Graz University of Technology

Institute for Computer Graphics and Vision

PhD Thesis

Tracking and Visual Quality Inspection in Harsh Environments

Markus Heber

Graz, Austria, January 2013

Thesis supervisors

Prof. Dr. Horst Bischof

Prof. Dr. Margrit Gelautz

To my wonderful wife Lisa.

Always set the trail,
never follow the path!

N.N.

Abstract

This Thesis describes novel image template based methods for robust and real-time capable tracking and visual quality inspection tasks in harsh environments.

The presented tracking methods include three different constitutive approaches. First, a spline-regularized template tracking method applied for weld seam tracking in industrial robotic welding environments is presented. Second, a template tracking method that relies on incremental template blending updates is presented for more dynamic scenes, allowing for template tracking in harsh outdoor environments. Finally, a novel generic tracking fusion framework that allows for combining an arbitrary number of trackers that report diverse typically not directly combinable outputs is presented. In this way, advantages of different tracking cues can be combined in order to solve highly complex tracking problems. The methodological focus of all presented tracking methods lies on robustness and successful handling of diverse disturbing effects, environmental influences, and image noise, while remaining real-time capable.

Considering the task of image based quality assessment we present a visual quality inspection framework for industrial robotic welding tasks, consisting of a semi-supervised method that incrementally generates panorama images of entire welding processes, and of an unsupervised classification approach that automatically detects welding defects in an on-line fashion. Both methods rely on weld seam image templates provided by an underlying tracking approach. The focus here lies on high panorama image quality including a consequent suppression of industrial image noise as well as prevention of typical blending artifacts, and on highly accurate classification results at a

coincidental minimum amount of training costs or off-line preparatory tasks.

All presented methods are extensively evaluated and compared to state of the art methods to prove their functionality and their practical applicability. It is shown that the presented methods for both, tracking and visual quality inspection, clearly outperform the state of the art, while solving hard computer vision problems in harsh environments in real-time.

Keywords: computer vision, template tracking, tracking fusion, panorama image generation, defect detection

Kurzfassung

Diese Dissertation beschreibt sowohl neue Methoden für robuste und echtzeitfähige Verfolgung von Objekten in Videos, als auch neue Ansätze zur bildgestützten Qualitätsbestimmung in extremen Umgebungen.

In Bezug auf Objektverfolgung werden drei unterschiedliche Methoden präsentiert. Die erste Methode basiert auf Spline-regularisierten Korrelationsmessungen, und findet ihre Anwendung in der Verfolgung von Schweißnähten in industriellen Roboterschweißprozessen. Dabei besteht das Ziel darin, die sich ständig verändernde Schweißnaht korrekt zu detektieren und zu verfolgen, um so weitere Analysen zu ermöglichen. Die zweite Methode ist speziell auf hochdynamische Umgebungen ausgelegt, und basiert auf inkrementell überblendeten Objektbildausschnitten. Dies ermöglicht beispielsweise die Objektverfolgung im Freien unter sich ständig ändernden Beleuchtungsbedingungen. Schließlich wird eine generische Methode zur Fusion von unterschiedlichsten Objektverfolgungsalgorithmen präsentiert. Dieser Ansatz ermöglicht es, völlig unterschiedliche und daher nicht direkt vergleichbare oder kombinierbare Messwerte zu vereinen. Auf diese Art und Weise können die Vorteile der verschiedenen Ansätze so kombiniert werden, dass die Qualität und die Genauigkeit der Objektverfolgung sichtbar verbessert werden. Der Fokus bei allen Objektverfolgungsmethoden liegt speziell in Echtzeitfähigkeit, Robustheit und in einer erfolgreichen Bewältigung von unterschiedlichsten Störeinflüssen, umweltbedingten unvorhersehbaren Veränderungen und Bildrauschen.

Für das Problem der bildgestützten Qualitätsanalyse wird ein Framwork zur Bewertung von industriellen Roboterschweißungen, welches einerseits aus einem inkremen-

tellen Panoramabildgenerierungsverfahren, und andererseits aus einem automatischen Fehlererkennungsverfahren besteht, präsentiert. Beide Ansätze basieren auf Schweißnaht-Bildausschnitten, welche von einem robusten Objektverfolgungsalgorithmus ermittelt werden. Bei der bildgestützten Qualitätsanalyse liegt der Fokus auf höchstmöglicher Panoramabildqualität durch konsequente Unterdrückung von Störungen und Bildartefakten, und auf höchstmöglicher Klassifikationsgenauigkeiten in Bezug auf Schweißfehlererkennung bei gleichzeitig minimaler Anzahl an notwendigen Trainingsdaten zur Modellgenerierung.

Sämtliche Methoden wurden ausführlichst evaluiert, getestet und mit aktuellen Standder-Technik Methoden verglichen, um einerseits die Funktionalität, und andererseits auch die Anwendbarkeit in der Praxis zu zeigen. In dieser Hinsicht wird demonstriert, dass sämtliche Methoden sowohl für die Objektverfolgung als auch für die bildgestützte Qualitätsprüfung von industriellen Roboterschweißprozessen den jeweiligen Stand der Technik im Bezug auf Geschwindigkeit, Robustheit und Genauigkeit klar übertreffen.

Schlagwörter: Digitale Bildverarbeitung, Bildausschnitt-basierte Objektverfolgung in Videos, Fusion von Objektverfolgungsmethoden, Panoramabildgenerierung, Fehler- und Defekterkennung

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.

Place	Date	Signature

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.

Ort	Datum	Unterschrift

Acknowledgments

This work was supported by the COMET K-Project *Embedded Computer Vision* (ECV) and has been conducted in collaboration with Fronius Ltd. , the Austrian Institute of Technology (AIT), and Greenfinn Technologies.

First and foremost, I would like to thank my supervisor Horst Bischof for his valuable, indefatigable, and inspiring support, for always having time for discussions, for providing a working environment and just for supporting me in my day to day work. Further, I would like to thank Margrit Gelautz for acting as my second thesis supervisor.

This work would not have been possible without the support of several people from the ICG as well as from all the collaborating companies. In this respect, I wish to express my gratitude to my diploma student Robert Lanner who contributed to this work. I wish to thank my colleagues Matthias Rüther, Peter Roth, Christian Reinbacher, Martin Lenz, and Martin Godec from the ICG for the invaluable collaboration. Further, I wish to thank Günther Reinthaler and Helmut Ennsbrunner from Fronius Ltd. as well as Jürgen Biber, Hartwig Fronthaler and Gerardus Croonen from the AIT, who provided me with valuable data and support from side of the companies. A successful completion of this work would not have been possible without their contributions.

I am also grateful to my other colleagues from the ICG, Katrin Santner, Jakob Santner, Matthias Straka, Michael Maurer, David Ferstl, Michael Donoser, Manuel Werlberger, Martin Köstinger, Christian Bauer, Werner Trobin, Stefan Kluckner, and Christian Leistner, for their advices, for interesting and fruitful discussions, and for their support throughout my time at the institute. Especially, I would like to thank my colleagues in the Robot Vision Group for a very inspiring, interesting and extremely fruitful time.

Finally and exceptionally, I would like to thank my wife Lisa, my brother Stefan, my parents Beate and Diethard, and my family in law for their love, mental support, and encouragements.

Contents

List of Figures

List of Tables

List of Algorithms

1 Introduction

In the last decades various image guided sensors from infrared devices up to high dynamic range sensors have been developed and applied to solve diverse problems in the field of computer vision. Examples are video motion analysis, image based object detection, image based object classification, or highly accurate industrial measurement tasks. Coincidentally, a continuously increasing number of novel industrial manufacturing techniques claims for adequate and robust quality inspection cues. Due to the recent progress in imaging sensor design and development towards measurement variability, robustness, usability and cost-effectiveness (e.g., Microsoft Kinect Sensor[1]), more and more industrial quality inspection applications rely on computer vision algorithms for quality related measurements. Thereby, the formulated measurement problems are two dimensional signal processing problems like, e.g., pattern recognition, texture or color analysis, or distance measurements.

In this Thesis we focus on two dimensional measurement problems and corresponding quality inspection methods as well as related practical applications. Assuming a device under test being located at a fixed position, and with a camera calibration and camera extrinsic parameters given, robust and highly accurate image based measurements are feasible. However, if a probe or specimen undergoes some kind of motion while being inspected, robust detection or tracking algorithms are necessary to initially determine the exact object position in the image before performing further quality assessment. This becomes even more complex if neither a camera calibration nor extrinsic parameters are given and if both the target object and the camera undergo an unknown motion coincidentally. The computer vision discipline that explicitly addresses these

[1] http://www.xbox.com/kinect

problems is image based object tracking, which forms the center theme in this Thesis.

Although, image based tracking is a well studied problem in computer vision. However, there is still a lack of robust and coincidentally real-time capable tracking approaches that are able to perform tracking in highly dynamic scenes or harsh and noisy environments. In this Thesis we present novel image based tracking methods that are on the one hand significantly robust to large amounts of image noise, that are able to adopt to highly dynamic environments or significant object appearance changes without exhaustive parameterization or off-line learning tasks, and that nonetheless remain real-time capable. Thereby, we rely on the fundamental concept of image templates. This is mainly due to following reasons: **a)** Image templates can be used as features or descriptors, thus allowing for an appearance based description of an object to be tracked. **b)** Templates can be easily extracted from images without complex parametrization or computational efforts. **c)** Template based tracking allows for following an object in videos in real-time. However, there are also several drawbacks of template based approaches. Examples are the well known drifting problem [89] that template based tracking approaches suffer from, or given geometric restrictions as template tracking aims in mapping objects with typically planar surfaces from one plane to another. Thus, non-rigid objects or non-linear transformations are problems that template based approaches typically cannot cope with. Anyhow, we address these quite substantial limitations and show in this work how existing template based tracking can be improved such that **a)** significantly large amounts of noise can be handled in a robust fashion, **b)** the geometric restrictions can be relaxed, allowing for a tracking of non-rigid objects in dynamic environments, and **c)** advantages of other different tracking cues can successfully be assimilated into template tracking. From an application point of view, we then show that these novel template tracking methods can be successfully applied in reasonable real-world applications, allowing, e.g., for robust real-time tracking in harsh industrial robotic welding environments, or for successful tracking of non-rigid and non-planar objects in highly dynamic outdoor environments. However, before going into detail we need to introduce typical tracking complications that various tracking approaches struggle with, as well as scenes and environments that we here refer to as harsh environments.

Tracking complications can roughly be separated into object related and environment related influences. Object related complications typically include pose changes of the tracked object which might result in unseen observations or view angles, object deformations where an object undergoes highly complex non-rigid or even non-linear

transformations, resulting in significant shape changes, or object appearance changes where an object's texture or color changes over time. Environment related changes on the other hand are given by external influences that considerably affect the appearance of an imaged object. These include partial or even complete occlusions by other objects, continuous or even rapid illumination changes which are typical for highly dynamic outdoor scenes, or image noise which can be defined by any combination of several here mentioned complications. An example for a harsh environment would be an industrial manufacturing setting. Thereby complicating factors that we would refer to as industrial noise are given by sparks, spilling, smoke wads, evaporating water, gas disturbances or even small explosions. These influences could e.g. occlude objects in images, deform specimen to be analyzed, or significantly change texture or color of depicted objects. Another example for a harsh environment would be an outdoor environment where varying camera poses typically result in significant illumination changes due to a varying incident angle of sun light, suddenly occurring cast shadows, or occlusions of an object e.g. by animals. Figure 1.1 illustrates an object related example for tracking complications as the depicted object undergoes deformations, non-rigid pose changes, and appearance changes due to the closed and re-opened eye. Figure 1.2 on the

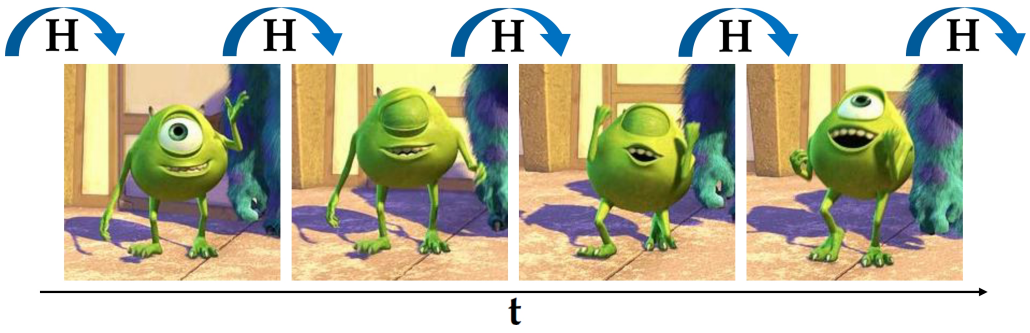


Figure 1.1: **Non-Rigid Motion and Appearance Changes:** Real-wold objects typically undergo non-rigid transformations, resulting in unforeseen appearance changes. These can be even more complex for synthetic videos as illustrated. Solving corresponding tracking problems requires thus for robustness in terms of non-rigid motion and appearance variations.

other hand gives an example for environment related disturbances. Although, the poses of both imaged persons remain nearly constant, rapid illumination changes caused by lightning and visible image noise caused by rain define some exemplary environmental influences that also belong form a facet of tracking complications in harsh environments. To summarize, we define harsh environments in term of image based tracking complica-

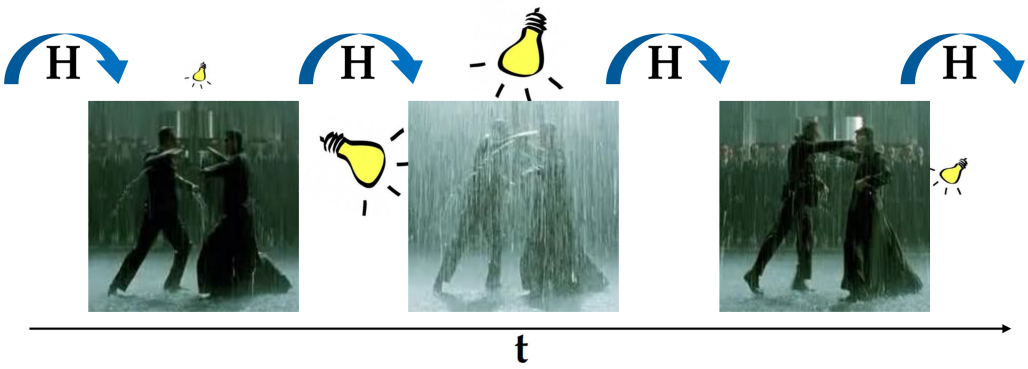


Figure 1.2: **Severe Illumination Changes and Natural Noise:** Environmental illumination changes caused by lightning and visible noise caused by heavy rain represent examples for environmental influences, thus contributing to our definition of harsh environments.

tions as the sum of influences, environmental changes, complicating factors, unforeseen variations, or even combinations of the latter, that severely alter or modify the typical appearance of imaged objects.

Contributions

The contributions presented in this Thesis considering image based tracking and visual quality inspection are manifold: **a)** A novel spline-regularized template tracking method for weld seam tracking in harsh industrial robotic welding environments is presented. Thereby a highly robust real-time template tracker that requires for a minimal number of parameters to be tuned is presented. **b)** We then present an image blending-based method for updating tracking templates, thus allowing for adopting to highly dynamic scenes and harsh environmental conditions while still remaining real-time capable. Moreover, this allows for relaxing the geometrical constraints of typical template trackers which normally estimate a mapping from plane to plane, as even non-rigid objects and more complex transformations can be successfully tracked. **c)** We present a novel tracking fusion approach that allows for combining an arbitrary number of diverse trackers, regardless on their reported outputs. In this way, individual strengths and advantages can be combined, resulting in the possibility for tracking realistic objects in highly dynamic real-world scenes while typically undergoing highly complex non-linear pose or appearance changes. **d)** Finally, we present two image template and consequently appearance based weld seam quality inspection approaches that are intent to be applied for the task of assessing the quality of industrial robotic welding processes.

The first method robustly generates panorama images free of typical industrial noise or disturbances like smoke wads, sparks or spilling in an incremental fashion. The second method is an unsupervised welding defect detection approach that relies on a minimum number of error-free training samples for on-line and real-time recognition of unusual welding regions.

Extensive experimental evaluations and comparisons with corresponding state of the art methods proof our presented concepts in terms of feasibility as well as functionality, and demonstrate the applicability of the methods to reasonable real-world applications.

Outline

This Thesis is organized as follows: Chapter 2 then presents the common notation used throughout the Thesis, as well as the theoretical background of image templates and template related methods, including linear transformations and their robust estimation in the two dimensional projective space. In Chapter 3 related work and the state of the art considering the computer vision disciplines of image based tracking and visual quality inspection are presented and discussed. Our central theme of image based template tracking is then presented in terms of three novel tracking approaches and of corresponding applications in Chapters 4, 5, and 6. In detail, a robust spline-regularized template tracking method, an image blending-based template tracking method, and a segmentation-based method allowing for fusing different trackers are presented. Chapter 7 then introduces a visual quality inspection framework for industrial robotic welding processes, which relies on weld seam image templates. In detail, an incremental welding process panorama image generation method and an unsupervised automatic welding defect detection method are presented. Chapter 8 finally presents our conclusions and an outlook on future work, followed by Appendix A presenting basic mathematical concepts and formulations utilized in methods and algorithms throughout the Thesis, a listing of acronyms and symbols in Appendix B, by an official list of peer reviewed publications in Appendix C, and by the Bibliography.

2 Templates in Computer Vision

In this Chapter we introduce the general concept of image templates and how they can be utilized in computer vision in high level methods and algorithms, presented later on in this Thesis. Typical examples for methods where image template find their appliance are visual pattern recognition, object detection, video tracking, matching, or object description. A detailed description of all these topics would definitely go beyond the scope of this Thesis. As the center themes that we address are given by robust template based tracking and image template based quality assessment, we present the basic concepts of image templates, an overview on different template matching strategies, different linear transformations in the two dimensional projective space, corresponding estimation algorithms, as well as a robust method that allows for coping with a large amount of mismatches and outliers in the following. However, first of all we need to introduce the common notation utilized throughout the remainder of this Thesis.

2.1 Common Notation

Throughout this Thesis, we use the here presented common notation within equations, definitions and for variables. Scalars are depict in italic font, e.g., a or x. Vectors and matrices that consist of several scalar elements are depict in bold font, e.g., $\mathbf{A}$ or $\mathbf{X}$. Vector spaces and numeric ranges like, e.g., the two dimensional projective or the three dimensional Euclidean spaces are depict in bold double-lined upper case letters, e.g., $\mathbb{P}^2$ and $\mathbb{R}^3$. Finally, mapping functions that describe a mapping or a transformation from an arbitrary vector space to another one are depict by calligraphic symbols like, e.g., $\mathcal{F}$ or $\mathcal{H}$. A summary on this common notation including examples is given in Table 2.1.

Entity Descriptions	Examples
scalar values	$a\ b\ x\ y$
vectors	$\mathbf{x} = \begin{pmatrix} x & y & z \end{pmatrix}^T$
matrices	$\mathbf{A} = \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix}$
vector spaces	$\mathbb{R}^1\ \mathbb{R}^2\ \mathbb{R}^3\ \mathbb{P}^1\ \mathbb{P}^2$
mapping functions	$\mathcal{F} : \mathbb{R}^3 \rightarrow \mathbb{R}^1$ $\mathcal{H} : \mathbb{P}^2 \rightarrow \mathbb{P}^2$

Table 2.1: **Common Notation:** A summary of the common notation used in equations, definitions and for variables.

2.2 Image Templates for Object Description

Image templates are generally given by image sub-regions, depicting e.g. some patterns of interest, an object to be tracked, or specific structures that could be used e.g. for object description. Figure 2.1 illustrate on the one hand how templates can be extracted from an image, and on the other hand how these could be utilized in this special case for describing an animated mouse. Either a single template can be used to describe the object, or multiple templates each describing a specific pattern or structure can be combined in a bag-of-words (BoW) model fashion for e.g. object description or categorization. Image templates define a very simple concept for object descriptors in computer vision. Thereby the descriptor is computed from the $n \times m$ pixels located inside the rectangular image sub-region, defining the template. Although, a single vector that contains all template pixel values could be used to describe the depicted structures, pixel neighborhoods are more interesting for describing an object as such additional relations are more discriminative. Thus, many high level methods in computer vision, such as e.g. object categorization, image retrieval, or template matching typically utilize image templates in terms of $n \times m$ pixel sub-regions for their individual purposes. As the center theme of this Thesis is given by template based tracking and quality inspection, we will focus on template matching and linear transformation estimation in the following.

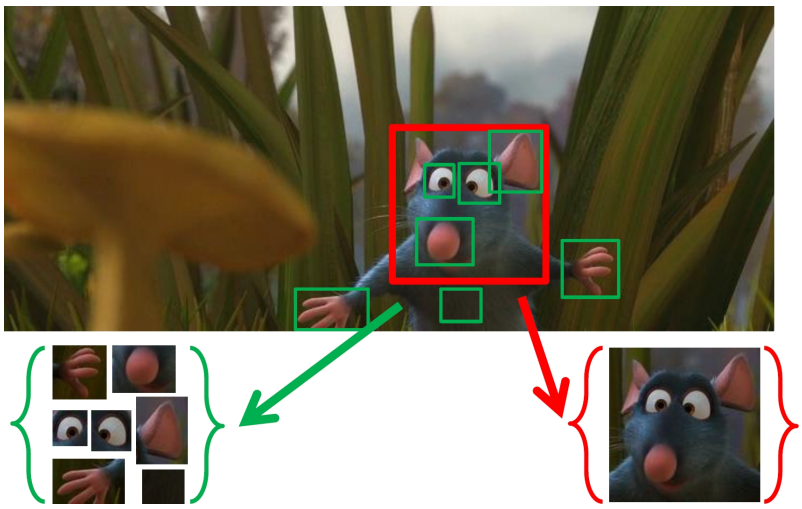


Figure 2.1: **Image Templates:** Image templates are generally given by sub-regions, depicting structures or patterns of interest. These can be used to describe an object using a single template (red), or multiple templates (green) each depicting a specific object pattern are used for object description e.g. in a bag of visual words fashion.

2.3 Template Matching

Template matching defines a basic strategy in computer vision for locating or identifying an object or a pattern in an image. The simplest approach in this course is given by a sliding window matching function to find a given template in an image. Thereby the aim is to exactly locate the object, described by the template as well as its pose, where any similarity measure could be used for intermediate matching computations. Figure 2.2 illustrates the simplest and basic template matching approach, where an object is assumed to exhibit similar scale and pose than the given template.

A more accurate approach is given by the full search algorithm (FSA) which additionally considers 360° rotations by rotating a given template, and arbitrary scales using a pyramid approach, while searching for the best match in a search area. Obviously this is extremely time-consuming, requires for large computational costs and also for large memory consumption. Thus, there is a need for fast reliable and robust template matching methods that do not rely on such exhaustive search approaches while though performing at similar accuracies. In the following we will discuss different matching functions for direct template matching, followed by an overview on different techniques to improve runtime performances and matching accuracy while avoiding exhaustive search.

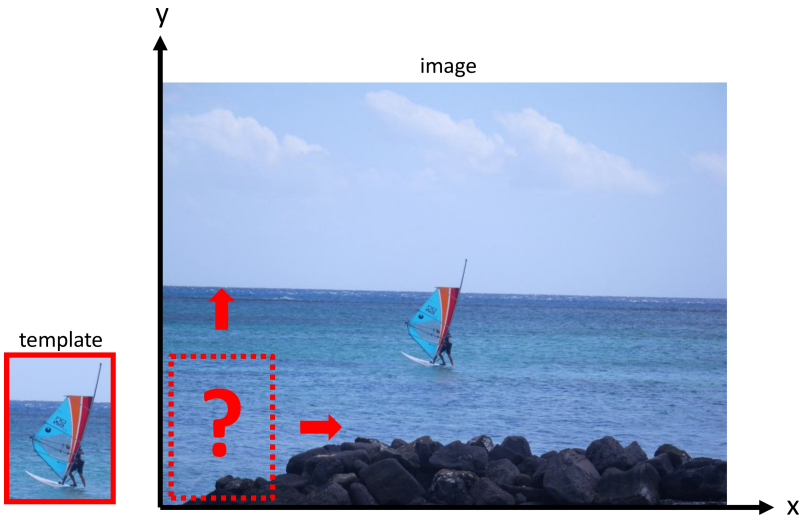


Figure 2.2: **Sliding Window Template Matching:** A given template is searched in an image via sliding the template window over the entire image, and by computing similarities for all possible image locations.

2.3.1 Similarity Measures and Image Correlation

For evaluating the similarity between an image region and a given template two different methods are commonly used in computer vision. In cases where a template is described by an array or vector of pixel values, the matching criterion is typically given by a distance metric or by the norm of vector differences, where a measure of match defines the degree of similarity. A typical example for a distance metric of vectors is given by the squared error defined by

$$d_{err^2} = \sum_{\forall i,j} \left(\mathbf{x}_{ij} - \mathbf{y}_{ij}\right)^2 . \tag{2.1}$$

The most common distance measure though is given by the Euclidean distance or L^2-norm between two vectors $\mathbf{x}$ and $\mathbf{y}$, which determines the distance between two n dimensional vectors in the corresponding vector space. It is defined according to

$$d_\epsilon = \|\mathbf{x} - \mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{n} \left(x_i - y_i\right)^2} . \tag{2.2}$$

However, the sum of pixel differences or the sum of absolute pixel differences are also commonly used. A distance measure that additionally incorporates the variance along each coordinate axis into the distance measure is given by the Mahalanobis dis-

tance between two vectors $\mathbf{x}$ and $\mathbf{y}$, which is defined by

$$d_{\Sigma} = \|\mathbf{x} - \mathbf{y}\|_{\Sigma} = \sqrt{(\mathbf{x} - \mathbf{y})^T \, \Sigma^{-1} \, (\mathbf{x} - \mathbf{y})} \,, \tag{2.3}$$

where Σ denotes the covariance matrix of the vector $(\mathbf{x} - \mathbf{y})$, which is generally defined for a vector $\mathbf{x}$ as

$$\Sigma_{mathbfx} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T - m_{\mathbf{x}} \, m_{\mathbf{x}}^T \,, \quad m_{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \,. \tag{2.4}$$

The measure of match M for a given distance function d can then be computed by evaluating the degree of similarity of the two arrays or images $\mathbf{x}$ and $\mathbf{y}$ according to

$$M_d = \frac{1}{1 + d\,(\mathbf{x}, \mathbf{y})} \,, \tag{2.5}$$

where a perfect match results in $M = 1$ and for increasing mismatch $M =\rightarrow 0$.

However, as mentioned above most high level computer vision methods rely on image templates in terms of $n \times m$ sub-regions. Thus, the second method for evaluating image or template similarities that additionally considers pixel neighborhoods in an image is given by different correlation functions. This allows for more robust and accurate matching. If image sub-regions or patches are used to define a template, the template matching or correlation task is mainly given by comparing image regions and by computing similarities or by evaluating correlation functions in between. A commonly used correlation measure which determines the similarity between two images $\mathbf{I}$ and $\mathbf{J}$ and which is invariant to illumination changes is the normalized cross correlation or $\mathcal{NCC}$ [79]. It is generally defined by

$$\mathcal{NCC}\,(\mathbf{I}, \mathbf{J}) = \frac{\sum\limits_{x,y} (\mathbf{I}\,(x,y) - \mu_{\mathbf{I}})\,(\mathbf{J}\,(x,y) - \mu_{\mathbf{J}})}{\sqrt{\sum\limits_{x,y} (\mathbf{I}\,(x,y) - \mu_{\mathbf{I}})^2 \cdot \sum\limits_{x,y} (\mathbf{J}\,(x,y) - \mu_{\mathbf{J}})^2}} \,, \tag{2.6}$$

where $\mu_{\mathbf{I}}$ and $\mu_{\mathbf{J}}$ are the mean intensity values of the corresponding images $\mathbf{I}$ and $\mathbf{J}$. The obtained correlation value is given in the range of $[-1, 1]$, where positive values indicate higher similarities. A second correlation or image distance measure is given by the root mean square distance ($\mathcal{RMS}$) which denotes a common measure of mismatch between

two digital images and which is given by

$$\mathcal{RMS}\left(\mathbf{I},\mathbf{J}\right)=\sqrt{\frac{1}{n}\sum_{x,y}\left(\mathbf{I}\left(x,y\right)-\mathbf{J}\left(x,y\right)\right)^{2}}. \tag{2.7}$$

The normalized sum of squared differences is another correlation measure that is robust to intensity differences between two images $\mathbf{I}$ and $\mathbf{J}$, but that requires less computations than the $\mathcal{RMS}$. The $\mathcal{NSSD}$ is given by

$$\mathcal{NSSD}\left(\mathbf{I},\mathbf{J}\right)=\frac{\sum_{x,y}\left(\mathbf{I}\left(x,y\right)-\mathbf{J}\left(x,y\right)\right)^{2}}{\sqrt{\sum_{x,y}\mathbf{I}\left(x,y\right)^{2}\sum_{x,y}\mathbf{J}\left(x,y\right)^{2}}}, \tag{2.8}$$

which defines a typical measure of mismatch between two images. The sum of absolute differences $\mathcal{NSAD}$ in contrast considers a linear penalizing of non-similar pixels, and is given by

$$\mathcal{NSAD}\left(\mathbf{I},\mathbf{J}\right)=\frac{\sum_{x,y}\left|\mathbf{I}\left(x,y\right)-\mathbf{J}\left(x,y\right)\right|}{\sqrt{\sum_{x,y}\mathbf{I}\left(x,y\right)^{2}\sum_{x,y}\mathbf{J}\left(x,y\right)^{2}}}. \tag{2.9}$$

Yet another example for a correlation function that defines the relation between two ranked variables is given by the Pearson correlation coefficient $\mathcal{PCC}$ given by

$$\mathcal{PCC}=\frac{\Sigma\left(\mathbf{I},\mathbf{J}\right)}{\sigma_{\mathbf{I}}\sigma_{\mathbf{J}}}, \tag{2.10}$$

where Σ again denotes the covariance matrix, and σ defines the variances of the images $\mathbf{I}$ and $\mathbf{J}$, respectively. Thus the Pearson correlation describes the linear dependence of two images in terms of a correlation measure.

Finally, the structural similarity or $\mathcal{SSIM}$ presented by Wang [128] defines an image similarity measure that is designed to additionally quantify errors between two signals $\mathbf{x}$ and $\mathbf{y}$. It is composed of a luminance component that considers the mean intensities given by

$$l\left(\mathbf{x},\mathbf{y}\right)=\frac{2\mu_{x}\mu_{y}+\epsilon_{l}}{\mu_{x}^{2}+\mu_{y}^{2}+\epsilon_{l}}, \tag{2.11}$$

a contrast component that considers the corresponding standard deviations defined as

$$c\left(\mathbf{x},\mathbf{y}\right)=\frac{2\sigma_{x}\sigma_{y}+\epsilon_{c}}{\sigma_{x}^{2}+\sigma_{y}^{2}+\epsilon_{c}}, \tag{2.12}$$

and a structural comparison component that is based on the correlation between the two unit vectors $(\mathbf{x} - \mu_x)/\sigma_x$ and $(\mathbf{y} - \mu_y)/\sigma_y$ given by

$$s(\mathbf{x}, \mathbf{y}) = \frac{\sigma_{xy} + \epsilon_s}{\sigma_x \sigma_y + \epsilon_s} . \tag{2.13}$$

Thereby, ϵ_l, ϵ_c, and ϵ_s are constants that ensure stability of the individual components. The structural similarity index between to signals $\mathbf{x}$ and $\mathbf{y}$ is then given by a linear combination of above introduced components.

$$\mathcal{SSIM}(\mathbf{x}, \mathbf{y}) = l(\mathbf{x}, \mathbf{y})^{\alpha} \; c(\mathbf{x}, \mathbf{y})^{\beta} \; s(\mathbf{x}, \mathbf{y})^{\gamma} , \tag{2.14}$$

where α, β and γ are power terms that control the impact or importance of each component. In oder to obtain a single similarity measure between to images $\mathbf{I}$ and $\mathbf{J}$, structural similarities are calculated in a sliding window approach. The final similarity measure is given by the mean $\mathcal{SSIM}$ as

$$\mathcal{MSSIM}(\mathbf{I}, \mathbf{J}) = \frac{1}{N} \sum_{i_1}^{N} \mathcal{SSIM}(\mathbf{x}_i, \mathbf{y}_i) , \tag{2.15}$$

where N denotes the number of considered local windows, and $\mathbf{x}_i$ and $\mathbf{y_i}$ denote the corresponding image contents of the i^{th} local window.

