Master's Thesis

Recovery of Depth Information Using Paired Optical and Thermal Images

Peter Pinggera

Institute for Computer Graphics and Vision

Graz University of Technology, Austria

Supervisor: Univ.-Prof. Dipl.-Ing. Dr.techn. Horst Bischof

Applied Mathematics and Computing Group, School of Engineering

Cranfield University, United Kingdom

Supervisor: Dr. Toby Breckon

Graz, October 2011

Abstract

This thesis deals with the recovery of dense depth information from thermal (far infrared spectrum) and optical (visible spectrum) images using computational stereo techniques. Systems which originally employ optical and thermal cameras separately could benefit from the obtained depth information based on the inherent stereo setup and without the need for additional hardware. However, the large differences in the characteristics of cross-spectral images make this task significantly more difficult than for the common optical stereo case. As a result no method has been proposed in previous work which is able to solve the considered problem. In this work we therefore investigate if a solution can be achieved by utilizing novel approaches as well as methods suggested in literature.

A modular framework based on a common taxonomy of stereo algorithms is implemented as a basis for the conducted experiments. The most crucial aspect is the definition of robust matching cost measures which are able to describe local similarities between the cross-spectral images. Furthermore powerful optimization techniques prove to be essential for the computation of valid depth estimates.

We implement, test and evaluate state-of-the-art robust matching cost methods and compare their performance with novel approaches. The influence of combinations with different types of optimization techniques is also investigated. Tests are performed on simulated as well as real cross-spectral stereo data, including both still images and video sequences. A qualitative evaluation and a comparison with standard optical stereo results shows that through the introduced approaches very coarse but largely valid dense depth estimates can indeed be achieved. We obtain best results by using distances between dense descriptors based on histograms of unsigned oriented image gradients (HOG and DAISY descriptors) as a matching cost in combination with semi-global matching optimization. In all our experiments this approach outperforms methods which have previously been suggested for use in such a scenario like mutual information or dense local self-similarity descriptors.

Kurzfassung

Die vorliegende Arbeit befasst sich mit der Rekonstruktion von dichter Tiefeninformation aus Wärmebildern (langwelliges Infrarot-Spektrum) und optischen Bildern (sichtbares Spektrum) unter Verwendung von stereoskopischen Berechnungsmethoden. Systeme, welche optische Bilder und Wärmebilder ursprünglich getrennt einsetzen, könnten basierend auf einer vorhandenen Stereo-Anordnung ohne zusätzlichen Hardwareaufwand von der resultierenden Tiefeninformation profitieren. Die Aufgabe wird jedoch aufgrund der stark unterschiedlichen Bildeigenschaften im Vergleich zu üblichen rein optischen Stereo-Anordnungen stark erschwert. Infolgedessen wurde bisher noch keine Lösungsmethode für das beschriebene Problem vorgestellt. In dieser Arbeit wird daher untersucht, ob eine Lösung des Problems durch die Anwendung von sowohl neuartigen Methoden als auch in der Literatur vorgeschlagenen Ansätzen erreicht werden kann.

Als Basis für die durchgeführten Untersuchungen wird ein modulares Framework basierend auf einer allgemeinen Systematik für Stereo-Algorithmen implementiert. Der wichtigste Aspekt ist die Definition von robusten Vergleichsmaßen, welche in der Lage sind, lokale Ähnlichkeiten zwischen den betrachteten spektral unterschiedlichen Bildern zu beschreiben. Darüber hinaus erweisen sich leistungsfähige Optimierungsmethoden als wesentlich für die Berechnung von gültigen Tiefenschätzungen.

Aktuelle robuste Vergleichsmaße werden implementiert, getestet und ihre Leistung mit neuartigen Methoden verglichen. Zusätzlich werden die Auswirkungen der Kombination mit verschiedenen Optimierungsmethoden untersucht. Für die Durchführung der Tests werden simulierte und tatsächlich spektral unterschiedliche Stereo-Daten verwendet, wobei sowohl Bilder als auch Videosequenzen berücksichtigt werden. Eine qualitative Evaluierung und ein Vergleich mit rein optischen Stereo-Ergebnissen zeigt, dass durch die vorgestellten Ansätze tatsächlich eine grobe aber weitgehend gültige dichte Tiefenabschätzung erzielt werden kann. In den durchgeführten Experimenten werden die besten Ergebnisse durch Distanzen zwischen Deskriptoren basierend auf Histograms of unsigned oriented Image Gradients (HOG und DAISY Deskriptoren) als Vergleichsmaß in Kombination mit Semi-Global Matching Optimierung erreicht. Dieser Ansatz liefert dabei bessere Resultate als in der Literatur für dieses Szenario vorgeschlagene Methoden wie Mutual Information oder Local Self-Similarity Deskriptoren.

Deutsche Fassung:
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008
Genehmigung des Senates am 1.12.2008

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am ....
(Unterschrift)

Englische Fassung:

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

....
date (signature)

Double Degree Programme

This thesis was created within the framework and regulations of the Double Degree Programme between Cranfield University and Graz University of Technology. Modified versions[1] of this work are submitted to both Universities in partial fulfillment of the requirements for the degree of Master of Science.

Cranfield University
UK

Supervisor:
Dr. Toby Breckon

Applied Mathematics and
Computing Group
School of Engineering

Graz University of Technology
Austria

Supervisor:
Univ.-Prof. Dipl.-Ing. Dr.techn.
Horst Bischof

Institute for Computer Graphics
and Vision

[1] Additional tests were performed at Graz University of Technology and the results and conclusions integrated into the present version of this thesis.

Acknowledgements

I would like to express my gratitude to my supervisor at Cranfield University, Dr. Toby Breckon, for his ongoing guidance throughout the whole course of this thesis. At the same time I would like to sincerely thank my supervisor at Graz University of Technology, Univ.-Prof. Dipl.-Ing. Dr. Horst Bischof, for his valuable advice and support.

My deepest gratitude goes to my family and especially my parents Elfriede and Edwin for supporting me throughout all my studies and making it possible for me to create this thesis.

Contents

List of Figures

List of Tables

Abbreviations

AD	Absolute Differences
DP	Dynamic Programming
DSI	Disparity Space Image
GC	Graph Cuts
HOG	Histogram of Oriented Gradients
IR	Infrared
LoG	Laplacian of Gaussian
LSS	Local Self-Similarity
MI	Mutual Information
NCC	Normalized Cross Correlation
SAD	Sum of Absolute Differences
SGM	Semi-Global Matching
SO	Scanline Optimization
SSD	Sum of Squared Differences
WTA	Winner-Takes-All
ZNCC	Zero Mean Normalized Cross Correlation
ZSAD	Zero Mean Sum of Absolute Differences

Chapter 1

Introduction

1.1 Motivation

Depth from Stereo Vision The perception of depth has been an important topic in the fields of computer vision for several decades. It has been the focus of very active research and a number of different approaches for the recovery of depth information of observed scenes have been investigated [48][49]. A very attractive and popular approach is computational stereo, which was originally inspired by binocular stereopsis utilized in the human visual system [50]. It is based on the perception of depth from observing a scene from two slightly different points of view. In contrast to active methods for depth recovery like radar, laser range finders or structured light techniques which may offer greater accuracy, the computational stereo method is a completely passive method for depth recovery. It is suitable in almost all application domains and apart from two cameras does not rely on additional special equipment. Depth information provided by computational stereo methods is commonly used to enhance the performance of many systems in application areas including:

- Robotic navigation and manipulation
- Obstacle detection and avoidance in autonomous vehicles
- Object/Person detection and tracking
- Industrial quality inspection
- Intelligent driver assistance

An introduction to the basic principles of computational stereo will be given in Section 1.3.

Cross-Spectral Stereo Vision The performance of standard optical camera systems can be severely affected by environmental conditions like low lighting, shadows, smoke and dust or when objects of interest have an appearance similar to the background [14]. A method to overcome such problems is to use camera systems operating in different parts of the electromagnetic spectrum. For example infrared images,

often called thermal images, are independent of visible light illumination and shadows, are relatively robust to dust and smoke and can often distinguish objects which look similar to the background in the visible spectrum. However, thermal images are often affected by ambient temperature and can offer difficulty in identifying objects with a similar temperature to the background (ambient temperature). As a result an attractive solution is the combination of both optical and thermal images in many common sensing and surveillance scenarios. In this way the advantages and complementary nature of both modalities can be exploited and the individual drawbacks can be largely compensated.
Numerous practical applications in both civil and military scenarios make use of this concept, including many of the areas listed in the previous paragraph (e.g. [47][14][11][12]). One possibility is to simply alternate between the use of optical and thermal cameras, for example to switch from optical to thermal imagery for night- or low-light-vision. More sophisticated approaches use both modalities simultaneously when the circumstances permit it and employ sensor fusion methods to combine the information acquired from the different images [14]. However, at the same time the inherent stereo setup resulting from the use of the two cameras is hardly ever exploited or a separate optical stereo setup is used to recover desired depth information. In fact the cross-spectral stereo setup often even forms a problem for sensor fusion and registration algorithms [32]. The direct recovery of depth information from the cross-spectral stereo setup could be used to improve applications such as obstacle avoidance or object detection and tracking approaches without the need for additional hardware which might be important for mobile sensing platforms/applications.

In the literature, the terms multi-spectral, cross-spectral or multi-modal are often used to generally refer to the use of systems combining images acquired in different spectral bands. In this work we will use the term cross-spectral for the combination of standard optical (visible) and thermal (infrared) images. Details on thermal imagery are given in the next section.

1.2 Infrared Imaging

Infrared radiation can be loosely defined as electromagnetic radiation with a wavelength longer than visible light and shorter than microwave radiation [36]. For the purpose of infrared imaging this loose definition needs to be refined and subdivided into several sub bands. The categorization utilized is based on sensor technology and atmospheric transmission properties and is summarized in Table 1.1. We can see the denomination of the different sub bands of the infrared spectrum and their respective wavelengths, bounded at the lower end by the visible spectrum.
Example images of a human head captured at different wavelengths are depicted in Figure 1.1, indicating the structures and features that are visible in the different spectral bands. Finer details can be seen in the visible and near infrared bands but shadows and specular reflections are also present. The mid- and long-wavelength infrared bands capture main facial features irrespective of the lighting conditions.

Spectral band	Spectral wavelength in μm
Visible light	0.4-0.78
Near infrared (NIR)	0.78-1.0
Short-wavelength infrared (SWIR)	1-3
Mid-wavelength infrared (MWIR)	3-5
Long-wavelength infrared (LWIR)	8-12

Table 1.1: General spectral bands based on sensor technology and atmospheric transmission [45].

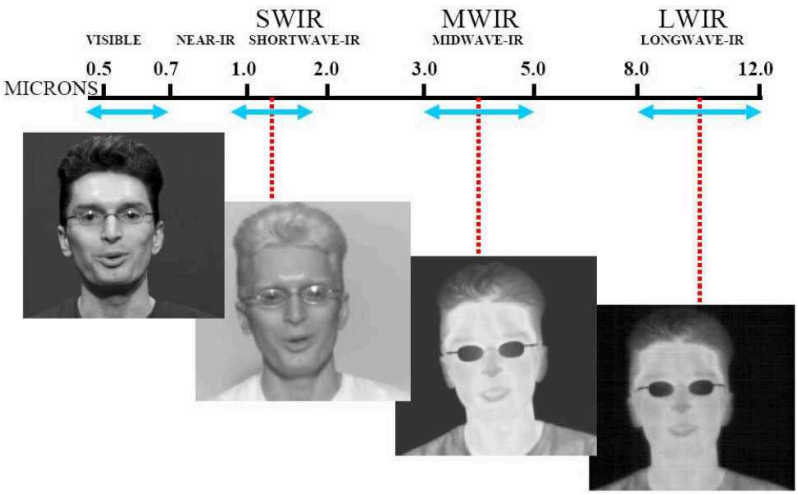

Figure 1.1: Images of a human head at different infrared wavelengths [1].

Practical applications of infrared imaging often use sensors sensitive in the long-wavelength band (LWIR), also called far infrared or thermal imaging. In this band the effects of reflections on objects are minimized and captured images mainly register emissions from the observed scene, independent of visible light illumination [22]. In this work we will consider images taken in the far infrared spectrum and will use the term infrared (IR) as referring to this specific band. Figure 1.2 shows two examples of outdoor images captured with our cross-spectral stereo setup consisting of an uncooled far infrared camera and a standard camera operating in the visible spectrum. The significant differences in the image characteristics can be seen clearly. Most optical texture details, shadows and specular reflections from the visible spectrum are not present in thermal images, while other features like thermal signatures can only be seen in the far infrared images.
Originally image processing on infrared images proved to be difficult due to the low resolution and high noise of early infrared cameras. However, in recent years improvements in sensor technology have allowed for images of higher quality and, while still expensive, good quality infrared cameras are likely to become more affordable in the near future.
Further details on infrared radiation and fundamentals of sensor technologies can be found in [36] and [45].

Figure 1.2: Examples of far infrared (left) and corresponding visible grayscale outdoor images (right).

1.3 Computational Stereo

Computational stereo is generally defined as the recovery of the 3D structure of a scene from two images taken from different points of view [2][10]. It is based on the fact that a point in the 3D scene is projected to different points in the two images. Given that these two corresponding points in the images can be identified and the relative camera positions are known, it is possible to reconstruct the coordinates of the original point in 3D space. Of course this concept is restricted to points which are actually visible in both images. It can be extended to consider more than two views of the scene, which is known as multi-view stereo. In this work we will only consider the standard two-view stereo case using cameras in a horizontal setup. Trying to solve the computational stereo problem involves three main stages [10]:

1. Calibration and rectification:
   The camera parameters are determined and the input images are prepared for the correspondence stage (Section 1.3.1).
2. Correspondence:
   Corresponding points in the images are searched for (Section 1.3.2).
3. Reconstruction:
   Depth information is computed from the determined correspondences and the camera parameters (Section 1.3.3).

The processing-pipeline relationship of these three stages is outlined in Figure 1.3 with respect to two images taken from a standard horizontal stereo setup (left and right images). These aspects of computational stereo are further discussed in the following subsections.

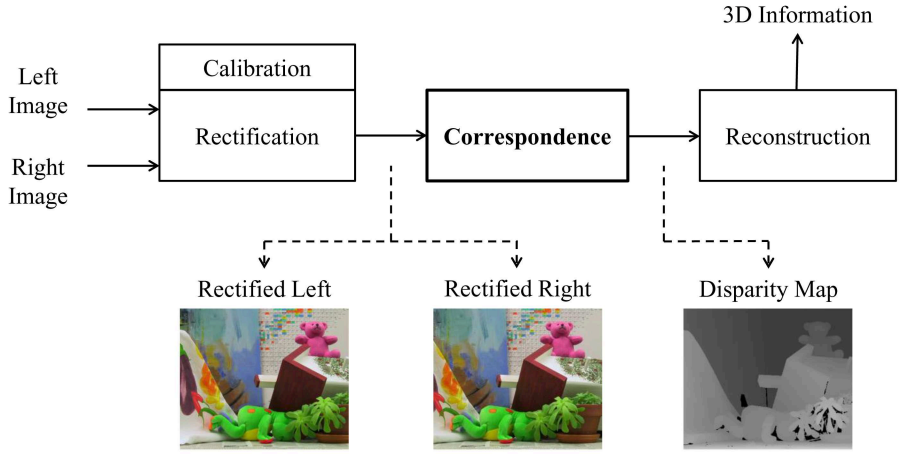

Figure 1.3: The computational stereo pipeline.

1.3.1 Calibration and Rectification

Calibration is used to determine the intrinsic and extrinsic parameters of a stereo camera setup. The intrinsic parameters of each camera include the focal length, the principal point, the skew coefficient and lens distortion coefficients. The extrinsic parameters of the stereo setup describe the relative positions of the cameras to each other. These are generally represented by a rotation matrix and a translation vector (see $\boldsymbol{R}$ and $\boldsymbol{t}$ in Figure 1.4).
A number of different methods to perform calibration exist, however the most common for use in stereo applications are usually based on control points from several images of a calibration target with known geometry (e.g. [59], see also Figure 4.4 in Section 4.1.2). A detailed explanation of the respective parameters and their computation can be found in [23] and [58].

Calibration is essential for both the correspondence and the reconstruction step. To understand its importance for the correspondence step, we first have to examine the well-known concept of epipolar geometry. The epipolar geometry is used to describe the relationship between corresponding image points in a stereo setup. As an example we assume that a point $\boldsymbol{p}$ in 3D space is projected to a point $\boldsymbol{x}_0$ in the left image and $\boldsymbol{x}_1$ in the right image (see Figure 1.4). We can see that the points $\boldsymbol{p}$, $\boldsymbol{x}_0$, $\boldsymbol{x}_1$ and the camera centers $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$ are coplanar. This plane, called the epipolar plane, can therefore be defined by the three points $\boldsymbol{p}$, $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$. Another way of defining the epipolar plane is to use the ray which is projected from one of the camera centers through the respective image point and the baseline, which is the connection between the two camera centers $\boldsymbol{c}_0$ and $\boldsymbol{c}_1$. The baseline intersects the two image planes at the epipoles $\boldsymbol{e}_0$ and $\boldsymbol{e}_1$ respectively. More detail on the underlying geometry can be found in [23].
The main benefit for computational stereo algorithms that can be drawn from the epipolar geometry is the resulting constraint for the search for corresponding points. As can be seen in Figure 1.4, the epipolar plane intersects the image planes at the

epipolar lines $\boldsymbol{l}_0$ and $\boldsymbol{l}_1$. If we now have to find a point $\boldsymbol{x}_1$ in the right image which corresponds to a given point $\boldsymbol{x}_0$ in the left image, we can restrict the search range to only include all points lying on the epipolar line $\boldsymbol{l}_1$. This simplifies the search for correct point matches and significantly reduces the computational complexity.
However, searching for corresponding points along arbitrary epipolar lines in two images is not convenient in practical applications. This problem can be solved by first warping the images, such that epipolar lines correspond to aligned horizontal scanlines of the images (i.e. horizontal image axes). In this way the correspondence search is confined to corresponding rows in the two warped input images (see Figure 4.6 for an example). The process of warping the input images is called rectification and the transformations necessary for rectification can be computed using the parameters previously determined in the calibration step [37][20]. The resulting virtually modified stereo setup has a so called standard rectified geometry [50] (see Figure 1.5). In addition to the rectification process, lens distortion effects of the two cameras can be removed using the previously determined distortion coefficients.

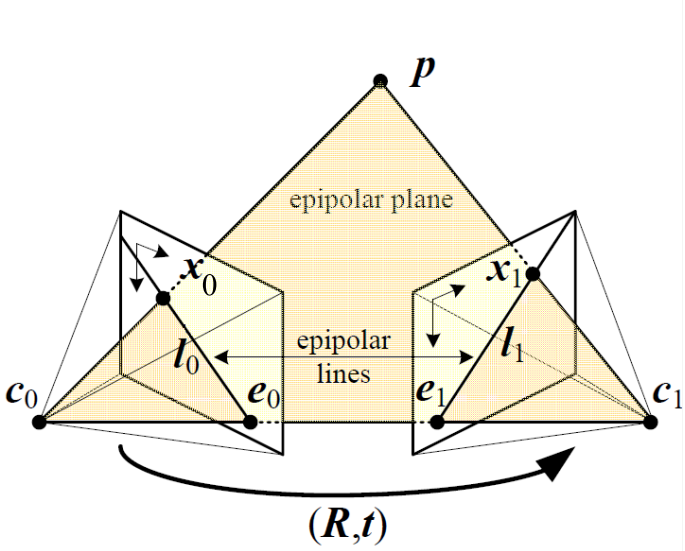

Figure 1.4: Epipolar geometry of a general stereo setup [50].

1.3.2 Correspondence

Assuming a pair of rectified stereo images as input, the correspondence step aims to identify the points in the corresponding rows of the two images which are projections of the same point in the 3D scene. The difference in x-coordinates of the corresponding points in both images is called disparity. Equation 1.1 shows this relationship between coordinates of corresponding points $\boldsymbol{x_0} = (x_0, y_0)$ in the left and $\boldsymbol{x_1} = (x_1, y_1)$ in the right image, where d is the disparity and $y_0 = y_1$.

$$x_1 = x_0 - d \tag{1.1}$$

The correspondence search is usually not carried out over complete image rows but restricted to the maximum disparity range occurring in the image pair. After calculating disparity values over the two images, a so-called disparity map can be created. The disparity values are inversely proportional to the actual 3D depth of the scene points thus objects closer to the cameras yield larger disparity values (usually displayed brighter in visualizations of disparity maps, see Figure 1.3).

Generally a fundamental distinction between sparse and dense correspondence has to be made. Sparse stereo correspondence algorithms are based on the matching of sparse features which have to be extracted from the input images first. Disparity values are then only calculated at these feature points. Many early stereo algorithms were based on this concept due to the reduced computational complexity but also to limit correspondences to the ones with high reliability [50]. Of course, this relies on the assumption that corresponding highly reliable features can indeed be extracted from both images. Furthermore, the incorporation of additional constraints on the disparities, like smoothness constraints, is hard for sparse features. The resulting sparse disparity values can be useful for certain applications but many modern scenarios require dense disparity information. Even though dense disparity maps can be interpolated from sparse matches, this represents a hard problem on its own and therefore the majority of current correspondence algorithms directly compute dense disparity maps [50][10][44][35]. The aim of this work is the creation of dense disparity maps where disparities are calculated for every image pixel and we will focus our investigations on the direct computation using dense correspondence algorithms.

In practice, solving the correspondence problem and identifying corresponding images points is the most challenging part of computational stereo algorithms and no general solution exists [10]. This is due to the ambiguity of potential correspondences in optical images, caused by e.g. lack of texture, repetitive patterns, radiometric differences and partial occlusions. In general solving the correspondence problem is seen as a computational optimization problem and many commonplace optimization techniques are employed for this task in state-of-the-art approaches [44][50]. The difficulty of correspondence search is further increased significantly if images of different spectra are considered, as is the topic of this work. Thus the main focus of this thesis lies on the correspondence problem and its solution across cross-spectral image pairs.

1.3.3 Reconstruction

The purpose of the reconstruction step is the computation of actual 3D structural information from the previously computed disparity values. Given the stereo setup in standard rectified geometry (Figure 1.5) which is achieved through calibration and rectification, the depth Z of a 3D point $\boldsymbol{p}$ in the scene can be calculated very simply using similar triangles:

$$Z = f\frac{B}{d} \tag{1.2}$$

where f is the focal length, B is the baseline and d is the disparity $x_0 - x_1$.

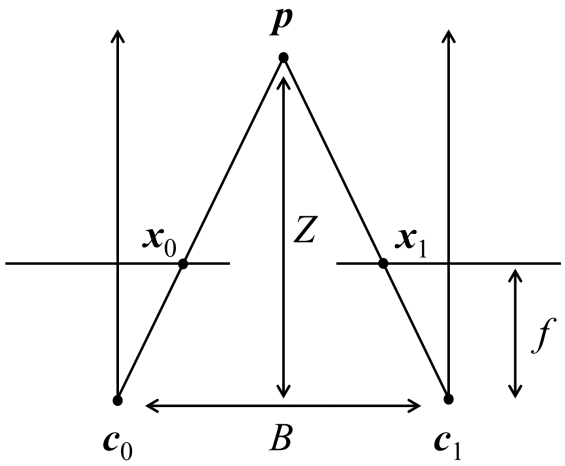

Figure 1.5: Stereo setup in standard rectified form (cf. [10]).

1.4 Problem Definition and Methodology

After this introductory chapter we can now refine the problem definition considered in the present thesis. Our goal is to recover dense depth information from cross-spectral stereo pairs (e.g. Figure 1.2), using dense stereo correspondence algorithms. Considering the possible application areas which motivate this goal, the need for computational efficiency becomes apparent. A desired algorithm would therefore (a) be able to actually produce valid dense depth estimates from cross-spectral stereo and (b) be computationally efficient (ideally being able to operate in real-time or near real-time).

Due to the challenging nature of the problem it has to be stated that we will focus on the creation of coarse depth estimates of a scene and do not attempt to produce highly accurate disparity maps of the kind that is required for applications such as image-based rendering or 3D model building.

Our general methodology in investigating the specified problem can be roughly described as follows. We will begin by implementing a general dense stereo correspondence framework as a basis for our experiments. We will then review, implement and test state-of-the-art correspondence methods which are able to deal with standard radiometric differences in optical stereo images. In the next step we will review, implement and test state-of-the-art correspondence methods which have been proposed to deal with extreme radiometric differences or actual cross-spectral stereo pairs. Finally we will implement and test novel correspondence methods for cross-spectral stereo and compare them with the state-of-the-art.

1.5 Thesis Outline

In Chapter 2 we will begin with a detailed review of general dense stereo correspondence algorithms and state-of-the-art robust methods in this field. We will then move to current research concerning cross-spectral stereo methods and their abilities and limitations. From the insights gained in this review we will then be able to select and define our own approaches more precisely. In Chapter 3 we will describe our implemented cross-spectral stereo framework and our chosen approaches. This is followed by testing and evaluation in Chapter 4, using different types of test data and setups. Finally we will summarize our conclusions in Chapter 5 and give an outlook on possible future work.

In a supplementary chapter (Appendix A) we will move away from our main topic of far-infrared-optical (cross-spectral) stereo and complement our work by also investigating the properties of the near-infrared-optical stereo case, represented by the i2iReader data of the 'MobiTrick' project at Graz University of Technology. We will apply the stereo methods and insights gained in the main part of the thesis to examine the suitability of different stereo methods in this scenario.

Chapter 2

Literature Review

In this chapter we will first provide a detailed review of general stereo correspondence algorithms for standard stereo applications. As a next step we will focus on robust correspondence measures for images with radiometric differences. Finally we will discuss cross-spectral stereo correspondence and the current research in this field.

2.1 Dense Stereo Correspondence Algorithms

In Section 1.3 we have given a short overview of the different parts of a general computational stereo algorithm. It can be stated that the calibration and reconstruction steps based on epipolar geometry are generally well understood and do not present major difficulties in stereo vision [44]. In contrast, the correspondence problem cannot be solved unambiguously and remains a topic of very active research. From this point on we will therefore assume the availability of a calibrated stereo setup[1], providing rectified input images to be able to focus on stereo correspondence algorithms which create a dense disparity map as an output.

In [44] Scharstein and Szeliski proposed a general taxonomy and provided an excellent review and evaluation of dense stereo correspondence algorithms. Together with this publication they also released stereo test data sets with ground truth data and an evaluation framework, freely available online at the Middlebury stereo vision website [43]. Since the publication of Scharstein and Szeliski's work it has become a main reference for the evaluation of dense stereo correspondence algorithms and the vast majority of newly proposed algorithms report their results on this evaluation framework allowing for easy and direct performance comparison. However, the Middlebury test data sets were created under very controlled conditions generally leading to very clean and high-quality images. This leads to the fact that the performance of algorithms on noisy real-world images taken in uncontrolled environments is often not evaluated and remains somewhat unclear.

[1] It should be noted that cross-spectral calibration presents its own specific difficulties. We will detail our practical method for cross-spectral calibration in Section 4.1.2.

An online evaluation and current ranking of all algorithms tested on the Middlebury framework is available at the Middlebury stereo vision website [43].

In this work and our implemented stereo framework we also adopt the general taxonomy proposed by [44] and reviewed in [50] and will therefore describe it as follows in more detail. According to [44] most stereo correspondence algorithms can be split into four steps:

1. Matching cost computation
2. Cost aggregation
3. Disparity computation/optimization
4. Disparity refinement and post-processing

Depending on the specific use and combination of these steps, correspondence algorithms can be generally classified into local and global methods. Local and global methods differ in the way that the final disparity values are computed and the type of constraints that are applied. More details on this classification will be made clear in the subsequent discussion.

Before starting with the matching cost computation, one image is defined as the reference image and the other one as the match image. The disparity values are computed with respect to the defined reference image, resulting in a final output disparity map which holds disparity values for each pixel of the reference image.

2.1.1 Matching Cost Computation

Matching costs are used to describe the similarity between two considered pixels in the left and right stereo images. This can be done by doing a purely pixel-based comparison, for example by calculating the squared or absolute difference of the pixel intensity values. However, such simple measures are very ambiguous, sensitive to noise and radiometric differences and therefore always have to be combined with local cost aggregation or global disparity optimization methods which enforce additional smoothness constraints.
Other types of matching costs implicitly describe pixels of interest using their local image neighborhoods and then use these local support areas for matching, making the cost computation more robust. Examples include traditional methods like normalized cross correlation, non-parametric measures like rank and census transforms [57] or local feature descriptors [53]. We will review robust matching cost measures in more detail in Section 2.2.

