. 10, pp. 2820–2827, 2006. 6.1

[135] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: A systematic survey," *IEEE Trans. on Image Processing*, vol. 14, no. 3, pp. 294–307, 2005. 6.1

[136] A. Lorette, X. Descombes, and J. Zerubia, "Texture analysis through a Markovian modelling and fuzzy classification: Application to urban area extraction from satellite images," *International Journal of Computer Vision*, vol. 36, no. 3, pp. 221–236, 2000. 6.2

[137] A. Ronsenfeld and E. B. Troy, "Visual texture analysis," in *Proc. UMR–Mervin J. Kelly Communications Conference*, 1970. 6.3