2.3.1.1 Matching and Matched Filtering

By applying above presented correlation measures to a search region that is typically larger than a given template for exhaustive template matching, a match surface $m(x, y)$ is generated. This generation can also be formulated as a convolution of a matching filter $h(x, y)$ with a given search area $s(x, y)$ according to

$$m(x, y) = s(x, y) \otimes h(x, y) . \tag{2.16}$$

Thereby matching filters are designed and derived to confirm some match fidelity criterion. A typical example is the signal to noise ratio ($\mathcal{SNR}$) given by the ratio of the filter response peal to the filter response for additive noise. Thereby the filter response $m(x, y)$ is defined as a the sum of image data $m_I(x, y)$ and additive noise $m_\lambda(x, y)$,

resulting in the $\mathcal{SNR}$ measure of a peak response at (x_0, y_0) given according to

$$\mathcal{SNR} = 10\log\frac{m\,(x_0, y_0)}{\sigma_\lambda^2}\,. \tag{2.17}$$

Matched filters in turn are optimal for detecting a signal in the presence of white noise, where the matched filter response at (x, y) is equivalent to the cross-correlation of template $t\,(x, y)$ and search region $s\,(x, y)$ at the corresponding coordinates. The least mean square filter gives yet another variation of a matched filter that is optimal in the presences of globally stationary noise, generally given by a zero-mean matched filter and thus equivalent to performing a zero-mean cross correlation.

2.3.2 Fast Matching Strategies

Although, the above presented correlation functions and similarity measures allow for robust appearance based matching under consideration of different aspects like illumination variations, image noise, or clutter, a common method for further improving template matching is to perform some preprocessing steps or to transform the template and the image to vector spaces that are more suitable for specific matching tasks. Examples would be usage of integral images, image binarizations, or pruning strategies, as well as utilizing the advantages of the frequency domain instead of matching in the spatial image domain.

Sibiryakov [112] proposed to densely transform template and image in binary code form by projecting and quantizing histograms of oriented gradients (HoG) [32]. They rely on HoG features as they exhibit an invariance to local object and appearance changes, as the distributions of intensity gradients do not significantly change in such cases. With the template and the image given in binary form, extremely fast matching can be performed based on Hamming distances denoted by $\mathcal{HAM}$, which are typically used in information theory. For two given strings, vectors, or images the Hamming distance equals the number of positions where the corresponding symbols are different:

$$\mathcal{HAM}\,(\mathbf{I}, \mathbf{J}) = \sum_{\forall x,y} \mathbf{I}\,(x, y) \neq \mathbf{J}\,(x, y) \tag{2.18}$$

Sibiryakov empirically showed that this derivative of template matching outperforms other correlation measures like $\mathcal{NCC}$ in terms of runtime and accuracy.

Shin et al. [111] proposed to use index tables which store image coordinates of pixels exhibiting similar gray values in terms of hash tables. For matching of a given template

a penalty based matching approach that considers small pixel errors allows for fast and robust matching in the presence of up to 30% Gaussian noise. Another approach for speeding up template matching is given by using integral images instead of the original image data. Jung et al. [66] demonstrated how integral images can be to boost the naive full search template matching scheme by using an integral image as search area instead of a block sum pyramid, resulting in reasonable speedups, less memory consumption, and an optional extension to non-square template matching.

Nguyen et al. [93] proposed a pruning scheme to remove image regions that are not matchable with only few simple computational operations beforehand, resulting in significantly increased runtime performances compared to standard template matching. The presented approach uses Haar-like features as weak features for identifying potential matching candidates, which are then compared to the given template using the $\mathcal{NCC}$ measure. In this way, runtime is reduced by ten times compared to standard methods e.g. based on the Fast Fourier Transformation.

Another traditional approach that allows for identifying image regions similar to a given template is the Fast Fourier Transform ($\mathcal{FFT}$) introduced by Cooley and Tukey [29] and typically defined by the Radix-2 Decimation-in-Time ($\mathcal{DIT}$) form according to

$$X_k = \sum_{m=0}^{\frac{N}{2}-1} x_{2m} e^{-\frac{2\pi i}{N}(2m)k} + \sum_{m=0}^{\frac{N}{2}-1} x_{2m+1} e^{-\frac{2\pi i}{N}(2m+1)k}\,,\; k \in [0, N-1]\,, \tag{2.19}$$

where the Discrete Fourier Transform ($\mathcal{DFT}$) of length N is described by two $\mathcal{DFT}$s of length $\frac{N}{2}$, given by the sum of $\mathcal{DFT}$s of even-indexed inputs E_k and of odd-indexed inputs O_k in terms of the general divide and conquer technique. Thus the $\mathcal{FFT}$ is finally given as

$$X_k = \begin{cases} E_k + e^{-\frac{2\pi i}{N}k} O_k & \text{if } k < \frac{N}{2} \\ E_{k-\frac{N}{2}} - e^{-\frac{2\pi i}{N}\left(k-\frac{N}{2}\right)} O_{k-\frac{N}{2}} & \text{if } k \geq \frac{N}{2} \end{cases}\,. \tag{2.20}$$

In terms of template matching, an image and a template are both first transformed to the Fourier space where complex convolution operations can be performed by simple matrix multiplications, using the Fast Fourier Transform. Aboutajdine and Essannouni [1] recently presented fast block matching algorithms that allow for performing $\mathcal{SSD}$, $\mathcal{SAD}$ and sum fourth order moment ($\mathcal{SFOM}$) correlation operations in the Fourier space, where computational costs are significantly reduced while obtaining similar results than with standard correlation functions.

Uenohara and Kanade [121] presented a fast pattern matching algorithm based on

the Fourier transform and the Karhunen-Loeve transform [92]. Thereby the eigenvectors derived by the Karhunen-Loeve transform are considered as patterns to be recognized. The presented approach addresses the task of pattern or template matching for objects with unknown distortions within a short period, where an object is given by multiple intensity patterns with different distortions generated using the Karhunen-Loeve eigenvectors. Transformation to the frequency domain using the Fourier transform, and subsequent normalized correlation between object patterns and the input image define the presented template matching approach, giving a significant speedup to standard spatial domain approaches.

A survey on different strategies on how to determine matches in the frequency domain at optimized reduced runtime performances can be found in the work of Fredriksson et al. [42].

For a review on even more advanced methods like Matched Spatial Filters, Synthetic Discriminant Functions, or low dimensionality representations for matching, including Principal Component Analysis (PCA), Independent Component Analysis (ICA), or Linear Discriminant Analysis (LDA), we would refer to the work of Brunelli and Poggio [18], which sets a special application focus on locating eyes in face images, to the report of Cox [30], and to the book of Brunelli [17] which extensively reviews template matching techniques in computer vision.

2.4 Template Matching in Video Tracking

A typical example for a high level computer vision method where template matching defines a basic and crucial component is video tracking. Thereby, an image template that depicts an object to be tracked is followed in an image sequence or video. Although, the simple sliding window approach and the different template matching methods described above might be useful in some tracking applications, the intuitive assumptions of object rigidness or of a simple translation as motion model between consecutive frames typically do not hold for real world situations. Options to cope with changes in size, rotations or even pattern distortions would be e.g. to transform the image to a standard size and orientation. However, this works only if there is no size or orientation variation given in the image. Another option would be to spatially scale and rotate the template in terms of the full search algorithm allowing for selecting the best matching scale and rotation. However, this results in high computational costs for large numbers of

scales and rotations. Thus, different more realistic object transformations that consider translations, rotations, scaling, shear, affine and projective transformations are required. Such transformations are typically computed from a set of corresponding image points, identified in consecutive or overlapping images. Thereby, the correspondence problem is solved by matching feature point descriptors like e.g. SIFT [81], SURF [6], or BRIEF [21], image templates, or interest points such as e.g. Harris corners [50] or FAST corners [102]. Taylor et al. [117, 118] rely on features based on histograms of pixel intensities, which together with a smart indexing scheme and a novel bit mask representation allow for feature matching of one or multiple targets in few microseconds.

In the following, we introduce a set of linear transformations in the two dimensional projective space $\mathbb{P}^2$, and corresponding Gold Standard estimation methods. These transformations allow for template tracking in more realistic real world situations, considering template distortions that range from rigid transforms to projective homographies.

2.4.1 Projective Geometry in 2D

In this Section, basic geometric relations and the algebraic formulation of two dimensional projective transformations are discussed, based on the formulations in the book of Hartley and Zisserman [51]. For further in-depth concepts and methods we would refer to the introductory chapters of [37], [51] and [85]. In general, the projective geometry represents an elegant way to model the perspective imaging concept, and provides appropriate mathematical representations in form of, e.g., linear matrix equations. The motivation for projective geometry and projective transformations of planes is given by the fact that the general imaging process by a camera is nothing else than the projection of three dimensional world points onto a two dimensional image plane under consideration of a specific projection model. Starting with a definition of the geometric primitives and the basic concept of homogeneous representations, different classes of transformations in the two dimensional projective space $\mathbb{P}^2$ are presented and discussed in subsequent Sections.

2.4.1.1 Geometric Primitives

The geometric primitives of projective geometry are given by linear entities in $\mathbb{P}^2$, such as points, lines and conics, and by linear entities in $\mathbb{P}^3$, namely points, lines, planes and quadrics. However, we focus on projective geometry in $\mathbb{P}^2$, as later on presented high level image processing methods and applications mainly rely on projective geometry in

the image plane. Geometric primitives in $\mathbb{P}^2$ are given by elementary entities like points, lines, or conics. These can be algebraically written as vectors or matrices in the projective space. Projective geometry is consequently defined by the geometric primitives as well as by geometric relations in between. A point in the two dimensional Euclidean space $\mathbb{R}^2$ is given by a coordinate pair (x, y), if $\mathbb{R}^2$ is considered as a vector space. A line in the plane is given by a homogeneous 3-vector $(a, b, c)^T$, and is inhomogeneously defined by

$$ax + by + c = 0, \tag{2.21}$$

where different values for (a, b, c) result in different lines. A point $\mathbf{x} = (x, y)^T$ lies on a line $\mathbf{l} = (a, b, c)^T$ if and only if $ax + by + c = 0$. Considering vectors, this may be written as an inner product, where an additional 1 is added as third coordinate to the point $(x, y)^T$ in $\mathbb{R}^2$. In this way, points are represented as homogeneous vectors, similar to the homogeneous vectors of lines. The arbitrary formulation of a homogeneous point $\mathbf{x}$ in the projective space $\mathbb{P}^2$ is thus given by $(x_1, x_2, x_3)^T$, representing the Euclidean point $(x_1/x_3, x_2/x_3)^T$ in $\mathbb{R}^2$. As mentioned above, a homogeneous point $\mathbf{x}$ lies on a line $\mathbf{l}$ if and only if $\mathbf{x}^T\mathbf{l} = 0$. In a similar way the intersection of two lines $\mathbf{l}$ and $\mathbf{l}'$ can be represented as a homogeneous point $\mathbf{x} = \mathbf{l} \times \mathbf{l}'$, where $\times$ defines the vector or cross product. At last, a line $\mathbf{l}$ connecting two homogeneous points $\mathbf{x}$ and $\mathbf{x}'$ is given by the cross product $\mathbf{l} = \mathbf{x} \times \mathbf{x}'$. Finally, the special cases of ideal points or points at infinity and of lines at infinity need to be defined. Finite points in $\mathbb{R}^2$ are defined by homogeneous points, where $x_3 \neq 0$. However, the projective space $\mathbb{P}^2$ also contains points where $x_3 = 0$. These points are known as *ideal points* or *points at infinity* denoted by $\mathbf{x}_\infty$. These points lie on a single common line $\mathbf{l}_\infty = (0, 0, 1)^T$, called the *line at infinity*. With this additional entities given, in $\mathbb{P}^2$ intersections of parallel lines are also defined in contrast to the standard Euclidean geometry of $\mathbb{R}^2$, where parallel lines form a special case and cannot be intersected. Table 2.2 summarizes the geometric primitives and entities in $\mathbb{P}^2$.

2.4.1.2 Projective Transformations and Mappings

Per definition, a projective transformation in $\mathbb{P}^2$ denoted by $\mathcal{F} : \mathbf{P}^2 \rightarrow \mathbf{P}^2$ is an invertible mapping from points in $\mathbb{P}^2$ to corresponding points in $\mathbb{P}^2$ that maps lines to lines. Hence, if points $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{x}_3$ lie on the same line, then the transformed points $\mathbf{x}'_1, \mathbf{x}'_2$ and $\mathbf{x}'_3$ also lie on a common line. In literature [51] such a transformation is referred to as projectivity, projective transformation, or homography. An algebraic definition of a

Entity Descriptions	**Algebraic Formulation**
homogeneous point	$\mathbf{x} = (x_1, x_2, x_3)^T$
homogeneous line	$\mathbf{l} = (l_1, l_2, l_3)^T$
Euclidean point	$\mathbf{x}_E = (x_1/x_3, x_2/x_3)^T$
intersection of two lines	$\mathbf{x} = \mathbf{l}_1 \times \mathbf{l}_2$
line connecting two points	$\mathbf{l} = \mathbf{x_1} \times \mathbf{x}_2$
line normal	$\mathbf{n} = (l_1, l_2)^T$
point at infinity	$\mathbf{x}_\infty = (x_1, x_2, 0)^T$
line at infinity	$\mathbf{l}_\infty = (0, 0, 1)^T$

Table 2.2: **Geometric Primitives in $\mathbb{P}^2$:** Basic geometric entities and their algebraic relations in the two dimensional projective space.

projective transformation in $\mathbb{P}^2$ is given by a 3×3 matrix $\mathbf{H}$ that transforms any point in $\mathbb{P}^2$ that is represented by a 3-vector $\mathbf{x}$ according to

$$\mathbf{x}' = \mathbf{H}\mathbf{x}\,, \tag{2.22}$$

where the projective transformation of a line $\mathbf{l}$ under the defined homogeneous point transformation is consequently given by

$$\mathbf{l}' = \mathbf{H}^{-T}\mathbf{l}\,. \tag{2.23}$$

Due to the homogeneous character of projective transformations only the ratio of the matrix elements is significant. As a projective transformation consists of nine matrix elements, eight independent matrix element ratios result in eight degrees of freedom (DOF). In the following different classes of two dimensional transformations exhibiting different geometric invariants are described. Thereby, we focus on the specific scalar invariants of the geometric configurations, where an invariant denotes a function of the configuration whose value is preserved by the specific transformation. Examples for such invariants would be, translations, rotations, or angles between lines. Depending on the number of unknown matrix entries and geometric configurations, different degrees of freedom (DOF) and hence different invariants are obtained for different classes of

transformations.

Isometry Transformations

Transformations of a plane in $\mathbb{R}^2$ that preserve Euclidean distances and that are called isometries or rigid body transformations are defined according to

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} \epsilon\, cos\theta & -sin\theta & t_x \\ \epsilon\, sin\theta & cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad \text{with} \quad \epsilon = \pm 1\,. \tag{2.24}$$

Depending on the sign of ϵ the transformation remains either orientation preserving $(+1)$, or reverses the orientation (-1). In general, isometries are given by a composition of a translation $\mathbf{t}$ and a rotation $\mathbf{R}$. Thus it exhibits three degrees of freedom (DOF), one for the rotation angle, and two for the translation in $\mathbb{R}^2$.

$$\mathbf{x}' = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x} \tag{2.25}$$

The corresponding invariants obtained by a planar isometry transformation are the distance between two points, the angle between two lines, and the area spanned by at least three circular connected points.

Similarity Transformations

An isometry that is additionally composed of an isotropic scaling s is called similarity transformation. In matrix representation a similarity transform is given as

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} s\, cos\theta & -s\, sin\theta & t_x \\ s\, sin\theta & s\, cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \tag{2.26}$$

Planar similarity transforms exhibit four degrees of freedom (DOF). Similar to isometry transformations one for the rotation angle, two for the translation vector in $\mathbb{R}^2$, and an additional fourth degree for the scaling. The matrix block form of a similarity transformation is given according to

$$\mathbf{x}' = \begin{bmatrix} s\,\mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x}, \quad \text{with} \quad \mathbf{R}^T\mathbf{R} = \mathbf{I}\,. \tag{2.27}$$

The invariants of similarity transformations are given by the ration of lengths, the ratio of areas, the angles between lines, and by parallel lines that remain parallel. Thus, a similarity transformation is shape preserving.

Affine Transformations

An affine transformation is defined by the composition of a non-singular linear transformation with an additional translation. The matrix representation of an affinity is given according to

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} , \tag{2.28}$$

where the number of degrees of freedom (DOF) is six according to the six unknown matrix elements. The block form of an affine transformation is given as

$$\mathbf{x}' = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \mathbf{x} , \tag{2.29}$$

where $\mathbf{A}$ can be decomposed into three component matrices using standard Singular Value Decomposition (SVD):

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T = \left(\mathbf{U}\mathbf{V}^T\right)\left(\mathbf{V}\Sigma\mathbf{V}^T\right) = \mathbf{R}_\theta \left(\mathbf{R}_{-\phi}\, \Sigma\, \mathbf{R}_\phi\right) , \quad \text{with} \quad \Sigma = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{2.30}$$

Compared to a similarity transformation, the two additional degrees of freedom (DOF) are obtained from the ratio of the two scaling parameters $\lambda_1 : \lambda_2$, and from the additional rotation angle ϕ, specifying the scaling direction. The invariants of an affine transformation are parallel lines that remain parallel as a point at infinity is mapped onto another point at infinity under an affine transformation. Second, the ratio of parallel lengths or of parallel line segments remains also invariant as the scaling along a specific axis remains the same for all lines with the same direction. Finally, the ratio of areas remains invariant as the area of any shape is scaled according to $\lambda_1\lambda_2$, which results in the area being canceled out as an invariant.

Homography Transformations

A projective homography transformation can also be seen as a generalization of an affine transformation. It is generally defined as a non-singular linear transformation of homogeneous 3-vectors, represented in matrix form as

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} . \tag{2.31}$$

In contrast to an affinity, a projective homography does not distinguish between orientation preserving and orientation reversing projectivities in $\mathbb{P}^2$. The matrix block form of a homography is give according to

$$\mathbf{x}' = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{v}^T & v \end{bmatrix} \mathbf{x}, \quad \text{with} \quad \mathbf{v} = (v_1, v_2)^T . \tag{2.32}$$

The number of degrees of freedom (DOF) for a homography sums up to eight, which results from two degrees for scaling, two for rotation, two for translation, and two for the line at infinity. Thus, a homography can be computed from four non-collinear point correspondences. The only resulting invariant is given by the cross-ratio or ratio of ratio of four points on a line or of lengths.

Figure 2.3 and Table 2.3 again summarize the introduced planar transformation types in $\mathbb{P}^2$. The distortions, the degrees of freedom (DOF) that rely on the number of unknown matrix elements, the number of required point correspondences, and transformation related properties that remain invariant under the specific mappings are depict and listed.

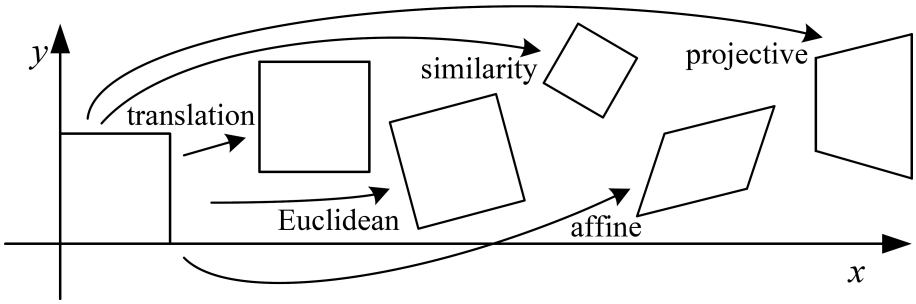


Figure 2.3: **Linear Projective Transformations in 2D:** Different distortions of planar transformations in $\mathbb{P}^2$. [115]

Transformation	Matrix	DOF	Points	Invariants
Isometry	$\begin{bmatrix} \epsilon\, cos\theta & -sin\theta & t_x \\ \epsilon\, sin\theta & cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$	3	2	length, area
Similarity	$\begin{bmatrix} s\, cos\theta & -s\, sin\theta & t_x \\ s\, sin\theta & s\, cos\theta & t_y \\ 0 & 0 & 1 \end{bmatrix}$	4	2	ratio of lengths, angle
Affinity	$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$	6	3	parallelism, ratio of areas
Homography	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$	8	4	cross-ratio (ratio of ratio)

Table 2.3: **Linear Projective Transformations in $\mathbb{P}^2$:** A summary of planar projective transformations in $\mathbb{P}^2$. The specific transformation matrices exhibit different degrees of freedom (DOF), depending on the number of unknown matrix elements. Consequently different numbers of point correspondences are required for their estimation. Transformation related invariants under the specific mappings are also specified.

2.4.2 Transformation Estimation

Transformation estimation methods generally describe how to establish a particular geometric model that maps shapes or geometric primitives from one image or plane to another. For tracking algorithms, which form a the central theme in this Thesis, planar transformation models are essential as they describe the way in which an object has been transformed from one frame to another. Depending on the number of model parameters, different *Gold Standard* methods exist. Isometries and similarities are typically computed by the *Procrustes Alignment* algorithm [46], which estimates a rigid motion in the image plane. Affinities and homographies on the other hand are computed by the *Direct Linear Transformation* (DLT) [51], which relies on algebraic relations between corresponding image points. As measurement errors in point correspondences typically do not follow a Gaussian distribution, the *Gold Standard* method for affinities and homographies further considers the robust *Random Sample Consensus* (RANSAC) [40] estimator. In this way mismatches and outliers are identified and not considered for the transformation estimation.

2.4.2.1 Rigid Motion Estimation

A rigid motion in image plane generally consists of a translation vector $\mathbf{t}$ and a rotation matrix $\mathbf{R}$. The generalized Procrustes analysis is a standard rigid motion estimation algorithm which can be dated back to the work of Gower [46]. In general, the Procrustes problem describes the task of estimating a rigid body transformation between N corresponding points $\mathbf{x}_i$ and $\mathbf{x}'_i$ by minimizing the mean squared distance

$$d = \frac{1}{N} \sum_{i=1}^{N} |\mathbf{R}\,\mathbf{x}_i + \mathbf{t} - \mathbf{x}'_i|^2\,, \tag{2.33}$$

where $\mathbf{x}_i$ and $\mathbf{x}'_i$ define corresponding image point matrices, containing one coordinate per row and one point per column respectively. The overall Procrustes Alignment consists of three consecutive steps. First, both point sets are normalized to their centers of gravity:

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \overline{\mathbf{x}}\,, \quad \text{with} \quad \overline{\mathbf{x}} = \frac{1}{N} \sum_{j=1}^{N} \mathbf{x}_j \tag{2.34}$$

Second, the rotation matrix $\mathbf{R}$ is determined by Singular Value Decomposition (SVD) of $\mathbf{A} \equiv \sum \tilde{\mathbf{x}}_i \mathbf{x}_i'^T = \mathbf{U}\Sigma\mathbf{V}^T$:

$$\mathbf{R} = \mathbf{V}\mathbf{D}\mathbf{U}^T\,, \quad \text{with} \quad \mathbf{D} = \begin{bmatrix} det\left(\mathbf{V}\mathbf{U}^T\right) & 0 \\ 0 & det\left(\mathbf{V}\mathbf{U}^T\right) \end{bmatrix} \tag{2.35}$$

Finally, the remaining translation vector $\mathbf{t}$ is computed:

$$\mathbf{t} = \overline{\mathbf{x}'} - \mathbf{R}\overline{\mathbf{x}} \tag{2.36}$$

The resulting Euclidean transformation mapping that consists of a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{t}$ can then be rewritten as a single transformation matrix $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{2.37}$$

2.4.2.2 Direct Linear Transformation

The Direct Linear Transformation (DLT) algorithm is a simple algorithm for determining a planar transformation given by $\mathbf{x}'_i\mathbf{H}\mathbf{x}_i$ from $n \geq 4$ image point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$. Considering the j-th row of matrix $\mathbf{H}$ as $\mathbf{h}^{j^T}$ and a point $\mathbf{x}'_i = (x'_i, y'_i, w'_i)$, the planar

transformation to be determined can be written as

$$\mathbf{x}'_i \times \mathbf{H}\mathbf{x}_i = \begin{bmatrix} y'_i \mathbf{h}^{3^T} \mathbf{x}_i - w'_i \mathbf{h}^{2^T} \mathbf{x}_i \\ w'_i \mathbf{h}^{1^T} \mathbf{x}_i - x'_i \mathbf{h}^{3^T} \mathbf{x}_i \\ x'_i \mathbf{h}^{2^T} \mathbf{x}_i - y'_i \mathbf{h}^{1^T} \mathbf{x}_i \end{bmatrix} . \tag{2.38}$$

According to $\mathbf{h}^{j^T} \mathbf{x}_i = \mathbf{x}_i^T \mathbf{h}^j$ for $j = 1 \cdots 3$, a system of equations in the entries of $\mathbf{H}$ of the form $\mathbf{A}_i \, \mathbf{h} = \mathbf{0}$ can be formulated for each point correspondence.

$$\begin{bmatrix} \mathbf{0}^T & -w'_i \mathbf{x}_i^T & y'_i \mathbf{x}_i^T \\ w'_i \mathbf{x}_i^T & \mathbf{0}^T & -x'_i \mathbf{x}_i^T \\ -y'_i \mathbf{x}_i^T & x'_i \mathbf{x}_i^T & \mathbf{0}^T \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0} \tag{2.39}$$

Although, there are three equations included in the system of equations above, only two of them are linearly independent. The third row is obtained up to scale from the sum of x'_i times the first row and y'_i times the second row. This results in two equations in the entries of $\mathbf{H}$ for each point correspondence. Thus in literature [51] the third equation is usually omitted, giving

$$\mathbf{A}_i \, \mathbf{h} = \begin{bmatrix} \mathbf{0}^T & -w'_i \mathbf{x}_i^T & y'_i \mathbf{x}_i^T \\ w'_i \mathbf{x}_i^T & \mathbf{0}^T & -x'_i \mathbf{x}_i^T \end{bmatrix} \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} = \mathbf{0} \, , \tag{2.40}$$

where $\mathbf{A}_i$ is a 2×9 matrix, the system $\mathbf{A}_i \, \mathbf{h} = \mathbf{0}$ is an equation linear in the unknown $\mathbf{h}$, and where the matrix elements of $\mathbf{A}_i$ are quadratic in the known point coordinates. In general, this system of equations holds for any homogeneous coordinate representation $(x'_i, y'_i, w'_i)^T$ of a point $\mathbf{x}'_i$, where (x'_i, y'_i) are measured image coordinates if $w'_i = 1$.

The desired homography transformation $\mathbf{H}$ is then computed with an over-determined system of equations given as $\mathbf{A}\mathbf{h} = \mathbf{0}$, where $\mathbf{A}$ is built up from the matrix rows $\mathbf{A}_i$ contributed from each point correspondence, and where $\mathbf{h}$ is a 9-vector that is made up of the entries of the desired transformation matrix $\mathbf{H}$.

$$\mathbf{h} = \begin{pmatrix} \mathbf{h}^1 \\ \mathbf{h}^2 \\ \mathbf{h}^3 \end{pmatrix} , \quad \mathbf{H} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \tag{2.41}$$

The result of this general DLT formulation for 2D homographies depends on the coordinate frame in which points are expressed, and is not invariant to similarity trans-

formations of an image. Hence, there exist coordinate systems that are better suited for these type of estimation methods. Hartley [52] proposed to apply a normalization consisting of a translation and a scaling of image coordinates to the data before applying the DLT algorithm, followed by an appropriate correction of the DLT result afterwards. In this way, they obtain the correct transformation matrix $\mathbf{H}$ with respect to the original coordinate frame.

For the normalization, first the coordinate systems of each image is translated, in order to bring the centroids of each point set to the origin. Then, the x and y coordinates of each point $\mathbf{x} = (x, y, w)^T$ are equally scaled by an isotropic scaling factor, such that the average distance of a point $\mathbf{x}$ from the origin is equal to $\sqrt{2}$. The normalized Direct Linear Transformation is summarized in Algorithm 1.

Algorithm 1 Normalized Direct Linear Transformation (DLT):

Input: $n \geq 4$ 2D to 2D point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$
Output: 2D homography transformation matrix $\mathbf{H}$ such that $\mathbf{x}'_i = \mathbf{H}\mathbf{x}_i$

(a) Normalization of $\mathbf{x}$ by estimation of similarity transformation $\mathbf{T}$, consisting of a translation and scaling, that takes points $\mathbf{x}_i$ to a new set of points $\tilde{\mathbf{x}}_i$ such that the centroid of the points $\tilde{\mathbf{x}}_i$ is the origin, and their average distance from the origin is $\sqrt{2}$.

(b) Normalization of $\mathbf{x}'$ by estimation of similarity transformation $\mathbf{T}'$ for the points in the second image, transforming points $\mathbf{x}'_i$ to $\tilde{\mathbf{x}}'_i$.

(c) Compute matrix $\mathbf{A}_i$ according to Equation 2.40 for all correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$.

(d) Assemble the n 2×9 matrices $\mathbf{A}_i$ into a single $2n \times 9$ matrix $\mathbf{A}$.

(e) Obtain the SVD of $\mathbf{A}$, where the unit singular vector corresponding to the smallest singular value (last column of $\mathbf{V}$ if $\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T$) is the solution $\mathbf{h}$.

(f) Determine matrix $\tilde{\mathbf{H}}$ from $\mathbf{h}$ according to Equation 2.41.

(g) Denormalization according to $\mathbf{H} = \mathbf{T}'^{-1}\tilde{\mathbf{H}}\mathbf{T}$.

2.4.2.3 Random Sample Consensus (RANSAC)

For typical real-world problems it cannot be assumed that a set of point correspondences used to estimate an arbitrary model only exhibits measurement errors that follow a Gaussian distribution. In fact, there is a high probability that the measurements in-

Algorithm 2 Random Sample Consensus (RANSAC):

Input: data set **S** that contains outliers, distance threshold ζ, inliers threshold Δ
Output: robust fit of model **M**

(a) Randomly select a subset **s** from **S** and estimate **M** from **s**.

(b) Determine the input data elements $\mathbf{S}_i$ that lie within the distance threshold ζ of the model, which coincidentally define the consensus set of the sample and the inliers of **S**.

(c) If the size of $\mathbf{S}_i$ is greater than the inliers threshold Δ, re-estimate **M** using $\mathbf{S}_i$ and terminate.

(d) If the size of $\mathbf{S}_i$ is less than Δ, select a new subset **s** and repeat the above.

(e) After N trials the largest consensus set $\mathbf{S}_i$ is selected to re-estimate the final robust fit **M**.

clude mismatches that are outliers to the Gaussian error distribution, and that severely disturb the estimated model. Hence, robust estimation methods that are able to identify highly suitable inliers, and that are tolerant to outliers or measurements that follow a different error distribution, are required.

The Random Sample Consensus (RANSAC) [40] algorithm is a general and very successful robust estimator that can cope with a large proportion of outliers. The algorithm first selects a random sample from the given input data which is sufficient to estimate the desired model. This sample is then used to fit a first estimate of the model. Next, the corresponding support is evaluated by the number of input data elements that lie within a specific distance threshold. This random selection is repeated a defined number of times, and the model that gives the highest support is finally chosen as the robust fit, where input data elements that lie within the distance threshold define the final set of inliers. This robust estimator is designed opposite to conventional smoothing techniques, because it aims to find the smallest feasible initial sample, which is subsequently enlarged with consistent data if possible. Algorithm 2 summarizes the procedure.