Matching costs are usually computed for all possible pixel pairings within a given maximum disparity range along an image row. The resulting costs for all rows can be stored in a three-dimensional matching cost volume, also termed Disparity Space Image (DSI) [44][4]. Positions in the DSI are indexed by (x, y, d), where (x, y) are the 2D coordinates of the reference image pixel and d is the disparity. Each point in the DSI holds the computed cost of matching a pixel at (x, y) in the reference image

with a pixel $(x - s \cdot d, y)$ in the match image. The parameter s can take the values ± 1 and depends on whether the left or the right image is defined as the reference image. In Figure 2.1 the three-dimensional structure of the DSI is visualized for images of size $W \times H$ and a disparity range of $[d_{min}, d_{max}]$. Figure 2.2 illustrates the concept of the DSI on an example. In this example we compute the matching costs using the sum of absolute differences of a local window and the disparity results by a simple Winner-Takes-All (WTA) selection (see Section 2.1.3). In Figure 2.2 the left and right input images as well as the resulting disparity map can be seen. The enlarged slices through the DSI at different y-coordinates are shown beneath, their positions are indicated by the three dashed lines in the images. Dark values in the DSI slices represent low matching costs and the computed disparity results are indicated in green. It can be seen that textureless regions cause large patches of ambiguous low matching costs which causes WTA to select wrong values.

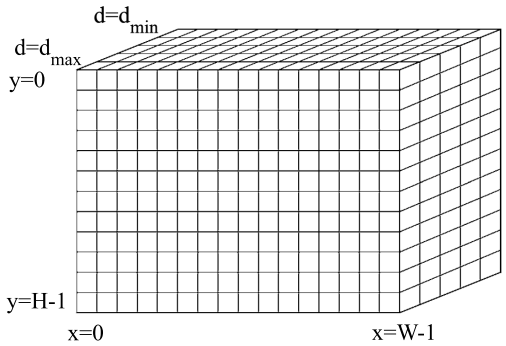

Figure 2.1: Structure of the three-dimensional DSI matching cost volume [40].

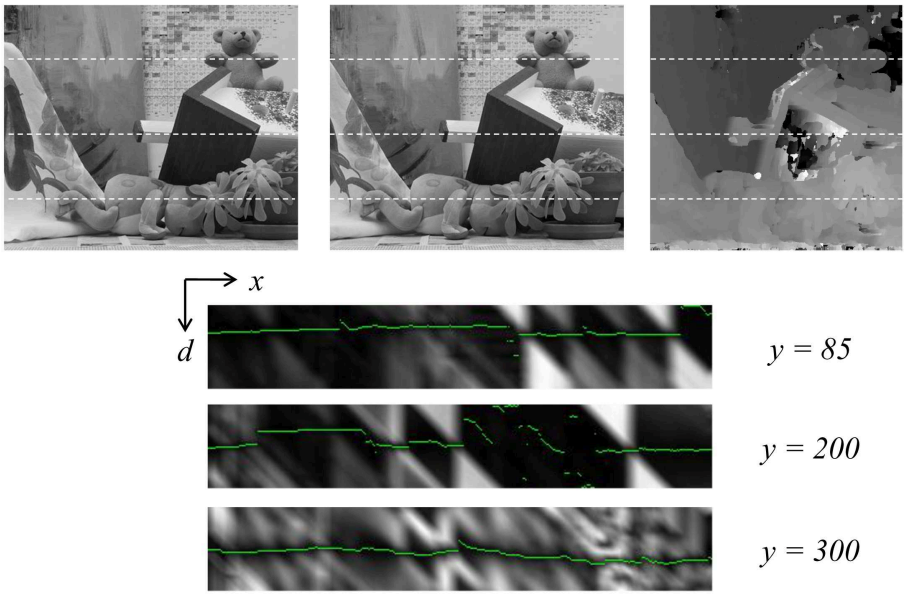

Figure 2.2: Examples of horizontal slices through a DSI computed from the Middlebury 'Teddy' images.

Ideally a corresponding pair of pixels (the true match) would yield the minimum matching cost of all considered pairings. However, in practical applications this is often not the case which makes the subsequent processing/optimization steps necessary.

2.1.2 Cost Aggregation

Cost aggregation is mainly used in local methods and is done by summing or averaging over a local support region in the DSI matching cost volume. In this way a local smoothness assumption is made, which helps to compensate for outliers and ambiguous matching cost values. Usually aggregation in the DSI is done in a two-dimensional way at fixed disparity, however also three-dimensional aggregation is possible to better support slanted surfaces [44].
The simplest and fastest way of aggregation at fixed disparities is done using square windows with equal or Gaussian weights. A disadvantage of these simple methods is the aggregation across disparity discontinuities, which leads to a blurring of disparity jumps. More advanced methods use shiftable windows, windows with adaptive sizes or windows with adaptive weights, where weights and sizes are usually adapted to input image colors/intensities and distance to the pixel of interest. See [50] and [54] for a review and evaluation of cost aggregation methods.

2.1.3 Disparity Computation/Optimization

Disparity Computation/Optimization is the process of determining the disparity values for every pixel in the disparity map using the previously computed and optionally aggregated matching costs.

Winner-Takes-All

Local methods enforce local smoothness constraints in the cost aggregation step and then use the simplest and fastest way of disparity computation by selecting for each pixel the disparity with the minimum matching cost. This approach is also called Winner-Takes-All (WTA). However, WTA approaches do not cope well with locally ambiguous regions in the input images, e.g. bland areas and repetitive patterns, or in general with unreliable matching cost values [44].

Global Energy Minimization

Global methods on the other hand often omit the cost aggregation step and perform a more sophisticated disparity optimization procedure. These methods are less sensitive to locally ambiguous regions and weak matches. Many global methods define

an energy minimization framework to find the optimal disparity function $D(x, y)$ inside the DSI. The optimal disparity function corresponds to a (local) minimum of an energy function $E(D)$, consisting of a data term $E_d(D)$ and a weighted smoothness term $E_s(D)$ [44][50].

$$E(D) = E_d(D) + \lambda E_s(D) \tag{2.1}$$

The data term is based on the previously computed matching costs:

$$E_d(D) = \sum_{(x,y)} DSI(x, y, D(x, y)) \tag{2.2}$$

The smoothness term $E_s(D)$ is used to enforce additional smoothness constraints and is often restricted to only consider the difference of disparities of adjacent pixels, however the layout of the considered neighborhood can vary [50]:

$$E_s(D) = \sum_{(x,y)} \rho(D(x, y) - D(x + 1, y)) + \rho(D(x, y) - D(x, y + 1)) \tag{2.3}$$

In this formulation ρ is a monotonically increasing function of the difference in disparities. Additionally, these terms can be modified to consider intensity edges in the input images and thus lower the smoothness cost for high intensity gradients. This is motivated by the observation that often disparity jumps coincide with intensity gradients in the input images.
In recent years numerous methods have been developed to minimize energy functions of this form based on regularization and Markov random fields (e.g. Graph Cuts (GC) or loopy belief propagation), but the details on this topic are beyond the scope of this work and we refer to additional sources [50][51][8][7][31].
Global methods based on such energy minimization frameworks have proven to produce very accurate results on the Middlebury stereo benchmark [43]. However, a drawback is the relatively high computational complexity which makes them less attractive for real-time or near real-time applications [35].

Dynamic Programming

Dynamic Programming (DP) is a different method to optimize the disparity values and is not a truly global optimization method as described in the previous paragraph [50]. Dynamic programming splits the 2D optimization problem into several independent 1D optimizations, making the process more computationally efficient. For each horizontal scanline a minimum cost path is calculated through the corresponding horizontal slice of the DSI, taking into account matching costs, smoothness and ordering constraints and explicit occlusion costs. Drawbacks of dynamic programming methods are the obvious lack of vertical or inter-scanline constraints and the right choice of occlusion costs. However, optimized dynamic programming implementations can operate in real-time and are therefore attractive for certain applications were computational efficiency is more important than high accuracy [50]. More details on dynamic programming will be given in Chapter 3.

Semi-Global Matching

The method of Semi-Global Matching (SGM) was first introduced in [26]. It approximates the global two-dimensional smoothness constraints applied by truly global methods by evaluating at each pixel a cumulative cost function from several one-dimensional paths at once. In this way many of the drawbacks of dynamic programming are removed and good results can be obtained more efficiently than with global energy minimization methods. More details on semi-global matching will also be given in Chapter 3.

2.1.4 Disparity Refinement and Post-Processing

The last step of Scharstein and Szeliski's taxonomy is disparity refinement and post-processing. For applications like image-based rendering a sub-pixel refinement of the discrete disparity values may be performed but we do not consider this step in our applications. Other post-processing steps include so called cross-checking, were the calculated disparity map is compared with results when reference and match image are switched. In this way occlusions (i.e. pixels that are only visible in the original reference image) can be identified and inconsistent disparity values can be discovered and invalidated [19]. Further post-processing steps can include simple median filtering or speckle removal, where small patches of isolated disparity outliers are removed [27].

2.2 Stereo with Radiometric Differences

After having reviewed the general structure and different building blocks of common dense stereo correspondence algorithms, we will now focus on the robustness of correspondence algorithms concerning differences in input images. Since our aim is to find stereo correspondences in cross-spectral images, which can be seen as the most extreme case of radiometric differences, we will first review common robust approaches in which radiometric differences and noise are being dealt with. Radiometric differences in standard stereo images can occur due to noise, differences in lighting, exposure or other camera settings and are very common in real-world applications.
An extensive review and evaluation of robust matching cost measures is provided by Hirschmueller and Scharstein in [28]. We will structure the first part of this section according to Hirschmueller and Scharstein's classification in preprocessing steps and parametric and non-parametric matching costs.

2.2.1 Preprocessing Methods

A number of different filter techniques can be used to reduce the amount of radiometric differences in two stereo images. The filters are applied to the input images followed by standard matching cost measures like absolute intensity differences. Table 2.1 summarizes the filter techniques described in [28] and lists their respective effects regarding the compensation of radiometric differences. Except for the gradient magnitude filter all listed filters can merely remove local intensity offsets which makes them unsuitable for more complex radiometric differences.

Name	**Method**	**Effect**
Mean filter	subtraction of mean intensity of local window from pixel of interest	Removal of local intensity offset
Laplacian of Gaussian (LoG)	Gaussian convolution and Laplacian operator (i.e. smoothed second derivative)	Removal of noise and local intensity offset
Background subtraction by bilateral filtering	subtraction of bilateral filtered image from original image	Removal of local intensity offset without blurring of edges
Gradient magnitude	computation of image gradient (first derivative) magnitude	Reduction to edge information

Table 2.1: Preprocessing methods for the removal of radiometric differences as described in [28].

2.2.2 Robust Parametric and Non-Parametric Matching Costs

Hirschmueller and Scharstein [28] compared several types of matching cost measures which are based on local image windows and robust to different types of radiometric differences. These can be split into parametric and non-parametric matching costs.

Parametric Matching Costs

- Zero mean sum of absolute differences (ZSAD):
  Before computing the sum of absolute differences for the two compared local windows, the mean of each window is subtracted from each pixel. This removes local intensity offsets of the whole window.
- Normalized cross correlation (NCC):
  Computing the normalized cross correlation between two windows results in a

matching cost which is robust to image gain changes and can handle Gaussian noise.

- Zero mean normalized cross correlation (ZNCC):
  The zero mean normalized cross correlation is the only parametric matching cost which is robust to both gain changes and intensity offsets within the window.

Non-Parametric Matching Costs

- Rank transform [57]:
  Before matching the rank transform computes the rank of the central pixel of a local image window with respect to the intensity values of all pixels within the window. Subsequently the rank values can be matched using standard pixelwise measures like absolute differences.
- Census transform [57]:
  The Census transform also creates an intensity ordering similar to rank but additionally stores the positions and relative rank of all pixels within the image window in a bit string. This gives it more discriminative power than the rank transform. The matching cost is then computed by calculating the Hamming distance between the compared bit strings. Both the rank and the census transform are robust to all types of radiometric differences which preserve the intensity ordering within the windows.

Hirschmueller and Scharstein [28] performed experiments on images of the Middlebury set with simulated and real radiometric differences. Depending on the type of radiometric manipulation very good results were achieved by the census transform, ZNCC and bilateral background subtraction with subsequent intensity difference matching.

However, while many of the presented robust matching costs can handle simple radiometric differences in standard stereo pairs well, their robustness is restricted to certain types of image transformations. The extreme differences in cross-spectral images do in general not conform to these restrictions which leads to the assumption that these methods cannot be successfully applied in this case. Nevertheless for comparison we will show their performance on images with complex radiometric differences in Chapter 4.

In addition to the methods mentioned above for matching cost computation in the presence of radiometric differences a mutual information based method was also investigated in [28]. We will discuss this method in detail in Section 2.2.3.

2.2.3 Mutual Information

Window-Based Mutual Information

In current applications and literature Mutual Information (MI) has emerged as a very popular method to measure similarities in images with complex intensity relationships. Mutual information is originally based on information theory and was first successfully applied by Egnal [15] as a window-based dense stereo correspondence measure. The similarity of two image windows w_l and w_r is measured by considering the marginal entropies of their intensity values ($H(w_l)$ and $H(w_r)$) and their joint entropy $H(w_l, w_r)$.

$$MI(w_l, w_r) = H(w_l) + H(w_r) - H(w_l, w_r) \tag{2.4}$$

where the entropies can be calculated from the respective intensity probability distributions, for example using the intensity histograms of the image windows. As the name suggests, mutual information measures the information that is shared by two random variables. In this case the discrete random variables being the image pixel intensities and MI measuring the statistical co-occurrence of pixel intensities.
Egnal [15] demonstrated the robustness of MI to radiometric differences and proposed an application to real cross-spectral stereo applications. However, no experiments to support this were conducted on real cross-spectral data as is studied in this thesis.

One drawback of window-based MI methods is the dependence on the matching window size. For small windows, the statistical power of the probability distributions is low due to the small sample size which leads to false matches. Larger windows can avoid this problem but result in stronger blurring of disparity discontinuities.
Fookes et al. [17][18] proposed approaches based on adaptive window sizes and the incorporation of hierarchical prior probabilities to improve the performance of MI-based correspondence. They also compared MI to other matching costs like ZSAD, NCC, ZNCC and rank and showed that, while MI gives worse results on standard stereo pairs, it performs well on images with simulated extreme radiometric differences where all other methods fail. For their experiments Fookes et al. [17][18] applied different intensity transforms to one image to simulate multi-spectral stereo data. Figure 2.3 shows a result from [18]. Here the left image of the standard stereo pair is unchanged while a complex non one-to-one intensity transformation is applied to the right image intended to simulate multi-spectral data. The resulting disparity map based on ZNCC matching costs and WTA disparity computation shows that this method fails in this scenario. Fookes' window-based MI method is able to produce a valid disparity map in which the general structure of the scene and the different objects and depth levels are well represented. Due to their results Fookes et al. [17][18] recommended MI for use in multi-spectral stereo applications but again did not provide any results on real multi-spectral data. Also, their definition of multi-spectral remains unclear, as for example results for combinations of visible spectrum and near infrared will differ significantly from results on combinations of visible and far infrared spectra.

Figure 2.3: Results from [18] on simulated multi-spectral data using both ZNCC and hierarchical window-based MI.

Energy Minimization and Mutual Information

To avoid the problems of window-based MI methods, Kim et al. [30] adapted the formulation of MI to fit into a global energy minimization framework. They approximated MI as a data term by using Taylor expansion to be able to convert it into a sum over pixels. Using this method operating over the complete images and not windows, an iterative solution can be achieved using energy minimization methods like graph cuts. For details regarding the conversion of MI into a global data term and optimization via graph cuts we refer to Kim et al's paper [30]. Kim et al. showed very good results for images from the Middlebury data set including complex non one-to-one intensity transformations. An example from [30] is displayed in Figure 2.4. Here also the left input image remains unchanged while an intensity transformation is applied to the right image. When compared to the ground truth, the proposed method proves to provide valid and relatively accurate results in this scenario of synthetically altered images simulating extreme radiometric differences.

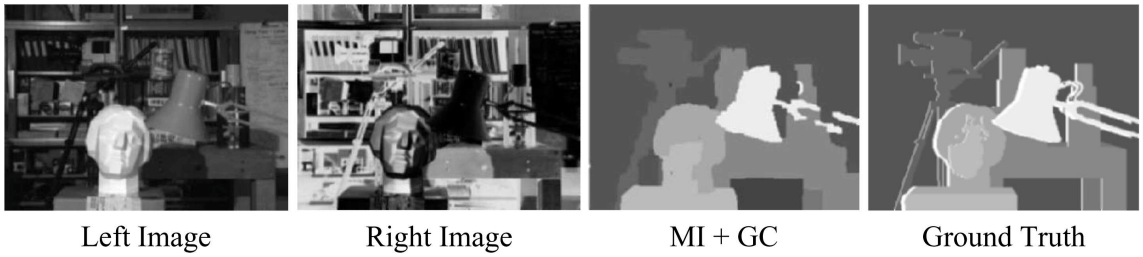

Figure 2.4: Results from [30] on a synthetically altered Middlebury 'Tsukuba' image pair using MI and Graph Cuts (GC).

Using the same representation of MI as a data term, Hirschmueller proposed in [26] and later refined in [27] a much faster hierarchical optimization method using semi-global matching. He also showed good results on synthetically altered stereo pairs. Figure 2.5 shows Hirschmueller's results on Middlebury test images with intensity transformations. As in the previous examples the right image is synthetically altered to demonstrate the robustness of the proposed method. A comparison to the ground

truth at a subjective level shows the good performance of Hirschmueller's approach.

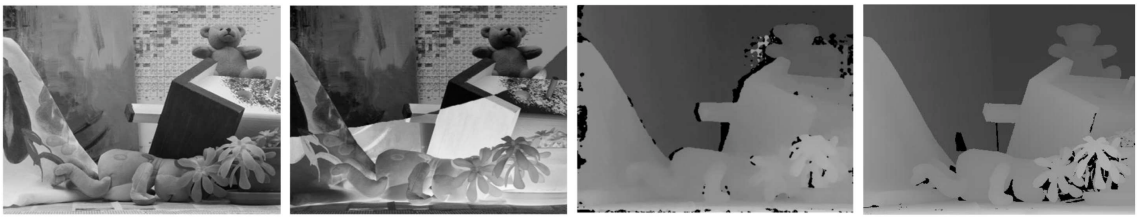

Figure 2.5: Results from [26] on a synthetically altered Middlebury 'Teddy' image pair using Hierarchical MI (HMI) and Semi-Global Matching (SGM).

The demonstrated results of both window-based and global MI methods on images with complex intensity relationships naturally make MI interesting for the application on real cross-spectral stereo pairs. In Section 2.3 we will review the existing previous work that has applied MI in real cross-spectral applications.

2.2.4 Dense Local Feature Descriptors

All matching cost measures discussed so far are based directly or indirectly on the relationship between pixel intensity values of the two stereo images. A different type of approach which we will consider in this work is the matching of dense local image features that are first extracted from the images and then compared using distances between their feature descriptors.
This method was originally used to match sparse image features in wide-baseline stereo applications (e.g. using SIFT [38]). Recently Tola et al. [52][53] proposed an efficient feature descriptor called DAISY for wide-baseline stereo which is suitable for dense correspondence computation. They also showed good results for short-baseline stereo and demonstrated robustness concerning image brightness and contrast as well as image resolution and quality. Similar to very popular feature descriptors like SIFT [38], DAISY describes feature points by using weighted histograms of local oriented gradients. However, it can be computed much more efficiently than SIFT, making it more suitable for dense stereo correspondence.

We will investigate the suitability of DAISY and other types of local feature descriptors for dense cross-spectral stereo matching, introducing necessary modifications of the original descriptors where necessary. Details on this will be discussed in Chapter 3.

2.3 Cross-Spectral Stereo

In the previous section we have reviewed the types of matching costs that are commonly used to deal with different types of radiometric differences up to extreme synthetic image intensity transforms simulating cross-spectral images. Now we will focus on the previous work that has been done on real cross-spectral stereo correspondence.
It has to be stated that to date only very little work has been done on the direct recovery of depth information from cross-spectral stereo images [32][33][55]. This might in part be due to the fact that until recently the availability of uncooled far infrared cameras with relatively good quality (in terms of resolution and signal-to-noise ratio) to the wider academic research community was limited [22].

Some very interesting work was done by Krotosky and Trivedi [33], who investigated the use of cross-spectral stereo for pedestrian detection and tracking. To motivate their approach they first analyzed the performance of state-of-the-art dense stereo methods on real cross-spectral images. More precisely they applied Hirschmueller's hierarchical MI method discussed in Section 2.2.3 to optical stereo images, infrared stereo images, synthetically altered optical stereo images with complex non one-to-one relationships and finally real cross-spectral stereo images. A selection of their results is shown in Figure 2.6. It can be seen that on infrared stereo images and synthetically altered optical images good disparity estimates were achieved, correctly depicting the scene structure and the different depths of the objects in the scene. Furthermore these disparity maps proved to be consistent with the results on the unmodified optical stereo pairs (first row in Figure 2.6). This is in accordance with the results shown in the original papers [26][30] and in Section 2.2.3. However, when applied to real cross-spectral data the method completely failed to produce valid disparity estimates (e.g. last row in Figure 2.6). Krotosky and Trivedi analyzed the reason for this in detail and showed that in cross-spectral stereo images there exists no global intensity transform that can be easily identified. This means that infrared and optical intensities are very uncorrelated and the MI energy term cannot be effectively minimized since good and bad matches produce similarly large values.

After gaining this insight, Krotosky and Trivedi returned to window-based MI methods to estimate disparities in cross-spectral images. Their aim was the registration of people at different depths in surveillance scenes using a cross-spectral stereo setup [33]. They solved the problem by first extracting regions of interest (i.e. people) by foreground extraction in the optical and intensity thresholding in the thermal image and then finding disparities between these foreground regions. This was done using a sliding correspondence window matching using mutual information and a disparity voting method. Good results were shown for the successful registration of previously extracted people but no depth information for any other objects in the scene was obtained.
Very recently Torabi and Bilodeau [55] described a very similar approach to the same problem but replaced mutual information by Local Self-Similarity (LSS) as a correspondence measure. LSS was originally proposed in [46] for object detection,

retrieval and action recognition in visually differing scenes. Torabi and Bilodeau reported a better performance of LSS than MI for this task and also suggested possible future work on the use of LSS as a correspondence measure for dense depth computation. We will investigate this idea and discuss LSS in more detail in Chapter 3. In a different type of application LSS has also been used to successfully align image patches in cross-spectral imagery [5].

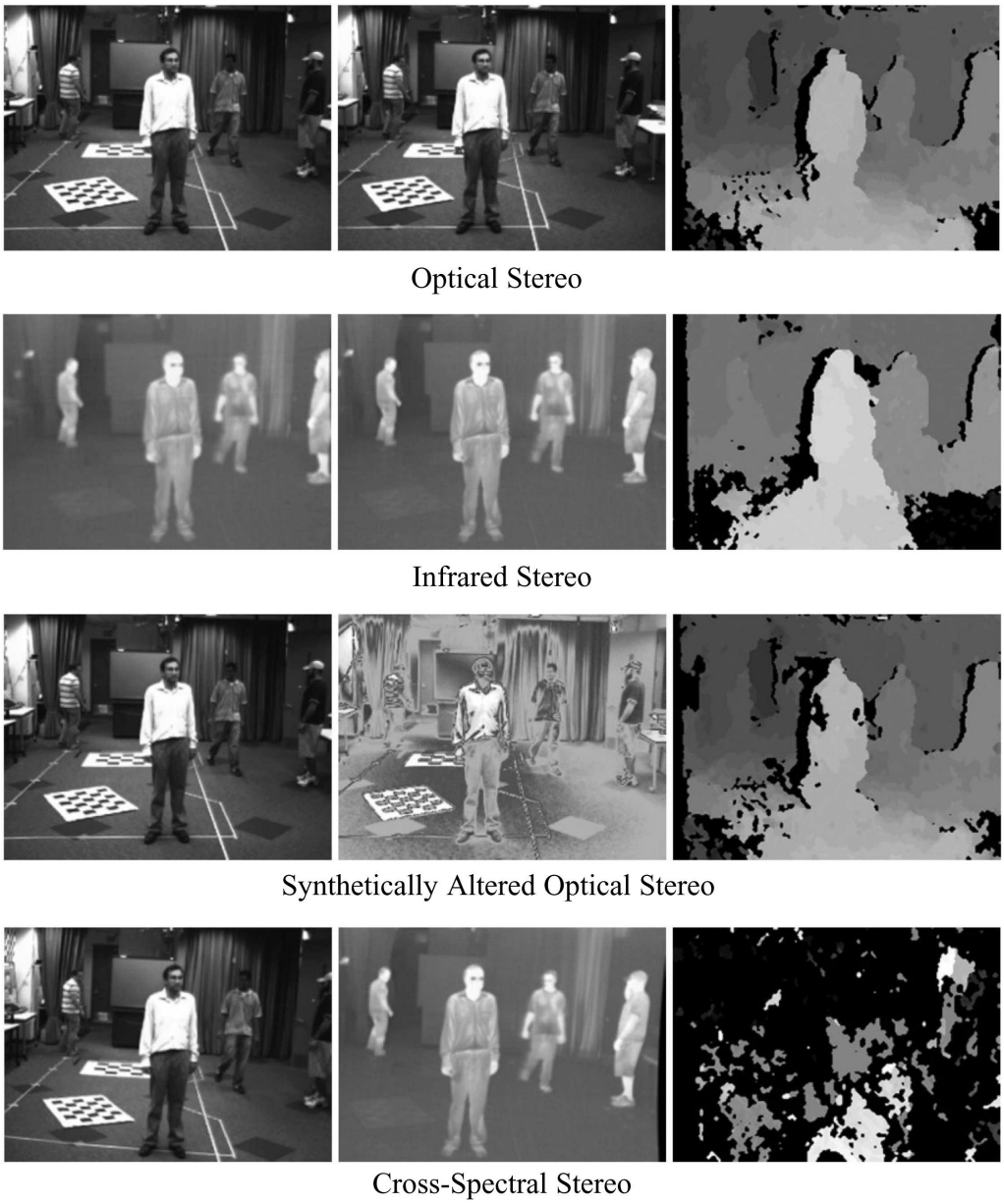

Figure 2.6: Results from [33]: Performance of MI as an energy minimization term on stereo setups combining different modalities.

Cross-Spectral Image Registration For the sake of completeness we will also shortly mention the topic of multi-modal or multi-spectral image registration as the computation of correspondences is also an issue here and a considerable amount of work has been done in this field. Main applications of image registration or alignment include medical imaging, remote sensing or long distance and overhead surveillance applications. Many of these applications assume a global affine transform between the images and are not designed to deal with more complex local transformations. Non-rigid registration methods can deal with locally varying geometric distortions by calculating transformation models based on a sparse number of matched features, but are more computationally expensive [61]. While it might be possible to adapt certain non-rigid registration methods to estimate disparity values of complex stereo scenes, the benefit of this is not apparent since they are designed for a very different type of application. We will restrict our investigations to established dense stereo correspondence methods taking advantage of the epipolar constraint and estimating disparities for every pixel.
For an analysis of previously used registration techniques in stereo scenes with objects at different depths and their limitations we refer to [32].

2.4 Summary

In this chapter we have reviewed the structure and building blocks of dense stereo correspondence algorithms and have considered common methods which are used by state-of-the-art algorithms. We have also discussed robust techniques that are commonly used to deal with standard and extreme radiometric differences in stereo image pairs. Finally he have presented an overview of previous work that used real cross-spectral stereo data and inferred depth information from this setup.

Several approaches have shown to produce good results on simulated cross-spectral stereo data, i.e. synthetically altered optical stereo pairs [15][18][30][26]. Applications that have used actual cross-spectral data have simplified the matching task by only considering certain regions of interest [33][55].
As a conclusion of our review, at the time of writing it appears that there exists no method that has been shown to produce dense depth estimates of a scene from real cross-spectral stereo images. However, some methods like window-based MI or LSS matching have shown relatively good performance for simplified applications and therefore justify further investigation.