2.5 Conclusion

In this Chapter we have presented the basic fundamental concept of image templates in computer vision as well as a short review on different template matching strategies. We

have shown how templates can be utilized in high level computer vision methods for object description, and image based tracking. As the center theme of this Thesis is given by robust template based tracking as well as image template based quality assessment in harsh environments, we have presented linear transformations, corresponding estimation methods, and a robust algorithm that allows for coping with large amounts of outliers or noise. To summarize, the tracking methods presented in Chapters 4, 5, and 6 and the quality inspection methods presented in Chapter 7 are based on a common set of image template related theoretical foundations and backgrounds, which have been presented in this Chapter in a common notation.

3

Related Work on Tracking and Visual Quality Inspection

The topics and concepts that are presented in subsequent Chapters generally deal with the computer vision disciplines of video object tracking and visual quality inspection, whereas the latter especially focuses on industrial robotic welding. Up to now we have presented the basics and fundamental concepts of image templates which form the center theme in this Thesis.

However, this Chapter provides an overview of related work and state of the art methods for the computer vision disciplines of **a)** image based tracking, including specific properties, advantages and weaknesses of diverse tracking methods and approaches, as well as of different tracking fusion concepts, and for the specific task of **b)** visual quality inspection in industrial environments, especially for industrial robotic welding, including an overview of diverse existing measurement cues, and an overview on related unusual event detection methods.

3.1 Image-based Tracking

In the last decades visual object tracking has been a vital field of research in the computer vision community. Several applications such as surveillance, augmented reality or assistance systems which benefit from the progress made in this area, as well as different novel tracking methods, concepts and strategies haven been proposed. As the field of tracking is very broad and diverse, we classify related approaches into several groups and consider them separately here, followed by a discussion of existing relevant fusion

techniques. Similar to the central theme considering tracking in this Thesis we start with an overview of template tracking methods, followed by learning based approaches and trackers that allow for successful tracking of non-rigid objects, and conclude with different concepts for fusing trackers to obtain more accurate tracking results.

Template-based Tracking

Template tracking represents the simplest tracking approach where image templates or sub-regions represent the target object. The fundamental idea can be dated back to the method of Lucas and Kanade presented in [83]. In template based approaches the tracking task is typically solved by detecting and matching or by correlating salient image features in consecutive frames, followed by an estimation of a suitable transformation matrix. Difficulties that arise are thereby partial or even complete occlusions, rapid or constant illumination changes, and object appearance or pose changes. Furthermore, tracking over long sequences typically suffers from the well known drifting problem, which is discussed in detail by Matthews et al. [89]. In order to successfully handle these problems and complications, the tracking template needs to be updated in some way. The naive update approach is to directly assign the new location of the tracked object to the template within each frame. As discussed in [89], this will result in a drift away from the object over time due to, e.g., object appearance changes, environmental changes or tracking failures.

Black and Jepson [10] presented a template or view based tracking approach that relies on Eigenspace techniques. They reformulated the Eigenspace reconstruction problem to a robust estimation formulation and incorporated a subspace constancy assumption which allowed for coincidental computation of an affine transformation between the image and the Eigenspace and of the view reconstruction. By using a robust formulation for subspace matching they showed that they can extend Eigenspace methods the tracking problems including occlusions, background clutter and noise. However, in their experiments they applied their method for the application of hand gesture recognition as well as for tracking of very simple objects like, e.g., coke cans. Although, few samples on above described complications are illustrated, extensive performance evaluations are not given. Moreover, the approach can only be applied for tracking of previously view objects and according to the authors the approach is far from real time as it requires several seconds for the reconstruction approximation.

Benhimane and Malis [8] introduced a homography based approach to image based visual tracking and servoing. They proposed to use an efficient second order minimiza-

tion method (ESM) in order to estimate a homography between consecutive images without using computationally expensive Hessian matrices. Their approach relies on an iterative minimization technique based on a Lie Algebra parameterization. Further, they utilized the estimated homography transformation matrix to derive and control a visual servoing control law. The authors demonstrate that their method provides a high convergence rate, with the specific advantage of avoidance of getting stuck into local minima. However, the presented approach lacks in robustness as it is quite sensitive to illumination changes as well as partial occlusions. Moreover, the approach is especially designed to track planar surfaces or planar objects which is typically not the case in real world scenarios where objects normally undergo some non-rigid transformations or deformations.

Hinterstoisser et al. [55] introduced two template tracking related methods that are based on template or patch rectification. In their approach a random fern based classifier [96] is applied for keypoint transformation estimation, followed by a perspective rectification using linear predictors [67]. The first presented method favors run-time and performance, and hence relies on grouping of keypoint transformations into several classes. Evaluation of four angles using further random fern classifiers then identifies the most suitable pose. The second presented method favors real-time learning and robustness, and consequently relies on a nearest neighbor classification of so called mean patches that cover small local pose variations, respectively. A fast computation of mean patches followed by an incremental learning step thereby result in robust tracking methods. However, the limitations of the presented approach are the underlying feature point detection as well as computationally expensive off-line preparatory learning tasks. Further, the authors state in their experimental evaluations that the methods are not applicable for long term tracking.

A recent template or patch based tracking method presented by Bolme et al. [12] is based on adaptive correlation filters. The authors propose to use a so called Minimum Output Sum of Squared Error (MOSSE) filter for image correlation in order to obtain stable and compact response peaks, representing target positions. The tracking algorithm is initialized from a single image, and correlation is performed in the Fourier domain for run-time advantages. Failure detection is done by evaluation of the Peak to Side-lobe Ratio (PSR). The authors demonstrated the functionality and real-time capability of their method on several standard video sequences used in relevant literature. However, most real-world objects are inherently non-rigid or perform complex deformations, e.g., due to viewpoint changes or out-of-plane rotations which are hard to cope with using tem-

plate based tracking approaches. Although, the authors state that the approach could be extended to estimate scale and rotation changes, this has not been done or evaluated, leaving room for improvement.

Another recent tracking approach that addresses direct visual tracking, thus finding the best match to a given image under consideration of all image pixel intensities in another image, is presented by Scandaroli et al. [107]. The proposed solution relies on the normalized cross correlation (NCC) as similarity measure. They explicitly address local illumination changes, specular reflections and partial occlusions, and derive a novel solution for the gradient transformation parameter estimation problem using inverse and forward compositional approaches. Experimental evaluations show that the improvements can handle partial occlusions even under severe illumination changes. However, the presented experimental results are obtained from tracking planar objects that undergo some linear transformations which is not a realistic scenario. Moreover, the approach cannot robustly handle large rotations or scale changes as can be seen from sample tracking images presented in the paper.

Most of the above presented approaches are highly applicable to the task of image template based tracking. However, the consideration of more complex objects that do not exhibit mainly planar surfaces, of non-rigid object transformations, and of typical real world tracking complications like occlusions, noise, dynamic scenes, or clutter is entirely missing.

Learning-based Tracking

Another branch of tracking approaches is based on on-line learning of different object representations, thus allowing for adoptions to, e.g., appearance changes over time. The patch-based representation of the above mentioned tracking methods presented by Hinterstoisser in [55] can thereby be seen as a hybrid approach between template tracking and learning based tracking.

Another method that is based on both, template tracking and on-line learning is the linear predictors approach presented by Jurie and Dhome in [67]. The approach is based on linear motion predictors and low order parametric models for image template motion. The approach relies on a hyperplane approximation instead of computationally expensive Jacobian approximations which is precomputed and further used dynamically, resulting in increased runtime performances. In the experimental evaluations the authors evaluate different linear transformations on few synthetic and real world scenarios and demonstrate that the hyperplane based approach outperforms the original

Jacobian approximation. However, the evaluations also showed that the approach is not robust to large rotations or large baseline motion. Moreover, only planar objects and surfaces which are rotated around the corresponding object center for the synthetic case are evaluated. More realistic scenarios including non-rigid objects, more complex transformations or any kind of tracking complications are neither considered nor evaluated.

A more recent on-line learning based template tracking approach is presented by Holzer et al. [59]. In their work they present a novel reformulation of the above discussed linear predictor approach presented by Jurie and Dhome in [67], allowing for on-line learning of linear predictors. They reduce the computational complexity of the original inverse-compositional tracker and additionally incorporate a multi-predictor approach that considers different parameter changes, respectively. The presented experiments show that the approach can be used for template tracking with varying template size, thus allowing the tracked template to grow over time. However, this can also be a disadvantage if objects with similar appearance are tracked. The authors also state that the approach is not robust against occlusions, which gives room for further improvements.

Javed et al. [63] and Avidan [3] were the first that used on-line learning for object detection and tracking. While Javed et al. [63] used on-line AdaBoost and holistic PCA based features, Avidan [3] already performed pixel-wise classification and used Mean-Shift to find the current object position. Additionally, to overcome the bounding box limitation, he incorporated a rejection scheme for pixels that cannot be classified. However, experiments and evaluations have been conducted on rather simple and short sequences giving a proof of concept but not demonstrating robustness and applicability to real world scenarios including noise, distortions or other complicating factors.

Based on the fundamental concept of Viola and Jones [124], Grabner et al. [47] used on-line AdaBoost to learn the object's appearance during runtime. They represented the object as a rectangular bounding box with a fixed aspect ratio and randomly placed Haar-like features. However, the bounding box limitation also introduces a large amount of noise as the objects typically do not cover the entire box. Moreover, Ada-Boost is very prone to noise, which increases the impact of errors during training and consequently also the possibility of tracking failures.

Leistner et al. [76] used different loss functions to improve the robustness of the Boosting algorithm. This concept has been further extended to semi-supervised learning by Grabner et al. [48], and to multiple-instance learning by Babenko et al. [4]. However, the bounding box limitation is still given in all these approaches.

Saffari et al. [105] used on-line Random Forests instead of Boosting, as they are known to be more robust to noisy training data. While this issue increased the robustness of the learning algorithm, the tree-growing scheme anneals the structure of the trees over time which makes adoptions to the current scene harder in long sequences. Godec et al. [88] further showed that even simpler structures than random forests or ferns could be used, yielding similar performances. They relied on a random naive Bayes classifier and adapted it for on-line learning using on-line random feature selection and histograms as weak learners in order to establish a robust and fast classifier. They empirically showed that this concept is applicable both, for incremental learning and tracking by detection. Nevertheless, tracking complications and noise, or dynamic environments are not considered, leaving room for further improvement, as it is not clear if the presented methods can cope with e.g. significantly large amounts of image noise.

Jiang et al. [64] recently presented a visual tracking method that addresses a solution to the structural order determination problem in metric learning based on sparsity regularization for metric learning. The aim of metric learning is to adaptively adjust a matching metric, which in this case is used for visual tracking, by projecting the actual features from their specific feature space to a new metric space, where the discrimination between target and candidate is maximized. Thereby the determination of the optimal order for the best metric adjustment meaning the ideal dimensionality of the new metric space defines the crucial factor. Jiang et al. proposed to use sparsity regularization for this task and showed that for visual tracking this leads to improved performances compared to other metric learning based tracking approaches [119, 130].

Although, the all presented approaches achieve impressive tracking accuracies, the bounding boxes that are utilized by most trackers introduce large amounts of noise and background during on-line learning. More accurate object descriptions given by, e.g., object contours, silhouettes, or segmentations would definitely improve the performances. Moreover, most approaches require some kind of off-line learning or preparatory tasks, which might not be suitable for many real-world applications.

Tracking of Non-Rigid Objects

Although, recently presented template trackers are already partially able to track objects changing their appearance or even highly dynamic objects, they only deliver a rough description of the object in form of a rectangular bounding box. Therefore, there is a recent trend to learn part-based object models to better cope with the corresponding

tracking complications.

Park et al. [97] recently presented a robust visual tracking approach based on autoregressive Hidden Markov Models (HMM). In their work they analyzed the probabilistic dependencies between consecutive object appearances within a learning phase, where target samples are clustered based on visual similarities. For on-line tracking, multiple appearance models are learning in terms of cluster specific classifiers. The best suitable appearance model is then determined by inferring the most probable model under consideration of model dependencies in the past. The approach is evaluated on several challenging scenarios. However, the proposed tracker is not real-time capable as the on-line learning stage requires on average several seconds per frame.

By using small blocks of appearance and shape descriptions Nejhum et al. [109] proposed a flexible part-based tracking model. They optimized the placement of the blocks during tracking but do not update their appearance. Thus, their approach is quite sensitive to appearance changes of the object over time. Utilization of object segmentations and therein located appearance blocks for tracking updates would definitely increase the overall performance.

However, approaches that directly use segmentation for tracking either require prior knowledge [31], which may not be available for the target objects, or perform some kind of off-line processing [49, 120] hampering, e.g., the tracking of unknown objects. Moreover, real-time capability is an issue that most segmentation or energy minimization approaches struggle with.

Another strategy for tracking of non-rigid objects is to utilize active contours approaches like e.g. level sets or snakes. An example would be the dynamic geodesic snakes approach by Niethammer and Tannenbaum [94]. They proposed a natural level set based approach for dynamic curve evolution which is based on an energy minimization functional that allows for integrating dynamics into the geodesic active contour framework. The method incorporates tracking state information in terms of normal velocity for every particle on a given contour, allowing for estimating velocity and position. Although, partial occlusions are considered as tracking complication in their formulations, only two very simple tracking scenarios are evaluated. Moreover, it is not clear if the approach is real time capable as information on runtime performance is not given.

Roth et al. [104] presented yet another energy minimization based tracking approach, based on learning of Gibbs distributions in a Bayesian tracking framework. They show how Gibbs energies can be effectively utilized as image likelihood, allowing for particle filter based tracking of humans. The approach relies on learning of a large training

dataset, which in turn allows for building a likelihood model that does not over-fit the distributions. In their experiments the authors show that the approach can be applied for tracking even in the presence of camera noise, while achieving better results than standard particle filtering based trackers. However, more extensive experiments in terms of non-rigid object tracking or handling of different tracking complications should be made as it is not clear how well the approach can be applied to tracking problems, other than human tracking. Moreover, no information considering runtime performance neither for the off-line training nor for on-line tracking is given, inducing that the approach is not real-time capable.

Bibby and Reid [9] overcome the runtime performance problem within a probabilistic framework. To handle the complexity of their theoretic framework and to allow for real-time tracking they separated the tracking of non-rigid objects into registration, level-set based segmentation, and on-line appearance learning. However, their appearance model is very simple and may not cope with complex objects and transformations.

To retrieve a more fine grained result, Godec et al. [45] used a subsequent segmentation step. They combined a large number of small parts in a voting style manner and used points with stable geometric relations to initialize a segmentation process. Besides the more appealing visualization of the result, the segmentation also improves the update process by decreasing the number of false positives. However, the segmentation is only used as a post processing step and does not respect a temporal smoothness of the object deformation, which may cause large over- or under-segmentations.

A similar approach has been proposed by Fan et al. [35], which used image matting to generate a more fine-grained object description. They combined salient image points, discriminative colors, and bag-of-patches to include short-term, mid-term, and long-term object appearance in their model, respectively. However, the model is based on heuristics to update the individual representation cues, which might result in overall tracking failures even if a single part of the model fails.

Another recent approach denoted as local orderless tracking (LOT) is presented by Oron et al. [95]. The proposed algorithm estimates the amount of local order in the object, allowing for tracking both, rigid and deformable objects in an on-line fashion. The underlying local orderless matching measures the similarity between two images based on the Earth Mover's Distance (EMD), attempting to explain a set of pixels or features as a noisy replica of another reference set. In a tracking context the local orderless matching is applied in a Bayesian tracking formulation using particle filtering for finding the unknown noise parameters. As solving the EMD problem is computation-

ally expensive, the approach further relies on super-pixels. The presented experimental results are comparable to other state of the art approaches, but the method again only considers bounding boxes instead of more accurate object descriptions.

Although, the presented approaches can successfully handle very challenging tracking scenarios and complex object transformations due to quite smart on-line learning techniques or the capability to adopt for appearance changes, still a significantly large amount of noise is introduced during on-line updates as the common object description is given by bounding boxes which are typically not entirely filled by the tracked object. Moreover, tedious off-line learning or preparatory tasks are necessary to reach high tracking accuracies presented in the discussed approaches. This might not be feasible for many real world applications or scenarios where, e.g., the object to be tracked might not be known a priori.

Fusion Concepts

To overcome the shortcomings of existing individual tracking methods, recently more complex scenarios and non-rigid object transformations are handled by a combination of several heterogeneous trackers. In fact, multiple observations or measurements can significantly improve the overall tracking performance. Usually, such a fusion of observations is either handcrafted and based on heuristics [68, 106], or is based on simple combinations of a large number of similar trackers [72, 83, 109]. Fusion of multiple different observations, cues or classifiers has a long tradition also in other disciplines. The research can be coarsely divided into two groups: **(a)** Classification fusion, where different cues are evaluated separately and the obtained decisions are fused (late fusion), and **(b)** Feature fusion, where different cues are combined and only relevant features are selected (early fusion). For a detailed survey we would refer to Mangai et al. [123].

Kwon and Lee [74] used a dynamic number of templates to describe the individual object parts. During tracking, they model the geometric relation of the parts and apply Basin Hopping Monte Carlo (BHMC) sampling to reduce the computational complexity. However, the overall performance of the approach is not satisfactory as it is far from real time. Moreover, the minimum size of the object is limited which can also be a disadvantage.

Cehovin et al. [24] split the tracking problem into two layers, consisting of global and local object models. The local model is given by a set of patches that adapt to the objects geometric deformation while the global model adapts to the overall appearance and adds or removes local patches. The overall tracking result is then given by the

convex hull of the local parts, which may not correspond to the actual object. Thus, an incorporation of only few local parts that are wrongly transformed might quite rapidly result in global tracking failures.

In [106] Santner et al. address the stability plasticity dilemma by combining three trackers having different adaptivity characteristics. Although, the basic concept of combining different trackers to cope with diverse facets of the tracking problem is good, the fusion of the trackers and of their individual outputs is based on a heuristic cascade and thus cannot be generalized to other tracking approaches. Moreover, it is not clear how the tracker can be extended to cope with, e.g., non-rigid motion or large significant pose variations.

Stenger et al. [113] recently investigated different combinations of tracking methods, where they try to learn which tracking approaches are useful for a given tracking scenario, and share how these trackers can be successfully combined. Again the basic fusion concept is elaborate, but the approach is based on an off-line training of possible combinations, which is sometimes not feasible in real-world applications.

Similar concepts haven been addressed in [5, 60, 91, 98, 129]. In contrast to [106] these approaches rely on parallel evaluation of different observers, followed by a combination of the individual outputs in terms of a late fusion. Thereby, [5] relies on an output combination concept that simply switches between different reported measures, depending on the individual performances. On the other hand, the approaches presented in [60, 91, 98, 129] combine individual outputs based on probabilistic concepts. The approaches assume that the trackers report a probability density function (PDF) or that a transformation of the outputs to a PDF is given. Thus, the most challenging task is here to obtain good confidence measures or probabilities for each contributing cue. However, different metrics are not considered.

Another related concept is the combination of a large number of similar measures or cues like, e.g., [24, 74, 109] discussed above, which are based on small patch-based trackers. Kalal et al. [68] proposed the combination of an adaptive Lukas Kanade tracker [83] for short-term tracking with a conservatively updated Random Forest for long-term re-detection. The final detector is thus based on a bounding box and is not well suited for non-rigid objects as a large amount of noise and background is still present within the reported bounding boxes.

Kwon and Lee [73] recently presented an extension of the general particle filter framework. In their visual tracking decomposition approach they utilize motion and observation models to explicitly cope with significant appearance changes caused by

pose, scale, or illumination variations as well as partial occlusions. However, the presented method does not perform an object segmentation or a labeling which results in considerable amounts of noise and background clutter that are incorporated during on-line model updates.

In contrast to that Wang et al. [127] presented a tracking method that relies on mid-level vision cues in terms of discriminative appearance models based on super-pixels. They initially train a discriminative model in order to distinguish between object foreground and cluttered background image regions. During on-line tracking they compute a confidence map based on a maximum a posteriori estimator, giving the actual most likely target location. The appearance model is then updated from obtained super-pixel segments around the target location, allowing for robust tracking in the presence of large appearance changes, shape deformations, occlusions and drift. Although, the approach reduces the amount of noise that gets incorporated during the on-line updates. However, the super-pixel segments are not accurate enough to completely eliminate the noise and background regions.

Markovic and Gelautz [87] presented a fusion-based approach for image segmentation, which in turn could further be utilized for robust segmentation-based object tracking. They combine image edges from intensity images with the locations of discontinuities in stereo-derived depth maps, and perform an active contour model based segmentation. In their experimental evaluations they showed that the fusion of the different features results in improved segmentation results in different real-world scenarios.

Another approach that is related to fusing different cues for tracking is given by the concept of modeling an object as a flock of features, presented by Hoey [57]. The author proposed a particle filter based on flocks of features for tracking objects under occlusions and distractions. Thereby, a flock is given by a loose collection of features which are moving independently, while still maintaining a consistent motion. The authors used so called color specks as underlying features, but denote that any other kind of feature could be used. Although, experiments including occlusions and object deformations are presented, hand tracking in hand washing sequences is not a convincing application. Further investigations on non-rigid object tracking in highly dynamic scenes or on how large amounts of noise could be successfully handled are entirely missing. Moreover the present approach lacks in comparison with other state of the art tracking approaches, and it is not clear if flock of features could be used for real-time tracking if more complex features are used.

A more recent approach presented by Matas and Vojir [125] aims in robustifying

the flock of trackers concept presented by Kalal et al. [69], which relies on different predictors used as trackers in a flock of trackers fashion. In their work Matas and Vojir introduced a so called cell flock of trackers that allows local trackers to drift to points that are good to track, as well as different tracking failure predictors relying on neighborhood consistency and past performance of individual trackers. Although, a real-time capability of the presented approach is demonstrated, it is not clear if the method is able to cope with tracking complications like large occlusions, significant amounts of image noise, or non-rigid object deformations, which are more realistic in terms of real-world object tracking.

Although the general concept of fusing different observations to obtain higher tracking accuracies is good, state of the art methods are not satisfactory, because the trackers are either coupled too tightly (e.g., by a cascade [106, 140]) or all trackers are run independently and only a late fusion is applied. Thus, the individual trackers do not benefit from each other. Obviously, it is reasonable to combine different tracking cues as the goal is to combine advantages of diverse approaches while compensating for individual weaknesses. However, this is not a trivial task due to diverse typically not directly combinable outputs.

3.2 Visual Quality Inspection

Within this Section we provide an overview of quality inspection methods for industrial robotic welding tasks as well as on unusual event detection approaches in computer vision.

Robotic Welding Quality Assessment

Industrial robotic welding is an important and widespread process in industrial production and especially in the automotive industry, which forms a core field of application for methods and algorithms presented in this Thesis. With a continuously increasing degree of automation in such a process, the aim is also to automatically evaluate the overall welding quality and to classify weldings, welding processes, or specific weld seam regions into either defective or error-free, if the simplest binary one-class-classification approach is considered. Thereby, an on-line classification during the welding task itself represents the ideal solution as defective weld seam regions could be, e.g., automatically repaired or the concerned specimen could be automatically sorted out without additional costly manual handling or inspection tasks afterwards.

Previous work in the field of weld seam quality analysis mainly relies on non-visual information like voltage, current, welding arc sound or brightness fluctuations [110, 126, 136]. The few works on visual inspection approaches primarily track the seam in front of the welding torch in order to correct the welding arc position. However, the final product, namely the welded seam itself, is not considered for additional quality assurance. In the following we present the few image based quality inspection approaches that we could find in relevant literature.

Ma et al. [84] introduced a seam tracking system based on a single camera, which is rigidly connected to the welding robot. In their approach the seam in front of the welding torch is tracked in order to continuously correct the welding arc position and to realign the robot to the seam center in case of deviations. The presented tracking approach thereby solely relies on image based edge detection and some geometric constraints for the alignment step. Evaluations are performed on a single straight weld seam by measuring the normal offset or gap between welding torch and tracked seam center as the crucial factor. However, the approach does not consider an inspection of the welded seam behind the welding torch. Furthermore, the conducted experiments do not consider typical image distortions like smoke wads, sparks or gas disturbances that usually occur in industrial welding environments and that make tracking in such a context really challenging.

Another approach also related to the above presented seam tracking concept is presented by Yan and Xu in [134]. The authors also introduced a system for automatic positioning of the welding torch during the welding process. The system consists of a camera that is rigidly connected to the welding robot and that observes the seam in front of the welding torch. The presented algorithm relies on extraction of image edges as well as of straight lines, in order to accurately detect the seam in a small sub-image in front of the welding torch. The conducted evaluations include deviation measurements of a straight welding process in order to proof their concept. Although, the approach seems to work for straight weldings, the geometric restrictions in terms of the additional straight line detection constraint significantly limit the applicability of the presented approach.

Xu et al. proposed a vision based sensor for seam finding and subsequent seam tracking in [133]. The presented sensor consists of a camera, a laser diode with a rotating lens for circular beam generation, and a narrow band interference filter that is used as scanning lens. The lens filters ambient light emitted from the welding arc as it provides a narrow bandpass of $0.99nm$ centered at an experimentally determined wavelength of

one arc light summit. The camera is rigidly connected to a welding robot and observes the seam in front of the torch. Here, the circular laser beam is projected through the scanning lens, followed by an image based analysis of the segmented laser lines in order to provide information on the actual seam position as well as on it's 3D shape. The method has been evaluated on two different weld seams representing two different seam shapes. However, the approach is again suited only for automatic welding and not for weld seam quality inspection as the welded seam is again not analyzed.

Schreiber et al. [108] presented a vision based weld seam quality inspection approach. They introduced a monocular tracking system with a camera that is rigidly connected to a welding robot. Other than in the previously presented approaches the camera observes the welding torch and newly welded seam directly behind the welding torch. The tracking approach is based on an estimation of affine transformation parameters between matched feature points in consecutive image frames. Considering weld seam quality analysis they compare local light distribution measurements around the welding torch, the weld seam width, and the weld seam position to manually annotated ground truth data as crucial factors. Their evaluations include measurements on two curved welding datasets, consisting of about 400 image frames each. Although the general concept of combining different measurement cues for welding quality assessment is good, the evaluations on solely two weldings is not sufficient as the variety of different weldings, welding processes and welding defects is not covered at all.

Another recent work that focuses on on-line monitoring and industrial quality inspection is presented by Fecker et al. [38]. The authors present a Bayesian adaption algorithm for creating an imperfection model for imbalanced training data sets. In their experiments on laser brazing images they show that the algorithm reaches performances comparable to the results that would be obtained in case of balanced training data. Thus, the underlying semi-supervised problem formulation is successfully transformed to a supervised one in terms of performance for the presented application.

Fennander et al. [39] introduced an optical system that automatically analyzes the regularity of the electric arc frequency and filler metal droplets in hybrid welding processes. Thereby, irregularities in the electric arc frequency are detected by fuzzy c-means clustering on image histograms. The filler metal droplet localization is performed using support vector machines (SVM) for a classification in combination with the principal component analysis (PCA). Droplet tracking is done with a Kalman filter [131]. The complete system has been trained on a subset of image sequences that exhibit similar image quality and droplet appearance. Hence, the system works as long as sequences without

large deviations or abnormalities are analyzed, resulting in necessary time-consuming training steps for individual welding processes.

In general limitations of state of the art industrial welding quality assessment approaches are still given by missing weld seam quality inspection during the welding in an on-line fashion. Most of the above introduced approaches are applied in an off-line manner or are not real-time capable. Thus, the quality inspection is performed in a separate and consequently time consuming costly task after the welding is finished. Moreover, experimental evaluations were mostly conducted on very few in most cases straight weldings. However, the most crucial shortcoming is given by the missing consideration of industrial noise, distortions, or environmental influences that are typical for industrial environments. Hence, from this point of view there is still a lot of room for improvements given.

Unusual Event Detection

Another topic in the field of computer vision that is also somehow related to the task of weld seam quality inspection is unusual event or outlier detection. Considering an algorithm that allows for continuously extracting images of newly welded seam, comparisons with an ideal model or with error-free welding images can be seen as an unusual event detection approach for industrial welding tasks. Unusual event detection, outlier detection or anomaly detection refers to the problem of detecting images, image regions, or patterns in a given data set that do not match to typical appearance or behavior. The techniques suitable and applied to this problem include image based classification, clustering methods, nearest neighbor approaches, information theoretical methods, or even spectral analysis. Typical unusual event detection methods thereby use large amounts of training data and in the majority of cases apply on-line learning techniques in order to adopt to new observations or unseen object poses. For a detailed survey on different outlier detection methodologies, techniques and applications we would refer to the works of Hodge and Austin [56] and of Chandola et al. [25].

Within the following overview on related work considering outlier detection we especially focus on one-class classification (OCC) as the aim of later on presented methods for welding quality assessment aims at separating unseen test images into defective or erroneous, and into error-free, thus defining a binary classification problem. Unusual event detection or one-class classification (OCC) aims at classification of input data into *usual* and *unusual* events. Thereby, we can roughly distinguish two approaches to this problem, namely **(a)** methods that are based on constant and previously trained models

of *normality* (e.g., [62, 65, 86, 139]), and **(b)** methods which try to adapt to new observed scenes in an on-line fashion or in real-time (e.g., [15, 99]).

In [15] Breitenstein et al. proposed to learn a model of normality by observing a scene with a static camera. New observations are then classified either as statistical outliers or as normal events, resulting in further adoptions to the existing model. The method was developed for natural scenes, where the normal activities exhibit a large variability. In contrast to this assumption industrial manufacturing tasks like robotic welding are highly repeatable. Hence, regular deviations are rather small in the error-free case, allowing to detect potential outliers at a more fine-grained level.

Unusual event detection without adaption during run-time is applied to a welding quality inspection task in [62] by Jäger et al. The method utilizes Hidden Markov Models (HMM) to account for the problem of weakly labeled training data sets. The authors apply the presented method in laser welding sequences in order to detect possible irregularities, where a camera monitors the emitted laser welding radiation. The quality inspection computations are then applied in an off-line manner, i.e., after the welding process has been finished, and has been evaluated on roughly 1000 welding image sequences, providing a good quantitative evaluation considering the variability of laser weldings.

Kenner [70] introduced a generic defect and outlier detection system for industrial applications in a more general context. The system is based on one-class classification and outlier detection. One-class training data is thereby learned and further adopted on-line with the well known AdaBoost [43] algorithm. In this way, a strong classifier is obtained from several weak classifiers for the outlier and defect detection task. Although, the system is designed in a very general context, robotic welding tasks and the visual inspection of weld seams could also be a field of application for the presented system.

Another even tracking related approach to unusual event detection has been presented by Ivanov et al. [61]. In their approach the authors rely on velocity and acceleration which are measured in terms of trajectories obtained from a segmentation and tracking algorithm presented by Cavallaro et al. [23]. The trajectories are then used to train a SVM classifier, as well as for further testing on more than a hundred trajectories obtained from different real-life scenes. Preliminary results showed that the proposed system is able to detect unusual trajectories from videos while keeping false alarm rates low.

4 Template Tracking in Harsh Industrial Environments

In this Chapter we present an appearance based tracking approach that is especially designed for tracking newly welded seam in harsh industrial robotic welding environments. Thereby, we assume large amounts of visible noise caused by, e.g., smoke wads, evaporating water, gas disturbances, or sparks and spilling. In our considerations we especially focus on high robustness, on a minimum of required parameterization or off-line preparatory tasks, and on a maximum runtime performance. Although, there exist various tracking techniques in literature, we rely on image templates for tracking as methods based on, e.g., probabilistic models [73], discriminative approaches [45], or kernel based tracking methods [28] are either prone to fail due to an undesired sensibility to the present image noise, or they do not conform with the mentioned real-time and parameterization requirements. Thus, we propose a prediction correction based template tracking approach for solving this tracking problem, as appearance based tracking is robust to image noise up to a certain degree, while the number of parameters is manageable and while remaining real-time capable. Figure 4.1 illustrates some tracking results obtained with mentioned state of the art tracking approaches for the task of weld seam tracking, demonstrating the discussed problems.

4.1 Robust Template Tracking in Robotic Welding

Robotic welding is an important and wide spread process in industrial production and automotive industry. It is fast, cheap, and accurate. Although, the degree of automa-

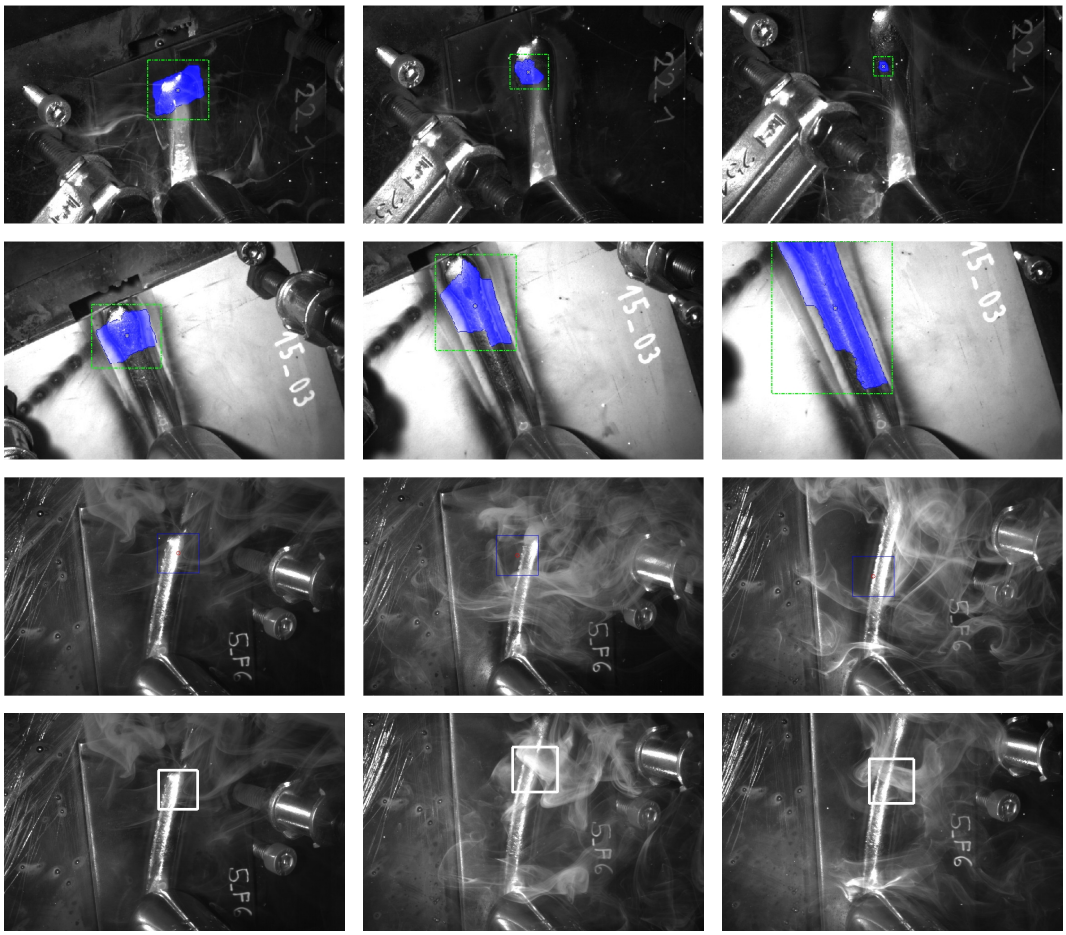


Figure 4.1: **Tracking in Harsh Industrial Robotic Welding Environments:** Discriminative Hough-based tracking [45] (upper two rows) either looses the target or results in undesirable over-segmentations due to similar foreground and background structures. Mean-Shift tracking [28] (third row) and Visual Tracking Decomposition [73] (bottom row) both cannot cope with the large amounts of image noise, resulting in a drift away from the specimen.

tion in such a process is high, quality assessment of welded seams is still mostly done manually by experts due to insufficient robustness of existing inspection methods. We present a robust weld seam tracking application that follows newly welded seam direct behind the welding torch, allowing for extracting axis aligned weld seam image patches for further quality assessment tasks. The challenges for robust weld seam tracking are thereby given by an unknown welding robot motion as well as by unknown welding trajectories, by typically large amounts of image noise including bright sparks, heavy smoke or evaporating water, and by appearance changes of the newly welded seam that

slightly morphs while cooling down. Figure 4.2 shows weld seam image patches from diverse welding processes to give an idea on the noisy and harsh environments that need to be coped with.

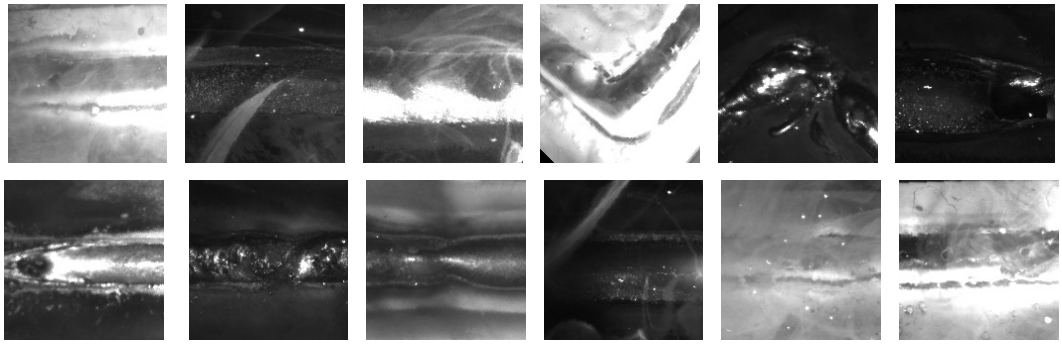


Figure 4.2: **Challenging Weld Seam Image Patches:** Diverse welding process configurations result in challenging welding scenarios that the tracking application needs to cope with, including smoke wads, bright sparks, specular reflections, evaporating water, or unforeseen welding defects.

4.1.1 Image Acquisition Setup

The *Q-Eye* image acquisition system by Fronius Ltd[1]. is a novel industrial imaging system developed for an on-line observation of newly welded seam during welding processes. The system enables new possibilities in process observation especially for capped manufacturing cells as they are used with laser hybrid systems. Of course it can also be used in any other arc welding application, where e.g. a welding engineer is endangered by the robot or high temperatures. The system consists of a camera unit, a robot proof hose-pack and a power supply unit, which realizes a high-voltage supply of a stroboscope assisted illumination unit. This allows for eliminating the emitted arc radiation, while coincidentally illuminating newly welded seam behind the welding torch. The system is rigidly mounted directly on the welding robot, and located next to the welding torch. This enables a good view of the newly welded seam while remaining protected against spatter and arc radiation. As welding is typically accompanied by high temperatures and as the high-power flash light unit also requires cooling, the system also includes a cooling circulation, where the generated cooling air flow is additionally utilized to protect the safety glasses of the flash light and camera units, respectively.

[1]http://www.fronius.com

The monochrome camera unit captures $752 \times 480px$ gray scale images, where the stroboscope gets simultaneously triggered every time an image is recorded. This results in the region behind the gas nozzle showing the weld pool and newly welded seam being highlighted. The system allows for recording images at up to $20fps$ at typical welding speeds of $50 - 100cm$ per minute. Figure 4.3 depicts the rigidly mounted acquisition system and a corresponding view.

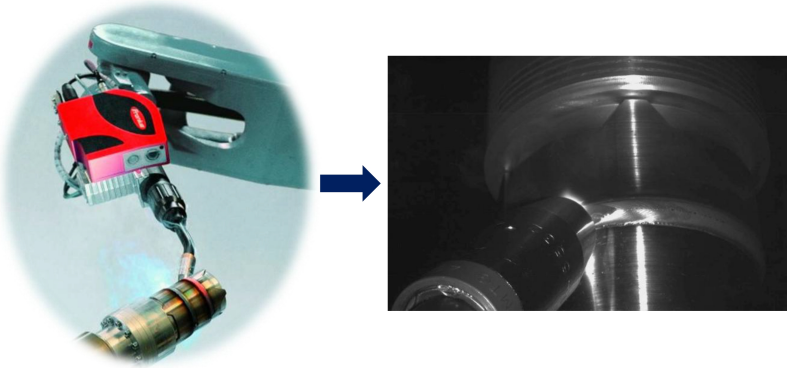


Figure 4.3: ***Q-Eye* Acquisition System:** The *Q-Eye* camera unit is rigidly mounted next to the welding torch, enabling a good view of newly welded seam. An exemplary view is shown on the right.

Considering the geometric relations of the setup, the working distance w between the camera unit and the observed weld seam amounts to $250mm$. The camera's focal length f is given by $12mm$ and the baseline between two consecutive frames amounts to approximately $8mm$ for an acquisition frame rate of $20fps$. Figure 4.4 illustrates these geometric relations between two consecutive camera views.

4.1.2 Weld Seam Tracking

The overall aim of our proposed weld seam tracking algorithm is to localize newly welded seam behind the torch in a sequence of images. Our proposed algorithm that solves this problem consists of four consecutive steps, applied at each incoming image. These are robust template matching, spline-based regularization, prediction of new weld seam points along an interpolated spline curve, and an appearance based correction. Although, the template matching could be easily replaced e.g. by any feature based registration approach, template matching exhibits essential runtime advantages. Moreover, image templates allow for a more detailed description of an object than e.g. local image features, as each pixel and its arrangement within the template can be seen

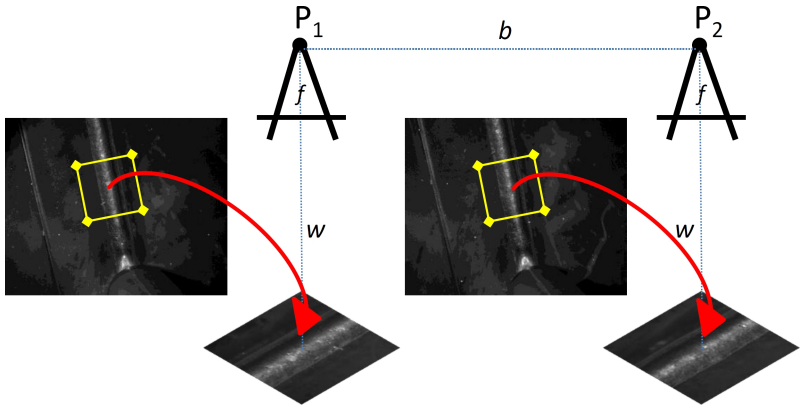


Figure 4.4: **Geometric Welding Image Relations:** Weld seam image patches are extracted at an approximately perpendicular view angle from consecutive camera views $\mathbf{P}_1$ and $\mathbf{P}_2$. This allows for neglecting an additional camera calibration or the consideration of computationally expensive image undistortion.

as specific feature. Local feature based approaches are prone to fail, especially in the presence of significant amounts of noise. Reasons could be too few or even missing detector responses or too many matching outliers resulting in wrong registration results. Figure 4.5 illustrates the proposed tracking method, its specific steps, and the data that is extracted from each image.

To estimate the relative scene motion between two images, image templates depicting the weld seam in the previous image $\mathbf{I}_{t-1}$ are matched with the actual image $\mathbf{I}_t$. The template matching results in a set of n support points at time t

$$\mathbf{S}_t = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \quad \text{with} \quad \mathbf{x}_i = (x, y)^T \,, \tag{4.1}$$

which exhibit high probabilities for lying on the tracked weld seam each. Within a subsequent regularization step a smoothing spline is fitted to the point set $\mathbf{S}_t$, as a spline best represents typical weld seam geometries, including curves or bending. Moreover, the spline is constrained to pass through the welding point $\mathbf{x}_w$, which exhibits a fixed position in the image due to the rigid connection between camera and welding robot. In order to overcome the well known drifting problem in template based tracking approaches [89], the spline gets corrected point-wise towards the center of the weld seam using a weld seam appearance model. Finally, we predict weld seam regions for the next image $\mathbf{I}_{t+1}$ by extrapolating support points at specific positions on the corrected spline.

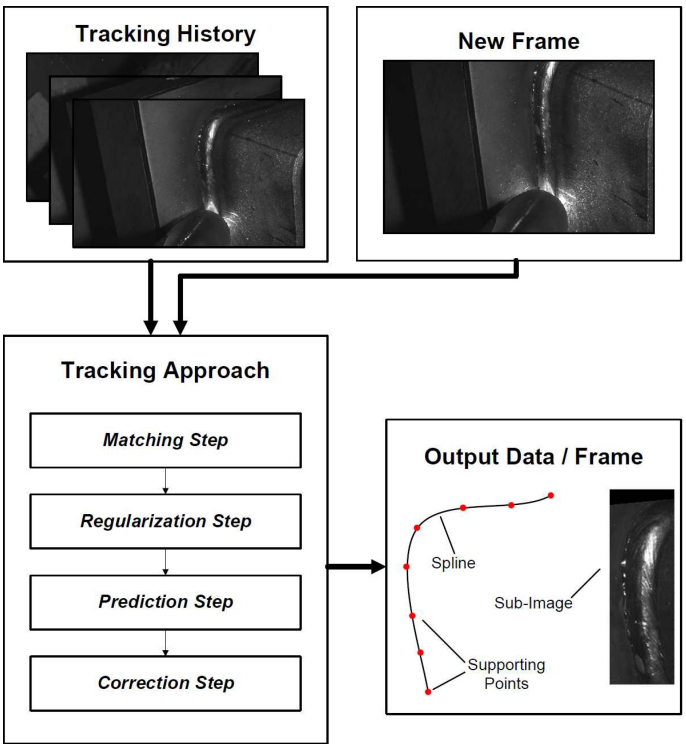


Figure 4.5: **Weld Seam Tracking Approach:** The proposed tracking approach is based on robust template matching of weld seam image patches, a spline-based regularization, prediction of weld seam points along an interpolated spline curve, and a final appearance based correction step.

Template Matching

We apply a robust template matching method in order to re-locate weld seam points $\mathbf{S}_{t-1}$ from a preceding image $\mathbf{I}_{t-1}$ in an actual image $\mathbf{I}_t$. Thereby, challenges like unforeseen inter-frame illumination changes, sparks, smoke, or local shape and appearance changes caused by cooling of the weld seam need to be considered. We define a weld seam templates as image patches located around weld seam points, respectively. They are chosen in a way such that the weld seam covers approximately 40% of the template, whereas the remaining area depicts background structures. For template matching any similarity metric $\mathcal{S} : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$ that allows a pairwise comparison of images might be used. For our specific task, the two dimensional normalized cross-correlation $\mathcal{NCC}$ [79], which is commonly used in computer vision and which is simple, fast, and more robust to lighting changes than the normalized sum of squared differences $\mathcal{NSSD}$, the normalized sum of absolute differences $\mathcal{NSAD}$, or the normalized mutual information $\mathcal{NMI}$ metric, performed best. Although, the $\mathcal{NCC}$ similarity measure is robust to illu-

mination changes, it is not invariant to rotations or scale changes. To account for image rotations we correlate the actual image $\mathbf{I}_t$ with rotated versions of each template, where the rotations are given by a discretized angular interval. The best correlation results are finally chosen as correct matches. A multi-scale approach based on e.g. scale space template pyramids could be used to additionally account for scale changes. However, due to the rigid connection between camera and welding torch and for runtime performance reasons we assume the tracked object to be depict in an image sequence at approximately equal scale.

Spline-based Regularization

In order to avoid the tracked templates moving in adverse directions, thus resulting in tracking failures, the naive template matching further needs to be regularized. Template matching results in a sparse representation of the tracked weld seam, given by a set of image center points located on the tracked seam. In order to regularize these sparse results, we fit a cubic smoothing spline through the point set, allowing for modeling a large space of possible object shape deformations without making any assumptions on the underlying geometry, material, or deformability. Due to welding limitations in curve angles and because the weld seam is typically not supposed to run in unforeseen zig zag course, a cubic spline function is sufficient for these purposes. The cubic smoothing spline function $\mathcal{F}(t)$ is given via minimization of

$$min\left\{ p \sum_{j=1}^{n} \mathbf{w} \left| y - \mathcal{F}(x) \right|^2 + (1-p) \int \lambda(t) \left| \mathcal{F}(t)'' \right|^2 dt \right\}, \tag{4.2}$$

where the term $\mathbf{w}$ denotes specific spline node weights. Nodes in our course are the elements of the given point set $\mathbf{S}_t$, weighted by their individual correlation results. λ defines the applied weighting function, and p is a smoothing parameter, determining the relative weighting of the spline bending energy and the nodes. Obviously, a large set of supporting points $\mathbf{S}_t$ would allow a more exact reconstruction of the tracked weld seam. However, due to visibility constraints and because of limited processing power, a reasonable number for the size of elements in $\mathbf{S}_t$ has to be found, thus enjoining a trade-off between accuracy and runtime performance on the tracking approach. Figure 4.6 shows exemplary welding images with corresponding spline curves and supporting points used for their interpolation.

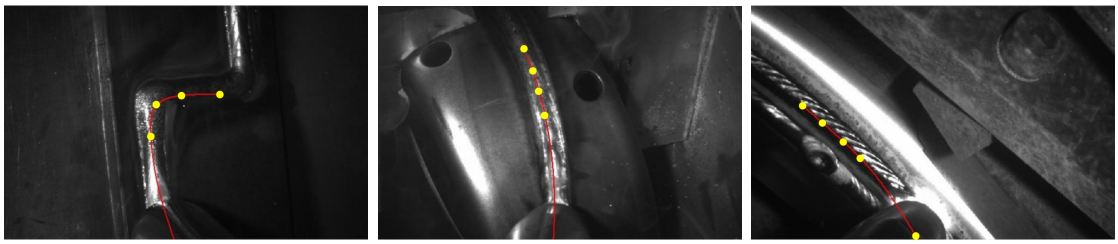


Figure 4.6: **Weld Seam Tracking Splines:** Exemplary welding images with corresponding splines and supporting points used for their interpolation. Depending on the visibility of the weld seam different numbers of support points are used.

Template Prediction

Due to the continuous but unknown robot motion weld seam regions will definitely leave the image sooner or later. Thus, we extrapolate new weld seam points located on the spline at given distances away from the fixed welding point $\mathbf{x}_w$, allowing for compensating the underlying robot motion. Figure 4.7 illustrates this compensation step. Template predictions are then obtained by extracting spline-aligned image regions around the new weld seam points.

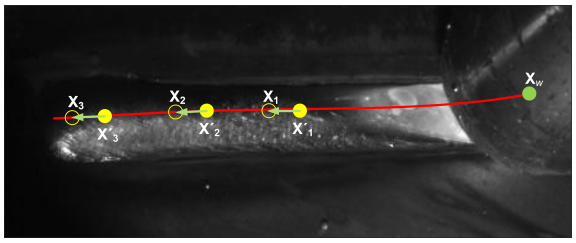


Figure 4.7: **Robot Motion Compensation:** A new set of weld seam points is computed within each tracking iteration by extrapolating spline points at fixed distances from the welding point $\mathbf{x}_w$, allowing for compensating the underlying robot motion.

Appearance-based Correction

As explained in detail by Matthews et al. in [89], adaptive visual tracking is subject to error propagation, which results in a drift away from the object of interest. This is especially true for generative or template based tracking approaches. In our specific case, the smoothing spline model might not accurately fit to the weld seam center due to a finite number of supporting points used for the interpolation. Consequently, this results in slightly inaccurate predictions, where an error would accumulates over time. To compensate for the drifting, we apply an appearance based correction of the predicted

templates, allowing for a repositioning of each template back to the weld seam center, based on similarity computations with a weld seam appearance model. Therefore, we perform a sliding window search in horizontal spline normal direction, using the $\mathcal{NCC}$ similarity measure. In a final step, new tracking templates are extracted around corrected weld seam center points. Algorithm 3 summarizes the presented robust weld seam template tracking algorithm.

Algorithm 3 Robust Weld Seam Tracking:

Input: actual image $\mathbf{I}$, previous templates $\mathbf{T}_{old}$, spline template distances $\mathbf{d}$, correction model $\mathbf{M}$, welding point $\mathbf{x}_w$

Output: actual templates $\mathbf{T}_{new}$, tracked weld seam points $\mathbf{x}_i^c$

(a) Relocate templates $\mathbf{T}_{old}$ in $\mathbf{I}$ by correlation-based template matching, giving tracked template center points $\mathbf{x}_i$

(b) Interpolate cubic smoothing spline through $\mathbf{x}_i$ and $\mathbf{x}_w$ using correlation scores as spline node weights

(c) Compensate for the unknown robot motion by extrapolating a new point set $\mathbf{x}_i'$ at given spline distances $\mathbf{d}$

(d) Correct $\mathbf{x}_i'$ back to the center of the weld seam by matching with correction model $\mathbf{M}$, giving a corrected point set $\mathbf{x}_i^c$

(e) Extract new templates $\mathbf{T}_{new}$ from $\mathbf{x}_i^c$ in spline normal direction, giving the actual track

4.1.3 Tracking Evaluations

In order to evaluate the robustness of the proposed template tracking method and its applicability to industrial robotic welding tasks, a qualitative evaluation on several thousand images from 674 different robotic welding tasks has been accomplished. Considering the evaluated data we first need to introduce the utilized wording. A *welding process* designates the material, welding parameters, and hardware depending process, accomplished by an industrial welding robot. A *welding sequence* designates the welding of a complete specific object, from the start to the end of the seam. Correct weld seam tracking consequently results in a set of axis aligned *weld seam image patches* extracted from the sequential images, depicting newly welded seam behind the welding torch, respectively. In factory automation, an industrial welding robot continuously repeats the

same process, e.g., in baseplate assembly welding in an automotive industry welding process. The data acquired from such repeated weldings is referred to as *welding process dataset*, where the collection of several welding process datasets is referred to as *welding test series*.

Under consideration of the introduced wording, the provided data can be separated into 3 welding test series and 25 welding process datasets. These datasets in turn consist of 674 welding sequences, and overall more than 176.700 welding images. The welding test series are denoted as test series α, β and γ. Test series α contains 21 welding process datasets, each consisting of 11 error-free and 10 consciously defective welding sequences. The welding processes represent a large variety of welding configurations, geometric welding trajectories and environmental conditions. Test series β and γ consist of industrial welding process datasets acquired in the automotive industry. Thereby, test series β includes 2 welding process datasets consisting of 73 and 99 welding sequences, respectively. The corresponding images reflect typical industrial welding situations. Test series γ represents a second welding test series acquired in the automotive industry. It consists of 2 welding process datasets, each including 21 welding sequences. Compared to the other test series, test series γ includes welding sequences that are on average twice as long and that include images from free-form surface weldings.

The crucial and performance effecting parameters of the template tracking approach are given by the size of the tracking templates, by the weighting of the spline points, and by the amount of template matching angles. For the template size, it turned out that it mainly depends on the resolution of the weld seam in the image. Best tracking performances have been achieved with templates that include both, the weld seam and few background structures on both sides. The distance between a template and the welding torch plays another major role as larger distances implicit already solid weld seam structures whereas short distances implicit hot and in many cases even still fluid weld seam regions as well as significantly more visible noise. Typical template sizes used in our experiments are thus given in the range of $100 \times 200px$. The spline weighting parameter strongly depends on the amount of visible noise present in processed welding images. The weighting generally implies how accurate the intermediate tracking result can be assumed, and thus how strong the tracked points influence the spline approximation. For welding processes that exhibit large amounts of noise like, e.g., severe smoke generation or significantly large amounts of sparks and spilling the weighting should be chosen small, implying weak tracking accuracies. On the other hand, for welding processes with fewer noise the parameter should be chosen higher. A typical value for

the later case is 0.9, whereas weightings in the range of $0.2 - 0.6$ should be chosen in case of large amounts of noise. The number of matching rotation angles evaluated for each tracking template mainly affects the overall runtime performance. The parameter is mainly constraint by the largest rotation that might occur between consecutive image frames, where large matching angle ranges imply lower frame rates and vice versa. In our experiments it turned out that for welding speeds of $50 - 100cm$ per minute at $10 - 20fps$ matching angles in the range of $\pm 15^\circ$ should be used if welding trajectories include narrow curves, whereas angles in the range of $\pm 2^\circ$ are adequate for straight or slightly curved weldings.

For the welding test series there is no ground truth in terms of tracking data or robot trajectories available. Thus, our evaluations considering achieved tracking accuracies rely on visual inspections. Tracking sequences are classified as correctly tracked if no drift away from the visible weld seam center occurred. Momentary outliers caused by, e.g., welding defects or noise are not considered as tracking error if the algorithm recovers within few frames. Figure 4.8 shows tracking images from three welding sequences, emphasizing the terms *correctly tracked*, *momentary outliers*, and *tracking drift*.

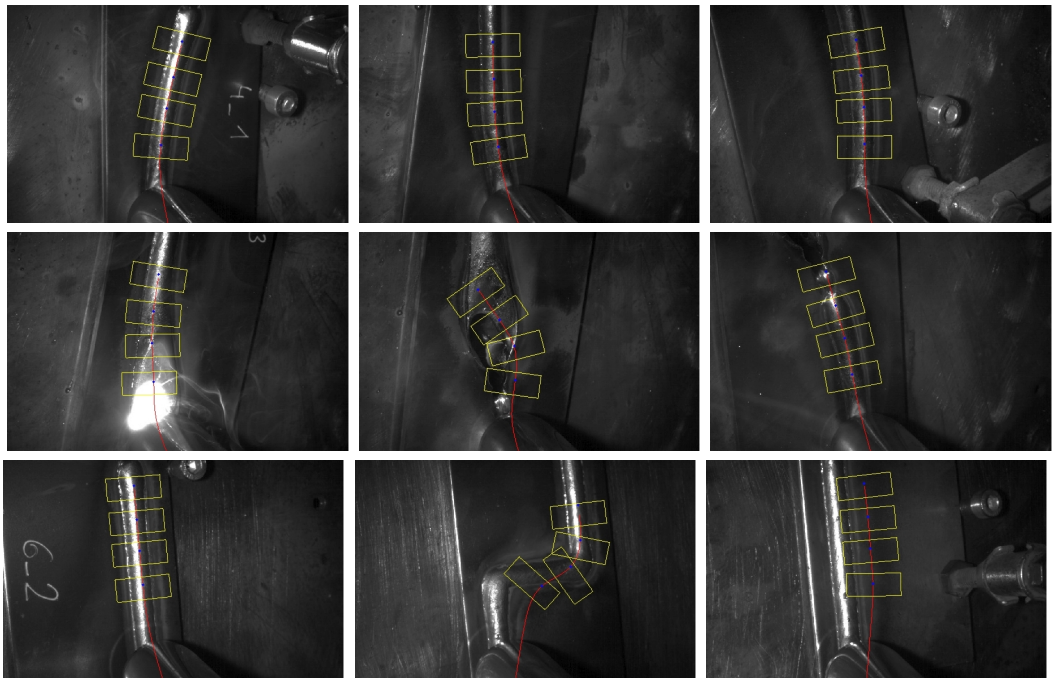


Figure 4.8: **Weld Seam Tracking Classification:** Weld seam tracking results from three welding sequences. The upper row shows a correctly tracked welding sequence. The middle row depicts images including momentary outliers caused by a welding defect. The bottom row shows a typical drifting example resulting in an overall tracking failure.

Series	Set	# Seq	Img / Seq	OK	Performance
α	01	20	252	19	95 %
α	02	20	254	19	95 %
α	03	20	254	20	100 %
α	04	20	259	19	95 %
α	05	20	259	20	100 %
α	06	20	281	13	65 %
α	07	20	289	19	95 %
α	12	20	286	20	100 %
α	13	20	275	19	95 %
α	14	20	274	20	100 %
α	15	20	282	19	95 %
α	16	20	275	19	95 %
α	17	20	273	19	95 %
α	18	20	297	18	90 %
α	19	20	289	18	90 %
α	20	20	267	20	100 %
α	21	20	277	20	100 %
α	22	20	134	16	80 %
α	23	20	135	20	100 %
α	24	20	137	19	95 %
α	25	20	292	20	100 %
					94 %
β	01	73	225	63	86 %
β	02	99	236	97	98 %
					92 %
γ	01	21	573	19	90 %
γ	02	21	325	21	100 %
					95 %

Table 4.1: **Qualitative Weld Seam Tracking Results:** Evaluations on 25 datasets from 3 different welding test series including of 674 welding sequences. The number of welding sequences per dataset, images per sequence, successfully tracked sequences, and the corresponding tracking performances are presented.

Table 4.1 presents qualitative results in terms of successfully tracked welding sequences and achieved average tracking performances, respectively. Although, the welding test series include several challenging scenarios tracking performances of 94% on test series α, 92% on test series β, and 95% on test series γ have been achieved, giving an overall performance of 93.7%. For a quantitative evaluation of our tracking approach we measured the repeatability in terms of tracking trajectory scatter for 4 welding process

Series	**Set**	**Seam Width** [px]	**std** [px]	**mean** [px]	**error** [%]
α	13	53.50	8.48	4.05	7.57
α	14	50.50	5.91	2.58	5.11
α	20	53.50	13.38	5.85	10.93
α	21	50.50	18.97	7.12	14.10

Table 4.2: **Weld Seam Tracking Repeatability:** Welding trajectory deviations for 21 repeated welding sequences of 4 welding process datasets, respectively. The repeatability is measured in terms of trajectory standard- and mean deviations , whereas the error gives the ratio between the weld seam width and the average deviation.

datasets from test series α. An average weld seam width given in pixels thereby allows for indicating a percentage error rate between weld seam width and mean trajectory deviations. Table 4.2 presents the numerical results for the repeatability experiment. A maximum repeatability error of 14.1% in terms of tracking drift has been achieved. Although, this value is quite high, the overall performance is adequate as the corresponding tracking was correct according to visual inspection.

4.2 Conclusion

In this Chapter we have presented a template tracking algorithm that relies on a spline-regularized prediction correction concept, and that has been designed for being applied in harsh industrial environments. Thereby, we especially focused on robustness to diverse kinds of typical industrial noise, on real-time capability of the tracking approach, and on a minimum of necessary parameterization or time consuming off-line preparatory work. We have then presented robust weld seam tracking as a suitable industrial application for our template tracking method. The experimental results on several hundred welding datasets clearly showed that template tracking is robust enough to be applied in harsh industrial robotic welding environments, as an overall qualitative tracking performance of more than **93%** has been achieved where other state of the art tracking approaches like [28, 45, 73] either fail due to noise sensitivity and do not comply with the given requirements. However, tracking drift experiments in terms of repeatability evaluations have shown that although correctly tracking the welded seam a quite large drifting occurs. Although, this is not a problem for the presented weld seam tracking application, for tracking problems in more dynamic scenes including e.g. highly dynamic illumination conditions, significant object appearance changes, or other tracking complications, the here presented template tracking method might not be sufficient and

robust enough. Thus more sophisticated methods for the template update are required, allowing for adopting to such dynamic scenes.

This issue is addressed in the subsequent Chapter, where we present a more advanced template tracking approach that additionally considers dynamic and adaptive updates of the underlying tracking template.

5

Template Tracking in Harsh Outdoor Environments

In the previous Chapter we have shown that templates are suitable for robust tracking in harsh environments. However, for tracking in more dynamic scenes simple template updates become insufficient due to inaccuracies and drifting. Assuming dynamic environments with, e.g., changing illumination conditions or significant object appearance changes, the underlying tracking template needs to be dynamically adopted to these changes. In this Chapter we present an appearance based tracking method that relies on image template blending and that is designed for robust tracking in highly dynamic scenes like outdoor environments, where moreover significantly large amounts of visible image noise might occur. Image templates and consequently object appearance are chosen as tracking cue as they can successfully cope with large amounts of visible image noise and clutter, where other state of the art tracking approaches like, e.g., the visual tracking decomposition approach by Kwon and Lee [73] or the Hough-based tracking approach by Godec et al. [45] are prone to fail. Especially for applications where significant amounts of image noise or tracking complications are assumed, local feature based approaches are not a good choice as the amount of outliers or mismatches increases, resulting in worse tracking accuracy or even in tracking failures. The usage of object appearance in terms of tracking templates allows for successful handling of image noise as each pixel in the underlying template can be seen as a separate feature. With given object appearances or templates from one or even several preceding views noisy regions can be easily identified and considered in particular during tracking, while the overall run-time is not affected, e.g., by additional and time consuming feature extraction or

matching computations. Figure 5.1 illustrates a sample comparison of appearance based template tracking to the above mentioned approaches on an exemplary agricultural outdoor tracking sequence, where image noise is given by rapidly changing illumination conditions or by greenery that is flying through the camera's field of view, resulting in partial occlusions and thus significant appearance changes. In order to demonstrate the applicability of the proposed image blending-based template tracking method, we present a specific application that is based on this appearance-based approach, namely agricultural tracking in harsh outdoor environments.