We will now continue in the following chapter with a discussion of our implemented cross-spectral stereo framework, starting with the selection of already reviewed and also novel matching cost measures.

Chapter 3

Implementation of a Cross-Spectral Stereo Correspondence Framework

This chapter describes our implementation of a complete dense stereo correspondence framework for both standard and cross-spectral stereo images. In the first section we will discuss and justify the selection of the implemented methods and give implementation details in the subsequent sections.

3.1 Selection of Methods

3.1.1 Matching Cost Computation

The crucial requirement to the solution of our problem of dense cross-spectral correspondence is to find matching cost metrics which are able to measure similarities between cross-spectral images as reliably and efficiently as possible. We have seen in our review in Chapter 2 that matching costs which are used for standard stereo pairs in state-of-the-art algorithms are unsuitable for this task. Methods that have shown promising results are window-based Mutual Information (MI) [33][18][15] and Local Self-Similarities (LSS) [55]. We therefore implement variants of both of these methods in our framework to investigate performance on real dense cross-spectral stereo.
In addition we investigate novel matching cost measures for this task based on a prior visual analysis of our cross-spectral stereo data as shown in Figure 1.2. While there is clearly no direct relation between pixel intensity values as is exploited by standard stereo algorithms, obvious similarities exist on a higher level considering objects and object boundaries. Many corresponding object boundaries and edge fragments appear in both spectra, enabling a human observer to easily match corresponding objects in the images. From this observation we motivate our approach

of using statistical local shape features based on image gradient orientations as a dense correspondence measure. This concept is also used for dense wide-baseline stereo by the DAISY descriptor of [52][53] mentioned in Section 2.2.4. Experiments in [52] and [53] showed that DAISY outperforms other local feature descriptors such as SIFT and SURF [3] in terms of both speed and accuracy. However, in cross-spectral images the orientations of image gradients do not correspond unambiguously because bright regions in the optical image can be dark in the thermal image and vice versa. As a result we have to base similarity on unsigned gradient orientations which allows for bright-dark intensity changes being also matched to dark-bright intensity changes of the same orientation. The restriction to unsigned gradient orientations was also suggested for a different type of application by Dalal and Triggs [13], who proposed a dense descriptor of Histograms of Oriented Gradients (HOG) for human detection. Their goal was the creation of a descriptor optimized for *"dense robust coding of spatial form"* [13]. Dalal and Triggs showed that for their purpose of human detection unsigned gradient orientations give better results due to the range of different backgrounds and clothing colors. Dense HOG descriptors can be computed efficiently (see Section 3.2) and we include them in our implementation to test their performance in the very different task of describing cross-spectral similarity.
It has to be noted that the dense descriptors LSS, DAISY and HOG are precomputed for each image before the actual correspondence search. Similarity is then measured by computing either the L1 or L2 distance between the descriptors.

In addition to the matching cost measures we consider for cross-spectral stereo, we also implement and test standard matching cost measures as described in Section 2.2.2. This allows for a verification of the correct functionality of our complete correspondence algorithms on standard stereo data. Furthermore a direct comparison on real or simulated cross-spectral data can be made to demonstrate the limitations of traditional robust matching cost measures. In combination with this we also implement the preprocessing methods described in Section 2.2.1.

To summarize, the following list gives an overview of the matching cost measures included in our implementation:

- Absolute Differences (AD) [28]
- Sum of Absolute Differences (SAD) [28]
- Zero mean Sum of Absolute Differences (ZSAD) [28]
- Normalized Cross Correlation (NCC) [28]
- Zero mean Normalized Cross Correlation (ZNCC) [28]
- Rank transform and AD [57]
- Census transform and Hamming distance [57]
- Mutual Information (MI) [15][18]
- Local Self-Similarity descriptors (LSS) and L1/L2 distance [46][55]
- DAISY descriptors and L1/L2 distance [52][53]
- HOG descriptors and L1/L2 distance [13]

These measures can be combined with the preprocessing steps [28]:

- Mean filter
- Laplacian of Gaussian filter (LoG)
- Background subtraction by bilateral filtering
- Gradient magnitude computation

3.1.2 Cost Aggregation

We implement different cost aggregation methods to investigate their influence on the performance of local as well as global correspondence methods. We only consider aggregation at constant disparity, i.e. aggregation is performed separately on each vertical slice of the Disparity Space Image (DSI). The simplest aggregation methods consist of a rectangular window of fixed size with equal or Gaussian weighted contributions of the matching costs within the window. More advanced methods use adaptive weights, multiple windows or windows with adaptive shapes based on pixel intensities and distances in the input images. While significantly improving the performance of local correspondence methods, most of these approaches are computationally expensive [54]. Recently Rhemann et al. [42] proposed a very fast method using adaptive weights for cost aggregation based on guided image filtering [24]. They reported very good results on the Middlebury test data [43] and real-time capability. For this reason we also implement this method in our framework to test its applicability to our various matching costs. Furthermore we implement a fast aggregation method based on adaptively weighted vertical windows which was proposed in [56] for use in combination with dynamic programming.

Overview of the cost aggregation methods included in our implementation:

- Rectangular window with equal weights
- Rectangular window with Gaussian weights
- Rectangular window with adaptive weights using guided filtering [42]
- Vertical window with adaptive weights based on pixel intensity and distance [56]

3.1.3 Disparity Computation/Optimization

In addition to the chosen matching cost the disparity result of correspondence algorithms depends heavily on the disparity computation/optimization method [44][28]. Compared to standard stereo images, cross-spectral images can be expected to produce more ambiguous or false matches as well as weaker correct matches. Weaker correct matches can be caused by the difficulty of matching cost metrics to cope with naturally different appearance in the different spectra. False or ambiguous matches can occur due to the increased presence of bland areas as well as the visibility of features and patterns in one spectrum which are absent in the respective other spectrum (e.g. see Figure 1.2 and Figure 2.6). It is therefore important to investigate how different optimization techniques can compensate for these difficulties.

The simplest method we implement is the Winner-Takes-All (WTA) method. Here the disparity producing the minimum matching cost is chosen.
The next, somewhat more advanced method is a dynamic programming approach which enforces additional constraints along the image rows and is computationally efficient. We also test a variation of dynamic programming introduced by [44] called scanline optimization which in contrast to traditional dynamic programming does not explicitly account for occlusions and does not enforce the ordering constraint.
Furthermore we include a variation of Hirschmueller's semi-global matching [26][27] which is also computationally efficient and provides a better approximation of global disparity smoothness constraints than dynamic programming.
Finally we also test the performance of global optimization based on graph cuts on our calculated matching costs [51]. We include graph cuts in our framework to see if significantly better results can be obtained in this way even at the cost of longer runtimes.

Overview of the disparity computation/optimization methods included in our implementation:

- Winner-Takes-All (WTA)
- Dynamic Programming (DP) [4][44]
- Scanline Optimization (SO) [44]
- Semi-Global Matching (SGM) [26][27]
- Graph Cuts (GC) [51]

3.1.4 Disparity Refinement and Post-Processing

Our main goal is to investigate the basic ability of the previously described methods to create valid dense disparity maps, independently of possible post-processing enhancement. However, to be able to detect occlusions and remove grossly invalid disparity regions in our final results we include the following post-processing steps in our framework:

- Left-right consistency check (cross-check) [19]
- Speckle removal [27]

3.1.5 The Complete Correspondence Algorithm

In our framework all methods of the different building blocks described above can be combined in any combination to form the complete correspondence algorithm. This includes combinations which can be seen as unusual considering current literature on the topic (e.g. [50][28]). For example window-based methods like ZNCC or MI are usually only used in local methods with WTA disparity computation. However, we also experiment with using these window-based matching costs in combination with global disparity optimization methods.
Figure 3.1 shows a schematic diagram of our complete dense stereo framework. The

program takes as an input either a single stereo pair of still images or alternatively two video files taken by the left and right cameras. In case of video input, the program automatically extracts and processes video frames at a desired frame rate starting at the desired frame number. All input images are converted to standard 8-bit grayscale intensity images. The input images or video frames are then rectified using the previously computed calibration data. The calibration process utilized for our application will be described in detail in Section 4.1.2. If the input images are already rectified, this step is omitted. After an optional preprocessing step the previously described core components of the correspondence algorithm matching cost computation, cost aggregation and disparity computation/optimization are executed. The optionally post-processed disparity map is saved as a single output image or as a frame in an output video file. The maximum disparity range is stretched to the full 8-bit grayscale range for improved visualization.

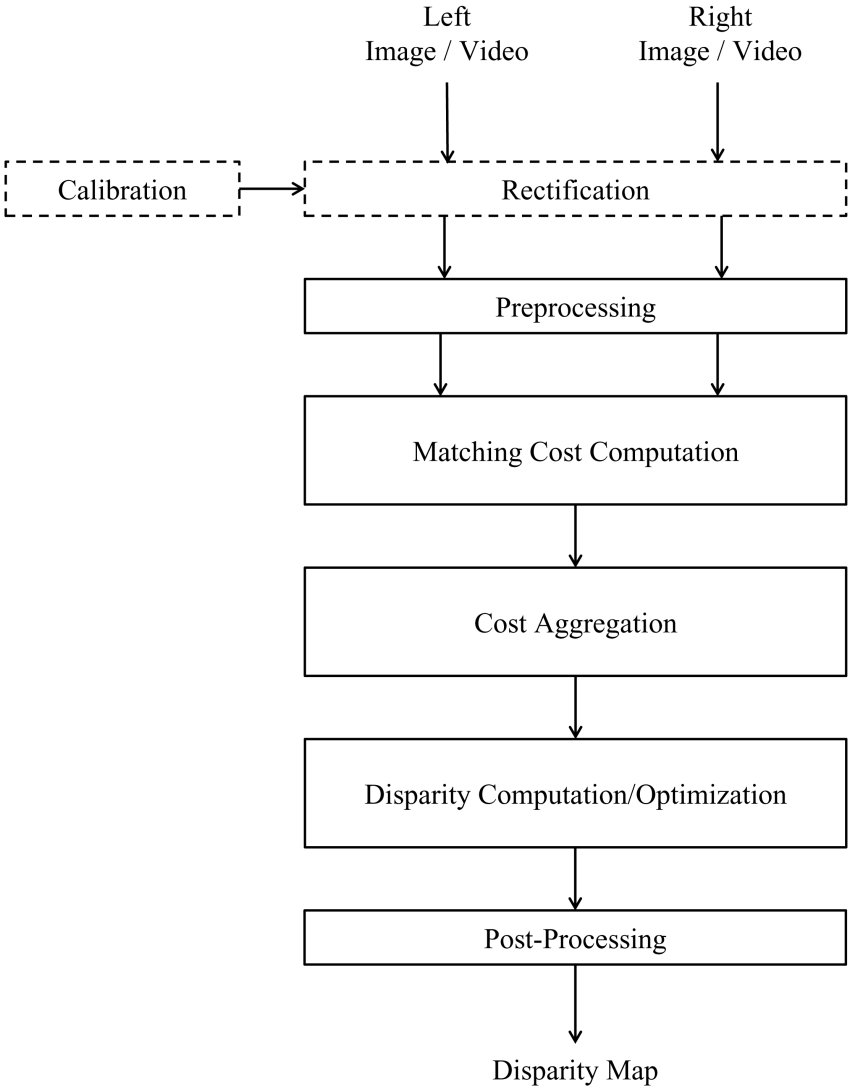

Figure 3.1: Schematic diagram of our implemented dense stereo correspondence framework.

3.2 Implementation

In this section we will provide details regarding our implementation of the different components we selected for our dense stereo correspondence framework.
Our complete framework is written in C++ and is based on the functionality provided by version 2.1 of the Open Source Computer Vision library (OpenCV)[1] [9][34].

3.2.1 Preprocessing

The filters used in the preprocessing step (see Sections 2.2.1 and 3.1) are straightforward to implement and we base our implementation on the description given by Hirschmueller and Scharstein in [28].

3.2.2 Matching Cost Computation

Standard Methods

We implement the selected standard matching cost measures (see Sections 2.2.2 and 3.1) as defined by Hirschmueller and Scharstein in [28]. For the respective mathematical definitions we therefore refer to [28].

Mutual Information

We implement a window-based MI correspondence measure as described in [15]. The MI between a window in the left image w_l and a window in the right image w_r can be calculated using the entropies of the intensity values of the windows as shown in Equation 2.4. As in [15] this equation can be expressed in terms of the joint and marginal probability density distributions of the intensity values:

$$MI(w_l, w_r) = \sum_{(w_l, w_r)} p(w_l, w_r) \log \left(\frac{p(w_l, w_r)}{p(w_l)p(w_r)} \right) \tag{3.1}$$

The joint probability distribution function $p(w_l, w_r)$ can be calculated from the 2D histogram $h(w_l, w_r)$ of the intensity values of the two windows

$$p(w_l, w_r) = \frac{1}{N} h(w_l, w_r), \tag{3.2}$$

where the 2D intensity histogram $h(w_l, w_r)$ is simply computed by splitting the possible intensity range (in our case 255 for standard 8-bit grayscale images) into n bins and filling a $n \times n$ matrix with the occurrence-count of the corresponding intensity pairs. The histogram is normalized by the number of pixels in one window

[1] http://opencv.willowgarage.com (May 2011)

N. The marginal probability distribution functions $p(w_l)$ and $p(w_r)$ can then be computed by summing over the rows and columns of the joint distribution function respectively. Before further processing the calculated MI values are normalized so that maximum MI values are converted to minimum matching cost values and vice versa.

In addition to the basic window-based MI method we also implement a variation of the approach proposed by Fookes et al. [17][18] which we discussed in our review in Section 2.2.3. As described in Section 2.2.3 small windows lead to a low statistical power of the probability distribution functions. Fookes et al. showed that this problem can be reduced by including global prior probabilities into the MI computation. The joint probability distribution function of two image windows is then computed as follows:

$$p(w_l, w_r) = \lambda p_{window}(w_l, w_r) + (1 - \lambda) p_{prior}(w_l, w_r) \tag{3.3}$$

The joint probability distribution of the windows $p_{window}(w_l, w_r)$ is computed as described above. The prior probabilities $p_{prior}(w_l, w_r)$ are computed using the joint probability distribution of the intensities of the complete input images. As in [17][18] we also combine this concept with a hierarchical matching approach using a two-level Gaussian pyramid to improve the accuracy of prior probabilities and enhance speed. In the first stage the correspondence search is performed on the downscaled images with a relatively large window size, in the second stage the full-sized images are used. The result of the first stage allows for a refinement of the prior probabilities for the second stage as the corresponding intensity values for the global 2D histogram are then computed using pixels shifted by the disparity result from the first stage. Furthermore the disparity search range in the second stage can be restricted to an interval centered on the disparity results from the first stage. Fookes et al. [17][18] use a simple WTA disparity computation in their approach, combined with a so-called *2D match surface*. This 2D match surface is used to enforce uniqueness of matching cost minima and left-right consistency constraints but we do not include this method in our implementation. Also in contrast to the original implementation in [17][18] we use fixed window sizes and no adaptive windows.
In our evaluation we will investigate how the basic window-based MI matching method combined with disparity optimization methods performs in comparison with our implemented version of a hierarchical method using WTA as proposed by Fookes et al.

Local Self-Similarity Descriptors

Our implementation of a Local Self-Similarity (LSS) descriptor is based on the paper of Shechtman and Irani [46] who first proposed this similarity measure for object detection and recognition, image retrieval and action recognition. The aim of LSS is to measure the internal geometric structure of local similarities within one image and then use descriptors for matching these self-similarities across images. The LSS descriptor associated with a pixel p in an image is computed as follows. A small image patch (e.g. 5x5 pixels) centered on the pixel p is compared to all patches in

a larger surrounding region (e.g. 40x40 pixels) by sliding the patch over the region and computing the Sum of Squared Differences (SSD). The SSD values are then normalized and transformed into a so-called *correlation surface* S_p as per [46] using Equation 3.4.

$$S_p(x,y) = exp\left(-\frac{SSD_p(x,y)}{max(var_{noise}, var_{auto}(p))}\right) \tag{3.4}$$

In the denominator used for normalization var_{noise} is a constant value representing the level of image noise and has to be selected empirically for best results. var_{auto} corresponds to the structure and contrast of the image patch and is computed as the variance of the differences of all patches within a radius of one pixel around the central patch. The descriptor is then constructed from the correlation surface S_p by creating a log-polar grid centered at p and binning the correlation values into this grid. Each bin is assigned the maximum correlation value of all pixels within the grid cell. Finally the descriptor is linearly stretched to the interval $[0, 1]$ for additional normalization.
For their application in [46] Shechtman and Irani use ensembles of dense LSS descriptors to match whole image regions between images. To make the descriptor ensembles more discriminative, they discard descriptors that are classified as non-informative. One type of non-informative descriptors are descriptors in bland or homogeneous regions which represent high self-similarity everywhere. These descriptors are identified using a specific sparseness-measure of the descriptor values. The second type of non-informative descriptors represent salient image patches where all self-similarity values lie below a given threshold.
Torabi and Bilodeau [55] recently applied LSS descriptors to a simplified dense stereo correspondence problem as seen in our review in Section 2.3. They use window-based matching to find corresponding regions of interest in cross-spectral images. In [55] the ensembles of LSS descriptors are composed of the informative LSS descriptors within the matching windows, the final matching cost between LSS descriptors is computed by using the L1 distances.
In our work we investigate the use of LSS descriptors as a truly dense stereo correspondence measure for cross-spectral stereo. We compute LSS descriptors for every image pixel and compute dense matching costs (via L1 or L2 distances) which can be applied to our disparity optimization methods. We also experiment with filtering non-informative descriptors as described above and the resulting effect on the disparity computation (see Chapter 4).
Figure 3.2 shows three enlarged examples of informative LSS descriptors computed at corresponding locations in infrared and optical stereo images using our implementation. The descriptors in this example are computed using a 5x5 pixel image patch and a 35x35 pixel surrounding region (marked as white squares in Figure 3.2), the log-polar grid is divided into 4 radial and 12 angular intervals resulting in a descriptor of size 48. It can be seen that despite significant differences in pixel intensities in the two images, on a qualitative level the LSS descriptors appear to describe the similar local layout structure in both images relatively well.

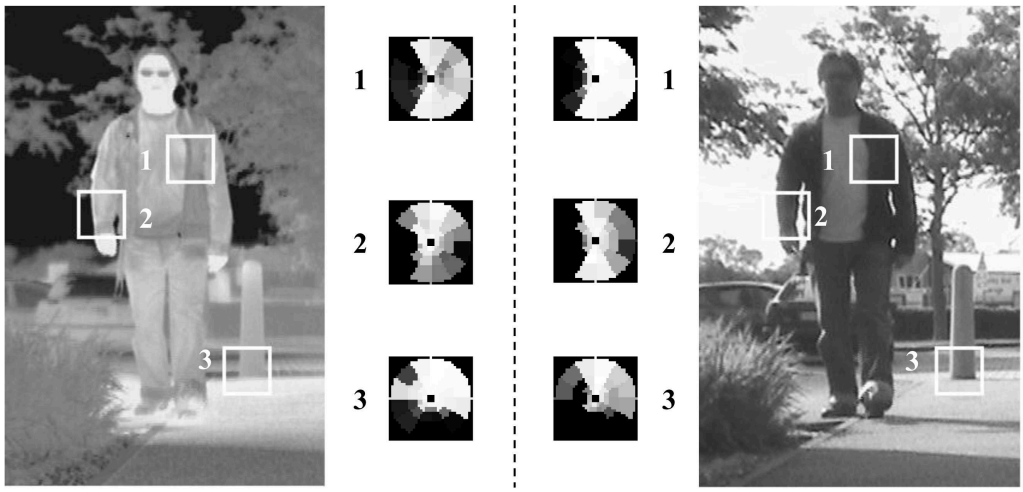

Figure 3.2: Examples of corresponding informative LSS descriptors extracted from infrared (left) and optical (right) images.

DAISY Descriptors

We implement the DAISY descriptor as proposed by Tola et al. [52][53] and based on a sample implementation provided by the authors[2]. The DAISY descriptor uses efficiently computed histograms of oriented gradient norms to describe local image regions. As a first step H so-called *orientation maps* are computed from the input images which hold the norm of the image gradients at each pixel with the respective orientation. The gradient orientations are quantized to distinguish H different orientations. As per the original sample implementation we use centered gradient filters of the form $[-1, 0, 1]$ and $[-1, 0, 1]^T$ on Gaussian smoothed input images for the computation of image gradients. The contribution of the image gradients to the quantized orientations is then computed from the results of these filtering steps as

$$G_\theta = \cos(\theta)\frac{\partial I}{\partial x} + \sin(\theta)\frac{\partial I}{\partial y}, \tag{3.5}$$

where G_θ is the orientation map with the orientation angle θ and $\frac{\partial I}{\partial x}$ and $\frac{\partial I}{\partial y}$ represent the image gradients in directions x and y respectively. The orientation maps are then convolved with Gaussian kernels of Q different scales to create $H \times Q$ *convolved orientation maps*. These Gaussian convolutions can be computed very efficiently and result in convolved orientation maps where each pixel holds a Gaussian weighted aggregation of neighboring oriented image gradients. The DAISY descriptor itself is constructed by reading and concatenating values from the convolved orientation maps. The values are sampled around the central pixel in concentric circles at T angular directions where the level of Gaussian smoothing (i.e. the level of convolved orientation maps) depends on the distance from the central pixel. The principal layout of a DAISY descriptor can be seen in Figure 3.3 with $Q = 3$ and $T = 8$. The crosses represent locations where the individual histograms are created by

[2]http://cvlab.epfl.ch/∼tola/daisy.html (June 2011)

sampling values from the convolved orientation maps for each of the H orientations. The circles around the sampling points represent the standard deviations of the respective Gaussian kernels. The final descriptor size can be given as $(Q \times T + 1) \times H$, the distance of the outermost sampling points to the center is determined by the parameter R. Normalization for robustness against varying gradient magnitudes due to local radiometric differences is performed by either normalizing each histogram separately or the whole descriptor to L2 unit norm. For an analysis of the advantages of this particular layout and a comparison with other descriptors like SIFT and SURF we refer to [53].
As discussed in Section 3.1 for the use in cross-spectral stereo we have to modify the DAISY descriptor to only use unsigned gradient orientations. This means that instead of 360° we only quantize 180° into H intervals and compute the orientation maps using the absolute value of the image gradients. To enable direct comparison our framework allows for a simple switch between signed and unsigned gradient orientation.
For computing the matching cost between two DAISY descriptors we use the L1 or L2 distance between the descriptor values.

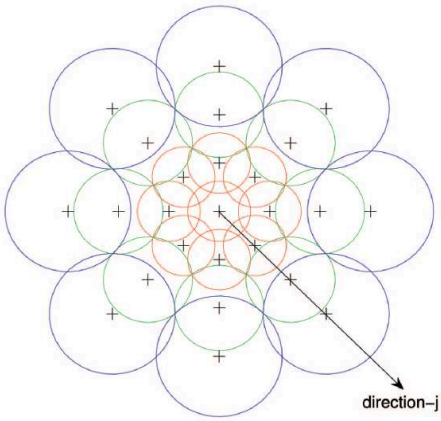

Figure 3.3: Layout of a DAISY descriptor [53].

HOG Descriptors

Our implementation of dense Histogram of Oriented Gradient (HOG) descriptors is a variation of the approach proposed in the original paper by Dalal and Triggs [13] for person detection. Similar to the previously described DAISY descriptor the HOG descriptor is based on histograms of oriented gradients in a local region around the pixel of interest. Here a rectangular block centered on the pixel of interest is divided into $n \times n$ cells and for each cell a histogram of oriented gradients is computed. The histogram values of all cells are then concatenated to represent the HOG descriptor. For the computation of image gradients we use centered gradient filters $[-1, 0, 1]$ and $[-1, 0, 1]^T$ which is also recommended by Dalal and Triggs. The image gradient orientations are then quantized into H intervals and the magnitudes of the oriented gradients are binned into the intervals for each cell. To increase invariance

to differing gradient magnitudes due to local radiometric differences we normalize the whole descriptor to L2 unit norm. Again we include both signed (0° to 360°) and unsigned (0° to 180° for cross-spectral stereo matching) gradient orientations. The final HOG descriptor size is given as $n \times n \times H$ and the matching cost between two HOG descriptors is computed using the L1 or L2 distance.
Dense HOG descriptors for every image pixel can be computed efficiently by using integral histograms. Similar to the DAISY descriptor computation we first create orientation maps for all H orientations. From these orientation maps we then calculate integral orientation map histograms. The histograms for every cell in the image can be calculated efficiently in constant time from the integral orientation map histograms [41]. This approach allows for a fast descriptor computation but prevents the use of spatial weighting (e.g. Gaussian) of the image gradients within one descriptor.

3.2.3 Cost Aggregation

Aggregation with Constant Weights

Our simplest and fastest implementation of a cost aggregation method is based on rectangular windows with constant weights. We include aggregation using a box filter with equal weights as well as a Gaussian weighted filter which can both be computed efficiently by convolution with the vertical slices of the DSI matching cost volume.

Aggregation with Vertical Windows and Adaptive Weights

We implement the adaptive aggregation method proposed by Wang et al. [56] for use in combination with dynamic programming optimization. Wang et al. perform adaptive cost aggregation along vertical windows only (i.e. single image columns) to achieve computational efficiency while increasing inter-scanline consistency and reducing the typical streaking effects of dynamic programming methods [50]. The weights of the matching costs in the vertical image windows of size $1 \times n$ are adaptively computed by taking geometric distance as well as pixel intensity differences in both input images into account.
In the following formulation from [56] we assume the left input image to be the reference image and denote the vertical window around the pixel of interest $p(x, y)$ in the left image as w_l and around the potential match $q(x - d, y)$ at disparity d in the right image as w_r. The aggregated matching cost in the DSI is then computed as

$$DSI(p,d) = \frac{\sum\limits_{k \in w_l, k' \in w_r} v(p,k)v'(q,k')DSI(k,d)}{\sum\limits_{k \in w_l, k' \in w_r} v(p,k)v'(q,k')}, \tag{3.6}$$

where the pixels k and k' are the corresponding pixels in the vertical windows and $v(p,k)$ and $v'(q,k')$ are the respective weight masks for the left and right window. The weight masks $v(p,k)$ and $v'(q,k')$ are computed in the same way and we state the equation for $v(p,k)$ as

$$v(p,k) = exp\left(-\left(\frac{\Delta c_{pk}}{\gamma_c} + \frac{\Delta g_{pk}}{\gamma_g}\right)\right), \tag{3.7}$$

where Δc_{pk} is the absolute difference of intensity values of p and k and Δg_{pk} represents the geometrical distance. The weighting constants γ_c and γ_g have to be adjusted empirically for best results.
Wang et al. [56] achieve real-time performance for this aggregation method by implementation on a GPU.

Aggregation with Adaptive Weights by Guided Filtering

We finally implement the very fast adaptive aggregation method recently proposed by Rhemann et al. [42] based on guided filtering. Rhemann et al. use a square filter window with adaptive weights that preserves edges and therefore disparity discontinuities to aggregate the matching costs in each vertical slice of the DSI. The fast filtering with adaptive weights is achieved by using a guided filter as introduced by He et al. in [24]. In this approach the adaptive filter weights do not have to be calculated explicitly but filtering can be implemented very efficiently using sequential summation by box filtering and integral images. For details on efficient guided filtering we refer to [24]. Rhemann et al. present a sample MATLAB implementation of their approach[3] which uses the guided filter implementation provided by He et al. We also adopt this implementation of guided filtering for our cost aggregation implementation. According to [24] the resulting filter weights $W_{i,j}$ can be expressed explicitly as

$$W_{i,j} = \frac{1}{|w|^2} \sum_{k:(i,j)\in w_k} \left(1 + \frac{(I_i - \mu_k)(I_j - \mu_k)}{\sigma_k^2 + \epsilon}\right), \tag{3.8}$$

where I is the so-called guidance image which is the stereo reference image in our application. μ_k and σ_k are the mean and the variance of I in the image window w_k centered at pixel k. $|w|$ represents the number of pixels in the window w_k and ϵ is a smoothness parameter.
The edge preserving property of the guided filter is explained in [24] by considering a simple 1D step-edge. The terms $(I_i - \mu_k)$ and $(I_j - \mu_k)$ in Equation 3.8 have the same sign if the pixels I_i and I_j lie on the same side of the edge resulting in a large numerator and averaging weight. On the other hand, if I_i and I_j lie on different sides of the edge the term $\left(1 + \frac{(I_i-\mu_k)(I_j-\mu_k)}{\sigma_k^2+\epsilon}\right)$ is small resulting in a low averaging weight of this pixel pair. The parameter ϵ controls the averaging strength, leading to an unweighted low-pass filter if $\sigma_k^2 \ll \epsilon$ [24].