Figure 5.1: **Tracking in Harsh Outdoor Environments:** Appearance based template tracking (upper row) outperforms visual tracking decomposition [73] (middle row) and a Hough-based tracking approach [45] (bottom row) which are state of the art tracking methods. Both approaches are prone to fail as they cannot cope with the significantly large amounts of visible noise.

5.1 Image Blending-based Template Tracking

The motivation for image blending-based template tracking is to successfully follow an initially marked object especially in harsh outdoor environments. Thereby, accuracy and the degree of robustness should be maximized, the number of necessary parameters to

be tuned should be minimal, and the runtime performance should allow for real-time tracking on standard PCs. Thus, we propose an appearance based tracking approach that relies on image templates of the tracked object. More precisely, the fundamental concepts are robust homography estimation and adaptive image blending for template updates. In order to initialize the tracking template, an object is marked in an initial image. This can be done by either manually marking object corner points, or by, e.g., applying any kind of detection algorithm that gives the location of the desired object in the image. The tracking template is then obtained by warping the marked image region from the image coordinate frame $\mathcal{I}$ to a predefined rectified template coordinate frame $\mathcal{T}$ by applying a projective transformation $\mathbf{H}$, as exemplary illustrated for an agricultural tractor hanger image in Figure 5.2. The size of the template in $\mathcal{T}$ is thereby fixed and mainly depends on the approximate size of the tracked object in $\mathcal{I}$ in pixels.

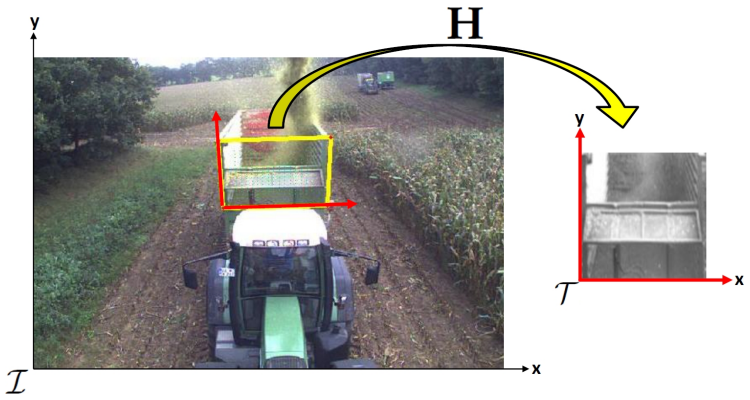


Figure 5.2: **Tracking Template Extraction:** An initially marked or detected image region that depicts an object to be tracked gets warped to a fixed template coordinate frame $\mathcal{T}$, giving the desired tracking template.

Template Tracking

Our addressed problem is robust object motion estimation between consecutive image frames in the presence of significant amounts of visible image noise. This is solved by robust image feature based homography estimation in the rectified template coordinate frame $\mathcal{T}$. The computed homography then allows for mapping image points from the preceding template coordinate frame to the actual one. Due to the assumed image noise we compute a projective transformation matrix $\mathbf{H}_p$ in a robust fashion using RANSAC [40] from established image feature correspondences. We chose small image patches extracted from the template as image features. However, also other robust approaches

for features or descriptors like, e.g., image corner points, SIFT [81], or SURF [6] could be used. Our choice of image patches as feature cue mainly results from the above mentioned real-time capability requirement. Many state of the art image feature and descriptor approaches do not guarantee real-time performance. Moreover, additional parameterizations which in turn must be adopted in case of changing environmental conditions might be necessary. As our objective is a robust real-time capable tracking approach that requires fewest parameter tuning steps, image patches are our choice for the underlying features. Similar to the approach presented in [58] we present a template in terms of a set of regularly placed equally sized image patches, where the template pose is given by the four template corner points. Figure 5.3 illustrates the image feature extraction from an exemplary template.

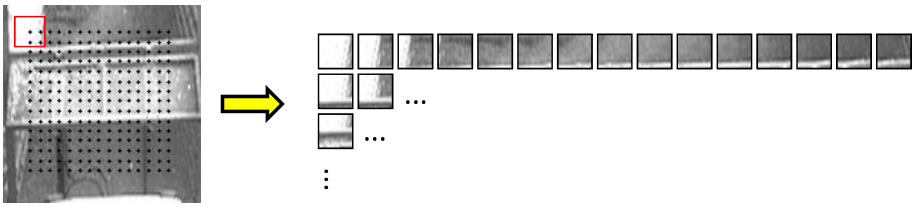


Figure 5.3: **Regular Grid Image Patch Features:** Small overlapping image patches are extracted from a regular grid. Each patch defines a single specific image feature.

Robust matching in the template coordinate frame $\mathcal{T}$ further results in a sufficient number of at least four [51] correct matches for the computation of a projective transformation. Thereby we rely on the normalized cross correlation ($\mathcal{NCC}$) as similarity measure due to its additional invariance to illumination changes compared to other similarity measures like, e.g., the normalized sum of squared differences ($\mathcal{NSSD}$), or the mean structural similarity measure ($\mathcal{MSSIM}$). The projective transformation $\mathbf{H}_p$ is robustly computed from the obtained correspondences based on RANSAC [40] and on the DLT [51] algorithm. With $\mathbf{H}_p$ given, the additional inverse transform $\mathbf{H}^{-1}$ from the template coordinate frame back to the image coordinate frame allows for aligning the existing tracking template to the actual image. Thus, the template corner points $\mathbf{p}_{t-1}^{\mathcal{I}}$ in the image coordinate frame $\mathcal{I}$ at time $t-1$ can be successfully updated according to

$$\mathbf{p}_t^{\mathcal{I}} = \mathbf{H}^{-1}\,\mathbf{H}_p\,\left(\mathbf{H}\,\mathbf{p}_{t-1}^{\mathcal{I}}\right), \tag{5.1}$$

giving the tracking result for the image at time t in the image coordinate frame $\mathcal{I}$.

Template Blending

As elaborately shown and discussed by Matthews et al. [89] template tracking suffers from a drifting problem over time. Furthermore, we are assuming significantly large amounts of noise and tracking complications that our template tracking approach needs to cope with. Thus, we propose to update the tracking template within each tracking iteration using an adaptive image blending strategy that allows for adopting image regions where the tracked object undergoes some changes, while the other regions remain constant. Although, there exist other different strategies for updating tracking templates, experimental drifting behavior evaluations presented later on clearly show that our blending strategy should be preferred. To conform with our initially stated requirements on real time capability, robustness and minimal requirements on parameterization, we propose a blending strategy that relies on multi-band blending presented by Zhao [138]. Although, multi-band blending is actually intended to be applied in image mosaicking and panorama image generation approaches, our approach of incremental template updates is also located in the scope of application. Moreover, the incremental design of the multi-band blending approach makes it the ideal blending strategy to be applied for our purposes as it is intended to process image sequences.

In terms of tracking, we extract an image template from an initial image and incrementally adopt variations of each pixel over time. Thereby image blending allows for smoothing transitions within template regions that include, e.g., illumination changes, object appearance variations, or object pose changes, while remaining object structures are preserved and remain constant. In contrast to the approach in [138] we use a weighting function that considers template registration quality weights denoted by $\mathbf{w_q}$, varying image exposure weights denoted by $\mathbf{w_e}$ and introduced by Mertens et al. [90], and temporal order weights denoted by $\mathbf{w_t}$. The resulting pixel-wise weighting function $\mathcal{W}$ for a new observation is thus given by

$$\mathcal{W}(x,y) = \mathbf{w_q}(x,y) \cdot \mathbf{w_e}(x,y) \cdot \mathbf{w_t}(x,y) \tag{5.2}$$

Image registration quality weights $\mathbf{w_q}(x,y)$ are defined by the L^2-norm between the template and the actual tracked image region, giving higher weights for template regions where the object undergoes some changes and small weights for regions where the object remains constant. Furthermore, moving background structures that do not belong to the tracked object get blurred over time as the constant blending of motion results in smoothed homogeneous image regions. The aim of exposure fusion is to em-

phasize unsaturated image regions in image sequences that exhibit varying exposures. It is based on an image contrast term C, an image saturation term S, and a so called well-exposedness term E for each relevant pixel. The contrast term is derived using a Laplacian filter, where the absolute value of the filter response corresponds to the actual contrast value. The saturation term is obtained by computing the per-pixel standard deviation within the given image color channels. Finally, the well-exposedness term reveals how well a pixel is exposed, allowing for suppressing over- and underexposure. This is achieved by weighting each image intensity denoted by i with a Gaussian curve, allowing for determining the degree of saturation for each pixel, respectively. The linear combination of C, S, and E gives the exposure fusion weights $\mathbf{w_e}(x, y)$. Constant temporal order weights are given by scalars q for the existing template and $(1 - q)$ for new observations. This allows for applying an image blending based forgetting function as new observations could be either directly incorporated into the tracking template using small values for q, or they are only slightly considered for the template update using larger values for q. The final choice for the value of q mainly depends on the underlying tracking application. The linear combination of the weighting components finally gives the weighting function $\mathcal{W}(x, y)$ for a new image $\mathbf{I}$:

$$\mathcal{W}(x, y) = (1 - q) \cdot \left(\| \mathbf{T}(x, y) - \mathbf{I}^W(x, y) \|_2 \right) \cdot \left(C^\alpha \cdot S^\beta \cdot E^\gamma \right), \tag{5.3}$$

where $\mathbf{T}$ defines the existing tracking template to be updated, $\mathbf{I}^W$ denotes the new warped image region or observation, and α, β and γ are weighting exponents that control the effect of the individual components. As a tracked object cannot be assumed to remain constant during tracking, and due to our choice of regular grid patches as underlying image features, an incremental update of the weighting function as proposed by Zhao [138] is not reasonable for our purposes. However, temporal order needs to be considered in order to apply the above discussed forgetting function for past observations. Thus, the weighting function $\mathcal{W}_{t-1}$ which coincidentally represents all observations so far, is given by the constant temporal order weighting scalar q, resulting in an equal weighting of each pixel from the existing template during the blending procedure. The additional geometry based weighting component that is used in [138] is not practicable for our approach, as the constraint of sharper focus in the center of the image does not hold for arbitrary warped image regions. The final multi-band template blending

formula for N Laplacian pyramid levels at time t is given according to

$$\mathbf{T}_t(x,y) = \sum_{i=1}^{N} \frac{\mathcal{W}_t(x,y)\ L^{\sigma_i}_{\mathbf{I}^W_{t_i}}(x,y) + \mathcal{W}_{t-1}(x,y)\ L^{\sigma_i}_{\mathbf{T}_{t-1_i}}(x,y)}{\mathcal{W}_t(x,y) + \mathcal{W}_{t-1}(x,y)}, \tag{5.4}$$

where $L^{\sigma_i}_{\mathbf{I}_i}$ denotes the i^{th} Laplacian pyramid level of image $\mathbf{I}$, depicting spatial frequencies in the specific range from σ_{i-1} to σ_i. To give an idea on how a tracking template evolves over time, Figure 5.4 illustrates the incremental blending results for three arbitrary tracking scenarios. Algorithm 4 summarizes the presented tracking approach for robust image blending-based template tracking in the presence of significantly large amounts of noise.

Algorithm 4 Image Blending-based Template Tracking:

Input: actual image $\mathbf{I}$, previous template $\mathbf{T}_{old}$, predicted corner points $\mathbf{p_{old}}$, warping transform $\mathbf{H}$

Output: actual template $\mathbf{T}_{new}$, updated corner points $\mathbf{p}_{new}$

(a) Extract the image region marked by the predicted corner points $\mathbf{p_{old}}$ and warp it to the template coordinate frame using $\mathbf{H}$

(b) Extract image patches from a regular grid and match them against the existing template $\mathbf{T}_{old}$

(c) Perform RANSAC based projective transformation estimation using the DLT algorithm, giving $\mathbf{H}_p$

(d) Update predicted corner points to $\mathbf{p}_{new}$ using $\mathbf{H}_p$ and the back-projection to the image coordinate frame via $\mathbf{H}^{-1}$

(e) Update the template by adaptive multi-band blending of the registered image region and $\mathbf{T}_{old}$, giving $\mathbf{T}_{new}$

5.2 Tracking in Agricultural Outdoor Environments

A real world application where the presented template tracking approach can be successfully applied is given by the task of robust tractor hanger tracking in agricultural environments. Thereby the hanger of a tractor which is following a combine harvester while harvesting crops should be tracked in order to automatically adjust a reloading arm that is filling the chaffed grain into the hanger following behind. The tracking

Figure 5.4: **Tracking Template Evolution:** Appearance changes caused by object motion, noise, or other influencing factors are dynamically incorporated into the tracking templates of three different tracked objects using the incremental image blending strategy.

challenges and complications which require for a robust appearance based tracking approach are given by abrupt illumination changes, back light cast shadows, and partial object occlusions caused by greenery that is flying through the camera's field of view. Moreover, object pose and scale changes need to be considered as the distance between the hanger and the harvester cannot be assumed as being constant, e.g., while driving along curves. Figure 5.5 shows few examples of challenging scenarios considering tractor hanger tracking.

5.2.1 Image Acquisition Setup

The image acquisition system consists of an industrial camera with a focal length of $4mm$ capturing images with a size of $752 \times 480px$ at a frame rate of $10fps$. The camera is rigidly mounted on the reloading arm of a combine harvester, which in turn is adjusted to directly reload chaffed grain into a following or laterally located tractor hanger. Using the hanger motion estimated by the tracking approach, a realignment of the reloading arm is possible. This allows for relocating the hanger back to the image center such that the chaffed grain can be directly reloaded at minimal loss.

Figure 5.5: **Agricultural Tracking Challenges:** Image based tracking of a tractor hanger behind a combine harvester while reloading chaffed grain is a challenging task due to object pose or scale changes, greenery that is flying through the camera's field of view resulting in partial occlusions of the tracked hanger, back light, cast shadows, or bad image quality.

5.2.2 Tracking Evaluations

In order to quantitatively evaluate the proposed template tracking approach, the method has been tested on 24 agricultural outdoor image sequences consisting of overall 8467 images from two different datasets. The first dataset contains 18 tracking sequences including different tractor hangers, diverse weather conditions, varying illumination, and typical noise like, e.g., greenery that is flying through the camera's field of view. The second dataset contains 6 specific sequences that represent further tracking challenges and complications like, e.g., a hanger driving along a curve. Table 5.1 presents short descriptions of the individual tracking complications, and obtained tracking performances of our proposed template blending approach denoted by **BT**, as well as of three state of the art tracking approaches denoted by **VTD** [73], **HT** [45], and **MS** [28], respectively. Thereby, we measure the tracking quality in percentage of correctly tracked frames, using the Pascal-VOC overlap criterion [34], defined as

$$score = \frac{R_T \cap R_{GT}}{R_T \cup R_{GT}}, \tag{5.5}$$

where R_T denotes the tracked area and R_{GT} defines the object area. A frame is marked as correctly tracked if the obtained score is greater than 50%, whereas we stop tracking once the tracker fails.

Our proposed template tracking algorithm achieved an average performance of more than 94% on the diverse challenging tracking sequences, while clearly outperforming

Set ID	Description	Images	BT	VTD	HT	MS
1-1	curve, noise	256	100	10	6	2
1-2	back light, scale	304	100	60	8	1
1-3	curve, few noise	512	100	12	4	47
1-4	rain, bad image quality	482	8	8	3	5
1-5	cast shadows	256	100	39	27	29
1-6	curve, bad image quality	618	100	99	4	9
1-7	back light, noise	686	100	9	3	2
1-8	multiple curves	81	100	100	27	5
1-9	illumination changes	990	100	8	6	8
1-10	curve, appearance changes	357	100	96	21	3
1-11	curve, greenery	312	100	72	11	6
1-12	appearance changes, scale	258	100	76	32	7
1-13	curve, scale	456	100	100	16	11
1-14	large occlusions, noise	120	100	19	29	68
1-15	curve, appearance changes	98	100	100	55	100
1-16	curve, bad image quality	160	54	5	43	6
1-17	curve, illumination changes	208	100	62	30	38
1-18	curve, cast shadows	483	100	7	17	2
2-α	appearance changes	408	100	100	12	6
2-β	noise, illumination changes	400	100	21	12	1
2-γ	cast shadows, background noise	256	100	39	27	29
2-δ	noise, back light	410	100	100	20	5
2-ϵ	curves, illumination changes	177	100	64	33	2
2-ι	scale, appearance changes	179	100	100	60	5
	Average	**353**	**94.25**	54.42	21.08	16.54

Table 5.1: **Agricultural Outdoor Tracking Results:** Individual challenges and achieved tracking performances of our template tracking method **BT**, visual tracking decomposition **VTD** [73], Hough-based tracking **HT** [45], and Mean-Shift tracking **MS** [28] on 24 different agricultural outdoor tracking sequences. The template blending approach achieved an average performance of **94.25%** correctly tracked frames, clearly outperforming the other methods.

the other approaches which are prone to fail due to the significantly large amounts of noise that are incorporated into the individual tracking models and states during their update steps. Thus, adaptive image blending based template tracking is definitely the best choice for this agricultural tracking application.

The low performance of only 8% on sequence 1-4 results from extremely bad weather conditions and several rain drops on the camera lens. Moreover, the depicted hanger is not in focus in several frames. The resulting blurry and homogeneous image re-

gions cause matching errors in the underlying feature matching which consequently resulted in an early tracking failure. Similar problems occurred in sequence 1-16 where a performance of more than 54% has been reached anyhow. Figure 5.6 illustrates some exemplary tracking results from three evaluated agricultural tracking sequences.

Figure 5.6: **Agricultural Template Tracking Results:** Illustrative image blending-based template tracking results from three different outdoor tracking scenarios including significantly large amounts of noise.

In a second set of qualitative experiments we evaluated the drift behavior of the presented template blending approach compared to other template update strategies as well as the runtime performance compared to the above mentioned state of the art tracking approaches.

For the drift behavior experiment we evaluated the drift of the template blending based approach and compared the results with mean template updates, where the template gets replaced by a mean template computed from the existing template and the actual tracked image region, and with keyframe template updates, where the template gets replaced by the tracked image region each 25^{th} frame on 6 video sequences. The incremental drift behaviors are illustrated in Figures 5.7 - 5.9. Numerical results are presented in Table 5.2. The incremental template blending clearly outperforms the other update strategies as only about one-tenth of the drift occurred in very long video sequences, including on average more than 900 images respectively. Considering the runtime performance experiment our template blending method reached an average frame rate of $15fps$ for a template size of 150×150 pixels and 144 partially overlap-

Figure 5.7: **Tracking Drift Evaluations:** Average center point drifts for template blending updates (yellow), mean template updates (magenta), and keyframe template updates (cyan) on the *book a* and *book b* sequences.

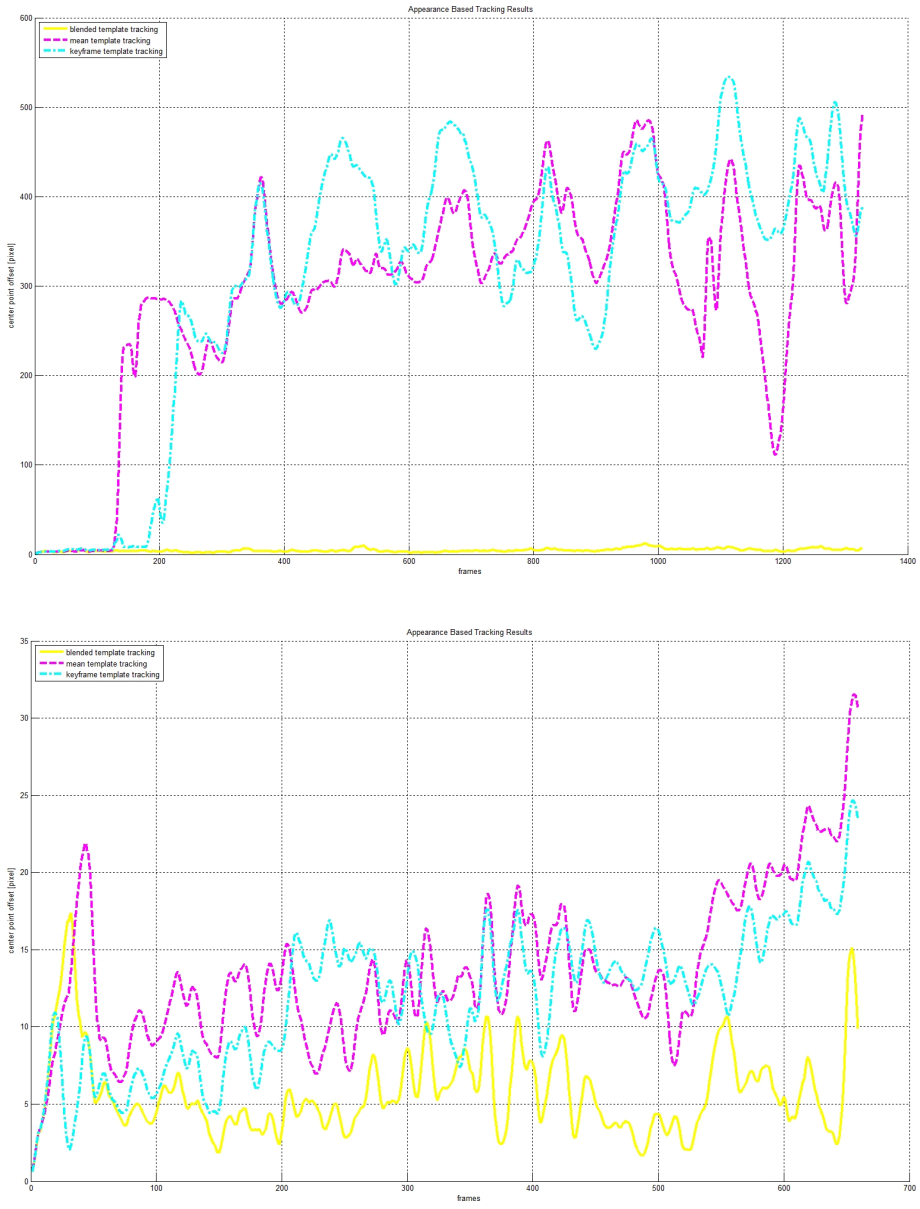


Figure 5.8: **Tracking Drift Evaluations:** Average center point drifts for template blending updates (yellow), mean template updates (magenta), and keyframe template updates (cyan) on the *book c* and *car* sequences.

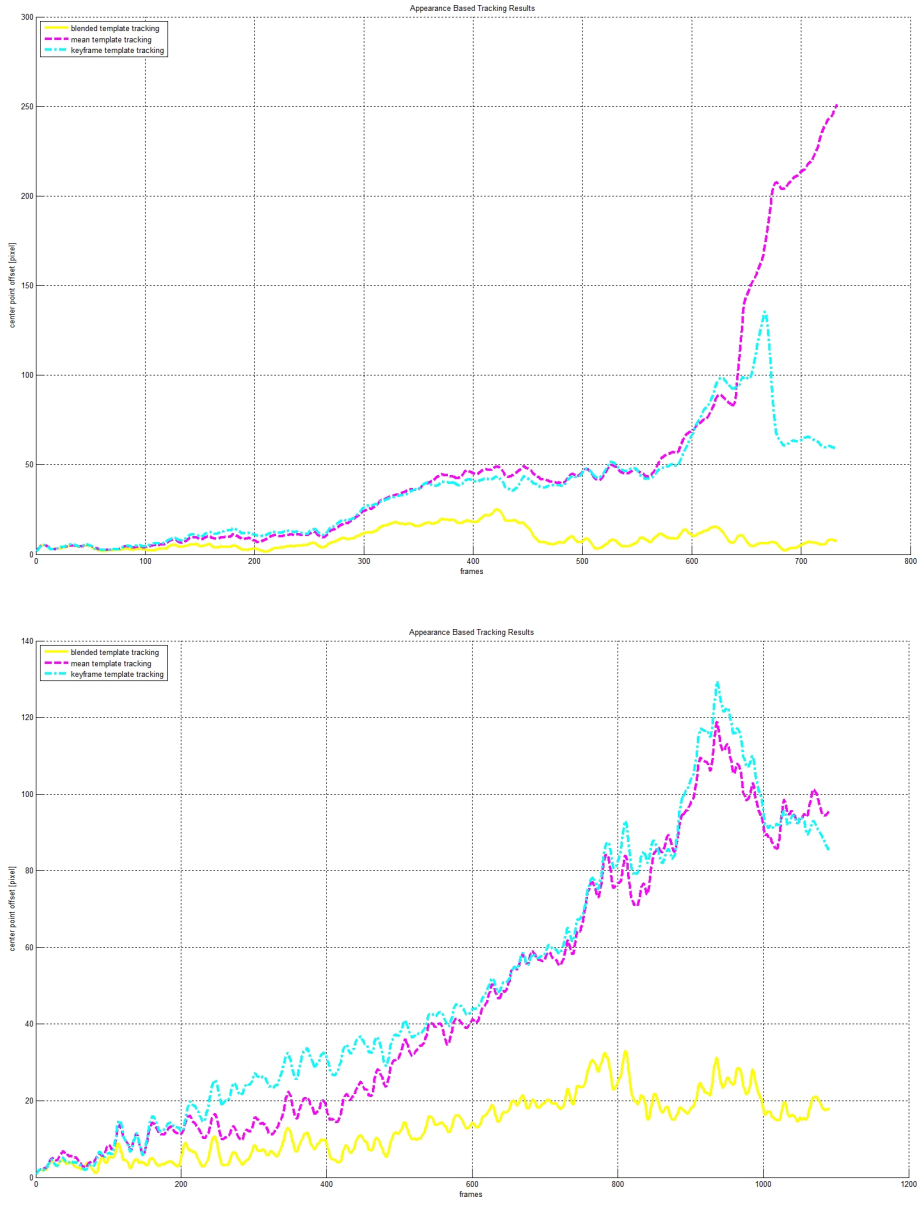


Figure 5.9: **Tracking Drift Evaluations:** Average center point drifts for template blending updates (yellow), mean template updates (magenta), and keyframe template updates (cyan) on the *greeting card* and *plant* sequences.

Sequence	**Images**	**blending** [px]	**mean** [px]	**keyframe** [px]
book a	1130	3.5	22.5	31.5
book b	691	4.8	10.5	19.5
book c	1327	5.5	212.0	254.5
car	659	7.5	19.5	17.5
greeting card	733	15.5	135.5	45.5
plant	1091	11.5	62.0	64.5
Average	938.5	**8.05**	77.0	72.2

Table 5.2: **Tracking Drift Evaluations:** Average center point drifts for template blending updates, mean template updates, and keyframe template updates on 6 different template tracking sequences. Our proposed template blending approach clearly outperforms the other strategies as only about one-tenth of the drift occurred.

ping image patches used as features for the underlying homography estimation on an Intel Core $i5$ processor with 2.6MHz and 4GB working memory. For the presented agricultural outdoor tracking application this allows for performing the tracking task in real-time as the underlying image acquisition system runs with $10fps$. The crucial and performance affecting parameters of the template blending approach are mainly given by the number of used image patches and by the global blending weight parameter q which controls the forgetting of previous views. The number of image patches mainly affects the runtime performance as each image patch needs to be compared and matched with the previous template. However, for patches in the range of $100 - 200$ we achieved robust and stable tracking results for the agricultural tracking application while still remaining real-time capable. Considering the blending weights, larger values in the range of $0.7 - 0.99$ should be used for tracking scenarios where severe and abrupt environmental changes occur, whereas for sequences with fewer noise and complications the forgetting factor should be chosen smaller. In our experiments on the agricultural tracking sequences we used a value of 0.85 for q as significantly large amounts of noise are present in the videos.

Considering the comparison with the other state of the art tracking approaches in terms of runtime performance, the visual tracking decomposition approach (VTD) requires 5-10 seconds per frame, whereas the Hough-based tracking approach (HT) requires 3 seconds per frame on the same machine. Indeed, the Mean-Shift (MS) algorithm reaches real-time performance. However, the corresponding achieved tracking accuracy of on average 20% is far from ours.

5.3 Conclusion

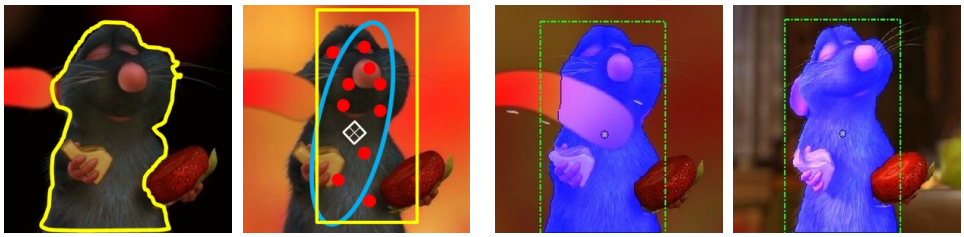
In this Chapter we have introduced a novel template tracking algorithm that relies on image blending based template updates in order be become robust against significantly large amounts of visible noise in tracking sequences. We have shown that this strategy allows for successful tracking objects where other state of the art approaches are prone to fail. Moreover, we have presented a real world application for the task of agricultural outdoor tracking, which relies on the proposed template tracking algorithm. In experimental evaluations on a large agricultural dataset we have shown that our approach is best suitable to solve the specific problem as we reached a considerably high average tracking performances of **94%** in terms of correctly tracked frames, as well as a comparable low drift behavior of on average only **8** pixels in long video sequences including on average more than 900 images, while clearly outperforming other state of the art tracking and template update strategies. Under consideration of the achieved experimental evaluations and results, we come to the conclusion that appearance based tracking approaches like the presented template blending algorithm are best suitable for tracking tasks where significantly large amounts of visible image noise are assumed.

However, template tracking aims in mapping an image plane yet to another image plane. This is an adequate assumption if objects with planar surfaces or with solid geometries are tracked, or if advanced template update strategies are applied, allowing for adoptions to slight 3D pose changes or object deformations. However, typical real world scenarios might include highly complex non-rigid objects, undergoing even more complex non-linear transformations between consecutive frames. In such cases, a template based tracking approach on its own is not sufficient any more. Thus, in the subsequent Chapter we address the problem of fusing different tracking cues, e.g., allowing for extending template based tracking approaches to above mentioned problems by combining them with other tracking cues.

6 Segmentation-based Tracking Support Fusion

In the previous Chapters we have presented template tracking approaches that are suitable to be applied in harsh industrial and outdoor environments. We have shown that templates are an adequate technique for robust object tracking in the presence of significant amounts of noise. However, as templates are intended especially for planar objects, there exist several challenging non-rigid scenarios, real world situations and tracking problems that template trackers typically cannot cope with. Facing such challenges like, e.g., non-rigid object transformations, severe appearance changes, abrupt illumination variations, or extremely fast motion, a recent trend is to combine several trackers, where each tracker solves a special facet of the overall problem. In this Chapter we thus present a novel method that allows for fusing heterogeneous trackers within a common, pixel-based representation as schematically illustrated in Figure 6.1.

Moreover, a tracker should also deliver a segmentation as proposed by Ren and Malik [100] on a super-pixel basis, and by Le and Hager [82] in terms of a bag of image patches two-class appearance model. This allows for an additional consideration of severe appearance changes and consequently for more precise updates of internal tracking states or models, thus allowing for an adequate handling of noise and background, e.g., during on-line learning, or update procedures. Moreover, object tracking in video sequences might require for accurate object segmentations, e.g., for extracting object contours for further analysis in different multimedia tasks. Typical examples would be video editing, video composition, or the combination of real video data with synthetic contents in terms of virtual realities.



(a) (b) (c) (d)

Figure 6.1: **Tracking Support Fusion:** Based on an initially delimited region (a) an object is tracked by fusing the diverse outputs of individual trackers (e.g., template bounding boxes, covariance ellipses, foreground pixels, object center votes) (b). The object's segmentation (blue overlay) computed from fused outputs in each frame (c - d) finally gives the track.

We define a mapping of the individual trackers' outputs to a common representation, and based on the individual tracking performances we apply a weighted fusion, followed by a regularization of the fusion result via an iterative energy minimization. The resulting segmentation of the tracked object then gets back-propagated to the individual contributing tracking approaches and is utilized in individual updating procedures. Thus, the trackers benefit on the one hand from the fusion as the combined advantages allow the individual trackers, e.g., to recover in error cases. On the other hand, the fine-grained object segmentation obtained in each frame allows for more precise state and model updates than by using, e.g., simple bounding boxes. Figure 6.2 summarizes the proposed tracking support fusion approach.

The major difficulty of any fusion method is that the individual tracking results (e.g., center point, bounding rectangle, kernel, or segmentation) strongly differ in their output and in the reported confidence values (e.g., probabilistic output, confidence range, error distances, or normalization). However, all visual tracking approaches have to be somehow coupled to the image domain which we can exploit to define a common representation that we refer to as *tracking support*, which is a quite general concept and defines a set of pixels in the image domain that support the trackers' result, and corresponding likelihoods. Typical examples would be individual pixels that exhibit high foreground probabilities, keypoints that exhibit similar motion with the object, image patches that are classified as foreground, or image regions that share appearance related similarities with the object (e.g., texture, color).

To fuse diverse results first each tracker individually finds the actual object position.

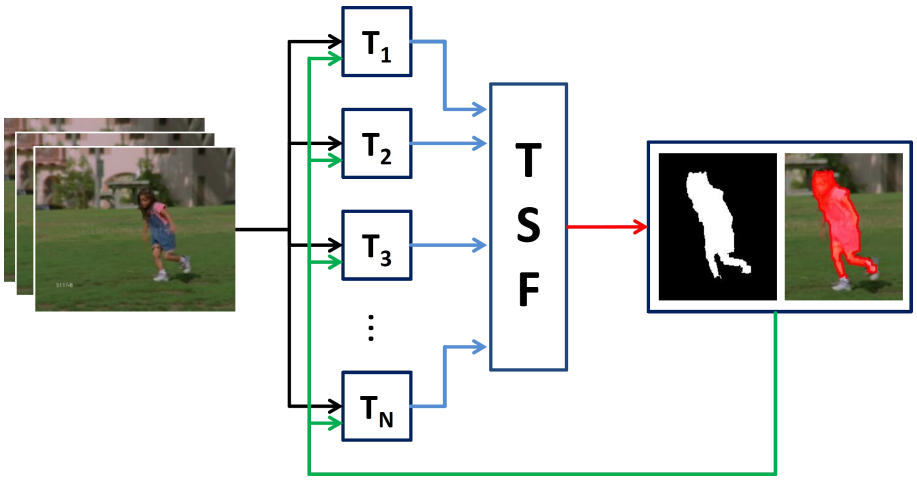


Figure 6.2: **Tracking Support Fusion Framework:** The diverse outputs of different trackers $\mathbf{T}_i$ are fused into a common pixel based representation, followed by an iterative segmentation. The looped back binary segmentations allow for more precise updates for each tracker, respectively.

The tracking output is then transformed into tracking support sets which are subsequently combined. However, instead of a simple union of the tracking support sets, the fusion additionally takes the recent performances of the individual contributing trackers into account, allowing for a weighting of individual trackers. The weighted support sets are then used within an iterative segmentation procedure. The segmentation determining both, the fusion result and the current tracking accuracies, gives the final desired tracking result. The segmentation is then provided to the individual tracking approaches to ensure high quality updates of their individual states. This design makes the proposed fusion scalable in the number of contributing trackers, and improves the granularity of the final result, while being completely parallelizable within the tracking stages.

As a proof of concept we demonstrate the fusion framework using three complementary tracking methods, namely the template blending based tracking method presented in Chapter 5, a recently presented discriminative tracking approach based on the generalized Hough transform [45], and a well known kernel tracking method based on feature histograms [28]. For these trackers we define the tracking support sets as projective homography inliers, as back-projected Hough votes that support the actual center object position, and as covariance ellipses around the object center point, respectively. These trackers have been chosen to given an idea on the variability of different trackers that the fusion framework can cope with. Of course they could be easily replaced by any

other tracking approaches. Subsequently, we define the estimation of the object's segmentation as an iterated energy minimization problem that is solved using an extended version of the GrabCut [22] algorithm.

6.1 Support-based Fusion of Heterogeneous Trackers

We propose a generic tracking fusion framework that we refer to as Tracking Support Fusion (TSF). On the one hand this allows for combining an arbitrary number of trackers, regardless of underlying methods and reported tracking outputs. On the other hand we explicitly address the problem of noisy updates by applying an iterative segmentation step, minimizing the amount of background and noise being incorporated into the individual updates. In particular, we show how the tracking results can be fused and how the finally obtained segmentation can be used to improve individual updates of internal tracking states, respectively. Figure 6.3 highlights the coarse block diagram of the proposed tracking support fusion approach and corresponding units, discussed in detail in the following.

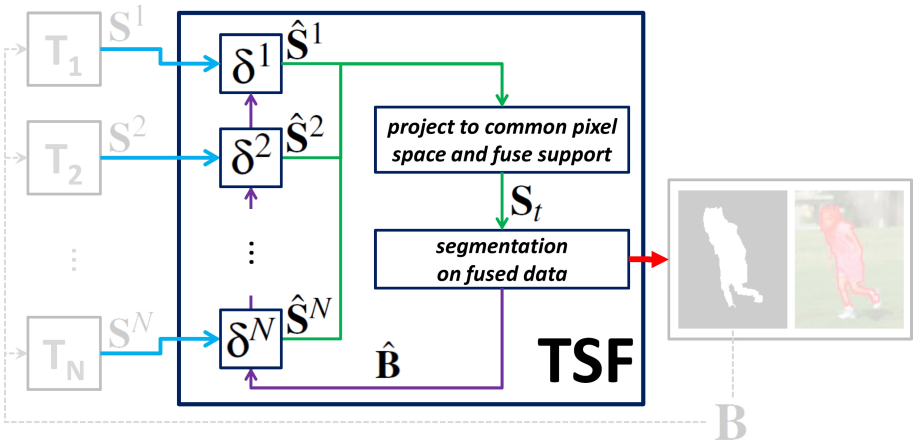


Figure 6.3: **Tracking Support Fusion (TSF):** The reported tracking outputs of N individual trackers $\mathbf{T}_i$ are fused on a pixel level by projecting the individually reported tracking outputs to corresponding tracking support sets $\mathbf{S}^i$ in the common image domain $\Omega_{\mathcal{I}}$. Iterative weighting of the individual support sets by δ^i based on the support overlap with intermediate binary segmentations $\hat{\mathbf{B}}$ allows for emphasizing the contribution of trackers that exhibit large overlaps, while dampening possible outliers. The final obtained segmentation $\mathbf{B}$ at time t can then be used for individual updates of $\mathbf{T}_i$.

6.1.1 The Fusion Framework

To combine diverse tracking methods that report significantly different outputs, we formulate a common generic output structure, the tracking support $\mathbf{S}$. For that purpose, we project the output of an arbitrary contributing tracker to the common image domain $\Omega_{\mathcal{I}}$ which is defined by the pixel coordinates of all pixels of the image. Depending on the tracker one or multiple potential target or foreground coordinates $(x_i, y_i) \in \Omega_{\mathcal{I}}$ and corresponding likelihoods $p_i \in [0, 1]$ are provided. Let an arbitrary tracker provide an output consisting of m target coordinates, the corresponding tracking support $\mathbf{S}$ is given according to

$$\mathbf{S} = \{(x_1, y_1, p_1), \ldots, (x_m, y_m, p_m)\}, \tag{6.1}$$

where $\sum p_i = 1$. Depending on the tracking method, the tracking support coordinates (x_i, y_i) are given by the coordinates of, e.g., homography inliers, discriminative votes, coordinates inside a reported rectangle or bounding box, or foreground pixels in the image domain $\Omega_{\mathcal{I}}$. The likelihoods p_i are derived from individual tracking foreground certainties like in the simplest case binary foreground votes, or in more sophisticated cases discrete labels, continuous probabilities, or confidence measures. As mentioned above we normalize the likelihoods of each contributing tracker to a simplex such that the overall sum of all likelihoods equals 1. Thus, in case of m binary foreground votes the corresponding likelihoods are given as $p_i = \frac{1}{m}$. For discrete labels the likelihoods are obtained in a similar way for each individual label, followed by a normalization to the number of labels. This normalization for each trackers' support allows for combining sparse and dense tracking supports as well as binary, discrete and real-valued likelihoods as the influence of each tracker is treated equal.

However, in practice diverse tracking approaches typically do not exactly coincide in their reported outputs. Therefore, we further incorporate so called tracking congruences δ^n, $n \in \{1, \cdots, N\}$, $N = \#$ trackers, which weight the different trackers' reported likelihoods according to the overlap of a tracker's support $\mathbf{S}^n$ with the final obtained tracking result given by an iteratively estimated binary segmentation $\mathbf{B}$. The congruences at time $t = 0$ are initialized with $\delta^n = 1$ in order to consider each contributing tracker equally at the beginning. They are then continuously updated within each tracking iteration according to

$$\delta^n = \frac{|\mathbf{S}^n \cap \mathbf{B}|}{|\mathbf{S}^n|}, \quad \delta^n \in [0, 1]\ . \tag{6.2}$$

In the first iteration of the segmentation step presented in detail in Section 6.1.2 no

binary segmentation $\mathbf{B}$ is yet available. Thus, we use the tracker's average congruency during the past M frames:

$$\delta_{\text{init}}^{n} = \frac{1}{M} \sum_{i=t-M}^{t-1} \delta_{i}^{n} \,. \tag{6.3}$$

In this way, trackers that performed well in the past are assigned high initial congruences, whereas trackers that exhibit worse performances are initially down-weighted or dampened. Once an iteration of the segmentation algorithm is passed, the overlap percentage of the tracking support $\mathbf{S}^n$ with the intermediate binary segmentation $\hat{\mathbf{B}}$ is evaluated until convergence, meaning $\hat{\mathbf{B}}$ does not undergo further changes. In this way, δ_t^n is tuned such that it finally reflects the maximum overlap of $\mathbf{S}^n$ with the final segmentation result denoted by $\mathbf{B}$. The fusion result $\mathbf{S}_t$ for N individual contributing trackers at time t is finally obtained by the normalized and weighted sum of the individually reported support sets $\mathbf{S}_t^n$:

$$\mathbf{S}_t = \frac{1}{N} \sum_{n=1}^{N} \delta_t^n \, \mathbf{S}_t^n \,. \tag{6.4}$$

Up to now, we have successfully projected individual tracking outputs to a common representation in a common space $\Omega_{\mathcal{I}}$, and transformed the corresponding pixel weights to common likelihoods. This allows for a successful combination of different heterogeneous tracking approaches that provide significantly different tracking outputs on a pixel level. In this way, the diverse tracking support sets haven been combined in terms of a Markov Random Field (MRF), where the unary potentials are given by accumulated tracking support likelihoods. However, the fusion of individual tracking supports on its own is not sufficient for successful tracking in challenging environments and scenarios, as fused tracking supports do not implicitly give a sufficient object description. Consider, e.g., homography inliers that are located on a different object that undergoes similar transformations. If no further information on the underlying image is available, outliers on the other object would also be fully considered for the fusion. Hence, we need to additionally consider the image data, e.g., color or contrast at the reported tracking support image pixels in terms of a Conditional Random Field (CRF). This allows on the one hand for combining diverse tracking outputs in a common space, and on the other hand for further considering, e.g., image data or image gradients in a local neighborhood. Thus, we finally apply a probabilistic iterative image segmentation algorithm that uses the fused tracking support set $\mathbf{S}_t$ at time t as input for the unary potentials. We therefore transform the fused tracking support $\mathbf{S}_t$ to a continuous weighting map $\mathbf{P}$ in the range $[-1, 1]$. Thereby, $sign(\mathbf{P})$ denotes the label (foreground,

background) of the support, $abs(\mathbf{P})$ defines the weight of the specific pixel, and $\mathbf{P} = 0$ gives the neutral or unknown label. In this way, the fused tracking support can be directly used within a probabilistic segmentation approach like *GrabCut* [22] or *TVSeg* [122] for energy minimization.

6.1.2 Iterative Segmentation

To obtain a reasonable segmentation of the tracked object, we use the fused and transformed tracking support as unary potentials in a probabilistic segmentation algorithm. The data terms are consequently given by the corresponding image data which can either be given by gray scale values, RGB color values or by any other representation of the image data (e.g., CIELAB, HSV, YUV, etc.). We build on the well known GrabCut [22] algorithm. However, also other probabilistic energy minimization methods like, e.g., TVSeg [122] can be used. We adopt the algorithm such that it directly uses the fused tracking support as continuous unary term input. We then further extended GrabCut such that iteratively estimated tracking congruences derived from intermediate binary segmentations and individual tracking support sets are additionally considered during the energy minimization.

Similar to the definitions in [22] we use Gaussian Mixture Models (GMM) for modeling object foreground and background in the RGB color space. However, also other models like, e.g., gray scale or color histograms could be used. Instead of utilizing manually marked discrete labels, we use the fused tracking support $\mathbf{S}_t$ from N contributing trackers in terms of the fused weighting map $\mathbf{P}$ and incorporate it into the underlying Gibbs energy formulation. In our case, the Gibbs energy consisting of data term $\mathcal{U}$ and smoothness term $\mathcal{V}$ becomes

$$\mathbf{E}\left(\alpha, \mathbf{k}\left(\mathbf{S}_t \rightarrow \mathbf{P}\right), \theta, \mathbf{z}\right) = \mathcal{U}\left(\alpha, \mathbf{k}\left(\mathbf{S}_t \rightarrow \mathbf{P}\right), \theta, \mathbf{z}\right) + \mathcal{V}\left(\alpha, \mathbf{z}\right) . \tag{6.5}$$

α thereby defines pixel opacity values in the range of $0 \leq \alpha_i \leq 1$ with 0 denoting hard background and 1 denoting hard foreground, thus the desired segmentation. $\mathbf{k}\left(\mathbf{S}_t \rightarrow \mathbf{P}\right)$ represents GMM component assignments for foreground and background pixels, obtained from the fused tracking supports. The term θ defines the model parameters, and $\mathbf{z}$ defines the corresponding image pixel values. The data term $\mathcal{U}$ that considers GMMs based on fused tracking support sets $\mathbf{S}_t \rightarrow \mathbf{P}$ is defined as

$$\mathcal{U}\left(\alpha, \mathbf{k}\left(\mathbf{S}_t \rightarrow \mathbf{P}\right), \theta, \mathbf{z}\right) = \sum_n -log p\left(z_n | \alpha_n, k_n, \theta\right) - log \pi\left(\alpha_n, k_n\right) , \tag{6.6}$$

where the GMM component assignments represented by k_n are obtained from object foreground and background pixels in $\mathbf{P}$, $p(\cdot)$ is a Gaussian probability distribution, and $\pi(\cdot)$ are mixture weighting coefficients. The smoothness term $\mathcal{V}$ of the Gibbs formulation, that finally also incorporates the image data and the local pixel neighborhood constraints into the energy minimization, is given according to

$$\mathcal{V}(\alpha, \mathbf{z}) = \gamma \sum_{(m,n)\in\mathbf{C}} [\alpha_n \neq \alpha_m] \, e^{-\beta \|z_m - z_n\|^2}, \tag{6.7}$$

where $[\phi]$ denotes an indicator function taking values $\{0, 1\}$ for a predicate ϕ, $\mathbf{C}$ denotes the neighboring pixels, and $\|\cdot\|^2$ defines the Euclidean distance in the RGB color space [22].

As mentioned above, different trackers typically solve different facets of the overall tracking problem, resulting in different tracking outputs, respectively. These tracking outputs usually also include an unknown percentage of noise and outliers. Thus, in order to identify trackers that perform well, we incorporate above introduced tracking congruences into the iterative segmentation approach. To do so, we extend the GrabCut algorithm by an iterative tracker specific re-weighting scheme for each contributing tracker within each segmentation iteration. This allows for iterative down weighting of tracking support components that exhibit low congruences δ with intermediate segmentations $\hat{\mathbf{B}}$, and vice versa. Thereby, the congruency of a specific tracker is defined as the percentage of corresponding tracking support coordinates located inside the actual intermediate object foreground region according to Equation (6.2). The iterative and normalized re-weighting of a specific tracker's support set $\mathbf{S}^n$ to $\hat{\mathbf{S}}^n$ is then given according to

$$\hat{\mathbf{S}}^n = \mathbf{S}^n \, \hat{\delta}^n = \mathbf{S}^n \frac{|\mathbf{S}^n \cap \hat{\alpha}|}{|\mathbf{S}^n|}, \tag{6.8}$$

where $\hat{\mathbf{S}}^n$ gets iteratively re-computed until the intermediate segmentation $\hat{\mathbf{B}}$ does not undergo some further changes, thus converging. The final object segmentation denoted by $\mathbf{B}$ is then computed using MinCut [14] energy minimization, giving the global minimum of the actual tracking support based segmentation problem. Algorithm 5 summarizes the proposed tracking support fusion algorithm.

Algorithm 5 Tracking Support Fusion

1 Initialize tracking support fusion and individual trackers $\mathbf{T}_1, \ldots, \mathbf{T}_N$

1a) Coarsely mark object in the first image at time $t = 0$

1b) Initialize GMMs for object foreground and background regions

for each image at time $t > 0$ **do**

2 Retrieve tracking support sets $\mathbf{S}^n$ from all trackers $\mathbf{T}_1, \ldots, \mathbf{T}_N$

3 Compute fused weighting map $\mathbf{P}$ from $\mathbf{S}^n$

4 Assign GMM components $\mathbf{k}$ to object foreground and background regions in $\mathbf{P}$

5 Learn GMM parameters using corresponding image data

6 Estimate binary intermediate segmentation $\hat{\mathbf{B}}$ using GrabCut

7 Re-weight tracking support sets according to $\hat{\mathbf{S}}^n = \hat{\delta}^n \, \mathbf{S}^n$ based on overlaps of the tracking support sets with $\hat{\mathbf{B}}$

8 Repeat from 3 until convergence

9 Compute final binary segmentation $\mathbf{B}$ from converged $\mathbf{P}$ using MinCut

10 Update all trackers $\mathbf{T}_n$ with the final segmentation $\mathbf{B}$

end for

6.2 Fusion of Three Heterogeneous Trackers

As discussed in Chapter 3, there exist numerous trackers that exhibit diverse characteristics, methodologies and what is most important different typically not directly combinable outputs. Thus, we first want to give an overview of tracking object representations and how these could be transformed to suitable tracking support sets. Table 6.1 provides an overview of common object representations used in relevant tracking literature, as well as corresponding tracking support sets suitable for our tracking support fusion framework.

As our framework is open to arbitrary tracking approaches and as we want to demonstrate the applicability of the proposed tracking support fusion approach for several diverse trackers and diverse significantly different tracking outputs, we combine the output of three entirely different heterogeneous tracking approaches: **(a)** an image blending-based template tracker denoted by BT that delivers dense homography inliers,

Object Representation	**Tracking Support Definition**
Centroid	*Centroid pixel coordinates*
Points	*Pixel Coordinates for reported points*
Rectangles	*Pixels inside the rectangle*
Ellipses	*Pixels inside the ellipse*
Edges	*Reported edge image pixels*
Contours	*Reported contour image pixels*
Silhouettes	*Pixels defining a silhouette or pixels located inside a corresponding contour*
Features Points	*Pixel coordinates of image features*
Descriptors	*Pixels used for descriptor computation*
Segmentations	*Pixels being segmented (foreground)*
Motion Models	*Pixel coordinates of motion model inliers*
Articulated Shape Models	*Pixels inside the individual articulated parts*
Skeletal Models	*Reported object skeleton pixels*
Probability Densities	*Pixel coordinates of image points that exhibit high probabilities*
Templates	*Pixel coordinates of image points that exhibit high similarities with the template*

Table 6.1: **Object Representations and Tracking Support Definitions:** In tracking literature diverse object representations are utilized, depending on the underlying tracking methodology [135]. These individual outputs could be used in terms of tracking support in our proposed fusion framework. The corresponding likelihoods p are thereby either set to 1, or in case of given weights for an object representation, these could be used.

(b) a discriminative Hough voting based tracker denoted by HT [45] that delivers sparse object foreground votes, and **(c)** a feature histogram based Mean-Shift tracker [28] denoted by MS that delivers covariance ellipses around the object center locations. The significantly different tracking outputs rely on completely different cues and hence cannot be directly combined. From a tracking fusion point of view, each tracker solves a special facet of the overall tracking problem, namely geometric consistency (BT), feature space consistency (HT) and color consistency (MS), which further allows for solving challenging non-rigid object tracking problems.

6.2.1 Blending-based Template Tracking (BT)

The first tracking approach that we incorporate into our fusion framework is a template tracking method. It is based on the novel incremental template blending update scheme presented in Chapter 5 and provides tracking support in terms of dense projective homography inliers. The concept of blending-based template updates allows for incorporating regions where the object undergoes some changes, while ignoring noise and abrupt variations. This ideally suites into our tracking support fusion concept as object segmentations can be used as template update masks.

In a tracking context, we propose to incrementally update a template T by weighted multi-band blending [16, 20] of a projectively rectified image region I^{warp}, representing the actual target object. Different weightings thereby allow for successful suppression of typical blending artifacts [90, 138] as well as for applying a forgetting function during tracking. To register the actual image region with the template, we compute the optical flow [132] between them. As optical flow typically contains a large number of outlier vectors, we filter the flow field by linearizion with a projective homography. In this way a dense set of flow vectors that coarsely follow a linear homography transformation is obtained. The tracking support $\mathbf{S}^{BT}$ required for our tracking support fusion framework is consequently given by the image coordinates (x_i, y_i) of the obtained homography inliers. The corresponding measurement likelihoods p_i at the supporting image coordinates are set to 1 as neither the underlying homography estimation nor the optical flow estimation return suitable probability or confidence measures, resulting in the simplest representation of binary object foreground labels as likelihoods. The proposed weighted blending update approach allows for considering, e.g., appearance changes where unseen object regions need to become part of the tracked object, while noise or abrupt changes do not effect the template sustainable. Incorporation of the back-looped actual segmentation $\mathbf{B}$ further allows for accurately separating background and object foreground for more precise blending. Hence, foreground regions are incrementally updated within each tracking iteration, whereas background gets successfully dampened. Although in a different context, Bleyer et al. [11] also showed that the incorporation of segmentations into optical flow allows for increasing accuracy and thus robustness. Figure 6.4 exemplary illustrates the evolution of an incrementally blended template for a real-world tracking scenario, demonstrating adoptions of the template to overcome non-rigid object deformations, out-of-plane rotations and appearance changes, which typically cannot be handled by a template tracking approach. Figure 6.5 illustrates exemplary template blending tracking supports given by image pixels that correspond to

a set of homography inliers for three different tracking scenarios.

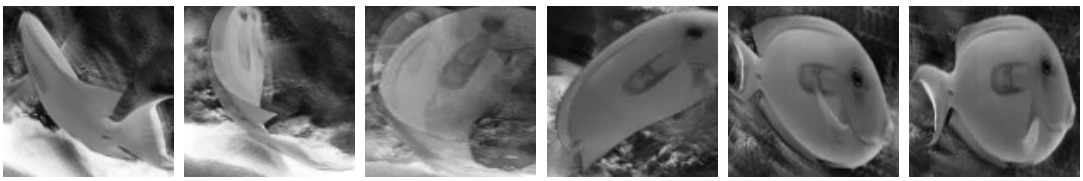


Figure 6.4: **Template Blending:** Dynamic template blending updates over time result in robustness to non-rigid object transformations, out-of-plane rotations, and significant appearance changes.

Figure 6.5: **BT Support:** Point-wise template blending support (red overlays) given by image pixels that belong to a set of homography inliers for three different tracking scenarios (cropped).

6.2.2 Discriminative Tracking based on Hough Voting (HT)

The second tracking approach that we incorporate into our tracking support fusion is based on HoughTrack proposed by Godec et al. [45]. The approach combines the generalized Hough transform with a rough segmentation to allow for tracking of non-rigid objects. However, we decouple the Hough-space based object detection from the subsequent segmentation and utilize the intermediate tracking outputs in terms of Hough back-projections for our tracking support fusion purposes. In the original HoughTrack approach the segmentation process is initialized by back-projection of the collected Hough votes in order to establish a set of image pixels that successfully voted for the current maximum in the image. However, this simple back projection does not consider the strength of the votes nor the foreground probability of the supporting pixel. Thus, we create the desired tracking support $\mathbf{S}^{HT}$ in terms of weighted supporting pixels

(x, y, p). The pixel weight is thereby given by

$$p = \sum_{t=1}^{T} \omega_{vote}(n_t) P^{+}(n_t)\,, \tag{6.9}$$

where $\omega_{vote}(n_t)$ defines the voting weight of the vector that voted to the center, $P^{+}(n_t)$ is the foreground probability of the leaf node n_t, and p is the resulting sum of these voting weights over all trees T, subsequently assigned as likelihood to the corresponding supporting pixel (x, y). The resulting set of supporting pixels is more fine grained than a simple unweighted point set used in HoughTrack.

Similar to the template tracking approach this discriminative tracking algorithm finally also benefits from the binary segmentation **B** that is obtained at the end of each tracking iteration. This is due to the segmented image pixel values being used for object foreground updates, and image pixel values outside the obtained segmentation being used for background updates, respectively. Figure 6.6 exemplary illustrates the sparse tracking support in form of back-projected Hough center votes generated from the Hough-based tracking approach for three different tracking scenarios.

Figure 6.6: **HT Support:** Point-wise Hough-based tracking support (blue overlays) given by back-projected object center votes for three different tracking scenarios (cropped).

6.2.3 Feature Histogram based Mean Shift Tracking (MS)

As a third tracking approach, we use the well known Mean-Shift algorithm proposed by Comaniciu et al. [28]. In their work they address the problem of tracking of non-rigid objects using feature histograms that are regularized by spatial masking with an isotropic kernel. Using the basin of attraction of the similarity function, the tracker locates the current object position and adapts the object model to the actual object appearance. In order to integrate this tracking approach and especially its output given by covariance ellipses into our tracking support fusion framework, we reformulate the reported track-

ing support using a Gaussian kernel with the same covariance matrix than the ellipse reported by the Mean-Shift tracker. Further, we scale the standard deviation σ, such that the ellipse radius corresponds to 3σ, resulting in more than 99% of the weight of the kernel being located inside the reported region. Consequently, the desired tracking support $\mathbf{S}^{MS}$ for the Mean-Shift tracker is given by image pixel coordinates (x, y) located inside the reported ellipse as well as by corresponding Gaussian weights for the likelihoods p_i. The benefits of this tracking approach gains from the final obtained segmentation $\mathbf{B}$ which allows for more precise updates of the underlying feature histograms. In fact, only the object foreground image pixels located inside the segmentation are considered for the update instead of all pixels inside the reported ellipse. Consequently, the back-looped segmentation significantly reduces the amount of background noise during the model update step, resulting in more accurate tracking results. Figure 6.7 exemplary shows the Mean-Shift tracking support derived from image pixel coordinates inside a reported covariance ellipse for three different tracking scenarios.

Figure 6.7: **MS Support:** Mean shift tracking support given by image points inside a reported covariance ellipse reformulated by a Gaussian kernel (green overlay) for three different tracking scenarios (cropped).

6.3 Experiments

We evaluate the tracking support fusion framework using the three heterogeneous tracking approaches summarized in Section 6.2 on different real world scenarios, movie scenes[1] and on tracking sequences used in relevant literature [4, 45, 74, 105]. The evaluated dataset contains gray scale and color sequences including complex non-rigid object transformations, severe appearance and illumination variations, abrupt object motion as well as motion blur, and different object pose variations. In more detail, the dataset includes comic and animation sequences showing complex non-rigid object transforma-

[1]Downloaded from the on-line video portal *YouTube*.

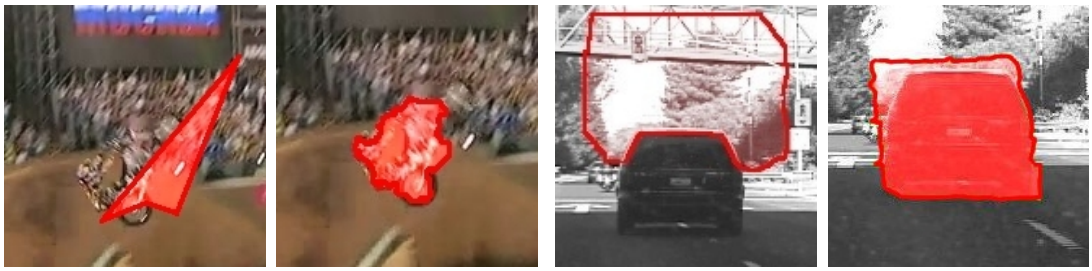
tions (e.g., Transformer, Monster AG, and Ratatouille sequence), sports scenarios including camouflaged foreground and background regions (e.g., Mountain-bike, Motocross 1 and 2 sequence), nature sequences exhibiting severe motion blur and bad image quality (e.g., cheetah, monkeydog, and birdfall sequence), sequences with abrupt foreground, background or appearance changes (e.g., Cliff-dive 1 and 2, car4, and david sequence), or a sequence showing a mirror-symmetric object (Coffee Mug sequence). Overall, the collected dataset consists of 26 sequences including more than 7500 frames. The obtained results are denoted as **TSF**, respectively.

In our evaluations we consider a narrow-band around the last object position instead of performing the computational costly segmentation on the entire image. The bandwidth has been set to ± 15 pixels in our experiments which is a reasonable value relative to the average object diameters in most tracking sequences. Of course this limits the performance on sequences with large motion baselines between consecutive frames. However, this can be resolved by increasing the narrow bandwidth but at the cost of additional runtime performance. Considering the runtime performance our achieved frame rates mainly depend on the individual performances of the contributing tracking approaches. Our Matlab implementation reaches $2-3$ frames per second without code optimization or parallelization of the contributing the trackers on an Intel Core i5 notebook processor with 2.6MHz and 4GB working memory.

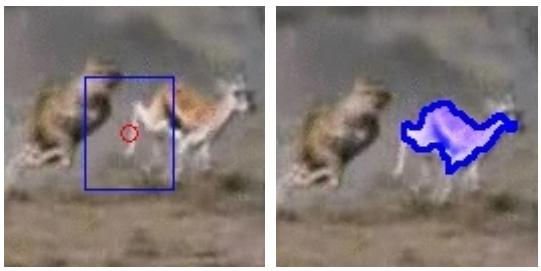
6.3.1 Fusing the Advantages

Before showing a quantitative evaluation, we demonstrate that the tracking support fusion allows for tracking objects in sequences where the individual contributing trackers fail. A common problem of template based methods such as the applied template blending tracker (BT) is tracking of highly non-rigid objects. This issue can be resolved via fusion with a non-rigid tracker. On the other hand the discriminative Hough based tracking method (HT) uses local patches to detect the object center. If these patches undergo large appearance changes from one frame to another, e.g., due to severe illumination changes or cast shadows, the detection may fail, which can also happen for Mean-Shift (MS). Figure 6.8 depicts individual cases where the fusion allows for successfully tracking objects where the individual trackers are prone to fail.