[3] http://www.ims.tuwien.ac.at/research/costFilter (July 2011)

3.2.4 Disparity Computation/Optimization

Before the optionally aggregated matching costs in the DSI are processed by the disparity computation/optimization algorithms, they are linearly normalized to a specified range. This allows for a more general application of the disparity optimization methods and an easier selection of optimization parameters, considering the different nature and ranges of our various raw matching costs.

Winner-Takes-All

The Winner-Takes-All (WTA) disparity computation is the simplest and fastest method and can be implemented by simply selecting the disparity with the lowest matching cost for each pixel.

Dynamic Programming

Our implementation of a Dynamic Programming (DP) optimization method is based on Bobick and Intille's work [4] as well as Scharstein and Szeliski's [44] modified version of this approach which is included in the Middlebury stereo benchmark implementation [43].
The goal of DP optimization is to find the minimum cost path through each horizontal DSI slice separately (i.e. as described in Section 2.1.3 the 2D disparity optimization problem is split into separate 1D optimization problems which can be solved efficiently). Our implementation processes each horizontal DSI slice and computes the minimum cost path through the slice while taking matching costs, occlusion costs and the ordering constraint into account. During the computation of the cost path a point in the DSI slice can take on three different states: matched (M), diagonal occlusion (D) and vertical occlusion (V)[4]. A point in matched state M is considered visible in both images and is charged the matching cost of the respective pixel pair. A point in state D is only visible in the left image and occluded in the right image and thus marks a diagonal gap in the disparity path. State V represents visibility in the right image and occlusion in the left image and marks vertical disparity jumps in the DSI slice. Points in the occluded states D and V are assigned fixed occlusion costs in the cost path computation. In this way occlusions are handled explicitly by the DP optimization. The DP approach also enforces the ordering constraint which means that corresponding pixels must appear in the same order in the left and right image. This constraint can be violated in scenes with very narrow foreground objects but can in general be assumed to hold in the majority of scenes [4]. Due to the ordering constraint a valid cost path can only traverse through the DSI slice using a limited number of legal moves between the described states. From state M a path is allowed to either move horizontally to state M, vertically to state V or diagonally to state D. From state V a path can only move vertically

[4]Here the terms 'vertical' and 'diagonal' refer to directions within the DSI slice and not in the original images.

to state V or horizontally to state M. From state D a path can move diagonally to state D or horizontally to state M. For a more detailed explanation of the possible moves and cost paths we refer to [4].
The actual computation of the minimum cost path is executed in two stages. In the first stage the DSI slice is traversed from left to right, accumulating the minimum cost of all valid paths to every point. Only the overall minimum cost and the previous state transition is known at each point. In the second stage the final minimum cost path is backtracked from the rightmost column of the DSI slice. For images of dimensions $W \times H$ with a maximum disparity range of $|d|$ the complete DP optimization algorithm can be computed in $O(W \times H \times |d|)$ [4].
Drawbacks of DP approaches are the lack of vertical constraints as well as the dependency on the chosen occlusion costs. To reduce horizontal streaking effects in the disparity map caused by the lack of vertical constraints we apply the method proposed by Scharstein and Szeliski [44]. An additional smoothness cost is charged at transitions from state D to M and from state V to M. This smoothness cost is set to be dependent on intensity gradients in the reference image to bias disparity jumps to coincide with intensity edges. The basic smoothness cost is multiplied with a penalty factor if the intensity gradient magnitude lies below a given threshold.
As in [44] we fill the detected diagonal occlusion areas in the disparity map with the closest disparity values from the left.
To illustrate the different states M, D and V as described above we display an example DSI slice in Figure 3.4. Similar to Figure 2.2 the displayed slice is taken at $y = 200$ from the DSI of the Middlebury 'Teddy' image pair computed via simple absolute differences. The green line represents the minimum cost path (i.e. the disparity values) computed by our DP approach. In this example the detected diagonal occlusion areas are not filled for clearer display. The arrows indicate examples of continuous regions of the respective states in the minimum cost path.

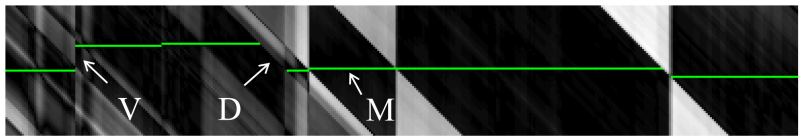

Figure 3.4: Illustration of DP states on a sample DSI slice.

Scanline Optimization

Scanline Optimization (SO) is very similar to traditional DP approaches and was proposed by Scharstein and Szeliski [44]. We also include this method based on the implementation in the Middlebury stereo benchmark [43] as an alternative to our DP optimization.
SO performs the same optimization of a cost path through each separate DSI slice as DP but does not explicitly take occlusions into account or enforce the ordering constraint. The cost of a cost path is computed using only the matching costs in the DSI slice and a smoothness cost. While traversing the DSI slice from left to

right all possible disparity jumps between adjacent columns are allowed. Disparity jumps are charged a smoothness cost which again depends on the image gradient magnitude of the reference input image. As noted in [44] the result of this method is equivalent to a WTA disparity computation if the smoothness cost is removed.
The computational complexity of the SO implementation can be given as $O(W \times H \times |d|^2)$ for images of dimensions $W \times H$ and a maximum disparity range of $|d|$. The fact that in contrast to DP optimization all possible disparity jumps are allowed results in a factor $|d|^2$ instead of $|d|$.

Semi-Global Matching

Our implemented Semi-Global Matching (SGM) optimization is based on Hirschmueller's original papers [26][27] and on the modified version which is available in the OpenCV library [9][34]. In contrast to DP optimization which optimizes a 1D problem for every scanline separately, SGM approximates a full 2D optimization by considering several 1D cost paths simultaneously at each pixel. As in [27] the optimization problem can be formulated as the search for a function of disparity values D through the DSI which minimizes an energy function $E(D)$. This is similar to the global energy functions described in Section 2.1.3. Equation 3.9 defines the energy function:

$$E(D) = \sum_{p} \Big(DSI(p, D_p) + \sum_{q \in N_p} P_1 T[|D_p - D_q| = 1] + \sum_{q \in N_p} P_2 T[|D_p - D_q| > 1] \Big) \tag{3.9}$$

In this formulation p denotes the current pixel of interest and q represents a pixel in the neighborhood N_p of p. The first term of Equation 3.9 represents the matching cost at pixel p at the respective value of the disparity function (denoted as D_p). The second term contributes a smoothness penalty P_1 for small disparity jumps (1 pixel) for all pixels in the neighborhood of p. The last term adds a smoothness penalty P_2 for disparity jumps larger than 1 pixel. The operator $T[]$ returns 1 if its argument is true and false otherwise.
The energy function $E(D)$ can be efficiently minimized using the SGM approach introduced by Hirschmueller [26][27] as follows. The overall cost for every pixel p at every disparity d is computed by summing the minimum cumulative cost paths $L_r(p, d)$ in 1D from all directions r around the pixel. In our implementation we use 8 directions r (i.e. the eight cardinal directions) but the use of 16 directions is also possible [27]. The minimum cost of a path $L_r(p, d)$ from direction r ending in p can be computed recursively as

$$\begin{aligned} L_r(p,d) = & DSI(p,d) + min\Big(L_r(p-r,d), \\ & L_r(p-r,d-1) + P_1, \\ & L_r(p-r,d+1) + P_1, \\ & \min_i L_r(p-r,i) + P_2\Big) \end{aligned} \tag{3.10}$$

which corresponds to the formulation in Equation 3.9 for this path. The overall cost $S(p,d)$ at the pixel p and disparity d is then defined as

$$S(p,d) = \sum_r L_r(p,d) \tag{3.11}$$

The final disparity function D is determined by simply selecting at each pixel the disparity d yielding the lowest value of $S(p,d)$.
For images of dimensions $W \times H$ with a maximum disparity range of $|d|$ the complete optimization algorithm can be efficiently implemented with a complexity of $O(W \times H \times |d|)$. For further details including an analysis of the computational complexity we refer to [26] and [27].

We base the SGM optimization in our framework on the implementation available in the OpenCV library [9][34]. However, this implementation is originally designed to match local image blocks using sampling-invariant intensity differences. We therefore extract only the optimization-core of the OpenCV implementation and use it for pixel-wise optimization of our different matching costs.

Graph Cuts

For our experiments with iterative global disparity optimization via Graph Cuts (GC) we use the energy minimization library (MRF library) released together with a survey paper on energy minimization methods for Markov random fields by Szeliski et al. [51]. The MRF library is available online[5] and includes contributions from several authors [8][7][31]. We use the supplied stereo matcher frontend and adapt it for use with our different matching cost measures.
In Section 2.1.3 we have already given a short description of how global energy minimization methods are used to find local minima of energy functions of the form

$$E(D) = E_d(D) + \lambda E_s(D) \tag{3.12}$$

with a data term $E_d(D)$ and a smoothness term $E_s(D)$ depending on the disparity function D (i.e. the labelling of pixels with disparity values). The data term consists of the matching cost values as defined in Equation 2.2. In the used MRF framework of [51] the smoothness term for stereo correspondence is defined as

$$E_s(D) = \sum_{\{p,q\} \in N} w_{pq} V(|D_p - D_q|) \tag{3.13}$$

[5] http://vision.middlebury.edu/MRF (July 2011)

where p and q are adjacent pixels in the considered pixel-neighborhood N (here a standard 4-connected neighborhood). The smoothness cost function $V(\Delta D)$ is a nondecreasing function of the disparity differences at p and q and w_{pq} are multiplicative weights for each considered pair. As per [51] the smoothness cost function $V(\Delta D)$ is defined as

$$V(\Delta D) = \min(|\Delta D|^k, V_{max}) \tag{3.14}$$

with $k \in \{1, 2\}$. The weights w_{pq} of the smoothness term can be increased if the image gradient magnitude lies below a given threshold.
In our experiments we use the default *expansion-move* graph cut algorithm of the MRF library due to its good overall performance reported in [51]. For details on energy minimization for Markov random fields and graph cuts in particular we refer to the respective literature [51][8][7][31][50].

Comparison of Optimization Methods

To illustrate the different properties of the implemented disparity optimization techniques described above, Figure 3.5 shows sample results on the 'Teddy' image pair of the Middlebury dataset. We use grayscale input images of size 450x375 pixels with a disparity range of 64 pixels and compute the matching costs via simple absolute differences. The shown disparity maps are computed with reference to the left image and limited by the maximum disparity range with respect to the left image border. The top row of Figure 3.5 displays the input images and the available ground truth. The middle row shows disparity computation results using WTA with prior cost aggregation using a square box filter as well as a guided filter. The rightmost result in the middle row is created using SO optimization and vertical aggregation with adaptive weights. The bottom row shows results of DP optimization with the vertical aggregation method as well as SGM and GC optimization without any prior cost aggregation. In Figure 3.5 the problems of WTA methods in bland and textureless regions can be seen clearly. The cost aggregation via guided filtering gives better results in comparison to simple aggregation via box filters and disparity discontinuities are preserved well. However, our experiments indicated that the performance of the guided filter aggregation depends heavily on the structure of the matching cost values in the DSI. We verified this observation by varying the matching costs and truncation values used in the sample MATLAB implementation provided with the original paper [42]. In this way we achieved results very similar to the ones in our own implementation. However, we could only reproduce the very good results reported in [42] by using the exact setup of the sample implementation (i.e. color images, matching costs and truncation values and postprocessing steps). For the result in Figure 3.5 we truncated the absolute differences before the guided filtering. Furthermore it can be seen that the SO and DP optimization methods can deal better with bland regions but despite vertical adaptive aggregation lead to the mentioned horizontal streaking artifacts. The SGM and GC based optimization methods can overcome textureless regions and do not create artifacts like DP and SO. Our used SGM optimization gives smoother results while the GC results appear blockier

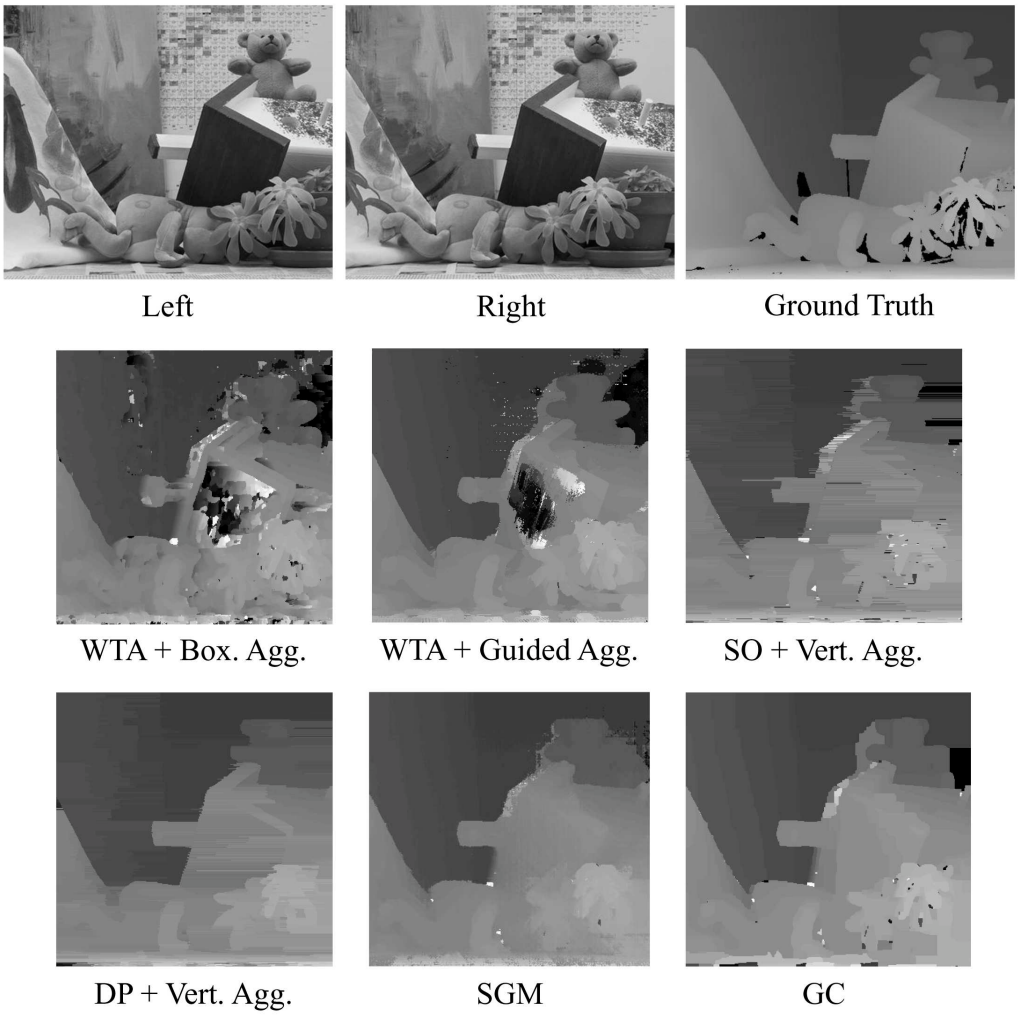

Figure 3.5: Example of disparity optimization results on the Middlebury 'Teddy' image pair.

but preserve discontinuities slightly better. However, in this example in combination with absolute differences all implemented optimization methods have problems with the homogeneous region at the top right of the images. It has to be noted that this is just a qualitative example and we achieve better results by using e.g. the census transform matching cost. Also the optimization results naturally depend on the selected parametrization which we adjust here to achieve best results illustrating the different overall properties of the optimization methods.
The optimization runtimes of our SO and DP implementations are approximately 4 seconds and 2 seconds respectively on a standard laptop (2.4GHz Intel Core i5 CPU). The SGM optimization takes 1 second while the iterative GC algorithm takes approximately 30 seconds and five iterations until convergence.

3.2.5 Post-Processing

Left-Right Consistency Check

The left-right consistency check (or cross-check) is often used by local WTA-based methods to detect occlusions and disparity outliers [19][10]. Hirschmueller also performs a cross-check in his semi-global matching algorithm [26][27]. Two disparity maps for left-right and right-left matching are computed and compared. If disparity values differ by more than a given small threshold they are considered as invalid and marked as such. This is mostly the case in occluded image regions but can also be caused by other types of inconsistent matches. The detected invalid regions in the disparity map can be interpolated from adjacent valid disparities (e.g. as in [27]) but we do not include this interpolation step in our implementation.

Speckle Removal

Speckle removal is performed to remove isolated patches of invalid disparity values which can be caused by ambiguous matching costs [27]. We utilize the speckle removal method available in the SGM implementation of the OpenCV library [9][34]. The disparity map is segmented into patches, where the disparity values within one patch are allowed to differ by a given value. Patches smaller than a given threshold are considered as disparity outliers and removed. Again as a possible refinement step the invalid regions can be interpolated as in [27].

3.3 Summary

In this chapter we have described our implemented dense stereo correspondence framework. We have given details on the building blocks of the framework and have discussed our implementation of the various methods that can be used in the different stages. Our framework allows for a wide range of combinations of matching cost measures, aggregation methods and disparity optimization methods. In this way we are able to perform extensive experiments to test and evaluate the performance of the different combinations on different types of test data, providing valuable information concerning our investigations towards a method for dense cross-spectral correspondence. In the next chapter we will describe our experiments in detail and will provide a discussion and evaluation of our results on both simulated and real cross-spectral stereo images.

Chapter 4

Results and Evaluation

In this chapter we describe our tests and evaluation of our implemented dense stereo algorithms on different types of test data. It has to be noted that before beginning with the actual evaluation we verified the correct functionality of the algorithms on standard optical stereo images. This was done to remove possible errors in our implementation and we will not describe this process in detail here.
In the first step of the evaluation we perform tests on simulated cross-spectral stereo data, i.e. standard stereo images with complex intensity transformations. This is a common approach to gain insight into the robustness of matching cost measures in the absence of real cross-spectral data as seen in our review in Sections 2.2.3 and 2.3. Furthermore this allows for a comparison of the results from the simulated cross-spectral tests with results from real cross-spectral data.
In the second step of the evaluation we focus on the main problem of our work, stereo correspondence between real cross-spectral images.
In the following section we will give an overview of the used test data sets and describe our method of acquisition and preparation of real cross-spectral stereo data. Subsequently we will move to the actual evaluation procedure and describe and discuss our results in detail.

4.1 Test Data

4.1.1 Standard Stereo Data

For the verification of the correct functionality of our algorithms as well as for the simulated cross-spectral stereo tests we use publicly available and commonly used stereo data. In particular we use images from the Middlebury data set [43] and from the CMU JISCT data set[1]. Figure 4.1 shows the respective left images of the used stereo pairs. The Middlebury 'Teddy' image pair (top left) is of size 450x375 pixels with a disparity range of 64 pixels. The Middlebury 'Tsukuba' data (top

[1] http://vasc.ri.cmu.edu/idb/html/jisct/index.html (June 2011)

right) is of size 384x288 pixels with a relatively small disparity range of 16 pixels. As mentioned previously the used Middlebury data represents images taken in a controlled environment with low noise. The bottom row in Figure 4.1 shows the 'Parkmeter' (left) and 'Shrub' (right) images from the CMU data set. These images of outdoor scenes are slightly noisier than the Middlebury images and are both of size 512x480 pixels with a disparity range of 32 pixels.

Figure 4.1: Standard stereo test data.

4.1.2 Cross-Spectral Stereo Data

Data Acquisition

In contrast to various standard stereo data sets consisting of pre-rectified image pairs which are provided online by academic institutions (e.g. Middlebury [43], CMU) real cross-spectral stereo data is currently not readily available. For this reason we set up a custom cross-spectral stereo rig to gather suitable test and evaluation data.
Our cross-spectral stereo rig consists of a *Miricle 307k* uncooled far infrared camera with a spectral response of 8μm-12μm and an optical camera of type *Visionhitech VC57WD-24*. Both cameras provide resolutions of 640x480 pixels. We experimented with static stereo setups for indoor use as well as mobile setups for outdoor data collection. In our test and evaluation process we will focus mainly on outdoor data due to the fact that most possible application areas (see Section 1.1) are based on real-world noisy outdoor scenarios. Furthermore the often very homogeneous temperature profile of our air conditioned laboratory rooms showed to make the acquisition of thermal images with sufficient contrast and detail for dense stereo difficult. Figure 4.2 depicts the different mobile cross-spectral stereo setups we

utilized for data collection. We experimented with the use of a four-wheeled trolley (Figure 4.2 left) with the cameras mounted at a baseline of approximately 10cm. However, the mounting possibilities of the cameras proved to be too unstable during mobile operation leading to low quality data and bad image alignment. As a result we switched to a more advanced mobile platform and achieved much better results by using an electric powered mobile robot of type *Mobile Robots Pioneer 3-AT*. For a first data acquisition phase we equipped the robot with our cross-spectral stereo camera pair (infrared left and optical right) with a baseline of 30cm (Figure 4.2 center). In a second phase we added an additional optical stereo camera pair with the same baseline (Figure 4.2 right) to be able to directly compare the cross-spectral results with standard stereo results of the same scenes. Figure 4.3 shows the final dual stereo setup on our mobile robot in the laboratory (left) and during a data collection run on Cranfield University Campus (right).

Figure 4.2: Different mobile cross-spectral stereo setups.

Figure 4.3: The four-camera dual stereo setup on our mobile robot.

We used a laptop mounted on the mobile robot to capture synchronized video input from the stereo camera pair using a custom-made capture software utilizing the OpenCV library [9][34]. For the acquisition of four video inputs from the dual stereo setup we resorted to the use of two laptops, requiring a subsequent manual synchronization of the two separate stereo data videos. Overall we collected more than two hours of cross-spectral stereo video data of real-world outdoor scenes. This includes data taken at different times of day (morning/afternoon) and under different weather conditions (sunny/cloudy).

Calibration

In Section 1.3.1 we have already described the concept of calibrating a stereo setup and rectifying the input images. A very common method is as we have seen the use of several images of a planar calibration target of known geometry at different orientations. For standard stereo rigs often a black and white checkerboard pattern with known pattern size is used. Calibration procedures like [59] use the corners extracted from the pattern in all image pairs to compute the camera parameters. Implementations for camera calibration are available for example in form of the *Caltech Camera Calibration Toolbox for MATLAB* [6] or the corresponding implementation in the OpenCV library [9][34]. These solutions are mainly based on the work of Zhang [59][60] and Heikkilae and Silven [25].
To be able to apply common calibration methods to a cross-spectral stereo setup a calibration pattern which is visible in both spectra is needed. To solve this problem we use a metal board on which a checkerboard pattern is marked with adhesive tape. Before taking the series of calibration images we heat up the calibration board using a commonplace blow-dryer or a high-power halogen lamp. The different temperatures of the materials of the board and the tape allow for thermal images with reasonable detail which can be used in the calibration process. The temperature has to be adjusted with care because a too hot or cold board can result in images with thermal halos or too little detail respectively. Figure 4.4 shows example images of our cross-spectral calibration board (top) and a standard optical calibration board (bottom). The standard optical calibration board was used to calibrate the separate optical stereo camera setup we used for comparison of our results. It has to be noted that the images in Figure 4.4 are an example of the best quality we could achieve for our cross-spectral calibration images. In general the generation of cross-spectral calibration images with sufficient accuracy and quality in both spectra turned out to be much more challenging than taking calibration images for standard optical cameras.
To perform the calibration of both cross-spectral and standard stereo setups we use the *Caltech Camera Calibration Toolbox for MATLAB* [6]. The toolbox provides a graphical user interface in which the outer corner points of the calibration pattern in each image have to be marked manually for initialization followed by an automatic extraction of the inner corners. The implementation available in the OpenCV library provides a fully automatic corner detection feature which performs well for standard optical stereo calibration patterns. However, the corners in our cross-spectral cali-

bration images are in general too indistinct for fully automatic operation and require the manual initialization step. Subsequently the calibration toolbox for MATLAB performs calculation and optimization of both intrinsic and extrinsic parameters of the stereo setup based on the extracted image corners. Figure 4.5 shows the reconstructed configuration of the cross-spectral stereo rig as part of the final dual stereo setup on our mobile robot. The shown configuration is computed from the extrinsic stereo parameters determined in the calibration. The two cameras are displayed in red on the left and the reconstructed positions of the used calibration board can be seen on the right (distances are given in millimeters). A detailed documentation of the calibration procedure and parameters of the used toolbox is available online [6].

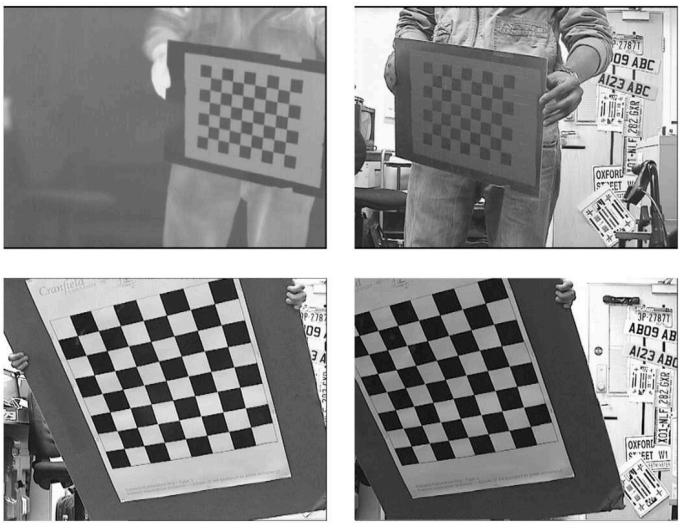

Figure 4.4: Cross-spectral (top) and standard optical (bottom) stereo calibration boards.

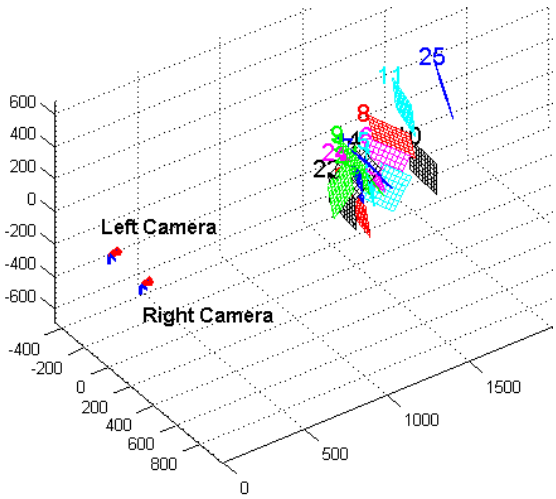

Figure 4.5: Cross-spectral stereo setup reconstructed from extrinsic parameters.

The intrinsic and extrinsic parameters of the stereo rigs computed in the calibration procedure are saved and used by our implemented stereo correspondence framework to rectify and undistort the input stereo data. Figure 4.6 shows an example of a cross-spectral input image pair before and after rectification. To illustrate the effect of the rectification the images are overlaid with sparse horizontal scanlines. It can be seen that in the raw input images (top) corresponding points in the scene do not lie on corresponding scanlines. After rectification and undistortion (bottom) corresponding points in the scene lie on the same horizontal scanlines. On the right of Figure 4.6 a zoomed region of the example images is shown where this result can be seen more clearly.

Figure 4.6: Example of a cross-spectral image pair before (top) and after (bottom) rectification.

4.2 Simulated Cross-Spectral Stereo

4.2.1 Methods and Parameters

For an initial assessment of the robustness of the implemented matching cost measures we perform tests on synthetically altered optical stereo images. We have seen in Sections 2.2.3 and 2.3 that this method is commonly applied in literature to demonstrate invariance of matching cost measures to radiometric differences or to simulate cross-spectral stereo data.
We perform the tests on the data shown in Section 4.1.1 and transform the left images of the stereo pairs as

$$I^* = 255 \left| \cos\left(I \frac{\pi}{255} \right) \right|, \tag{4.1}$$

where I represents the intensity values of the original image and I^* the transformed intensity values. This non one-to-one intensity transformation is very similar to the one used by Fookes et al. [17][18].