Figure 6.9 illustrates specific tracking performances in terms of tracking congruences over time for two different tracking scenarios. The congruency curves in Figure 6.9 (a) illustrate individual trackers' failures for a sports scenario that are successfully passed due to the fusion with the other trackers. In frame 17 HT and BT tend to fail, in frame



(a) #18 BT (b) #18 TSF (c) #213 HT (d) #213 TSF



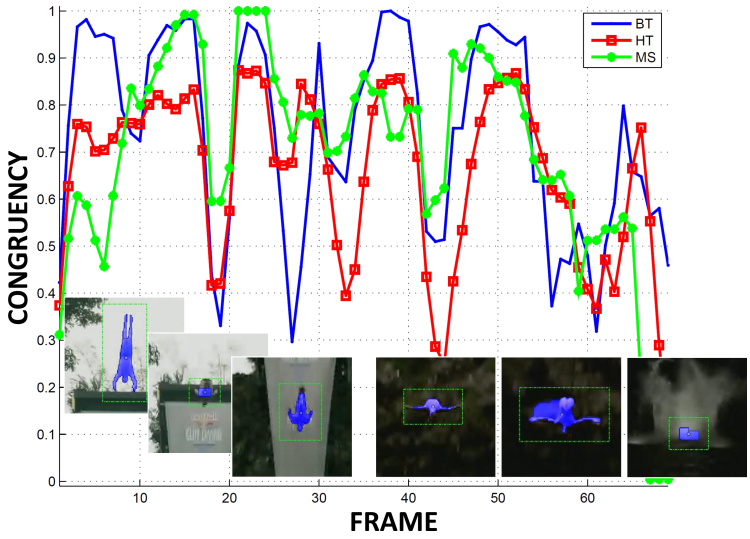
(e) #19 MS (f) #19 TSF

Figure 6.8: **Fusion of Tracking Advantages allows for Solving Individual Failure Cases:** Failure cases of BT (a), HT (c), and MS (e) can be resolved by fusing the individual trackers and their specific advantages, (b), (d), and (f).

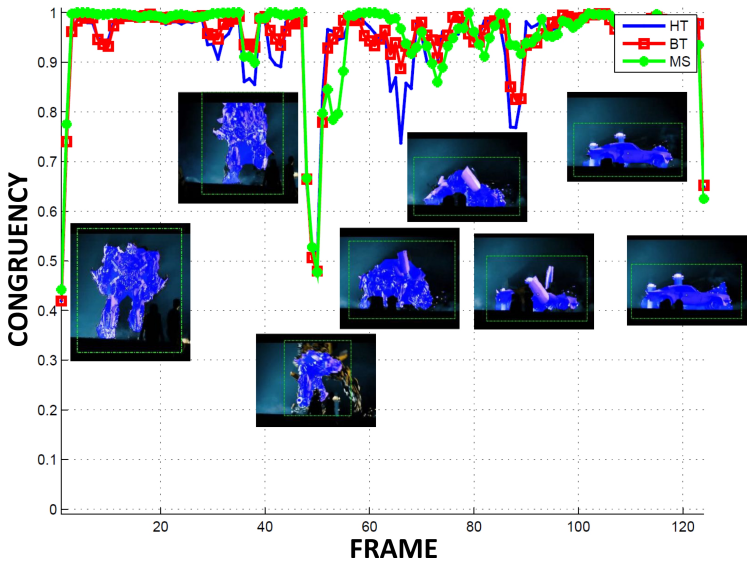
26 BT tends to fail, and in frame 60 MS fails. However, the overall tracking succeeds due to the weighted fusion of the individual tracking advantages. Thus, the fusion of the individual trackers clearly helps to successfully track objects and to recover individual trackers if they failed. The congruences in Figure 6.9 (b) on the other hand show that even if all three trackers coincidentally perform with only $\approx 50\%$, successful tracking is still feasible as each tracker solves another facet of the overall tracking problem. In this way, the trackers obviously benefit from each other, resulting in an overall successful track. However, the fusion does not implicitly guarantee a successful tracking as will be shown in further experimental evaluations later-on.

6.3.2 Tracking Fusion Performance

We demonstrate the strength of our proposed tracking support fusion framework by robustly tracking objects that undergo highly non-rigid transformations. We compare our obtained tracking fusion results to the results of the individual trackers denoted by BT, HT, and MS. Moreover, we compare our results to a recent state of the art feature



(a)



(b)

Figure 6.9: **Tracking Congruences:** Tracking performances of BT, HT, and MS in terms of individual congruency curves for two different tracking scenarios, demonstrating that the fusion of diverse trackers allows for successful tracking (see cropped segmentation results) where one or multiple individual trackers tend to fail (frames 17, 26, 45, 65 in (a)), or where even all trackers' performances coincidentally tend to break down (frame 49 in (b)).

Sequence	**Images**	**TSF** [%]	**BT** [%]	**HT** [%]	**MS** [%]	**ORF** [%]
david	462	100	100	100	8	95
dudek	1145	100	100	100	72	100
car4	659	100	100	29	21	28
faceocc1	886	97	97	100	100	100
Average	787	**99.25**	**99.25**	82.25	50.25	80.75

Table 6.2: **Tracking Results on Standard Sequences:** Percentage of correctly tracked frames for standard sequences evaluated in relevant tracking literature. The individual trackers (BT, HT, MS) and a state of the art bounding box tracker (ORF) designed for these type of sequences are outperformed by the proposed tracking fusion method (TSF).

template fusion approach denoted by Visual Tracking Decomposition (VTD) [73], and to Online Random Forests (ORF) [105] bounding box tracking approach. Thus comparisons with state of the art trackers based on single cues and multiple fused cues are presented. Considering further state of the art trackers it should be mentioned that HT [45] outperforms MIL [4], ORF [105], and BHMC [72]; VTD [73] outperforms MIL [4], MS [28], MCMC [71], and IVT/OAL [101]; and MIL [4] outperforms FragTrack [2] on sequences that we also use for our evaluations, respectively.

We measure the tracking quality in percentage of correctly tracked frames, using the Pascal-VOC overlap criterion [34], which is defined as

$$score = \frac{R_T \cap R_{GT}}{R_T \cup R_{GT}}, \tag{6.10}$$

where R_T denotes the tracked area and R_{GT} defines the object area. A frame is counted as correctly tracked if $score > 0.5$, whereas we stop tracking if the tracker fails for the first time. Table 6.2 presents our obtained TSF results obtained by fusing BT, HT, and MS on rigid object tracking sequences evaluated in relevant literature and denoted by standard sequences. Comparisons with the individual trackers BT, HT, and MS as well as ORF representing a typical bounding box tracking approach are shown.

Table 6.3 presents obtained TSF results for non-rigid object tracking dynamic sequences, including highly complex object transformations, fast motion, abrupt appearance changes and camouflaged foreground and background structures, obtained by fusing BT, HT, and MS. The TSF results are compared to the results of the individual trackers BT, HT, and MS as well as to the results of VTD, representing a state of the art fusion based non-rigid object tracking approach.

Sequence	**Images**	**TSF** [%]	**BT** [%]	**HT** [%]	**MS** [%]	**VTD** [%]
Cliff-dive 1	76	100	100	100	100	100
Cliff-dive 2	69	100	33	100	52	27
Motocross 1	164	100	21	100	6	9
Motocross 2	23	100	79	100	74	70
Mountain-bike	228	100	18	100	100	87
Volleyball	500	58	26	100	59	57
Skiing	81	54	15	60	52	6
Transformer	124	100	100	100	100	53
High Jump	122	53	17	100	8	60
Diving	231	43	43	35	100	19
Gymnastics	767	90	33	10	100	92
parachute	51	100	100	100	100	100
girl	21	100	100	100	100	100
monkeydog	71	100	21	100	94	21
penguin	42	100	100	5	4	100
birdfall	30	100	73	100	60	16
cheetah	29	100	100	100	58	43
Ratatouille	87	100	100	64	100	100
Rango	241	100	64	100	100	40
Monster AG	372	100	53	87	100	31
Harry Potter	43	100	100	100	100	64
Coffee Mug	492	100	19	70	85	70
Average	175	**90.82**	59.77	83.57	75.09	58.74

Table 6.3: **Tracking Results on Dynamic Sequences:** Percentage of correctly tracked frames on dynamic sequences also evaluated in [45, 73, 120] and on sequences acquired by ourselves or downloaded from *YouTube*. Our fusion approach (TSF) clearly outperforms the individual trackers (BT, HT, MS) and the state of the art fusion based non-rigid object tracker (VTD).

6.3.3 Segmentation Quality

Although, our segmentation-based tracking fusion method does not rely on highly accurate segmentations, we also evaluate TSF on the sequences presented in [120], and compare our obtained segmentations with the results of HT [45], Tsai et al. [120] (MCT), and Chockalingam et al. [27] (AFT). Table 6.4 presents our obtained numerical results in terms of wrong segmented pixels (E), as well as the corresponding numbers for false negatives (FN) and false positives (FP). It can be seen that we perform best in 4 out of 6 sequences and that the rough segmentations give reasonable segmentation quality if compared to the ground truth.

Figure 6.10 depicts initial coarsely marked image regions and some illustrative seg-

Sequence	Images	TSF [px]			HT [px]	MCT [px]	AFT [px]
		E	FN	FP			
parachute	51	**219**	167	52	350	235	502
girl	21	4550	3817	733	3301	**1304**	1755
monkeydog	71	**427**	348	79	651	563	683
penguin	42	5040	4975	65	16097	**1705**	6627
birdfall*	30	**217**	92	125	271	252	454
cheetah	29	**855**	816	39	1037	1142	1217

Table 6.4: **Segmentation Quality:** Average number of wrong segmented pixels (E = error, FN = false negative, FP = false positive) per frame as also presented in [120]. The bold numbers mark the approach that performs best. *(* Note that TSF performed without MS as MS failed within the first few frames)*

mentation results for two different sports scenarios, obtained by the proposed tracking fusion approach. Both sequences are mainly characterized by camouflaged foreground and background regions as well as non-rigid object deformations.

Figure 6.10: **TSF Segmentation Results:** Initial coarsely marked tracking regions (left-most) and subsequent object segmentation results obtained by TSF for two different sports scenarios mainly characterized by camouflaged foreground and background regions and non-rigid object deformations.

6.3.4 Individual Improvements

In a final experiment we show that the iterative tracking support based segmentation and the corresponding back-propagation together increase the overall performance of the individual trackers without being fused with any other approach. This is mainly due to precise model and state updates that are feasible with the help of object segmen-

tations in each tracking iteration, allowing for successful suppression of noise and background structures during on-line updates. Table 6.5 presents two exemplary tracking scenarios for HT, BT, and MS where the iterative tracking support based segmentation significantly increases the individual tracking accuracy, compared to the performance of the original tracking approaches. Again the tracking accuracy is measured according to the Pascal-VOC overlap criterion [34] and by visual inspection for sequences without exact ground truth labels.

HoughTrack (HT)			
Sequence	**Images**	**HT** [%]	**TSF**$_{HT}$ [%]
penguin	42	5	100
Monster AG	372	87	100
Blending Tracker (BT)			
Sequence	**Images**	**BT** [%]	**TSF**$_{BT}$ [%]
Coffee Mug	492	19	100
Cliff-dive 2	69	33	49
Mean-Shift (MS)			
Sequence	**Images**	**MS** [%]	**TSF**$_{MS}$ [%]
david	462	8	25
High Jump	122	8	100

Table 6.5: **Individual Tracking Accuracy Improvements:** Percentage of correctly tracked frames of the individual trackers (HT, BT, MS) and the corresponding performances of the trackers using tracking support based iterative segmentation results for individual model and state updates. The improved update concept results in significantly increased tracking accuracies in several challenging scenarios.

In the following we present further illustrative results from diverse challenging tracking scenarios to demonstrate the robustness and flexibility of our approach. The sequences in Figures 6.11 - 6.15 contain real-world scenarios acquired by ourselves, movie scenes downloaded from the on-line video portal *YouTube*[2], as well as standard tracking sequences used in relevant literature [4, 45, 74, 105].

6.3.5 Discussion

We have shown that the fusion of completely different tracking outputs which are typically not directly comparable is feasible and definitely beneficial for all contributing trackers. Further we have demonstrated that using a segmentation step significantly in-

[2]http://www.youtube.com

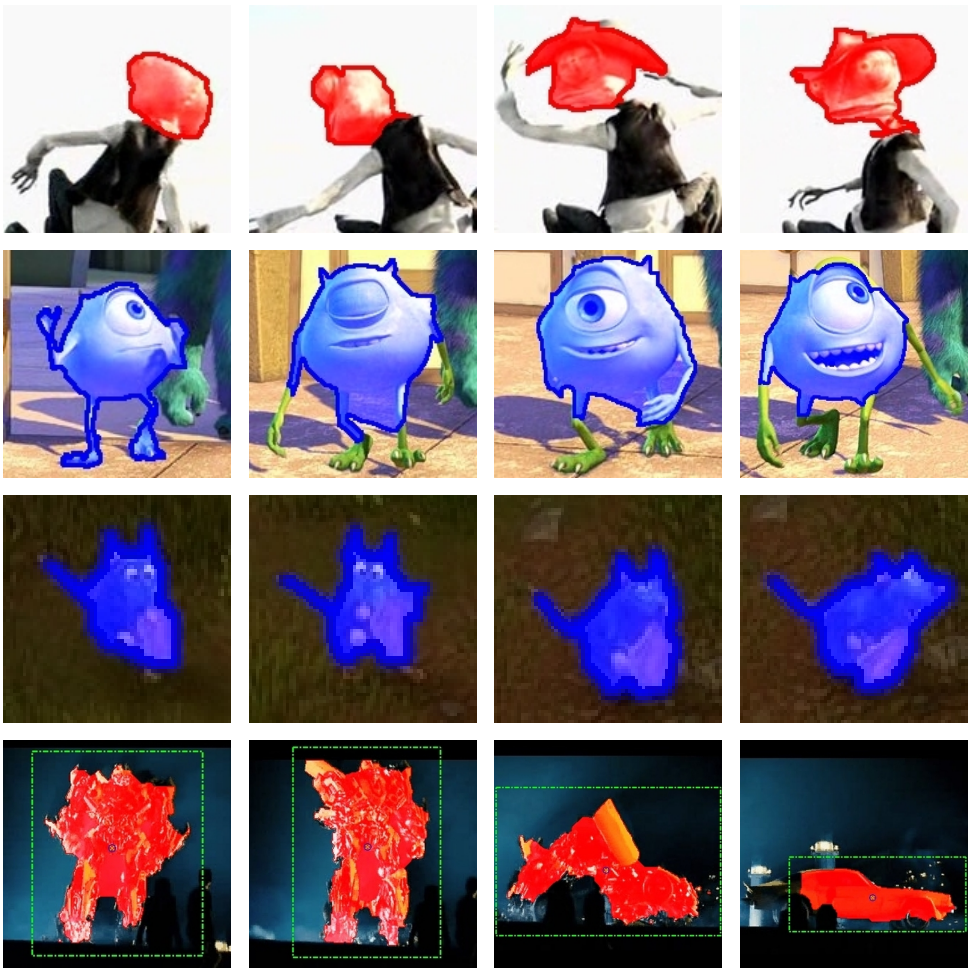


Figure 6.11: **Illustrative Tracking Results A**: Obtained (cropped) results from four animated movie sequences with non-rigid moving objects and severe appearance changes (*transformer* sequence) tracked.

creases the performance of each tracker as the amount of noise and background present in the data used for the updates gets significantly decreased.

For the three chosen tracking methods we have pointed out that the fusion of individual completely different tracking outputs increases the tracking accuracy, even if one or even more individual trackers temporally fail. Moreover, the fusion allows for rapid recovery of the individual trackers due to its iterative optimization character. Thus, the proposed fusion approach obviously outperforms, e.g., simple majority vote approaches or confidence based selection from different cues (late fusion) as presented in [5], as well as the fusion of several similar cues as presented in [73] due to the additional perfor-

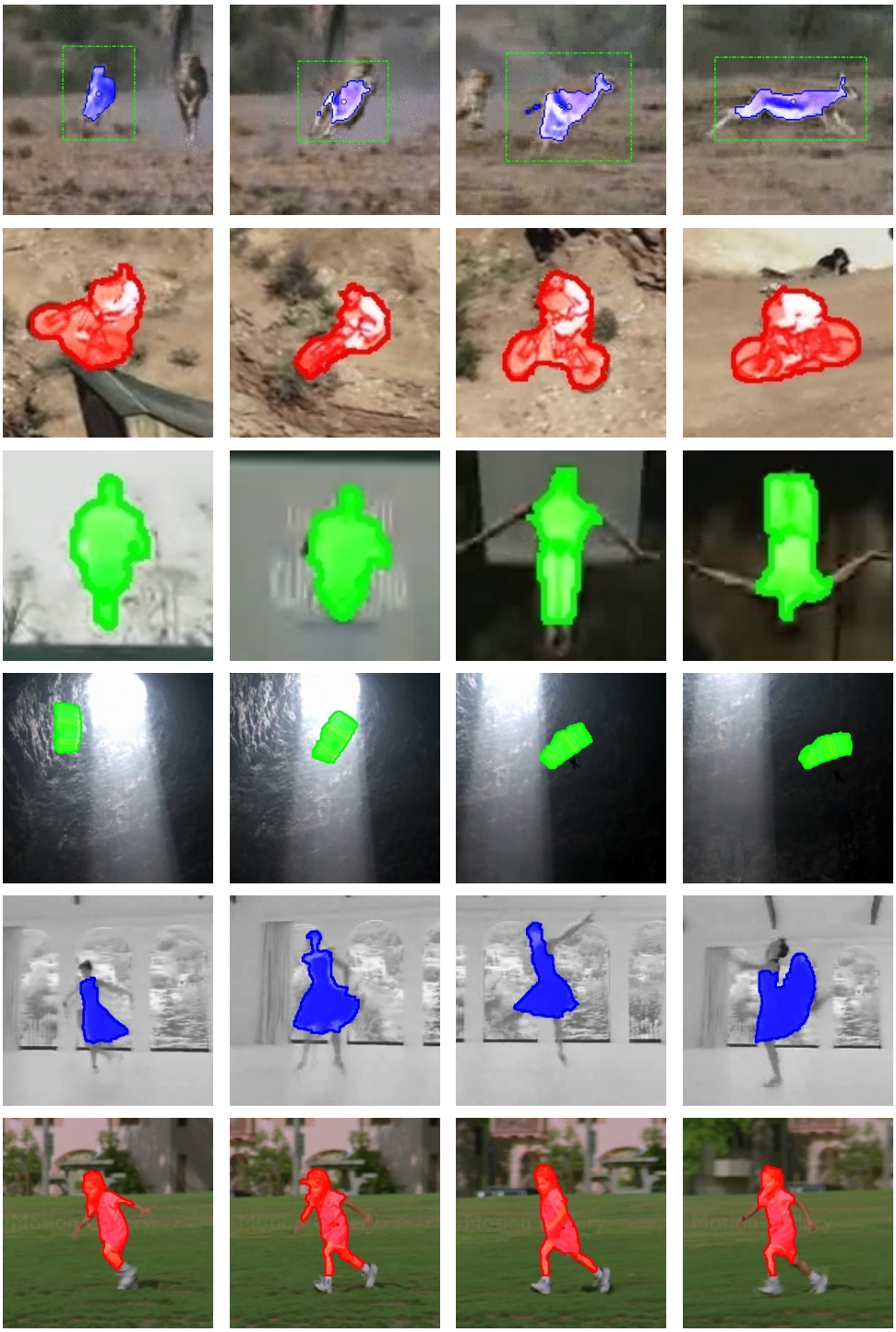


Figure 6.12: **Illustrative Tracking Results B**: Obtained (cropped) results from six real-world and sports tracking scenarios that are characterized by large baseline motion, camouflaged foreground background regions, and severe appearance changes due to out of plane rotations.

Figure 6.13: **Illustrative Tracking Results C**: Obtained (cropped) results from an exemplary dynamic non-rigid object tracking scenario.

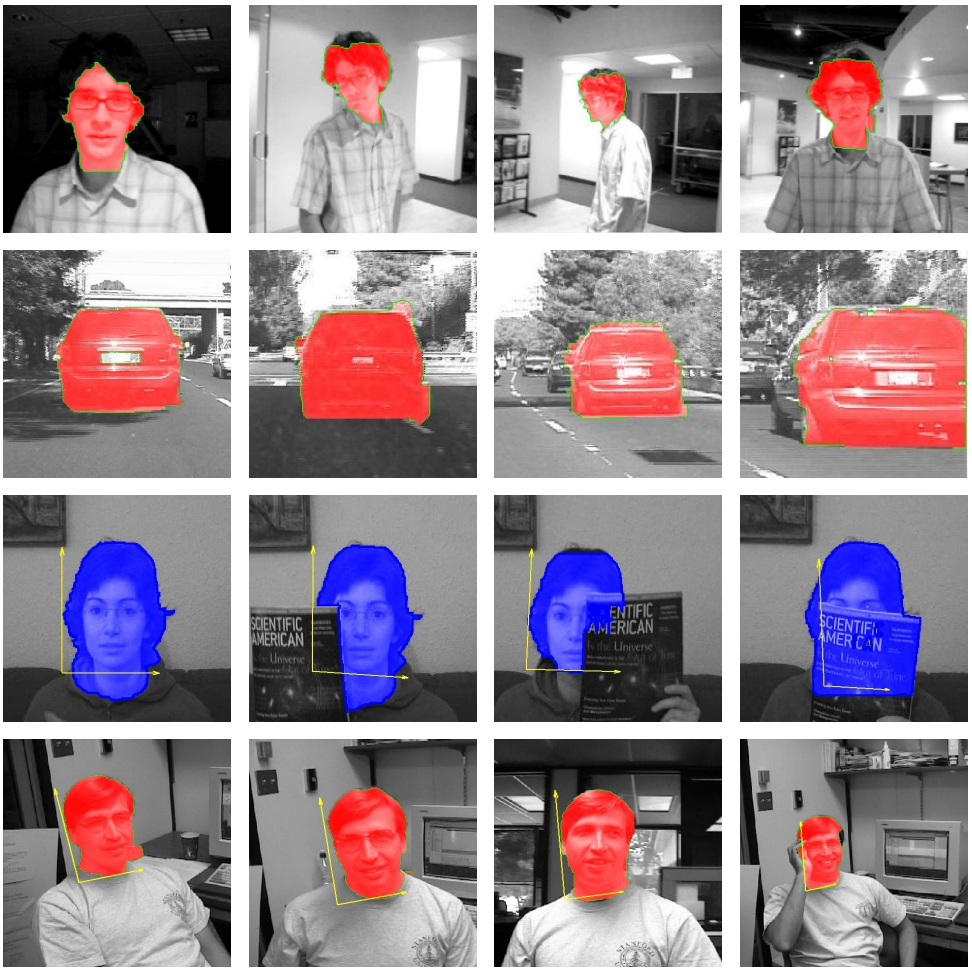


Figure 6.14: **Illustrative Tracking Results D**: Obtained (cropped) tracking results from standard gray scale tracking scenarios.

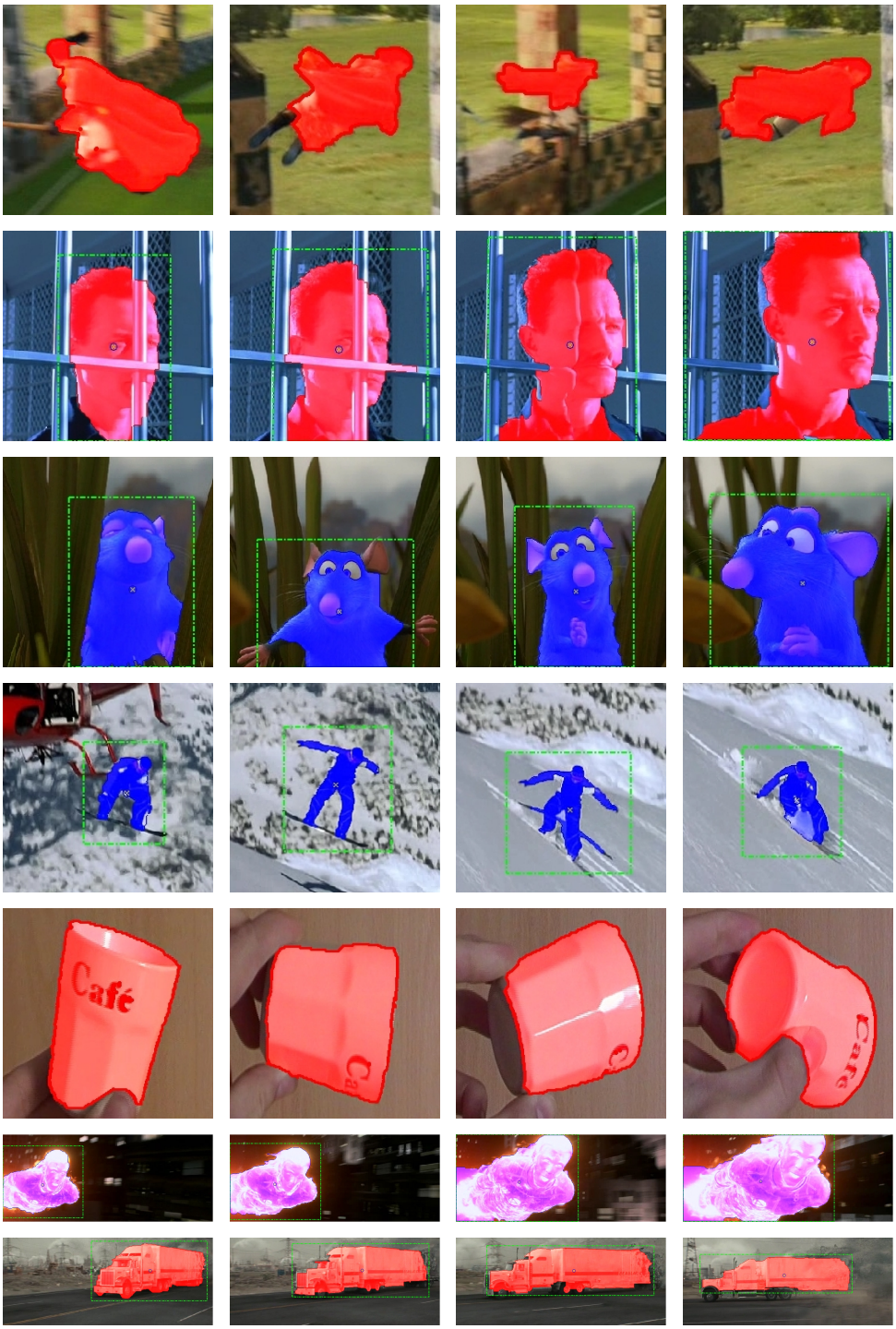


Figure 6.15: **Illustrative Tracking Results E**: Obtained (cropped) tracking results for severe motion, camouflaged foreground and background, homogeneous and symmetric object surface (*coffee mug*), and severe appearance change scenarios.

mance based iterative weighting scheme. The reasons for the obtained accuracy and robustness are on the one hand the fusion of the individual advantages of the contributing tracking approaches (according to different features, representations and degrees of flexibility), and on the other hand the beneficial back-propagated object segmentation which allows for more precise object model and state updates, respectively. In more detail, the fusion of geometric consistency obtained by the template tracker, object feature consistency obtained by the discriminative tracker, and color consistency obtained by the kernel tracker results in an overall very robust tracking approach that exhibits a high degree of invariance as shown in the different experimental evaluations. Moreover, also individual improvements for the different tracking approaches on their own are feasible.

The tracking failure in Table 6.2 for the faceocc1 gray scale sequence derives from an over-segmentation, which we classified as tracking failure (see Figure 6.16). The lower TSF tracking accuracies in Table 6.3, e.g., for the Diving or Skiing sequences compared to the individual results of all contributing trackers result from coincidental mismatches of a majority of the contributing trackers over several consecutive frames. In detail, the lower tracking results originate from low image resolutions (sequences Diving and Skiing) and similar foreground and background color or gray values (sequences faceocc1 and Gymnastics), resulting in wrong or missing segmentations and consequently tracking failures according to the Pascal-VOC overlap criterion used as performance criterion. The High Jump sequence and the Volleyball sequence are not fully tracked due to heavy occlusions, which are not explicitly handled by any of the fused tracking approaches. The integrated segmentation step is able to handle partial occlusion, but the combined trackers do not support a re-detection of the object once a full occlusion occurred. Figure 6.16 exemplary depicts some of the here discussed tracking failure cases.

Moreover, if the underlying segmentation algorithm is not able to find a reasonable result, the tracking fusion fails due to incorrect tracking state and model updates caused by the looped back segmentations. This can happen if e.g. foreground and background exhibit highly similar structures, texture or color. Figure 6.17 illustrates such segmentation failures for the task of weld seam tracking, where both, background and welded seam are similar in color and texture.

Figure 6.18 illustrates further segmentation errors for the specific task of agricultural hanger tracking. Thereby similar foreground (hanger) background (grain) colors and blurry or homogeneous image regions result in over- or under-segmentations. Although, the hanger is tracked correctly, the obtained intermediate segmentations are not

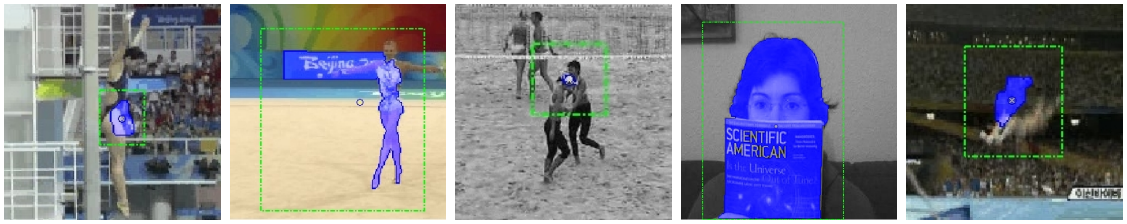


Figure 6.16: **TSF Failure Cases:** Tracking failures of the presented fusion approach are caused by either missing or heavy over-segmentations and background bleeding. This happens if the majority of contributing trackers coincidentally fails for several consecutive frames, thus providing wrong support sets including significant amounts of noise and background image pixels.

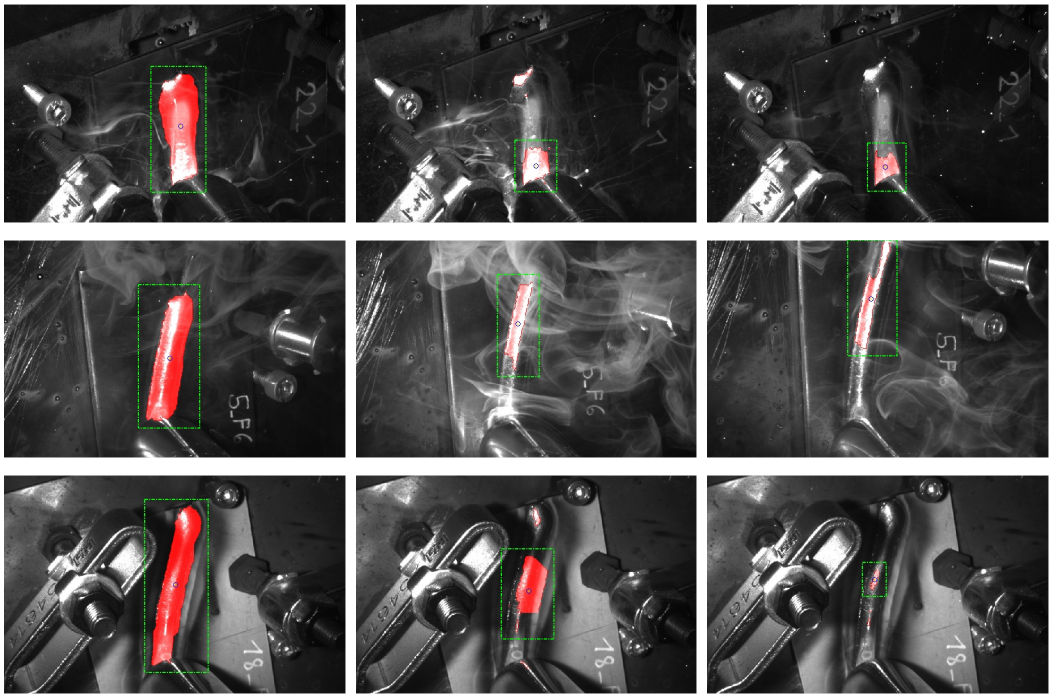


Figure 6.17: **Segmentation Failures in Harsh Industrial Environments:** If the underlying segmentation algorithm is not able to compute a reasonable segmentation, in this case for the task of weld seam tracking where welded seam and background are alike in terms of color, texture and similar structures, TSF fails due to incorrect tracking state and model updates caused by the looped back segmentations.

convincing or convenient.