In the following we present the results of our tests on the simulated cross-spectral images in three steps. In a first step the standard robust parametric and non-parametric matching cost measures ZSAD, ZNCC, rank and census as described in Section 2.2.2 and in [28] are applied to the transformed image pairs. We use windows of size 11x11 pixels for the computation of the window-based matching costs and a 7x7 local region for the Census transform.
In the second step we investigate the effect of different basic preprocessing methods as described in Section 2.2.1 and in [28] in combination with standard robust matching costs.
Finally we consider the more advanced and also novel robust matching cost measures as selected in Section 3.1 and described in Section 3.2.2. For our hierarchical MI method based on the work of Fookes et al. [17][18] we use windows of size 15x15 pixels and 16 histogram bins to compute the probability distribution functions. As recommended in [18] we select a value of $\lambda = 0.4$ which defines the weight of prior probabilities and probabilities of the local windows (see Section 3.2.2). Furthermore we restrict the disparity search range in the second stage of the hierarchical computation to half of the original range, centered on the results of the first stage.
For our standard MI method we use the same basic parameters as for the hierarchical method. However, we also test a value of $\lambda = 1$ which means that no prior probabilities are taken into account, making it equivalent to the window-based MI method proposed by Egnal [15].
For the computation of LSS descriptors we use patches of 5x5 pixels, surrounding regions of 35x35 pixels and a log-polar grid with 4 radial and 12 angular bins. For the DAISY descriptors we use unsigned gradient orientations and the parameters[2] $R = 5$, $Q = 3$, $T = 4$ and $H = 8$ (see Section 3.2.2 and [53]). In our tests we

[2]Note: Through additional testing and evaluation these parameter values were improved from the ones stated in the version of this thesis submitted to Cranfield University. The present results and conclusions are based on this improved set of parameters.

achieve better results by normalizing the complete descriptor instead of each histogram separately. For the dense HOG descriptors we use local windows of 18x18 pixels split into 3x3 histogram cells and bin the unsigned gradient orientations into 9 intervals. The matching costs between the respective descriptors are computed by the L1 distances for LSS and HOG and the L2 distances for DAISY descriptors.

In the presented tests on the simulated cross-spectral data the disparity values are computed using WTA to display the robustness of the matching cost measures independently of more advanced disparity optimization methods. The matching costs are aggregated using a box filter of size 11x11 with equal weights to smoothen the disparity values and filter outliers. Only for the hierarchical MI implementation we omit the cost aggregation as in [18]. It has to be noted that our main focus lies on the basic ability of the matching costs to produce valid disparity maps. We therefore accept inaccuracies and blurring at disparity discontinuities in favor of an overall valid dense disparity map.
In the following we show only a selection of our results but similar results were achieved for all of the considered images presented in Section 4.1.1. Here we discuss the Parkmeter and Shrub stereo pairs as an example of real-world outdoor scenes. Figure 4.7 shows the original stereo pairs and their respective disparity maps computed using ZNCC as a reference. Figure 4.8 shows the left images of the stereo pairs after transforming the intensity values as defined in Equation 4.1.

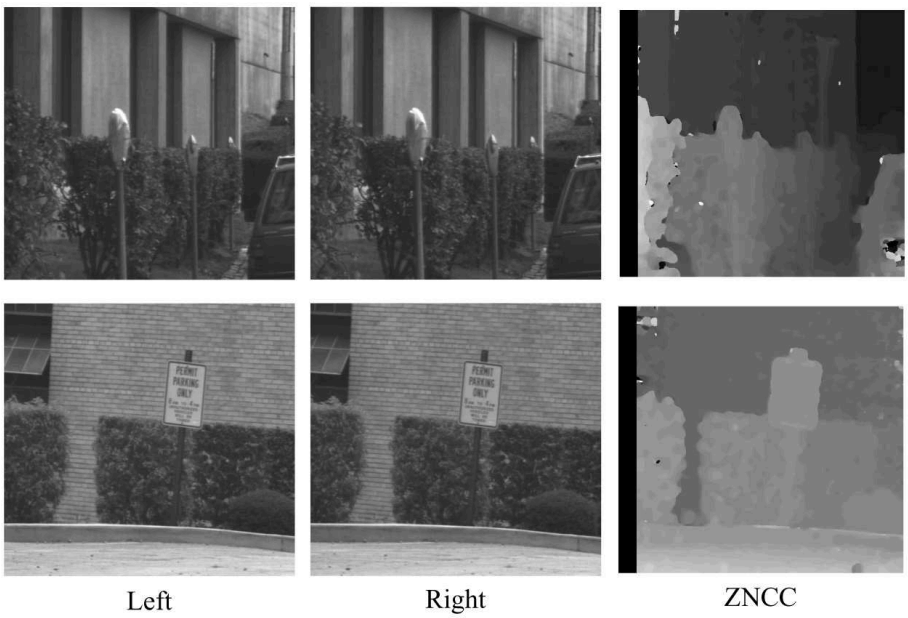

Figure 4.7: Unmodified Parkmeter and Shrub stereo pairs and respective disparity maps.

Figure 4.8: Transformed left images of the Parkmeter and Shrub stereo pairs.

4.2.2 Performance Comparison

In the first step as described above we test the matching costs ZSAD, ZNCC, rank and census on the simulated cross-spectral data. Figure 4.9 shows the resulting disparity maps for the Parkmeter stereo pair. It can be seen that none of the methods can cope with the complex intensity transformation and all fail to produce valid disparity values. This is an expected result as the limitations of the intensity transformations which these methods are invariant to are not met by the applied transformation.

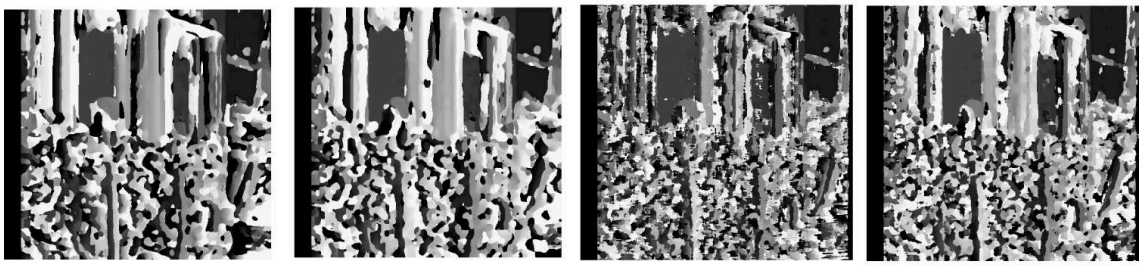

Figure 4.9: Disparity maps of standard robust matching cost methods for the transformed Parkmeter stereo pair.

In the next step we investigate if the shown results of standard matching cost methods can be improved by the preprocessing steps described in Section 2.2.2. Figure 4.10 shows the disparity results for the transformed Parkmeter stereo pair using ZNCC after application of the preprocessing methods. It can be seen that mean filtering, Laplacian of Gaussian (LoG) filtering and background subtraction by bilateral filtering (BilSub) are not able to compensate for the complex intensity transformation. This is due to the fact that these methods are designed to only remove local intensity offsets as described in Section 2.2.1 and in [28]. It is notable that the computation of the image gradient magnitude and the subsequent ZNCC computation leads to a valid disparity map with relatively few artifacts as shown in the last row in Figure 4.10. The basic gradient magnitude information of the images appears to offer enough invariance to the complex intensity transform to yield largely unique ZNCC matching cost minima.

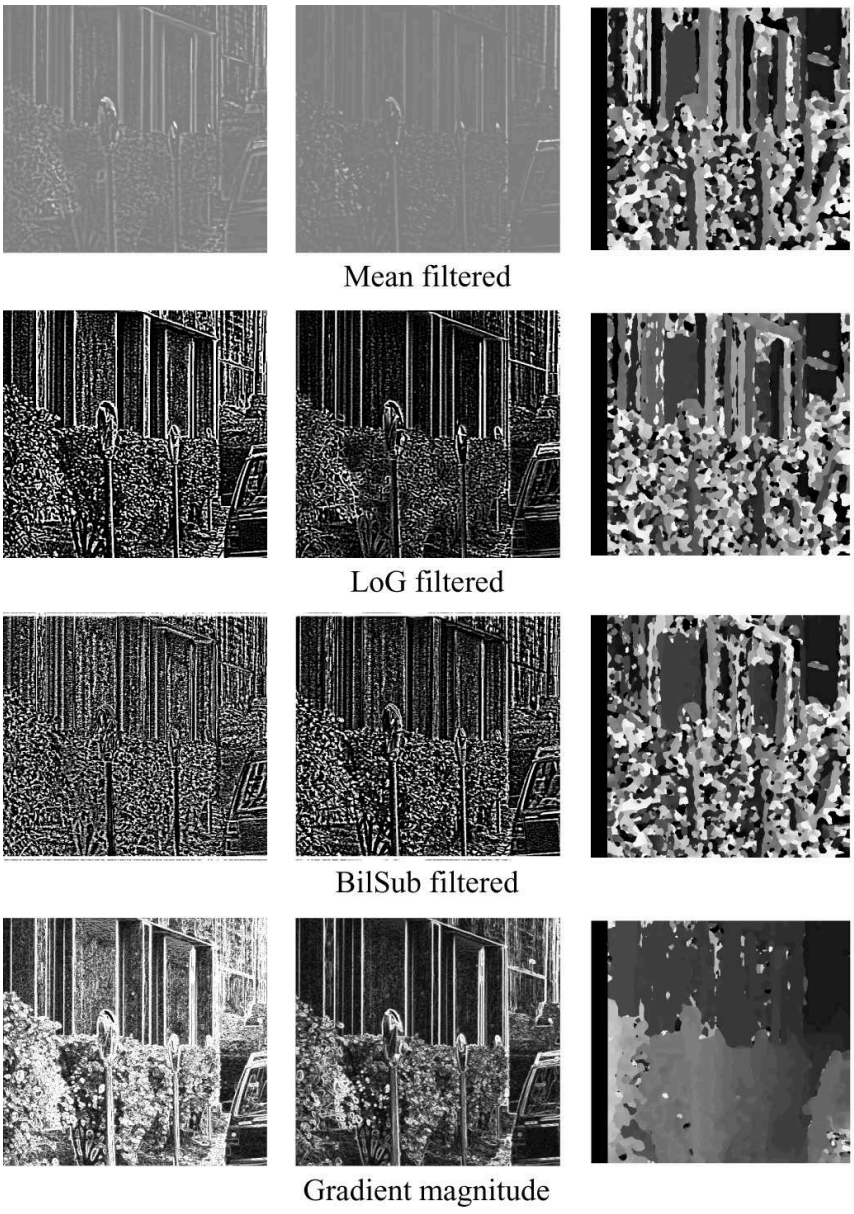

Figure 4.10: Intensity transformed and preprocessed Parkmeter stereo images and the respective disparity maps computed using ZNCC.

Finally we move to the more advanced robust matching cost measures MI, LSS, DAISY and HOG. Figures 4.11 and 4.12 show the disparity results of these methods for the transformed Parkmeter and Shrub stereo pairs. In both figures the top left image shows the disparity map computed using our hierarchical MI method based on the work of Fookes et al. [17][18] and described in Section 3.2.2. The center and right images in the top row show results computed with our standard MI method using different values of λ. It can be seen that the results in the center image are very similar to the result of the hierarchical MI implementation. The bottom left image shows the results using distances between dense LSS descriptors as a matching cost and the bottom center image shows the results using DAISY descriptors. Finally the bottom right image shows the results using distances between dense HOG descriptors. The slightly different sizes of the computed disparity maps are a consequence of the border handling we employ for the different matching costs.
It can be seen in Figures 4.11 and 4.12 that all methods are able to produce largely valid disparity maps using simple WTA disparity computation. However, different levels of artifacts and disparity outliers are present in the results. The MI methods perform better on the Shrub than on the Parkmeter stereo pair and it can be seen that the consideration of prior probabilities, even in the non-hierarchical methods, can reduce the amount of disparity outliers. These results of our window-based MI implementations are in accordance with the results reported in the respective literature (Section 2.2.3, [15][17][18]). In the disparity maps computed using LSS several small artifacts can be seen, especially in the areas of very fine repetitive texture in the Shrub images. DAISY also introduces some disparity outliers in these regions but produces good results otherwise. The dense HOG descriptors show a relatively constant performance and provide good results for both image pairs. However, they also result in a noticeable blurring of disparity discontinuities.

Performance Comparison Summary

We have seen in our experiments on simulated cross-spectral stereo data (i.e. optical images with complex intensity transformations) that standard robust matching cost measures like ZSAD, NCC, ZNCC, rank and census transform cannot cope with this scenario. Furthermore common preprocessing steps for the compensation of standard radiometric differences prove to be unsuitable for this task. Only the computation of image gradient magnitudes showed to create input images which can be matched by robust matching cost measures like ZNCC. The tests of more advanced and novel matching cost measures like MI, LSS, DAISY and HOG showed their ability to produce valid matching cost values with the potential for optimization by common disparity optimization techniques.

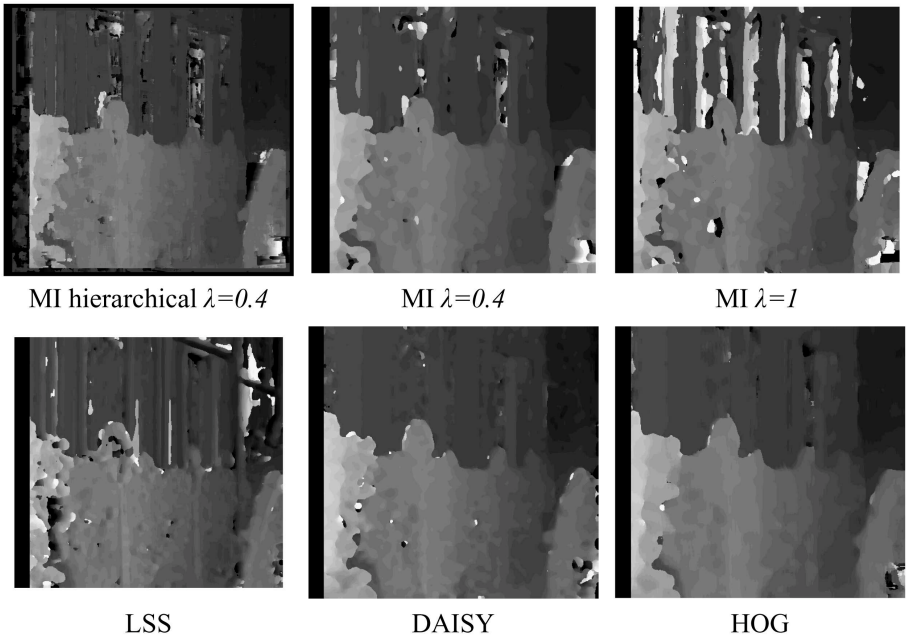

Figure 4.11: Results of advanced robust matching cost methods for the transformed Parkmeter stereo pair.

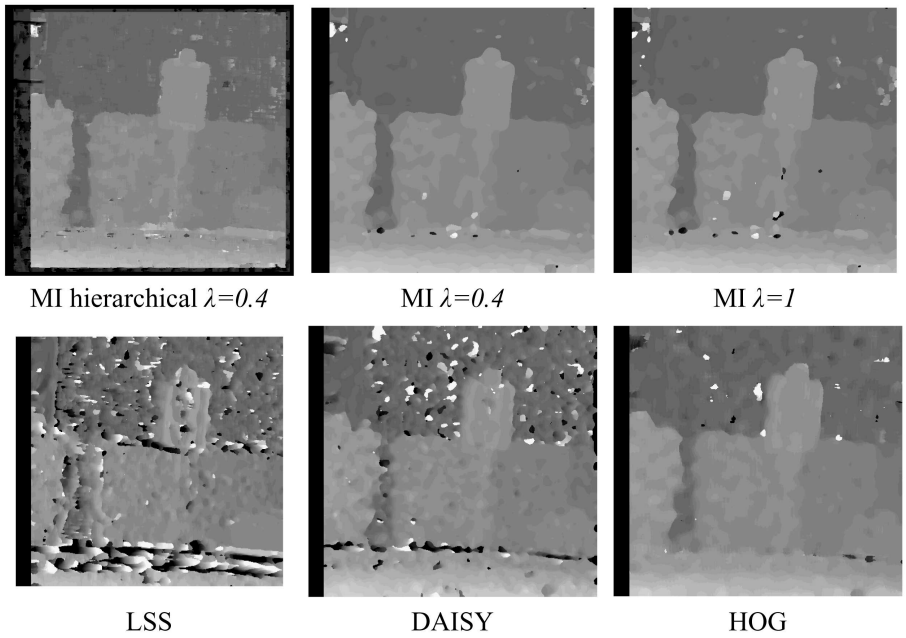

Figure 4.12: Results of advanced robust matching cost methods for the transformed Shrub stereo pair.

4.3 Real Cross-Spectral Stereo

4.3.1 Methods and Parameters

In this section we will now focus on our ultimate goal of computing dense stereo correspondences between real cross-spectral images. We have seen in the previous section that several different matching cost methods are able to produce valid results on simulated cross-spectral images. Based on these results we will further investigate the performance of MI, LSS, DAISY and HOG matching cost measures on real cross-spectral images. We also include ZNCC in combination with prior gradient magnitude computation due to the results achieved in Section 4.2. However, as already discussed in Section 2.3 real cross-spectral images present a much more challenging problem than the synthetically altered optical images considered so far. Figure 4.13 shows an example of a cross-spectral image pair taken with our stereo setup and Figure 4.14 shows the corresponding disparity maps computed using the selected matching costs and WTA as in the previous section. It can be seen that none of the methods can provide matching costs which are unambiguous enough for simple WTA disparity computation. As a result no dense disparity map which is consistent with the structure and depth of the scene can be computed. This fact also becomes clear by analyzing the DSI matching cost volume. Figure 4.15 shows examples of horizontal slices through the DSI at *y=100* and *y=300*, here the matching cost values are linearly stretched for better visualization. The positions of the DSI slices are also indicated by the horizontal lines in Figure 4.13. This example shows that in general the matching costs are able to produce notable minima around prominent scene features present in both spectra (e.g. body of person, concrete pillars). However, homogeneous regions and regions which appear very differently in both spectra (e.g. details present in one spectrum and not in the other) produce ambiguous values and wrong minima in the DSI. Figure 4.15 illustrates the different structure of the various matching costs and how they are affected by the described problematic image regions.

Figure 4.13: Cross-spectral stereo example scene.

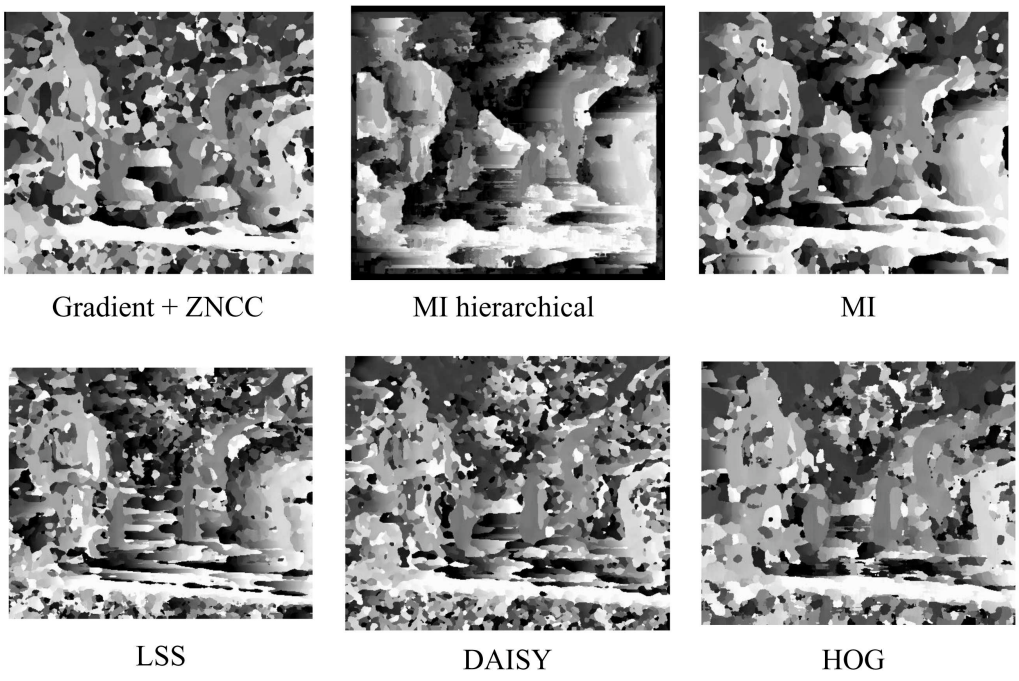

Figure 4.14: Results on cross-spectral images using WTA disparity computation.

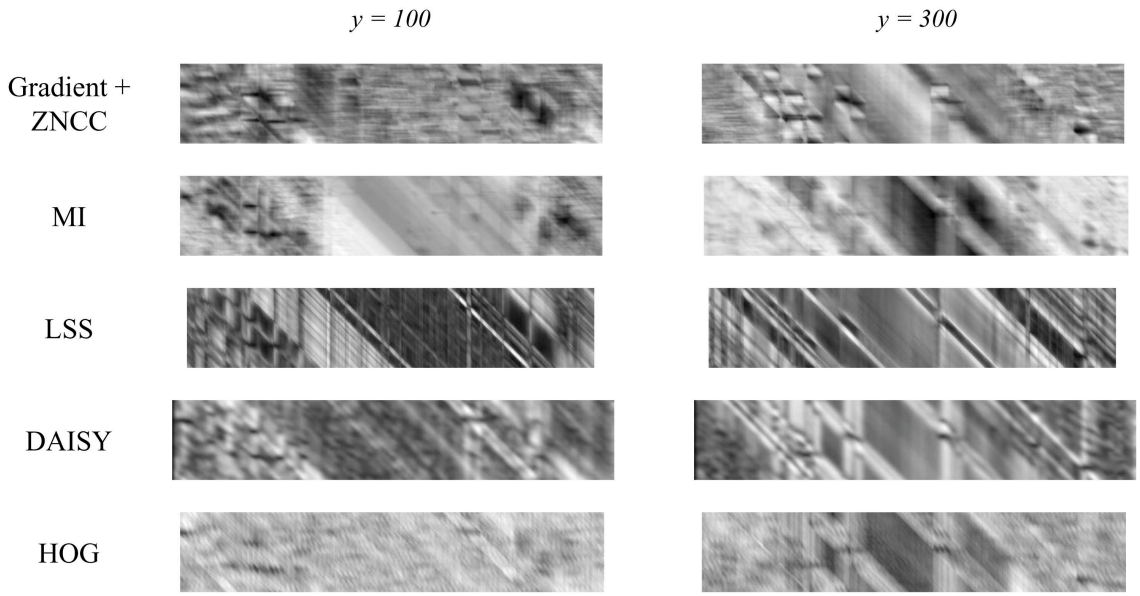

Figure 4.15: Example DSI slices of different matching costs computed from cross-spectral images.

These observations also indicate why approaches considering only regions of interest (e.g. people) [33][55] for cross-spectral window-based disparity estimation work well. For example the approach of Torabi and Bilodeau [55] (see Section 2.3 and Section 3.2.2) uses filtered LSS descriptors for a sliding-window disparity voting to register people in cross-spectral stereo images. They consider only informative LSS descriptors which are determined by the criteria proposed in [46] and mentioned in Section 3.2.2. Figure 4.16 illustrates the regions which yield informative LSS descriptors according to these criteria on the sample scene shown in Figure 4.13. It can be seen that in both spectra many informative descriptors are located around the prominent scene features mentioned above, especially on the depicted person. These regions also largely correspond to the valid disparity values that can be recovered using WTA as shown in Figure 4.14. The approach in [55] relies on the assumption that a sufficient number of informative descriptors are available in both spectra for the window matching which is true for the selected region of interest (i.e. person). However, large regions of the whole scene yield descriptors classified as uninformative which of course makes a dense disparity estimation of the scene more difficult.

Figure 4.16: Regions of filtered informative LSS descriptors (white) on thermal (left) and optical (right) images.

In consideration of the observed weak matching cost values we investigate how the implemented optimization methods can be utilized to improve the results and if the computation of a valid dense disparity map is indeed possible. As already stated in Section 3.1.5 the combination of window-based matching cost methods like MI or ZNCC with advanced disparity optimization techniques can be seen as quite uncommon as these optimization techniques are usually applied to purely pixel-based matching costs [50][28][44]. However, as we have seen in our review in Section 2.3, state-of-the-art robust pixel-based matching cost and disparity optimization methods fail at the problem of cross-spectral stereo. Also our experiments have shown that robust window-based matching costs like MI fail in combination with simple WTA disparity computation. This motivates our approach of investigating the performance of all implemented matching cost measures in combination with disparity optimization methods.

Before we present and discuss results on a number of different test scenes we will in the following describe our chosen methods and parameters. For the window-based matching costs MI and ZNCC we use large windows of 21x21 pixels to increase

discriminative power. For the MI method we use prior probabilities but due to the larger window size we only weight them with 30% ($\lambda = 0.7$). For the matching costs computed via HOG, DAISY and LSS descriptors the same settings as described in the previous section (Section 4.2.1) are used. We also experimented with different layouts, sizes and distance measures for all descriptors. For example for the LSS descriptor computation we also tested the use of larger local regions (correlation surfaces) of 45x45 pixels and 55x55 pixels and tried different log-polar as well as rectangular grid arrangements. However, we could not achieve consistently better results than with the selected parameters. The same parameters for all methods were also used for the results shown in Figure 4.14. The implemented hierarchical MI method is less suitable for combination with our disparity optimization methods and due to the observed similar results we will only consider the standard window-based MI method in combination with the disparity optimization methods.
Regarding the implemented disparity optimization methods we manually tuned all parameters of each method in combination with the different matching costs (for the respective parameters see also 3.2.4). In the absence of ground truth data we used a number of sample scenes for the parameter tuning and tried to qualitatively minimize the amount of artifacts and disparity outliers while aiming for disparity maps consistent with the objects and depth levels in the scenes. Additionally we used the corresponding disparity maps computed from the standard optical stereo pair as a visual reference. After setting the parameters we kept them constant for our experiments and the results shown in the following discussions.
We found in our experiments that the results of Scanline Optimization (SO) were mostly similar to or slightly worse than the results using traditional Dynamic Programming (DP) optimization. For this reason and due to their very similar nature we will show and discuss only the results of DP optimization in the presented test scenarios. Figure 4.17 shows an example of the results of SO and DP optimization applied to the HOG matching costs for the scene shown in Figure 4.13. Occlusions detected in the DP method are filled with the nearest disparity values from the left.

Figure 4.17: Disparity maps of SO (left) and DP (right) optimization methods using HOG matching costs.

We also compared the performance of Semi-Global Matching (SGM) optimization and global disparity optimization via Graph Cuts (GC) on the different computed matching costs. Depending on the matching cost we found that the used GC

optimization often lead to extreme smoothing and very blocky results. Trying to avoid this by varying parameters or reducing the smoothness weight however resulted in excessive artifacts. SGM produced smoother and more stable results on our test data but in general lead to a number of small patches of disparity outliers. However, these outliers can be reduced by post-processing steps like cross-checking and speckle removal. Figure 4.18 shows an example of GC and SGM optimization without any post-processing for the sample scene shown in Figure 4.13 and HOG matching costs. This example illustrates the best results we could achieve with the used GC optimization. However, depending on the scene the GC optimization took between three and four minutes to compute the results for a disparity range of 96 pixels while SGM optimization took approximately 1.5 seconds. In the following we will therefore present and discuss our results using SGM optimization in combination with the implemented post-processing steps. Cross-checking is applied to detect occlusions and inconsistent disparities and speckle removal is used to remove isolated disparity outliers. Pixels marked as occlusions or invalid disparities are set to zero in the result images.

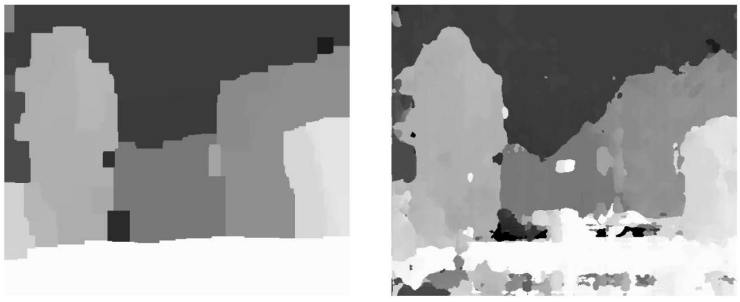

Figure 4.18: Disparity maps of GC (left) and SGM (right) optimization methods using HOG matching costs.

In addition to the optimization methods we investigated the use of the implemented cost aggregation methods (see Section 3.2.3). The adaptive vertical cost aggregation method showed to improve the results of SO and DP optimization and we utilized it in all our tests with a window size of 1×31 pixels. We also considered the weighted cost aggregation method using guided filtering on our matching costs. However, as already discussed at the end of Section 3.2.4 this aggregation method showed to be very sensitive to the structure of the matching costs. We experimented with different scalings and truncation values of the matching cost values but could not achieve improved results through this aggregation method. In the shown results using SGM optimization a simple Gaussian weighted cost aggregation (size 11x11, $\sigma = 2.2$) to smoothen the disparity maps is applied.
To remove noise in the input images we use a 3x3 median filter as a preprocessing step which showed to slightly improve results.
In the following subsection we will now present and discuss our test results on a selection of different scenes taken with our mobile cross-spectral stereo setup. All input images are of size 640x480 pixels and we use a disparity range of 96 pixels for all shown experiments. It has to be noted that due to the setup of the stereo

rig on our mobile robot the position of the cameras is very close to the ground. As a result the maximum disparity of scene points on the ground close to the robot have a true disparity larger than the considered disparity range. Furthermore the ground regions in most of our scenes present extremely homogeneous regions in the thermal (and also optical) images causing a large amount of false disparities. In the subsequent discussion we will therefore concentrate on the recovered disparities of the main structure and objects of the scene and put less focus on disparity outliers in the described homogeneous ground regions.
The shown disparity maps are computed with reference to the left input image (the thermal image) and limited by the maximum disparity range with respect to the left image border.
We also present the disparity maps computed from the images taken with our separate optical stereo setup for comparison. It has to be noted that due to the limited mounting possibilities of these additional cameras and the vibrations caused by the moving robot the alignment of the images in several scenes is imperfect. Furthermore the quality of the images captured by the used optical cameras in terms of contrast and noise is relatively low. The DAISY descriptor turned out to cope well with these problems and therefore we apply it in standard mode (signed gradient) with parameters $R = 5$, $Q = 3$, $T = 4$ and $H = 8$ which are also shown to give good results for standard stereo in [53].

4.3.2 Performance Comparison

Still Images

We discuss the performance of the different matching costs in combination with DP and SGM disparity optimization on six different test scenes. The first scene ('Scene 1') is the previous example scene shown in Figure 4.13. This scene was taken with our first cross-spectral setup on the mobile robot and therefore no optical stereo results are available for comparison. The cross-spectral input images of the other five scenes ('Scene 2' - 'Scene 6') are shown in Figure 4.19. The left column displays the images from the left thermal camera and the right column the images from the right optical camera. Additionally Figure 4.20 shows the optical stereo images of the respective scenes taken with the separate optical stereo setup. The computed disparity maps for the test scenes using the considered matching cost methods gradient magnitude and ZNCC, MI, LSS, DAISY and HOG are shown in Figures 4.21 - 4.26. The left column of the figures shows the results using DP optimization and the right column using SGM optimization.
As already noted, ground-truth data is not available for our cross-spectral test data to allow for a quantitative comparison of matching performance. We therefore qualitatively evaluate the different methods by assessing the ability to recover the basic scene structure and the amount of obvious disparity outliers, also considering the invalid disparities removed by the left-right cross-check which are set to zero (i.e. black). For comparison the results obtained from the standard optical stereo images are shown in the respective first row of each figure for Scenes 2 to 6.

Figure 4.19: Cross-spectral stereo images of the discussed test scenes.

Figure 4.20: Optical stereo images of the discussed test scenes.

It can be seen in all scenes and for all matching cost methods that in a qualitative comparison SGM gives much better results which are more consistent with the objects and depth levels of the scenes than DP. The characteristic horizontal streaking artifacts of DP are clearly visible, distorting the results depending on the respective scene and matching cost. Only on the optical image pairs the DP optimization performs well and produces results of similar quality to SGM. This demonstrates the importance of the additional 2D smoothness constraints of SGM for dealing with the weak and ambiguous matching costs of cross-spectral images. The exclusively horizontal 1D constraints of DP optimization appear to be too weak to create results of good quality which are consistent over different scenes. While the results using DP optimization still provide some insight into the performance and robustness of the different matching costs, we focus mainly on the analysis of the results computed through SGM.

In Scene 1, which has relatively good contrast and details in both spectra, all matching cost methods are able to produce disparity maps which are largely consistent with the structure and depth of the scene. A different amount of artifacts and invalid disparities are present in the results but the contours and depth levels of the main objects (person, shrubs, concrete pillars) are discernible in all methods. It has to be noted that the nature of the implemented matching costs which all rely on local image regions leads to a blurring of disparity discontinuities. This is further increased by the relatively strong smoothness constraints which have to be applied for valid results. However, as already stated we accept these inaccuracies in favor of an overall valid dense disparity estimation.

In the results for Scenes 2 to 6 the different characteristics of the matching costs can be seen more clearly. The straightforward method of gradient magnitude computation with ZNCC matching turns out to perform well in recovering main objects, in most scenes even doing better than MI and LSS. However, ZNCC also introduces a considerable amount of artifacts in some scenes. The person in Scene 4 and the car in Scene 6 are distorted by artifacts and invalid disparities. Furthermore the DP optimization performs particularly bad here.

The window-based MI method results in significant artifacts in most scenes, for example the person in Scene 4 and the trees in Scene 5 cannot be recovered at all. Similarly LSS produces large regions of invalid disparities leading to inconsistent results. While the results for Scene 1 are still acceptable, many of the details and the structure in the other scenes are lost.

Both the DAISY and HOG matching costs successfully recover the basic structure of all scenes. DAISY appears to preserve shape details slightly better and results in less blurring of disparity discontinuities. However, several small speckles of invalid disparities are present in the results (e.g. Scene 6). HOG gives slightly more stable and smooth disparity maps but leads to a stronger blurring of object borders (i.e. object fattening). This effect can be seen well for the persons depicted in Scenes 2 to 4.

Apart from Scene 5 all results of DAISY and HOG in combination with SGM optimization appear to be largely in accordance with the results from the optical stereo images. In Scene 5 the difficulties of cross-spectral stereo correspondence are highlighted when considering the input images. The rightmost tree can be seen clearly in

the optical input image but even for a human observer is hardly distinguishable from the background in the thermal image. As a result no method is able to successfully recover the disparity values for this problematic region.

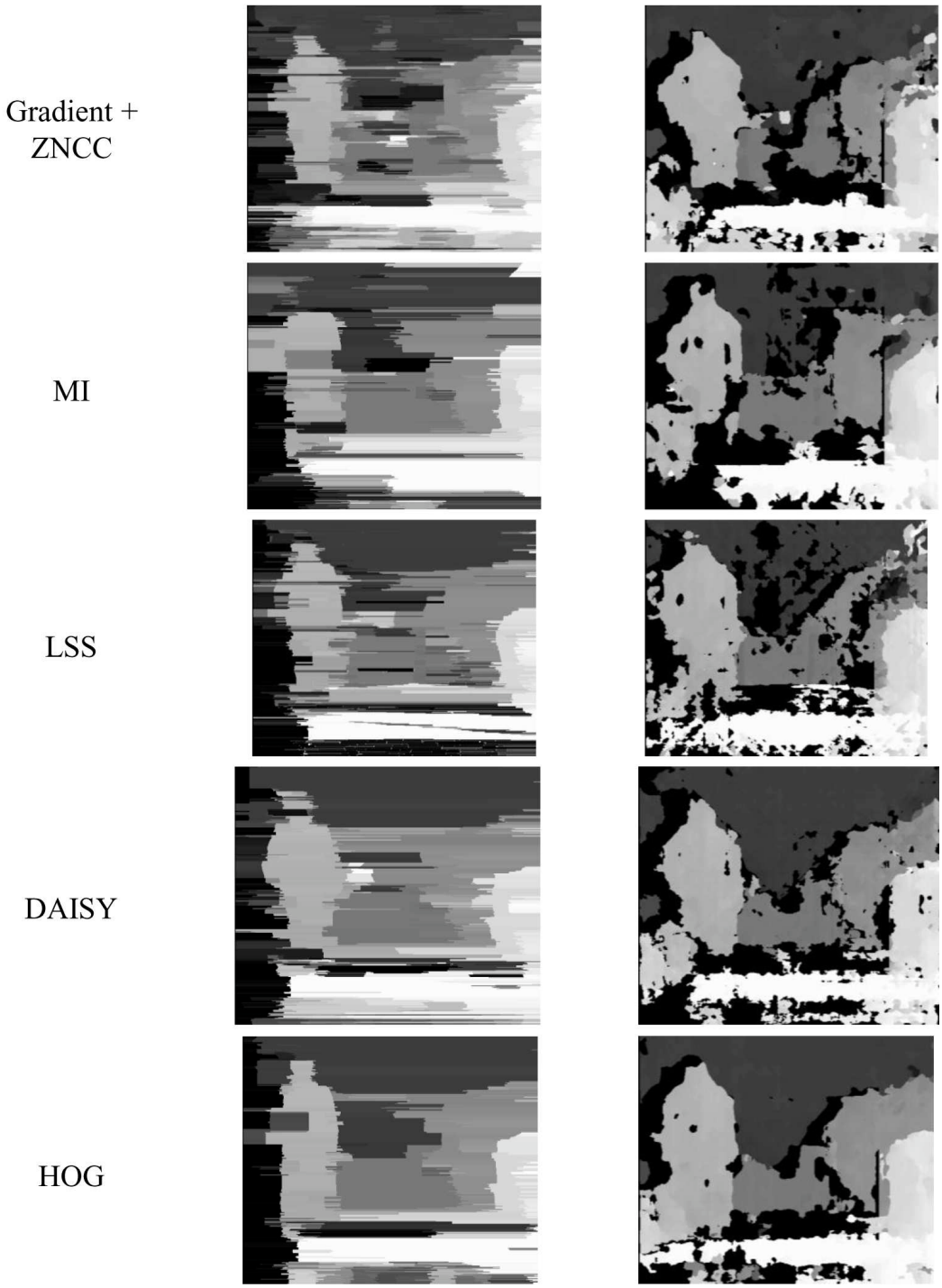

Figure 4.21: Results for Scene 1

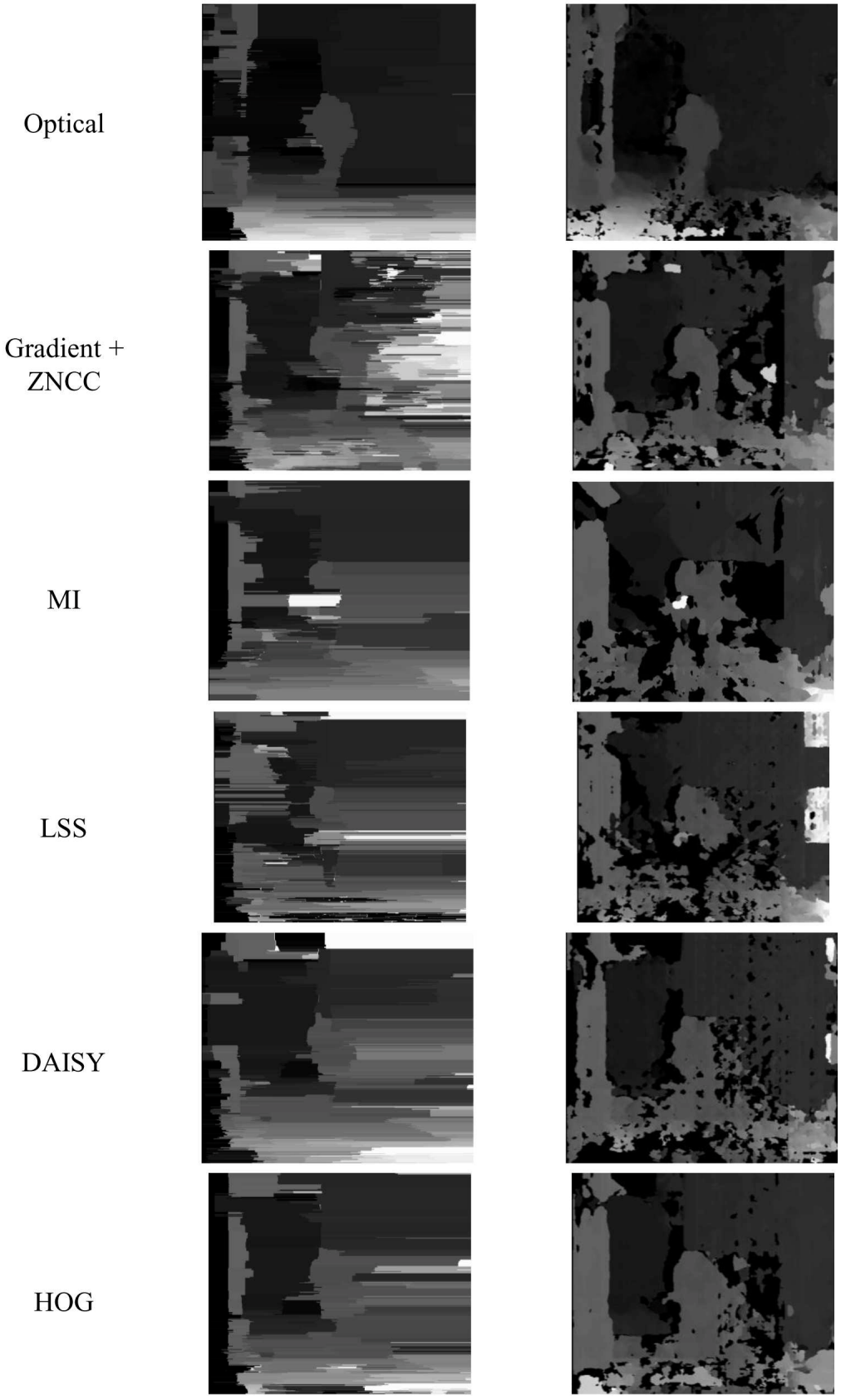

Figure 4.22: Results for Scene 2

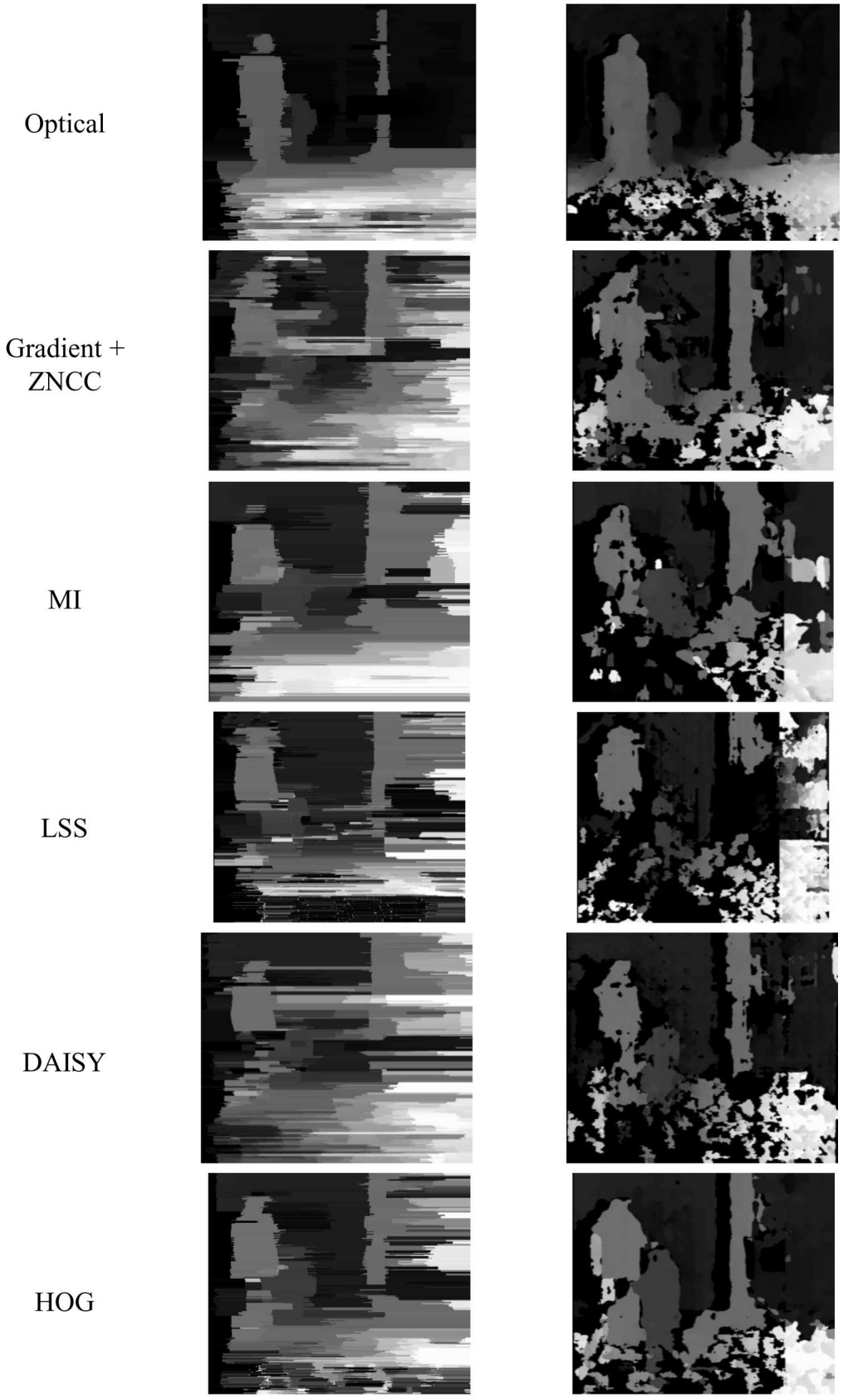

Figure 4.23: Results for Scene 3

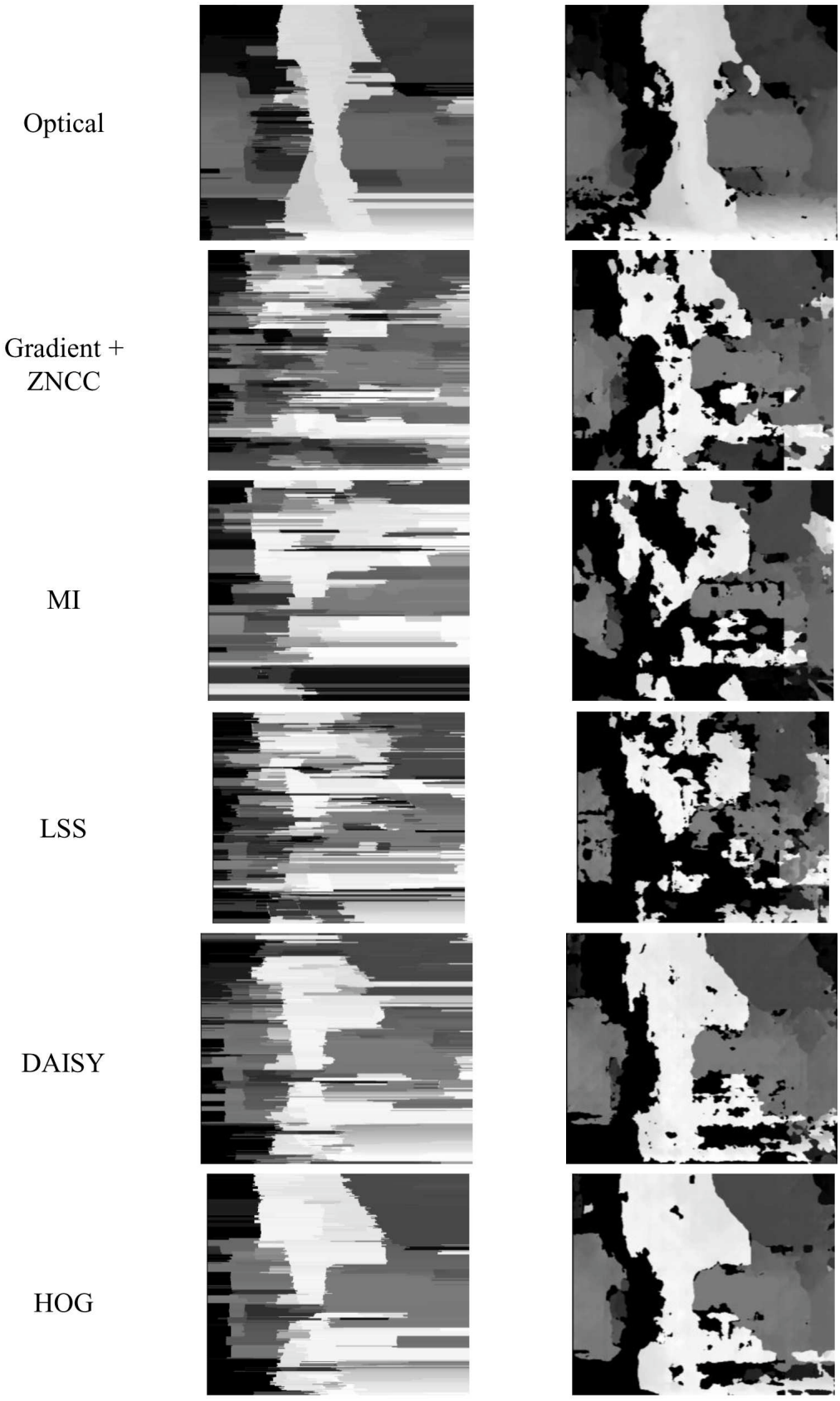

Figure 4.24: Results for Scene 4

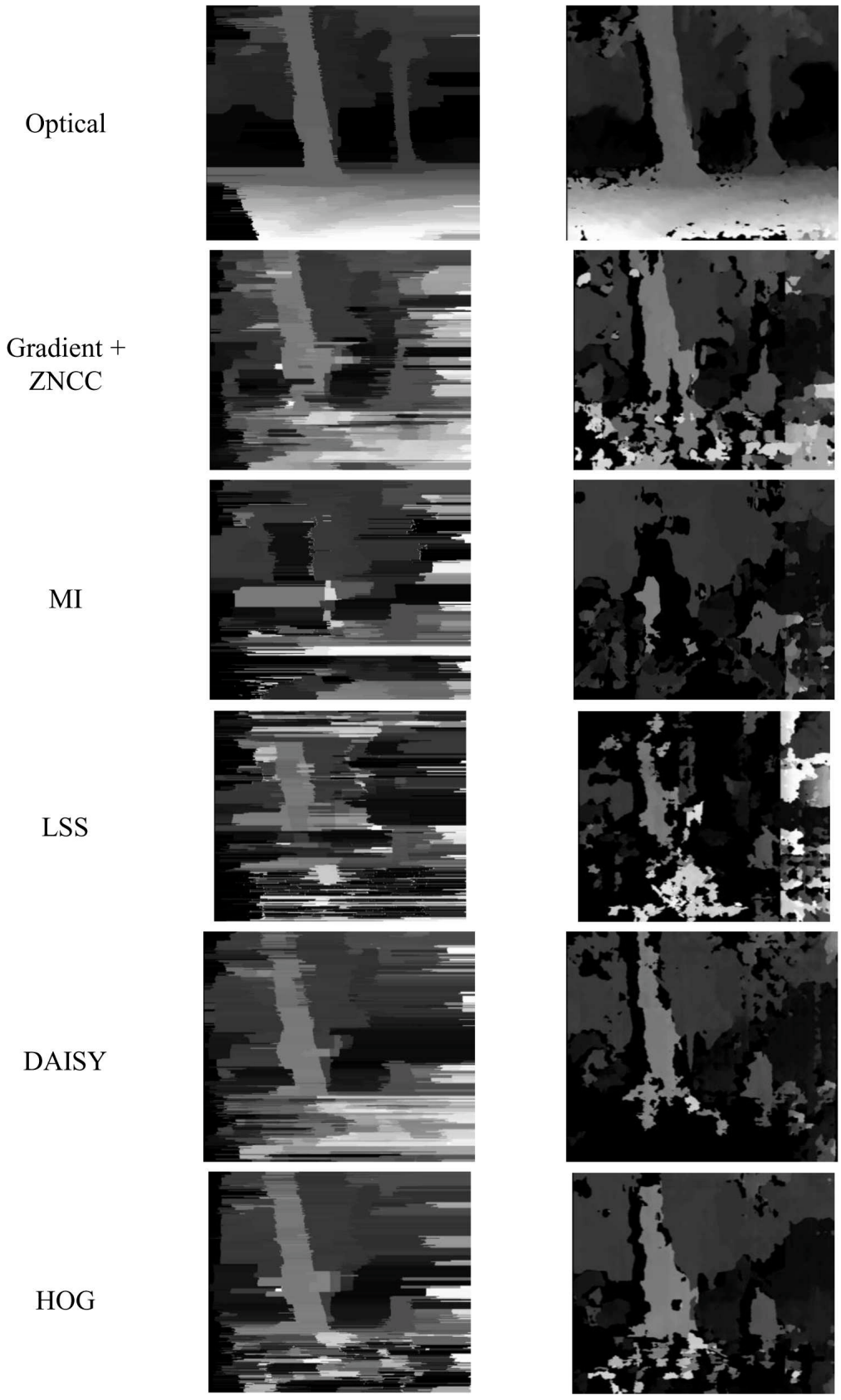

Figure 4.25: Results for Scene 5

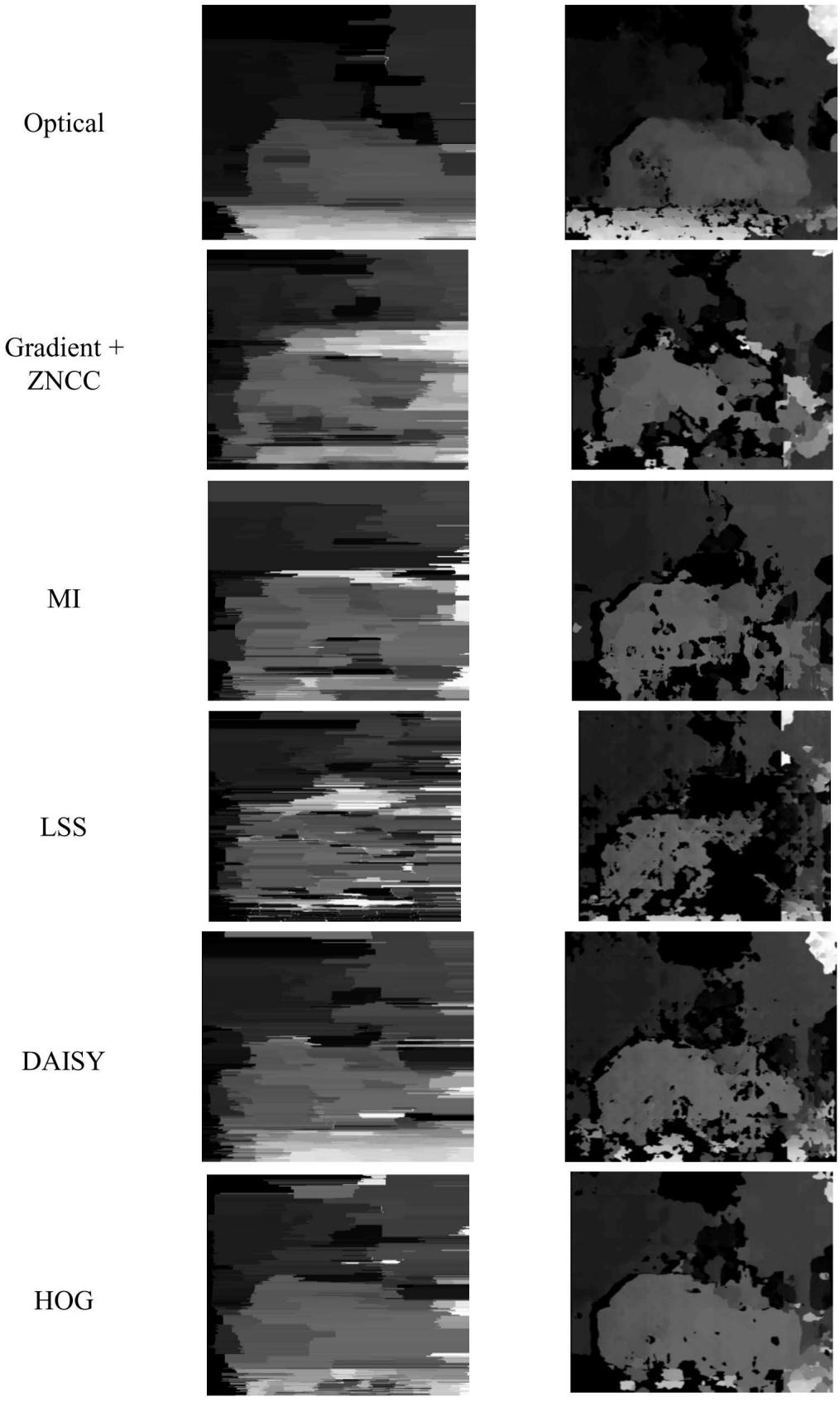

Figure 4.26: Results for Scene 6

Video Sequences and Temporal Consistency

In our experiments on still cross-spectral images of various different scenes we have found that HOG and DAISY matching costs with SGM optimization produce better and more consistent results than all other implemented methods. To now further investigate the properties of these two methods we also test them on a number of cross-spectral video sequences. In this way we examine the temporal consistency of the results and if the disparity maps change in accordance with a moving scene. Figures 4.27-4.30 each show six sample frames extracted from four different cross-spectral video sequences and the respective disparity results using HOG and DAISY matching costs in combination with SGM.
In Video 1 (Figure 4.27) the robot is moving slightly and a car passes by. The shown frames are extracted in intervals of one second. It can be seen that for both HOG and DAISY the car and its changing distance are recovered well. In the video a flickering of the disparity values appears between some consecutive frames but the overall depth of the car is largely consistent as can be seen in the shown sample frames. The results for HOG appear more stable and smoother, better representing the true shape of the car. Also the trees in the background are picked up by the disparity computation for both methods.
Figure 4.28 shows frames extracted from Video 2 also with an interval of one second. Here the robot is stationary and a person walks through the scene. The rough shape and the distance of the person is identified well and is consistent with the changing scene for both HOG and DAISY. The cars in the background are also recovered well. The disparity maps for HOG appear smoother with a more compact representation of the person's body.
Video 3 (Figure 4.29) shows frames with an interval of 0.25 seconds displaying a person walking away from the stationary robot. In this sequence DAISY performs slightly better than HOG, more accurately representing the shape of the body and producing less invalid disparities on the background wall.
Video 4 (Figure 4.30) represents a relatively complex scene in 1.66 second intervals where the robot moves over a bumpy surface and people are walking by. The main objects of the scene can be distinguished roughly by both HOG and DAISY and their disparity fits with the structure of the scene. However, in row four of Figure 4.30 an 'outlier-frame' can be seen where the consistency is temporarily lost but is recovered again in the subsequent frames. Here again HOG appears to produce more stable disparity maps with more compact objects than DAISY.

Considering the results obtained from the video sequences a possible area of future investigation would be the integration of temporal information into the disparity computation. The integration of disparity values from adjacent frames might be exploited to stabilize the computed results (e.g. see [39]).

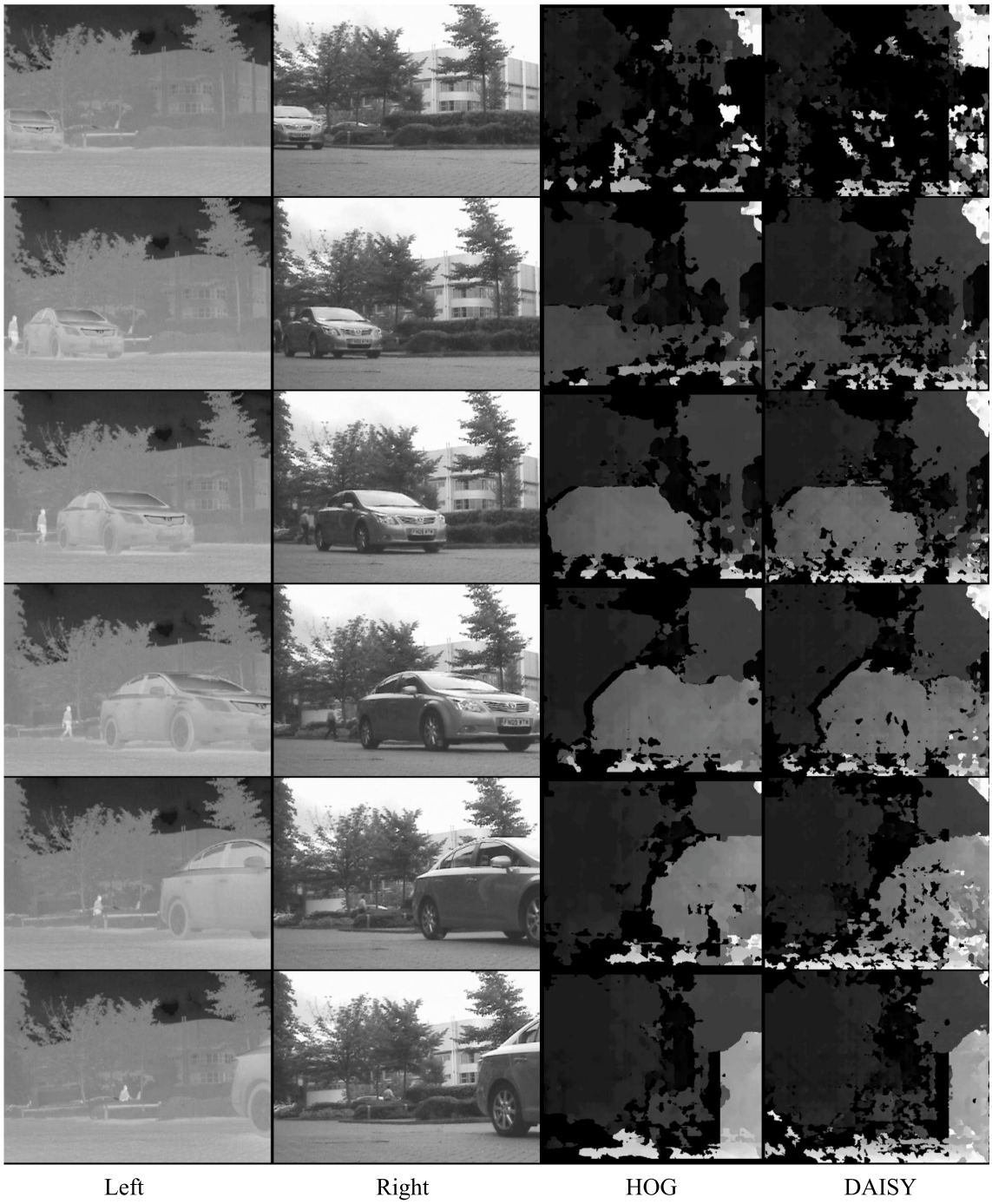

Figure 4.27: Result frames from Video 1 using HOG and DAISY matching costs with SGM.

Figure 4.28: Result frames from Video 2 using HOG and DAISY matching costs with SGM.

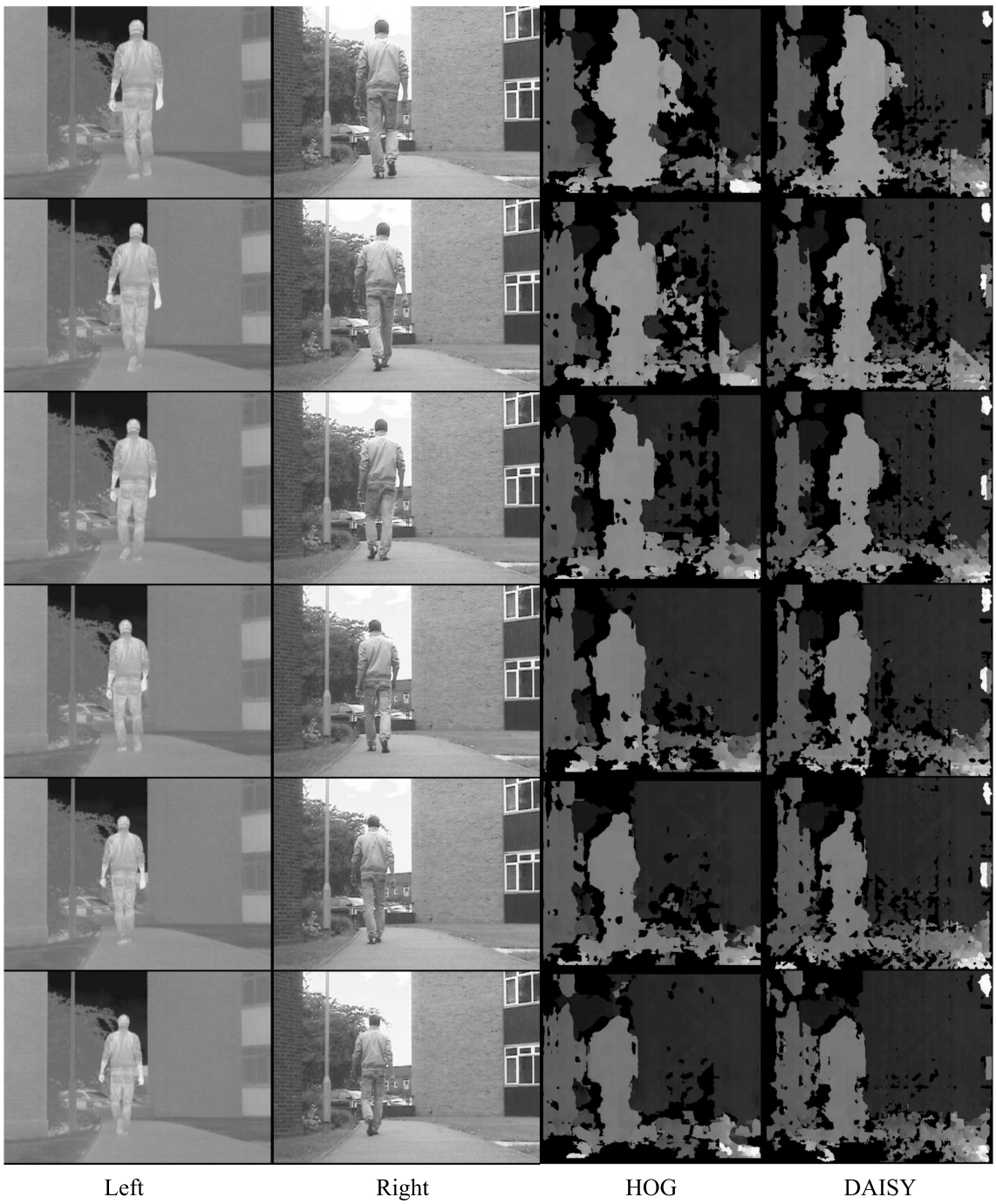

Figure 4.29: Result frames from Video 3 using HOG and DAISY matching costs with SGM.

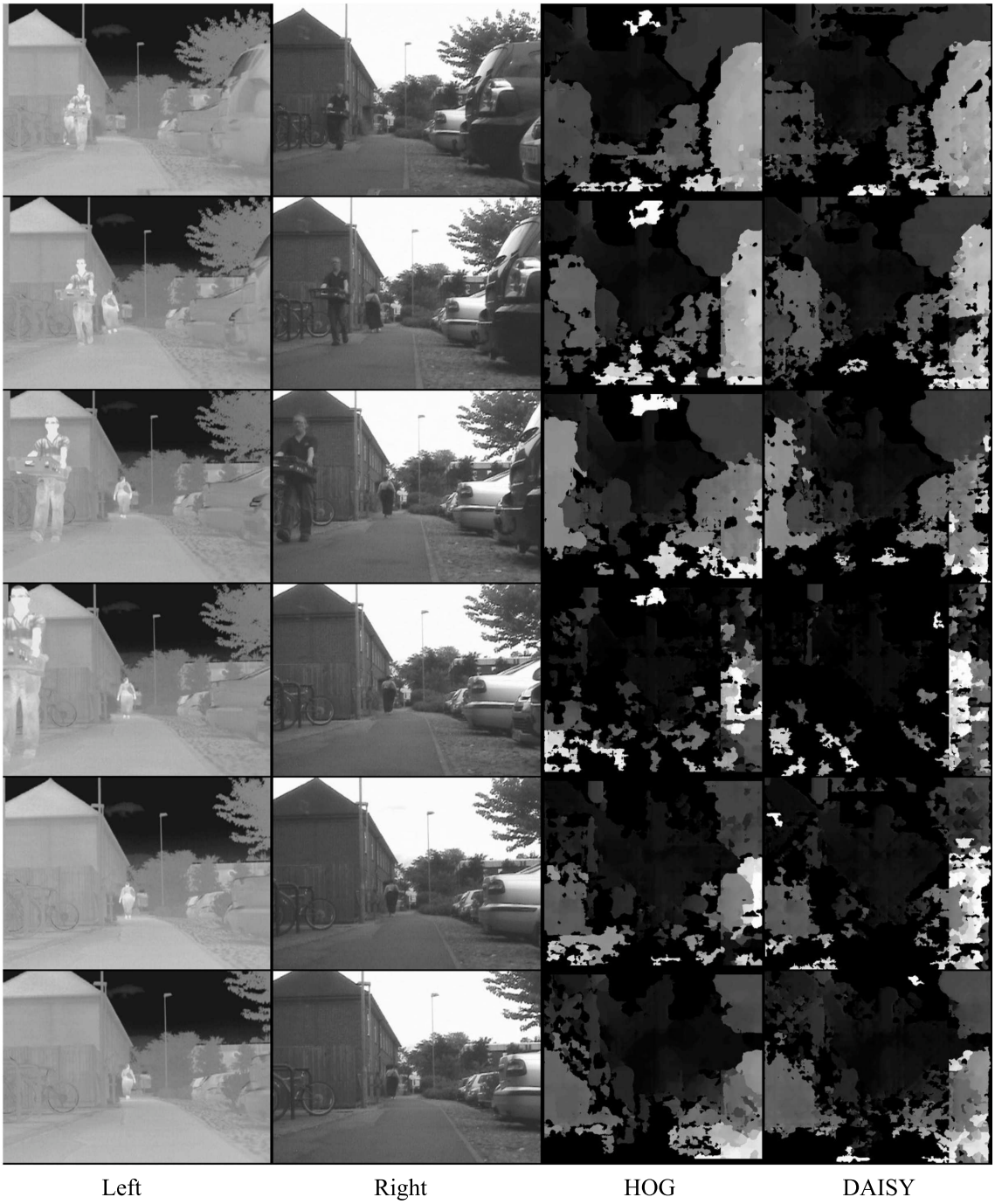

Figure 4.30: Result frames from Video 4 using HOG and DAISY matching costs with SGM.

Performance Comparison Summary

Overall it can be said that the implemented approach of combining robust local matching costs with disparity optimization methods can indeed achieve valid but coarse dense disparity estimates from cross-spectral stereo. From our experiments and a qualitative evaluation we find that the investigated matching cost methods differ noticeably in their ability to produce valid disparity estimates and in the stability and consistency of the results over different scenes. The approach of using dense descriptors based on histograms of unsigned oriented gradients (HOG/DAISY) as a matching cost in combination with SGM optimization gives best and most consistent disparity results on our cross-spectral data. Tests on different video sequences show that on many scenes these matching costs give good results which are largely consistent with the rough structure and depth of the scenes changing in time. Artifacts most often occur in regions which are hard to match such as the ground or the sky. In many cases the novel use of HOG as a stereo matching cost results in more stable disparity maps and more compact (i.e. blob-like) representations of objects. However, in some scenes DAISY yields less artifacts and preserves object boundaries in more detail and with less blurring. For an accurate comparison of the performances of HOG and DAISY matching costs a quantitative comparison based on ground-truth data would be necessary. In this context also the effects of exhaustive tuning of descriptor and disparity optimization parameters and post-processing steps have to be considered. Finally an important aspect is the intended area of application and the respective requirements (more compact and stable objects versus less boundary blurring).

4.3.3 Runtime

In the previous section we have presented comparative results on the performance of the different matching cost methods. We will now shortly discuss the runtime of our implementation of the different matching costs used in the experiments. In general it has to be noted that we did not optimize our implementation regarding runtime but focused on the tuning for best disparity results. As a consequence the following discussion represent only a coarse guideline for comparing the runtimes of the different methods.

The window-based MI computation is the slowest of our matching cost implementations since 2D histograms for each window pair have to be computed for the estimation of the joint probability distribution. This high computational complexity is also reported by Egnal [15] and he suggests a possible speed-up by memorization during the computation of probabilities. However, we did not investigate this proposition and thus our MI computation based on Egnal's [15] and Fookes' [18] descriptions takes approximately six minutes on our test laptop (2.4GHz Intel Core i5 CPU) for the considered parameters (640x480 pixels with 96 disparity levels and 21x21 windows).

For the ZNCC computation we use a basic window-based approach without optimization as described in [28] leading to a matching cost computation runtime of

approximately two minutes on the test data. However, this could be reduced significantly and made independent of the matching window size by fast correlation techniques as in [16].
The computation of dense LSS descriptors for every image point in our implementation takes approximately 30 seconds per image. Here the main workload lies in the computation of the correlation surfaces where each small image patch has to be compared to its surrounding region in a sliding window approach. This could also be accelerated by fast SSD computation techniques. The computation of the implemented dense DAISY and HOG descriptors for each image takes roughly 0.5 and 0.8 seconds respectively. For these methods the most time in our implementation is spent on the distance calculation when creating the Disparity Space Image (DSI), taking approximately 5 seconds. However, the simple operations in this step can be optimized and could also be massively parallelized.
A naive implementation of the left-right cross-check shown with the SGM optimization doubles the overall runtime as two disparity maps have to be computed. This can be avoided by reusing the stored matching costs in the DSI (e.g. see [40]). In combination with SGM optimization Hirschmueller [26][27] also proposes an approximated cross-check to avoid repeated optimization computations.

4.4 Summary

In this chapter we have described our tests of the implemented stereo correspondence framework using different types of test data and discussed the results of our experiments. Before an evaluation can be performed a fundamental prerequisite is the acquisition of real cross-spectral stereo data and we have given details on our utilized methods for this task in Section 4.1.2. Subsequently we have presented results on simulated cross-spectral stereo data in Section 4.2, showing the failure of standard robust matching cost methods. Our implementation of more advanced robust matching cost methods MI, LSS, DAISY and HOG as well as ZNCC after gradient computation proved to be able to deal with the simulated cross-spectral data and produced largely valid results. A comparison with results on real cross-spectral data confirmed that this scenario presents a more difficult problem regarding the computation of dense disparity maps and requires new robust approaches. We therefore combined the robust matching costs with different cost aggregation and disparity optimization techniques. In Section 4.3.2 we have discussed the results of our evaluation of these approaches on various cross-spectral test scenes. We have shown that the use of descriptors based on histograms of unsigned oriented gradients (HOG/DAISY) as a matching cost in combination with SGM optimization allows for a valid dense disparity estimation on different scenes, outperforming the other investigated methods in both quality and consistency of results. This was further confirmed by examining the performance of HOG and DAISY on cross-spectral video sequences. Both methods yield very similar results with HOG appearing more stable on a subjective level considering all performed tests. However, for a more accurate distinction a quantitative evaluation considering application relevant test

data is essential.
We have also given an overview of the runtime requirements of our implementation in Section 4.3.3. Both the dense HOG and DAISY descriptors in combination with SGM optimization can be computed efficiently although the implementation is not capable of real-time operation.

Chapter 5

Summary

In this thesis we have studied the problem of recovering dense depth information from cross-spectral (i.e. optical and thermal) input images using computational stereo techniques. Solving the stereo correspondence problem for cross-spectral images is significantly more difficult than for standard optical images due to the extreme differences in image characteristics. The arguably most popular method in literature for computing correspondences in images with extreme radiometric differences is Mutual Information (MI). Window-based MI approaches as well as pixel-based approaches in combination with global optimization methods have been shown to produce very good results on simulated cross-spectral stereo images, i.e. optical image pairs with complex intensity transformations [18][26][30]. However, it has been shown in [33] that state-of-the-art pixel-based MI methods fail at real cross-spectral images due to the lack of correlation between global intensity values. As a result other approaches have simplified the problem definition by using window-based MI and only computing disparity values for certain predetermined salient regions of interest in cross-spectral stereo images (e.g. people) [33]. The same concept has also been shown to work well by replacing MI with Local Self-Similarities (LSS) [55] as a matching cost.
In view of the fact that currently no solution appears to exist for the given problem we investigated in this work if the computation of dense disparity values for whole images is possible. We have implemented a general framework for dense stereo correspondence computation to be able to test and evaluate the suitability and performance of numerous existing and novel approaches on both simulated and real cross-spectral data. We summarize our obtained findings in the following.

5.1 Conclusions

Based on experiments on simulated cross-spectral stereo images it becomes clear that common preprocessing methods and robust matching cost measures (ZSAD, ZNCC, rank, census) designed for standard radiometric differences in optical stereo

images fail in the presence of extreme image intensity differences. Notably only the raw computation of image gradient magnitudes as a preprocessing step allows for valid results with standard robust matching costs like ZNCC. Also window-based MI performs well which is in accordance with the results reported in literature. The further matching cost approaches for the novel use in dense cross-spectral stereo correspondence, LSS, unsigned DAISY and HOG descriptors also perform well in this scenario. The selection of these methods is based on a visual analysis of commonalities between cross-spectral images. While intensity values obviously differ strongly, local shape features which are visible in both modalities usually correspond well and therefore motivate our choice. LSS descriptors encode local correlation-based self-similarities while DAISY and HOG describe local shape based on statistical image gradient representations. The choice of DAISY and HOG is also supported by the observed good performance of ZNCC on simple image gradient magnitudes.

For our tests on real cross-spectral images we collect data using a custom cross-spectral stereo rig mounted on a mobile robot. The tests on cross-spectral data show that this case presents a more challenging problem than simulated cross-spectral stereo due to the naturally more complex differences in the input images. The increased amount of bland regions in thermal images and the difference in appearance of features result in weaker and more ambiguous matching cost values. Here a simple local WTA disparity computation approach cannot produce valid disparity maps, independent of the considered matching cost method. We therefore combine the robust local matching cost measures with disparity optimization techniques and achieve a significant improvement in performance. In a qualitative evaluation Semi-Global Matching (SGM) turns out to give very good results over all matching costs methods. Graph Cuts (GC) achieves similar results but depends heavily on parameter tuning and results in significantly longer runtimes. Scanline Optimization (SO) and Dynamic Programming (DP) lead to the characteristic horizontal streaking artifacts due to the lack of additional vertical constraints which prove to be particularly important in this case of very ambiguous matching cost values.
We perform a test series on different cross-spectral scenes and qualitatively evaluate the disparity maps computed from each matching cost method in combination with DP and SGM optimization. For the evaluation we also take reference results acquired with a separate standard optical stereo rig into account. In our experiments dense HOG and DAISY descriptors as matching costs outperform the other methods in terms of both validity of disparity values as well as consistency over different scenes. Further tests on cross-spectral video sequences confirm these results. HOG descriptors tend to produce more stable disparity maps with more compact, blob-like objects while DAISY results in less blurring of object boundaries. However, due to the similar results of HOG and DAISY descriptors the intended area of application (e.g. obstacle avoidance, object tracking) as well as additional parameter tuning and post-processing would have to be taken into account to select and optimize the more appropriate method.

In summary it can be said that through extensive experiments and investigations of new approaches we could achieve a coarse estimation of dense disparity maps from cross-spectral stereo. In all considered tests the combination of dense des-

criptors based on histograms of unsigned gradient orientations with SGM disparity optimization outperforms methods using MI or LSS matching costs which have been suggested in previous work for use in such a scenario. Notably the novel use of dense HOG descriptors for stereo correspondence leads to very good results. Although the implemented methods of HOG and DAISY matching are currently not capable of real-time operation they are based on computationally efficient descriptors and offer several possibilities for runtime optimization.

5.2 Further Work

A recommendable next step would be a further testing of the implemented approaches on additional cross-spectral data taken in application-relevant environments. The acquisition of ground truth data would be essential for an accurate quantitative evaluation of results and would also be necessary for a better tuning of the parameters of all methods (i.e. enabling an automatic exhaustive parameter sweep). In combination with this an improved cross-spectral camera calibration procedure would be desirable to allow for a more accurate calibration and a simpler and faster data acquisition. A fully automatic method possibly including an active infrared calibration target would of course be an extremely useful tool.

Aside from possible performance improvements regarding the quality of the disparity estimation, an interesting aspect would be the possible reduction of runtime through optimization techniques. Investigations on a possible speed-up through specialized hardware could also be considered.

Ultimately the integration of the proposed approach of using HOG or DAISY in combination with SGM into a real test environment would be needed to judge its usefulness in real-world applications. For example it would be interesting to see if the disparity results can actually be used to improve the performance of obstacle avoidance for mobile robots or object detection and tracking applications. However, it has to be stated that based on the current results a possible practical use in critical scenarios which require high reliability and speed (e.g. automotive, military) seems unlikely.

Appendix A

i2iReader Near-Infrared-Optical Stereo

In this supplementary chapter we perform and discuss additional experiments on data taken at the Institute for Computer Vision and Graphics at Graz University of Technology. In contrast to the main part of this thesis which focuses on far-infrared-optical cross-spectral stereo, this data represents the near-infrared-optical stereo case. We apply the implemented stereo framework and the insights gained in the main part of the thesis to analyze the properties of near-infrared-optical stereo data and the differences to far-infrared-optical stereo.
The considered stereo test data is captured using an EFKON i2iReader camera, courtesy of the Institute for Computer Vision and Graphics. This hardware has been developed within the Austrian FFG project 'MobiTrick' (8258408) under the FIT-IT program and is designed for license plate recognition in traffic enforcement applications. Although the camera is not originally designed for stereo applications, the inherent stereo setup allows for an estimation of depth information. The MobiTrick project could make use of the computed depth information and therefore we investigate which of the implemented stereo methods could be suitable for such a task.

A.1 i2iReader Test Data

The i2iReader houses one Near-Infrared (NIR) and one optical camera in an integrated unit. The relative positions of the two cameras are fixed with a relatively short baseline of approximately 4cm, resulting in a low disparity resolution and a respective uncertainty of results with increasing depth. The cameras are not aligned but rotated towards each other such that the optical axes intersect in front of the objects of typically considered scenes. This results in an inverted disparity between captured images, i.e. scene points with a larger distance to the cameras yield larger disparity values and vice versa. For the rectification of the input images already

available calibration data is used.
To investigate the suitability of different stereo methods for i2iReader NIR-optical stereo matching we capture images of different indoor test scenes (Figure A.1). The considered rectified images are of size 535x290 pixels with a maximum disparity of 16 pixels.

Figure A.1: i2iReader test scenes.

In a visual analysis of the test scenes it becomes clear that the near-infrared (NIR) and optical images have relatively similar image characteristics. This is in contrast to both the simulated (Section 4.2) and real (Section 4.3) far-infrared-optical stereo cases, where the differences between the images are much more extreme. In the NIR images some details and textures are less distinct than in the optical images (e.g. the tiger and the poster on the left in Scene 1) and in some areas contrast is poorer. However, other parts of the scenes appear sharper and in more detail than in the optical images and effects of overexposure and saturation are reduced (e.g. books on the right in Scene 3). Contrast inversion effects between the NIR and the optical images appear only very limited, one example can be seen well on the shirt of the person in Scene 3. In addition to these properties, bland and textureless regions in the test images present challenges for stereo matching.

A.2 Methods and Parameters

Considering the image characteristics observed on the i2iReader test data we investigate which of the implemented robust matching cost methods (Section 3.1) are able to produce valid disparity maps in this scenario. In the following we discuss experiments using the Zero mean Sum of Absolute Differences (ZSAD), Zero mean Normalized Cross Correlation (ZNCC), census, rank, window-based Mutual Information (MI), Local Self-Similarity descriptors (LSS), DAISY descriptors and HOG descriptors. We also include LoG filtering as a preprocessing step in combination with Sum of Absolute Differences (SAD) since it gave slightly better results than the other implemented preprocessing methods (Section 2.2.1) in our experiments.
For ZSAD, ZNCC and rank we use window sizes of 11x11 pixels, for census we use 7x7 pixels and for MI 15x15 pixels. Prior probabilities with a weight of 30% ($\lambda = 0.7$) are included in the MI computation. For the computation of LSS descriptors we use patches of 5x5 pixels, surrounding regions of 35x35 pixels and a log-polar grid with 4 radial and 12 angular bins. In our experiments with DAISY and HOG descriptors the use of signed gradients ($0° - 360°$) turned out to give better results than the use of unsigned gradients ($0° - 180°$). A reason for this is the limited amount of contrast inversion in the NIR images with the consequence that resulting errors due to signed gradient orientations are being outweighed in the overall scene by the increased discriminative power of the descriptors. We therefore show results of signed gradient orientations for HOG descriptors of 18x18 pixels split into 3x3 cells with 9 orientation intervals and DAISY descriptors with parameters $R = 5$, $Q = 3$, $T = 4$ and $H = 8$.
In a first step we perform tests with Winner-Takes-All (WTA) disparity computation to assess the performance of the different matching cost methods independently of more advanced disparity optimization methods. We combine the WTA computation with cost aggregation using a 11x11 box filter. In the seconds step we apply Semi-Global Matching (SGM) to compute the disparities from the matching costs due to the good performance of this method shown in Chapter 4. In both steps we apply left-right cross-checking and speckle removal, pixels labeled as invalid are set to zero in the result images.

A.3 Performance Comparison

Figures A.2-A.5 show the disparity maps computed in our test run. As mentioned in Section A.1 the disparity values are inverted, i.e. objects closer to the camera are displayed darker and vice versa. In the absence of ground truth data we attempt a qualitative comparison based on the computed disparity maps. However, due to the limited test data and the partly similar results a reliable analysis is hard and we will therefore only provide a rough interpretation of the available results.
First of all it can be observed that in contrast to the far-infrared-optical stereo case (Section 4) here also robust standard matching cost methods like census or ZNCC can produce valid results. Furthermore it can be seen clearly that, considering the

results for all scenes, SGM leads to significantly better and more stable results than WTA. The bland image regions combined with the observed image characteristics and quality cause simple WTA to produce large regions of inconsistent or invalid disparities.
Overall it appears that ZNCC, census, DAISY and HOG perform slightly better than the other considered matching costs. SAD with prior LoG filtering and ZSAD produce particularly bad results for Scene 3 and MI struggles with the bland regions in Scenes 2 and 3. The performance of LSS is consistent but produces relatively heavy blurring. Details and object borders in the scenes are preserved best by census and rank (e.g. tiger in Scene 1) but rank introduces noticeable disparity artifacts in Scene 2. ZNCC, DAISY and HOG produce stable and smooth disparity results over all scenes but introduce more blurring than census.

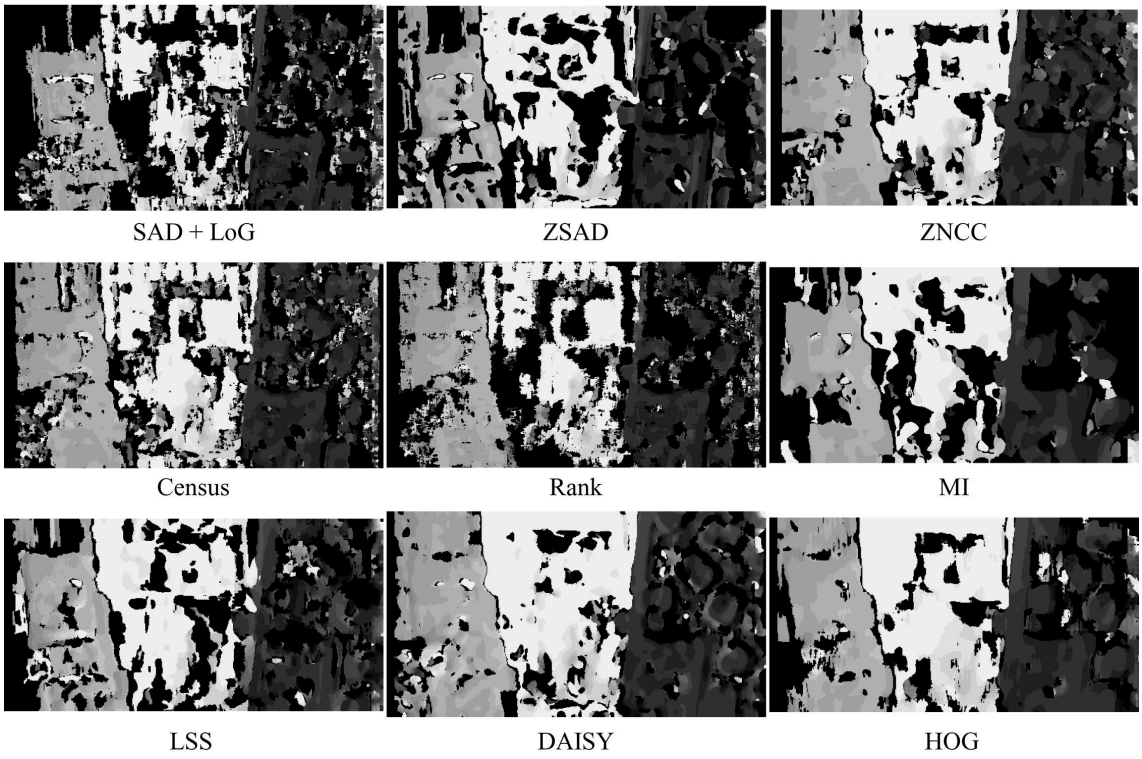

Figure A.2: Results for Scene 1 (WTA)

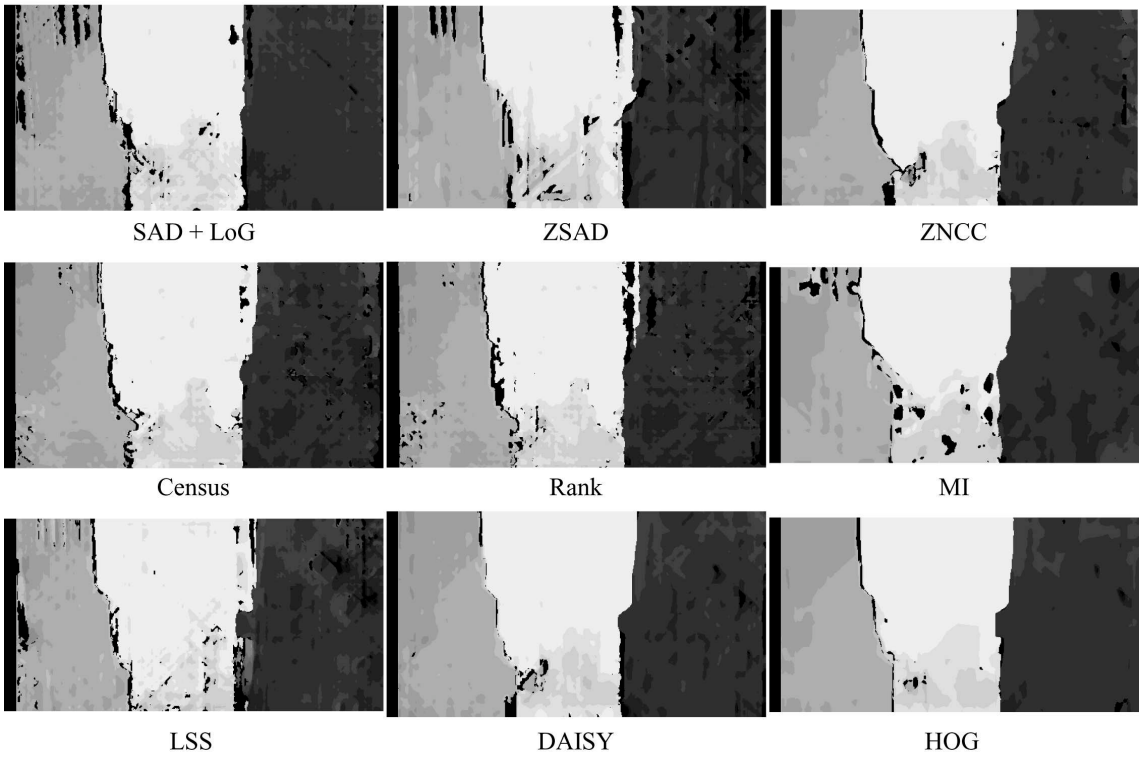

Figure A.3: Results for Scene 1 (SGM)

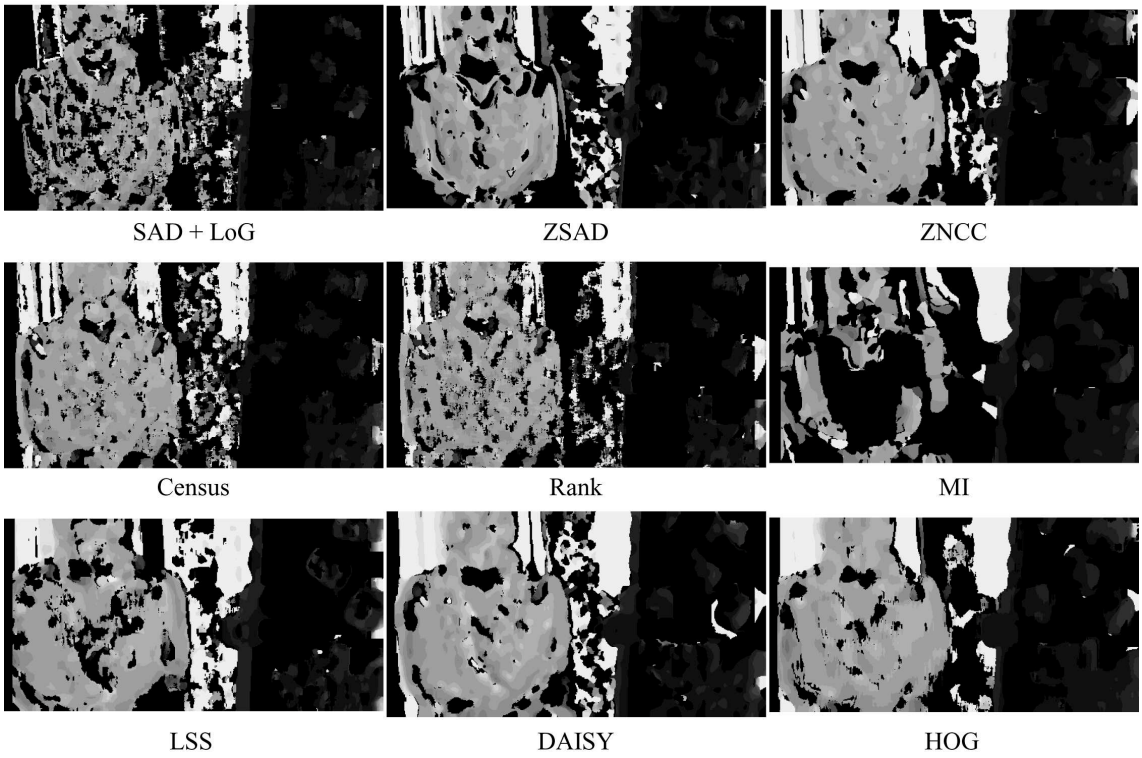

Figure A.4: Results for Scene 2 (WTA)

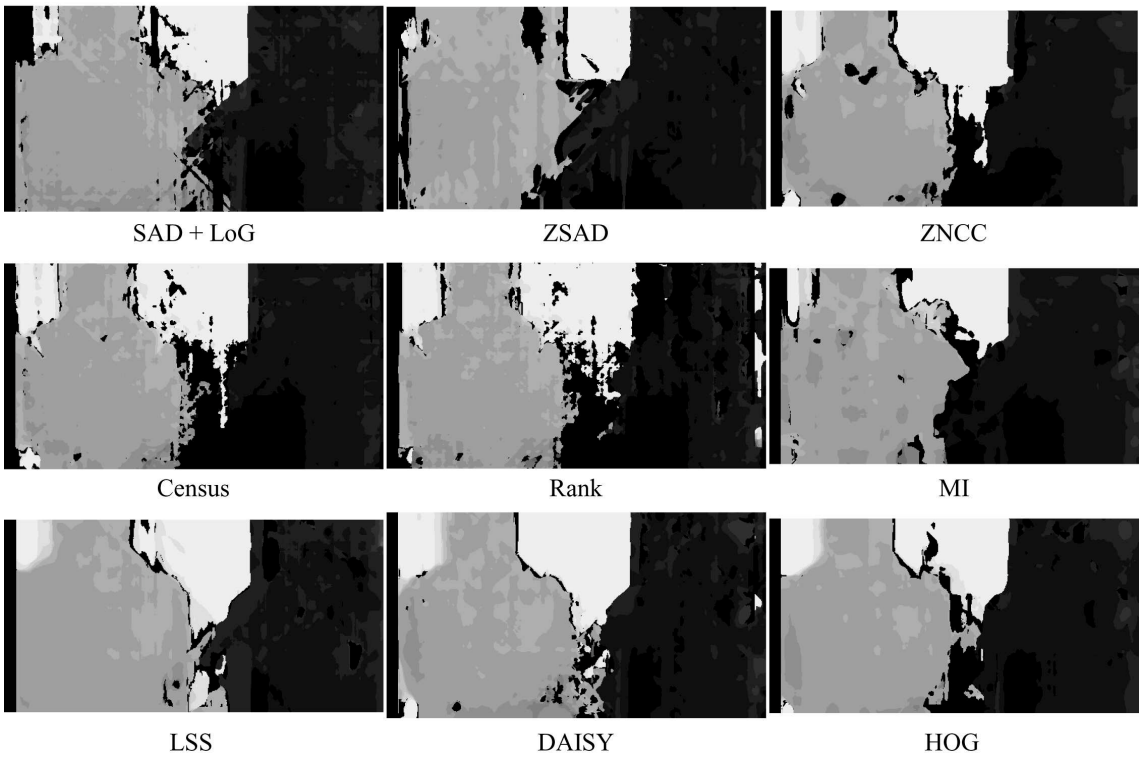

Figure A.5: Results for Scene 2 (SGM)

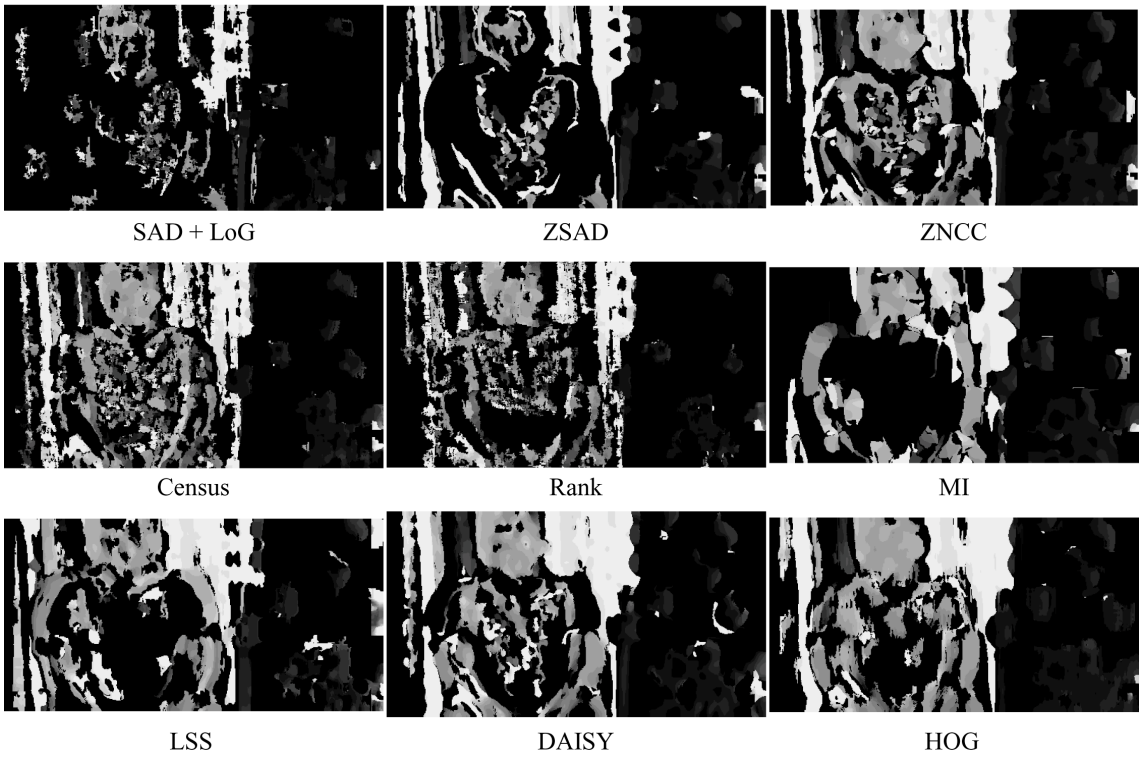

Figure A.6: Results for Scene 3 (WTA)

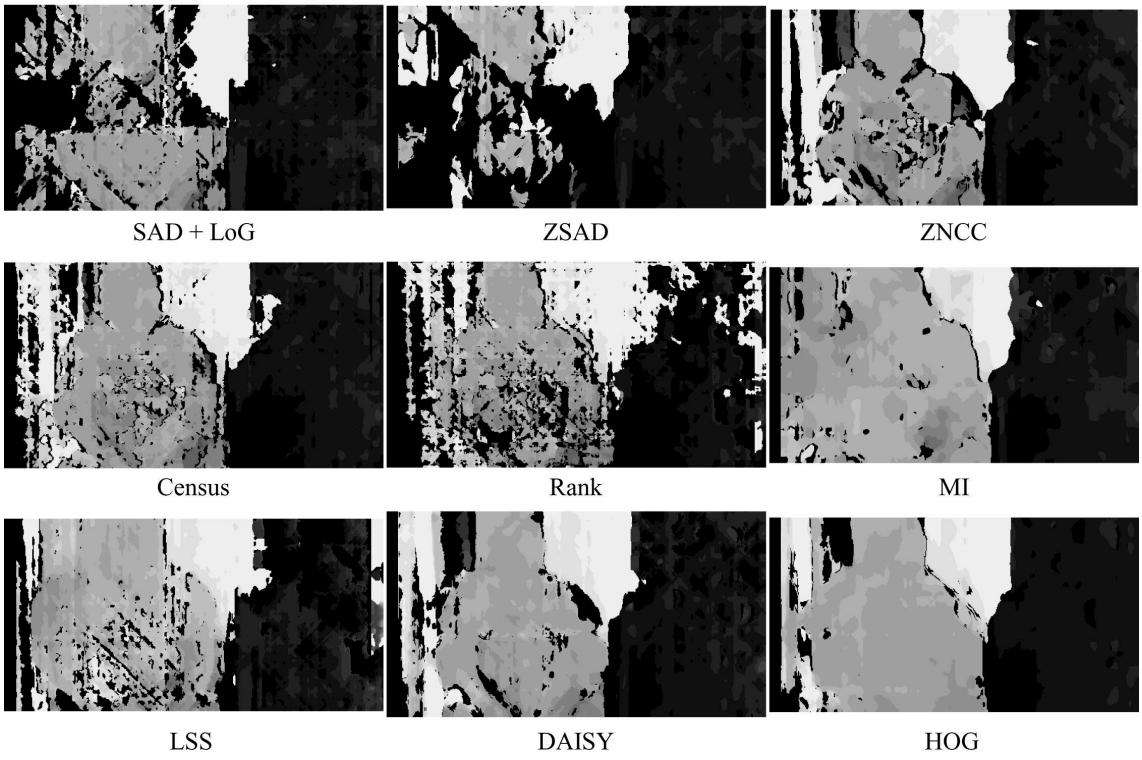

Figure A.7: Results for Scene 3 (SGM)

A.4 Practical Application Considerations

With regard to a possible application in the MobiTrick project also implementation details of the stereo algorithms would have to be considered. This is due to the fact that eventually the stereo matching should be performed on an embedded system as part of the integrated i2iReader camera unit. While DAISY and HOG descriptors can be computed relatively efficiently, correlation-based standard robust matching-costs like ZNCC and non-parametric costs like census are already commonly applied in such environments and have been implemented to operate in real-time (e.g. [16]). In combination with fast census matching variations of SGM which are suitable for implementation in embedded systems with general purpose processors have been proposed recently [21][29].
It has to be noted that additional test data would be necessary for a more reliable assessment of the performance of the different matching costs in this specific scenario. Data taken from the actual area of application (i.e. traffic enforcement) would be important in this context.

A.5 Summary

In this chapter we have described additional experiments on near-infrared-optical stereo data taken by the EFKON i2iReader cameras of the Institute for Computer Graphics and Vision. Even though the current camera setup is not optimized for stereo applications, depth estimates can be achieved and might be exploited in the MobiTrick project.
Tests on indoor scenes show that in contrast to far-infared-optical stereo in the near-infared-optical stereo case also standard robust matching costs can produce valid disparity maps. On the considered test images the use of optimization techniques like SGM shows to be essential for good results.
In a rough qualitative comparison the matching costs ZNCC, census, DAISY and HOG appear to produce the most consistent disparity results. Census preserves object borders best while ZNCC, DAISY and HOG produce smoother disparity maps with stronger blurring.
Regarding an integration into the MobiTrick project, matching costs like census or ZNCC might be preferred due to their known suitability for embedded implementations. Furthermore, suitable variations of SGM have recently been proposed for such applications. For a more thorough assessment of the applicability of the different matching cost methods additional data from the actual area of application would be necessary.

References

[1] F. Arnell. *Vision-based pedestrian detection system for use in smart cars.* Master's thesis, Royal Institute of Technology, Stockholm, Sweden, 2005.

[2] S. T. Barnard and M. A. Fischler. Computational stereo. *ACM Computing Surveys*, 14(4):553–572, 1982.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[4] A.F. Bobick and S.S. Intille. Large occlusion stereo. *International Journal of Computer Vision*, 33(3):181–200, 1999.

[5] C. Bodensteiner, W. Huebner, K. Juengling, J. Mueller, and M. Arens. Local multi-modal image matching based on self-similarity. In *Proceedings IEEE International Conference on Image Processing*, pages 937–940. IEEE, 2010.

[6] J.Y. Bouguet. The Caltech camera calibration toolbox for MATLAB. http://www.vision.caltech.edu/bouguetj/calib_doc, May 2011.

[7] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004.

[8] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[9] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library.* O'Reilly, 1st edition, 2008.

[10] M.Z. Brown, D. Burschka, and G.D. Hager. Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):993–1008, 2003.

[11] S. Colantonio, M. Benvenuti, M.G. Di Bono, G. Pieri, and O. Salvetti. Object tracking in a stereo and infrared vision system. *Infrared Physics & Technology*, 49(3):266–271, 2007.

[12] M. Correa, G. Hermosilla, R. Verschae, and J. Ruiz-del Solar. Human detection and identification by robots using thermal and visual information in domestic environments. *Journal of Intelligent & Robotic Systems (in press)*, ("Online First"):1–21, July 2011.

[13] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 886–893. IEEE, 2005.

[14] S. Denman, T. Lamb, C. Fookes, V. Chandran, and S. Sridharan. Multi-spectral fusion for surveillance systems. *Computers & Electrical Engineering*, 36(4):643–663, 2010.

[15] G. Egnal. Mutual information as a stereo correspondence measure. Technical report, University of Pennsylvania, 2000.

[16] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy. Real time correlation-based stereo: algorithm, implementations and applications. Technical report, INRIA, 1993.

[17] C. Fookes, A. Maeder, S. Sridharan, and J. Cook. Multi-spectral stereo image matching using mutual information. In *Proceedings Second International Symposium on 3D Data Processing, Visualization and Transmission*, pages 961–968. IEEE, 2004.

[18] C. Fookes and S. Sridharan. Investigation & comparison of robust stereo image matching using mutual information & hierarchical prior probabilities. In *Proceedings Second International Conference on Signal Processing and Communication Systems*, pages 1–10. IEEE, 2008.

[19] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6(1):35–49, 1993.

[20] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.

[21] S. K. Gehrig and C. Rabe. Real-time semi-global matching on the CPU. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition - Workshops*, pages 85–92. IEEE, June 2010.

[22] K. Hajebi and J.S. Zelek. Structure from infrared stereo images. In *Proceedings Canadian Conference on Computer and Robot Vision*, pages 105–112. IEEE, 2008.

[23] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2nd edition, 2004.

[24] K. He, J. Sun, and X. Tang. Guided image filtering. In *Proceedings Eleventh European Conference on Computer Vision*, pages 1–14, 2010.

[25] J. Heikkilä and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 1106–1112. IEEE, 1997.

[26] H. Hirschmüller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 807–814. IEEE, 2005.

[27] H. Hirschmüller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[28] H. Hirschmüller and D. Scharstein. Evaluation of stereo matching costs on images with radiometric differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(9):1582–1599, 2009.

[29] M. Humenberger, T. Engelke, and W. Kubinger. A census-based stereo vision algorithm using modified semi-global matching and plane fitting to improve matching quality. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition - Workshops*, pages 77–84. IEEE, June 2010.

[30] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 2, pages 1033–1040. IEEE, 2003.

[31] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[32] S. Krotosky and M. Trivedi. Mutual information based registration of multimodal stereo videos for person tracking. *Computer Vision and Image Understanding*, 106(2-3):270–287, 2007.

[33] S. Krotosky and M. Trivedi. Registering multimodal imagery with occluding objects using mutual information: application to stereo tracking of humans. In R.I. Hammoud, editor, *Augmented Vision Perception in Infrared: Algorithms and Applied Systems*, chapter 14, pages 321–347. Springer, 2009.

[34] R. Laganière. *OpenCV 2 Computer Vision Application Programming Cookbook*. Packt, 1st edition, 2011.

[35] N. Lazaros, G.C. Sirakoulis, and A. Gasteratos. Review of stereo vision algorithms: from software to hardware. *International Journal of Optomechatronics*, 2(4):435–462, 2008.

[36] S.S. Lin. Review: Extending visible band computer vision techniques to infrared band images. Technical report, University of Pennsylvania, 2001.

[37] C. Loop and Z. Zhang. Computing rectifying homographies for stereo vision. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 125–131. IEEE, 1999.

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

[39] S. Morales and R. Klette. Spatio-temporal stereo disparity integration. Technical report, University of Auckland, 2011.

[40] K. Mühlmann, D. Maier, J. Hesser, and R. Männer. Calculating dense disparity maps from color stereo images, an efficient implementation. *International Journal of Computer Vision*, 47(1):79–88, 2002.

[41] F. Porikli. Integral histogram: a fast way to extract histograms in Cartesian spaces. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 829–836. IEEE, 2005.

[42] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 3017–3024. IEEE, 2011.

[43] D. Scharstein and R. Szeliski. The Middlebury stereo vision website. http://vision.middlebury.edu/stereo, August 2011.

[44] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[45] D. Scribner, P. Warren, and J. Schuler. Extending color vision methods to bands beyond the visible. *Machine Vision and Applications*, 11(6):306–312, 2000.

[46] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[47] D.A. Socolinsky. Design and deployment of visible-thermal biometric surveillance systems. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–2. IEEE, 2007.

[48] M. Sonka, V. Hlavac, and R. Boyle. 3D vision, geometry. In *Image Processing, Analysis, and Machine Vision*, chapter 11. Thomson, 3rd edition, March 2007.

[49] M. Sonka, V. Hlavac, and R. Boyle. Use of 3D vision. In *Image Processing, Analysis, and Machine Vision*, chapter 12. Thomson, 3rd edition, March 2007.

[50] R. Szeliski. *Computer Vision: Algorithms and Applications*, volume 5 of *Texts in Computer Science*. Springer, 1st edition, 2011.

[51] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, 2008.

[52] E. Tola, V. Lepetit, and P. Fua. A fast local descriptor for dense matching. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[53] E. Tola, V. Lepetit, and P. Fua. DAISY: an efficient dense descriptor applied to wide-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5):815–830, 2010.

[54] F. Tombari, S. Mattoccia, L. Di Stefano, and E. Addimanda. Classification and evaluation of cost aggregation methods for stereo correspondence. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[55] A. Torabi and G.-A. Bilodeau. Local self-similarity as a dense stereo correspondence measure for thermal-visible video registration. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 61–67, 2011.

[56] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nister. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. In *Proceedings Third International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 798–805. IEEE, 2006.

[57] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *Proceedings Third European Conference on Computer Vision*, volume 2, pages 151–158, 1994.

[58] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–195, 1998.

[59] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Proceedings Seventh IEEE International Conference on Computer Vision*, pages 666–673. IEEE, 1999.

[60] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

[61] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21(11):977–1000, 2003.