Figure 6.18: **Segmentation Failures in Agricultural Outdoor Environments:** Similar foreground background colors or blurry and homogeneous regions result in either over- or under-segmentation results. Although, the hanger is tracked correctly, the intermediate segmentations are not convenient and satisfactory.

This clearly shows that the fusion of several diverse tracking approaches does not implicitly result in an improved overall performance. For instance, consider the fusion of tracking approaches that are robust to, e.g., large baseline motion and out-of-plane rotations. If such a fused tracking approach is then applied to scenarios where, e.g., severe object appearance changes or varying illumination conditions occur, the tracking will obviously fail. Thus, it is important to combine heterogeneous tracking approaches where the combined capabilities of the individual trackers cover the entire spectrum of scenarios that are addressed.

6.4 Conclusion

In this Chapter, we have presented a novel segmentation-based method to fuse heterogeneous single target trackers that report individual significantly different tracking outputs which typically cannot be directly combined. We have shown how to transform the different outputs to a common representation, where they can be successfully combined. Further we have shown that using object segmentations for individual trackers' updates results in robust and stable tracking especially for the non-trivial task of non-rigid object tracking, as the amount of noise and background present in the data utilized for the update step gets significantly decreased.

To demonstrate the feasibility of the presented fusion concept, we have chosen three complementary tracking approaches and fused their individual outputs: **(a)** a template tracking method that reports dense projective homography inliers, **(b)** a discriminative Hough-based tracking algorithm that delivers sparse foreground votes, and **(c)** a feature histogram based Mean-Shift tracker that reports a covariance ellipse around the target center location. The individual approaches struggle with typical tracking problems like heavy occlusions or abrupt illumination changes. However, in our experiments we demonstrate that the fusion of the individual trackers and consequently of their specific individual tracking outputs significantly increases the overall tracking robustness and quality. Moreover, we have shown that the performance based weighting of the trackers during the fusion allows for successfully tracking even if the majority of contributing approaches coincidentally fails.

Extensive evaluations on several dynamic real-world scenarios, well known standard sequences and challenging sequences from movies clearly show that our proposed tracking fusion approach outperforms the individual tracking approaches, state of the art single cue trackers as well as fusion-based methods. Although, we do not focus on accurate object segmentation, our evaluations show that we can even compete with the state-of-the-art in this respect. Thus, we come to the conclusion that the fusion of individual tracking outputs apparently increases the overall tracking accuracy as individual strengths are successfully combined. Furthermore, the back-propagation of the improved result significantly increases the performance of individual and of fused tracking approaches. However, a fusion of different trackers does not implicitly guarantee a successful tracking or accuracy improvements if an inappropriate selection of different tracking cues is made for a specific problem. Moreover, the segmentation based fusion of trackers does not work in scenarios where object and background exhibit similar colors and structures, like e.g. in the weld seam tracking scenario presented in Chapter 4. Although, different tracking supports can be successfully fused, the segmentation algorithm fails due to foreground and background similarities in color and texture. In order to perform a successful fusion for such kind of problem, trackers that provide dense supports should be used, whereas the regularization should be performed directly on the fused support sets instead of incorporating ambiguous image data using an image segmentation approach.

7

Template-based Visual Quality Inspection

In this Chapter we present an image based quality inspection pipeline that allows for assessing the quality of robotic weldings or welding processes, and that consists oft a semi supervised and an unsupervised quality inspection approach. Both methods thereby rely on image templates, provided by a tracking approach. The tracker allows for extracting axis aligned templates depicting newly welded seam behind the welding torch. Thus, in this Chapter we demonstrate how image templates can be successfully used for robust visual quality inspection in harsh industrial welding environments. The first semi-supervised quality inspection approach relies on manual weld seam classification by visual inspection of survey or panorama images of entire weldings composed from weld seam template sequences. Thereby, a welding expert classifies the complete welding process by visual inspection of the final noise free panorama image. The focus of this semi-supervised approach mainly lies on image noise reduction and on the accomplishment of high panorama image qualities with respect to high image resolutions and high signal-to-noise ratios as the input templates typically include a lot of visible noise, typical for industrial environments. The second approach is an unsupervised quality inspection method that relies on an off-line trained error-free weld seam appearance model. During on-line welding new weld seam templates are compared to the learned appearance model in real-time, where highly dissimilar images are reported as welding defects. We thereby present two different application scenarios that either automatically classify an entire welding or the individual local weld seam templates into either error-free or defective. The classification of the complete welding sequence should be applied

for robotic weldings where small punctual defects do not affect the overall welding quality, whereas the local inspection approach is useful, e.g., in automotive industry where even small welding defects require the specimen to be replaced or repaired.

7.1 The Weld Seam Inspection Framework

The proposed weld seam inspection framework consists of a weld seam tracking module and of two different image template based weld seam quality inspection methods. Basically, the track introduced in Chapter 4 allows for extracting axis aligned weld seam templates from welding image sequences, depicting newly welded seam. These templates are then passed to the two individual quality inspection modules as input. The first method which is presented in detail in Section 7.2 is a semi-supervised quality inspection approach that uses the weld seam template in order to incrementally generate a noise free high quality panorama image of the complete welding. The second weld seam quality inspection method is presented in Section 7.3. It is an unsupervised approach that analyzes the new incoming weld seam template and computes an appearance based similarity to an off-line learned weld seam appearance model. This allows for automatically classifying either the entire welding sequence or the individual weld seam templates into error-free or defective, depending on the underlying quality constraints and requirements. Figure 7.1 illustrates the weld seam inspection pipeline including some exemplary images.

7.2 Incremental Welding Panorama Image Generation

In order to perform welding quality assessment in a semi-supervised manner, we propose to utilize the weld seam image templates provided by the weld seam tracking approach for incrementally generating an overview or panorama $\mathbf{I}_p$ of the entire welding in an autonomous manner. Thereby, we especially focus on image denoising and prevention of image blending artifacts as the weld seam templates typically include a severe amount of visible noise. In this way we obtain high quality panorama images that allow for further determining the overall quality at a glance. Therefore, the final generated panorama image is presented to the operator or to a welding expert after the welding task is finished. This allows for detecting potential welding errors and defects in a single overview image, where typical image noise or distortions including smoke, sparks, evaporating water or small gas disturbances are effectively suppressed or re-

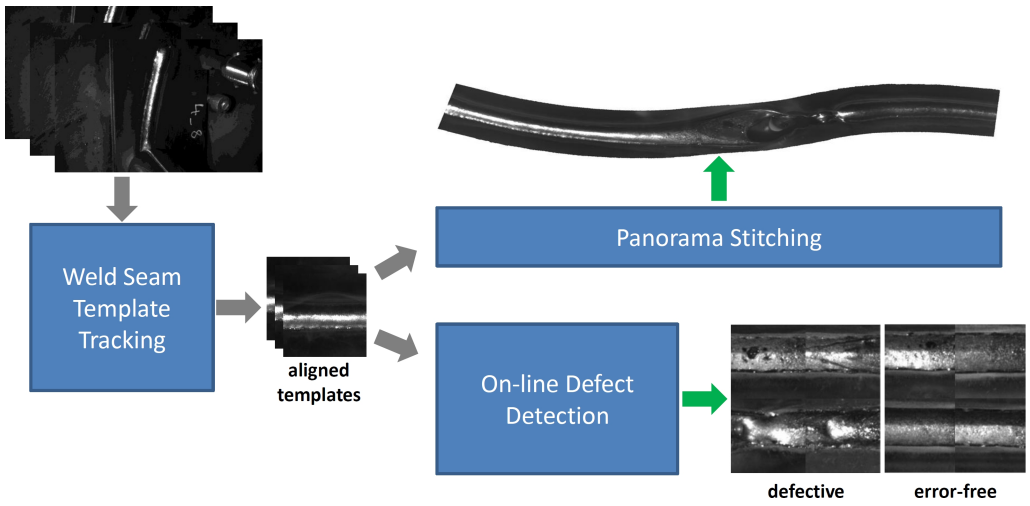


Figure 7.1: **Weld Seam Inspection Framework:** A robust weld seam tracker allows for extracting axis aligned templates from welding image sequences. The templates are then used as input for an incremental panorama image generation approach and for appearance based defect detection.

moved. Thus, the operator is able to determine the overall quality of the actual welding without performing additional material handling, which consequently reduces working time and prevents from additional costs. A second important advantage of this proposed quality inspection approach is that erroneous regions in $\mathbf{I}_p$ can be directly mapped to their temporal location on the weld seam. This in turn allows targeted repairing or re-finishing operations. Figure 7.2 shows several templates from a welding sequence that are successfully combined to a panorama image of the entire welding with significant reductions of the present image noise.

7.2.1 Weld Seam Templates

Weld seam templates $\mathbf{I}_1 \cdots \mathbf{I}_n$ obtained from an underlying weld seam tracker serve as input for the panorama image generation. As the tracker provides a quite accurate pose of the weld seam in each image, axis aligned templates $\mathbf{I_i}$ can be extracted directly around the weld seam in a region where adverse effects and distortions are minimal. Such effects usually appear in regions close to the welding arc or near the melt pool. Consequently, the required weld seam templates $\mathbf{I_i}$ used for the panorama image generation are ideally extracted at point $\mathbf{x}_\infty$ which exhibits the largest distance from the fixed welding point $\mathbf{x}_w$ and the fluid melt bath, while remaining completely visible in the actual frame. Figure 7.3 exemplary depicts three welding images with marked weld

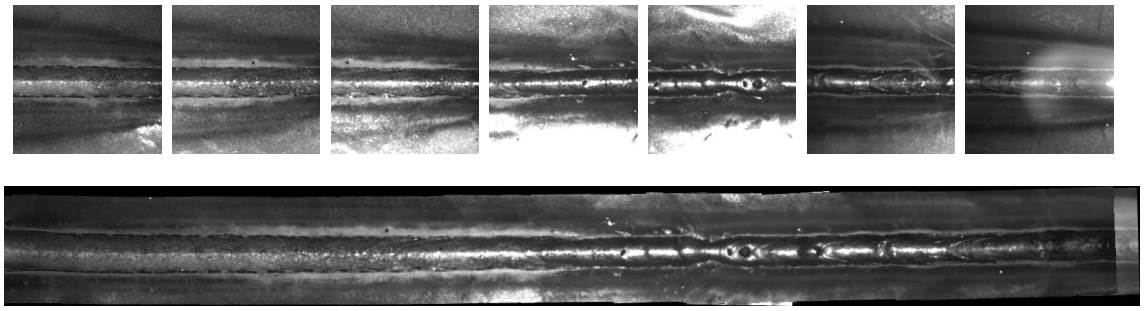


Figure 7.2: **Weld Seam Panorama Image Generation:** Incoming weld seam templates are combined to a single overview image, that allows a welding expert to classify the welding at a glance. Thereby, specific image preprocessing and blending strategies result in a significant reduction of typical industrial image noise, which makes the classification easier.

seam template locations.

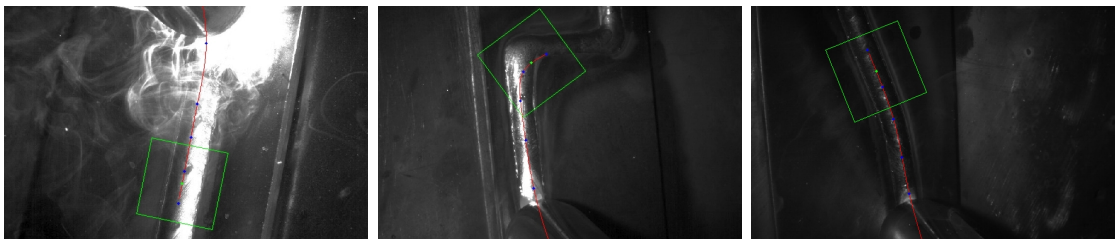


Figure 7.3: **Weld Seam Template Locations:** Weld seam tracking provides quite accurate localizations of the weld seam in the actual image. This allows for extracting weld seam templates (green rectangles) from a region where adverse effects and distortions are minimal.

The panorama image generation needs to cope with difficulties and complications such as severe noise and unforeseen shape and appearance changes that aggravate the image processing steps. To overcome the noise in the input templates, we exploit the high redundancy that is present in the incoming templates $\mathbf{I}_i$ due to their large overlap of up to 80%, depending on the underlying welding speed as well as on the corresponding robot trajectory. While straight welding results in continuous regular distances between consecutive weld seam images, welding along curves like, e.g., shown in the middle image of Figure 7.3 might result in varying distances and hence different amounts for the overlap between consecutive weld seam templates. Figure 7.4 depicts this issue for two different welding scenarios. The templates in the upper row exhibit larger overlaps than in the bottom row as the welding robot remains on a spatial position while performing

a rotation, whereas the straight welding results in regular and continuous distances.

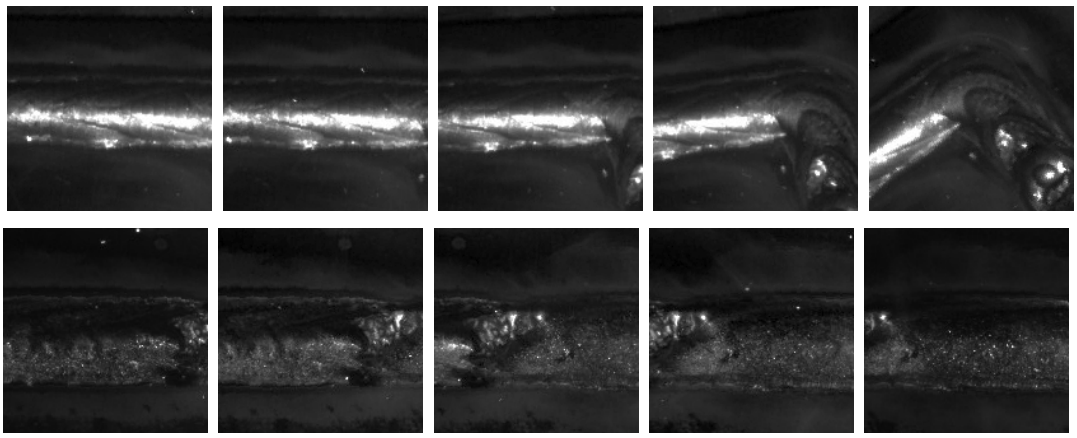


Figure 7.4: **High Overlap Redundancy:** High redundancies due to large overlaps of consecutive weld seam templates allows for successful suppression of noise in the panorama. Different welding speeds and varying robot trajectories thereby result in different and varying amounts for the overlap.

7.2.2 The Stitching Pipeline

The proposed stitching pipeline or welding panorama image generation consists of three iterative stages that are passed for each incoming template, and a final step that performs a global refinement. As the entire stitching pipeline works in an iterative manner, the growing panorama can be observed until the last template is added. Algorithm 6 presents a formal overview of the proposed panorama stitching pipeline and it's intermediate steps. As the proposed stitching pipeline is working in an iterative manner, an image or template sequence is assumed as input. Consequently, extracted axis aligned weld seam templates from consecutive welding images are suitable as input for the panorama image generation approach. The desired weld seam panorama image $\mathbf{I}_p$ gets initialized with the first incoming template $\mathbf{I}_1$. Subsequent templates are then registered to their corresponding predecessors in order to obtain a local rigid motion transformation matrix $\mathbf{H}_k$ that describes the mapping from $\mathbf{I}_k$ to $\mathbf{I}_{k-1}$, respectively. The computed local transformation matrices are iteratively accumulated to a global rigid motion transformation $\mathbf{H}_G$. This allows for correctly warping each incoming template to the global mosaic coordinate frame, resulting in an iteratively growing panorama image.

Algorithm 6 Incremental Weld Seam Panorama Image Generation:

Input: a sequence of weld seam templates $\mathbf{I}_1 \cdots \mathbf{I}_n$
Output: panorama image $\mathbf{I}_p$

(a) Initialize mosaic $\mathbf{M}_1$ with first template $\mathbf{I}_1$

(b) Initialize global rigid motion transformation $\mathbf{H}_G$ with identity matrix $\mathbf{I}$

for $k = 2$ to n **do**

(c.1) Perform image preprocessing of $\mathbf{I}_k$

(c.2) Register $\mathbf{I}_k$ to previous template $\mathbf{I}_{k-1}$ and compute local transformation $\mathbf{H}_k$

(c.3) Update global rigid motion transformation according to $\mathbf{H}_G = \mathbf{H}_G\,\mathbf{H}_k$

(c.4) Warp $\mathbf{I}_k$ to $\mathbf{I}_k^W$ using $\mathbf{H}_G$

(c.4) Compute $\mathbf{M}_k$ by blending $\mathbf{I}_k^W$ over $\mathbf{M}_{k-1}$ and by adding new template regions

end for

(d) Perform global refinement of $\mathbf{M}_n$

(e) Perform final blending on $\mathbf{M}_n$ to get $\mathbf{I}_p$

Image Preprocessing

In order to cope with the large amount of noise that is present in the incoming templates, we propose a twofold preprocessing. First, an adaptive histogram equalization that significantly improves the registration quality is performed, followed by an image dehazing procedure that enhances the image blending if global haze or smoke are visible in the incoming images. Typically, incoming weld seam templates do not exhibit high contrast especially in the presence of smoke or evaporating water in the images. Contrast-limited adaptive histogram equalization [141] allows for enhancing the contrast of gray scale images, which consequently results in more accurate and comprehensive feature registrations. Exhaustive evaluations in [75] have shown that the adaptive histogram equalization technique in [141] is the best way to significantly increase the number of feature matches and consequently the quality of the resulting image registration. Second, image dehazing presented by He et al. [53] and which actually intents

to reduce the amount of visible haze from outdoor images is applied. However, the underlying assumptions on light ray modifications while passing through haze or fog can also be applied for light rays passing through welding smoke. The dehazing algorithm relies on a dark channel prior and consists of a transmission estimation, a soft matting and a recovering of the scene radiance and the atmospheric light. The dark channel for a color image $\mathbf{J}$ is defined as

$$\mathbf{J}^{dark}(\mathbf{x}) = \min_{c \in \{r,g,b\}} \left(\min_{\mathbf{y} \in \Omega(\mathbf{x})} \left(\mathbf{J}^{c}(\mathbf{y}) \right) \right), \tag{7.1}$$

where $\mathbf{J}^c$ denotes a color channel of image $\mathbf{J}$ and $\Omega(\mathbf{x})$ is a local window centered at $\mathbf{x}$. The transmission can be estimated with a given dark channel prior $\mathbf{J}^{dark}$ according to

$$t(\mathbf{x}) = 1 - \mathbf{J}^{dark}, \tag{7.2}$$

where ω denotes an application based haze constant in the range of $0 < \omega \leq 1$. In [53] the authors propose to refine the transmission based on soft matting and based on a sparse linear system given by

$$(\mathbf{L} + \lambda \mathbf{I}) t_{refined} = \lambda t\,. \tag{7.3}$$

Thereby, $\mathbf{L}$ is a matting Laplacian matrix [77], $\mathbf{I}$ is an identity matrix with the same size as $\mathbf{L}$, and λ is a regularization parameter. However, soft matting and the calculation of the matting Laplacian matrix are computationally expensive operations and hence not tractable for our panorama image generation approach. Instead we apply a guided filter approximation proposed in [54], where the transmission estimation refinement is given by lower bound (t_0) restricted guided filtering of the hazy image.

$$t_{refined}(\mathbf{x}) = max\left(t_{refined}(\mathbf{x}), t_0\right) \tag{7.4}$$

The final scene radiance $\mathbf{J}(\mathbf{x})$ for an image $\mathbf{I}$ can be recovered according to

$$\mathbf{J}(\mathbf{x}) = \frac{\mathbf{I}(\mathbf{x}) - \mathbf{A}}{max\left(t_{refined}(\mathbf{x}), t_0\right)} + \mathbf{A}\,, \tag{7.5}$$

where $\mathbf{A}$ denotes the atmospheric light which is estimated from the most haze-opaque pixel. Thereby the dark channel is used for the estimation by first choosing the top 0.1% brightest pixels in the dark channel $\mathbf{J}^{dark}$. Among these pixels the corresponding one

with the highest intensity in the input image **I** is finally selected as atmospheric light. In this way, selection of, e.g., white objects as atmospheric light is neglected. Figure 7.5 illustrates the effects of image dehazing as a preprocessing step in our stitching pipeline. Smoke wads and global haze which are visible in several incoming weld seam templates are successfully dampened or removed in the resulting panorama image.

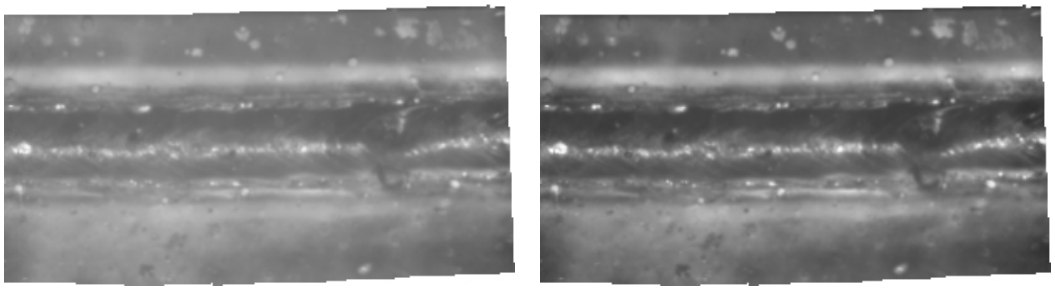


Figure 7.5: **Image Dehazing:** In the presence of smoke wads and global haze in several incoming weld seam templates dehazing significantly increases the quality of the computed weld seam panorama. The left image shows a partial weld seam panorama without image dehazing, whereas the image on the right depicts the positive effects of dehazing on the panorama quality. [75]

Local Registration and Global Transformation Update

The task of a local registration is to align incoming weld seam templates according to an underlying motion model with previous templates. An underlying weld seam tracking approach allows for extracting axis aligned weld seam templates. Moreover, we assume local planarity of the weld seam depict in the template. These assumptions and constraints justify the choice of an image feature based rigid motion model as the underlying motion model. Due to the large amount of image noise and the consequently large amount of potential feature matching outliers the robust RANSAC [40] estimator is applied for the rigid motion estimation.

The fundamental step for the local image registration is given by the extraction of image keypoints and by the computation of highly invariant and robust descriptors. Although, there exist various robust concepts like, e.g., SIFT [81], SURF [6], BRIEF [21], or FAST [103], exhaustive evaluations in [75] suggest to use a combination of FAST corners as keypoints and SIFT as descriptors. This combination on the one hand allows for extracting a larger amount of keypoints compared to other techniques, and on the other hand the SIFT descriptor exhibits the necessary invariance to rotations, translations,

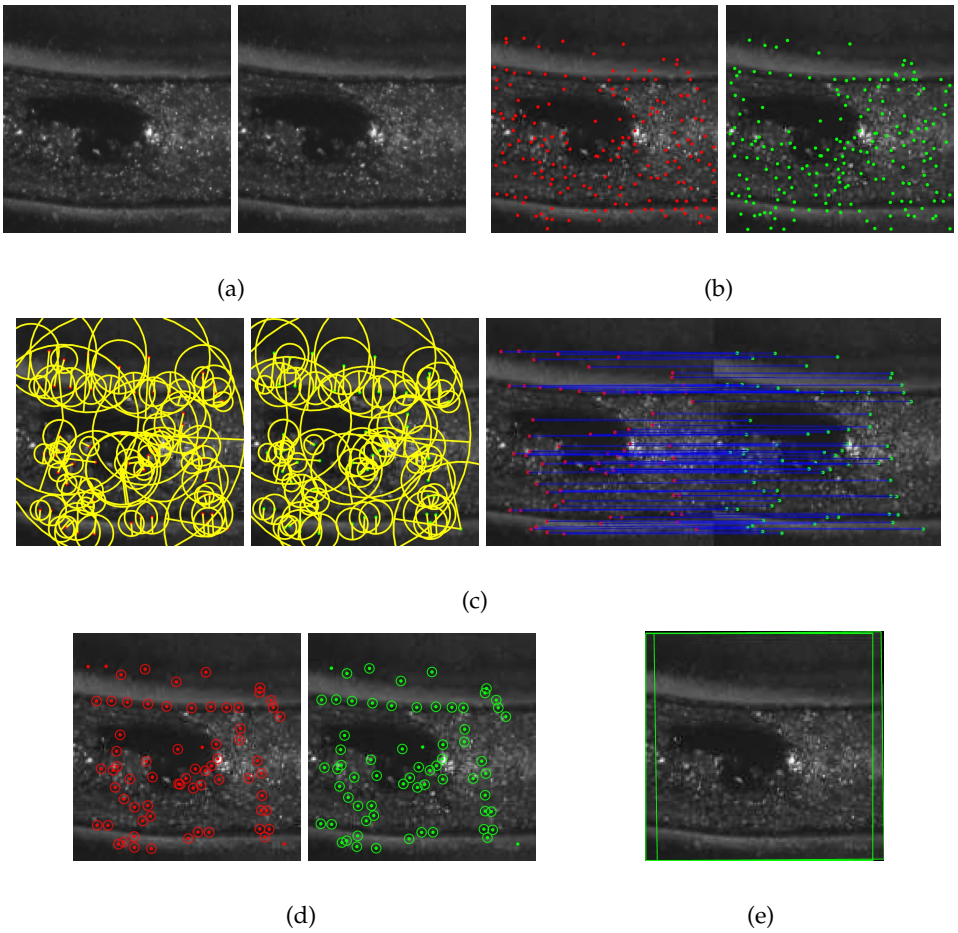


Figure 7.6: **Robust SIFT Feature-based Weld Seam Template Registration:** Incoming templates (a) are registered using SIFT (b). Obtained SIFT matches (c) are robustly filtered (d), in order to obtain an alignment (e). [75]

scale, and partially to illumination changes. Feature matching between extracted SIFT keypoints then relies on distance evaluations of corresponding SIFT descriptors. Two SIFT descriptors $\mathbf{d}_1$ and $\mathbf{d}_2$ match if the Euclidean distance d_ϵ in between multiplied by a threshold τ is smaller than the distances to n nearest neighbor descriptors $\mathbf{d}_i$.

$$\frac{d_\epsilon\left(\mathbf{d}_1, \mathbf{d}_2\right)}{d_\epsilon\left(\mathbf{d}_1, \mathbf{d}_i\right)} < \frac{1}{\tau}\ ,\ i = 3 \cdots n \tag{7.6}$$

We compute the local rigid motion for each incoming template to its predecessor via the generalized Procrustes alignment algorithm [46] in a robust manner using the RANSAC [40] estimator, resulting in a local rigid motion transformation $\mathbf{H}_k$ from image

$\mathbf{I}_k$ to $\mathbf{I}_{k-1}$ at time k. Figure 7.6 illustrates the robust local keypoint based registration as well as the resulting alignment. In order to align new incoming templates with an existing partial panorama an additional incrementally updated transformation to a global mosaic coordinate frame is required. Therefore, the first incoming template $\mathbf{I}_1$ coincidentally defines a global mosaic coordinate frame for the panorama. In order to be able to warp each incoming template to this global coordinate frame, a global transformation $\mathbf{H}_G$ up to time k gets continuously updated by accumulating local rigid motion transformations.

$$\mathbf{H}_G = \mathbf{H}_1 \, \mathbf{H}_2 \, \mathbf{H}_3 \, \cdots \, \mathbf{H}_{k-1} \, \mathbf{H}_k \tag{7.7}$$

Finally, a new template $\mathbf{I}_k$ can be aligned with an existing partial panorama by warping $\mathbf{I}_k$ to the global mosaic coordinate frame using the updated global transformation $\mathbf{H}_G$. Figure 7.7 again illustrates the local and global registration and transformation operations for a new incoming template $\mathbf{I}_k$.

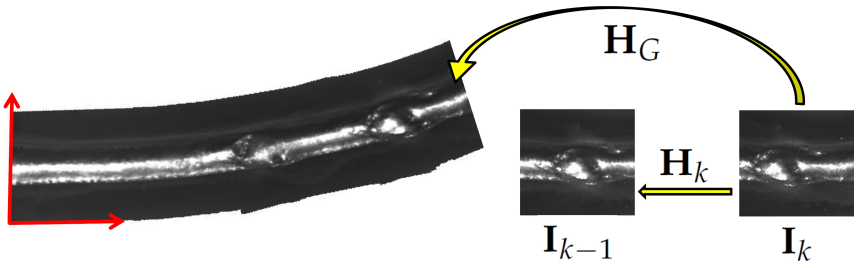


Figure 7.7: **Local and Global Registration:** Incoming templates $\mathbf{I}_k$ are first aligned with their predecessors in order to accumulatively update a global transformation $\mathbf{H}_G$ with the local rigid motion transformation $\mathbf{H}_k$. At last the template $\mathbf{I}_k$ is warped to a global mosaic coordinate frame and gets aligned with an existing partial panorama.

Table 7.1 presents the registration quality for repeated weldings of 5 different welding processes for SIFT [81], SURF [6], KLT [83] and the pyramid optical flow approach denoted by OFPyrLK [13]. The results clearly underline the robustness of SIFT as it outperforms the other approaches.

Incremental Image Blending

Once a template is aligned with the existing partial panorama, an adequate image blending strategy is required in order to obtain a high quality panorama image with minimal visible noise. Thus, we suggest a transition smoothing approach with an adaptive weighting scheme for image blending. Important aspects that need to be considered within the desired weighting scheme are the incremental behavior of the panorama

Approach	Standard Deviation [px]					
	x	y	x	y	x	y
Curved Welding a						
SIFT	2.69	2.01	4.13	8.87	4.21	18.66
SURF	2.50	3.31	4.33	10.90	4.22	24.02
KLT	2.67	2.86	4.46	9.87	4.66	19.28
OFPyrLK	10.71	1.54	10.93	9.49	10.55	26.81
Curved Welding b						
SIFT	8.10	7.01	14.86	21.17	15.71	54.21
SURF	8.84	17.25	26.11	41.14	28.09	91.11
KLT	4.69	14.03	25.07	25.68	24.91	69.70
OFPyrLK	79.99	6.77	83.36	25.80	83.13	65.12
Straight Welding a						
SIFT	1.46	2.17	2.36	6.76	3.52	12.69
SURF	1.74	2.86	3.20	8.73	4.70	19.95
KLT	1.98	5.30	4.39	20.45	8.44	49.77
OFPyrLK	1.73	5.26	3.62	15.10	5.02	34.02
Straight Welding b						
SIFT	1.00	1.72	2.20	4.35	4.30	7.84
SURF	1.17	2.35	3.22	6.64	5.19	13.77
KLT	0.85	4.21	2.17	11.31	4.44	20.20
OFPyrLK	1.08	1.82	2.86	6.05	23.55	12.51
Straight Welding c						
SIFT	2.23	2.09	6.54	6.94	6.85	15.55
SURF	8.29	5.10	21.58	20.53	24.71	47.15
KLT	1.88	2.48	5.56	8.78	6.06	20.13
OFPyrLK	4.07	4.58	17.55	13.93	17.36	27.13
Average						
SIFT	**3.10**	**3.00**	**6.02**	**9.62**	**6.92**	**21.79**
SURF	4.51	6.17	11.69	17.59	13.38	39.20
KLT	2.41	5.78	8.33	15.22	9.70	35.82
OFPyrLK	19.52	3.99	23.66	14.07	27.92	33.12

Table 7.1: **Registration Comparison:** Standard deviations evaluated at three different trajectory nodes. A node is thereby located at the central position of the relevant template in the mosaic coordinate frame. The five experiments each contain the average values from 19 sequences of a repeated welding task, resulting in overall $\approx$ 4200 pairwise registrations per experiment. [75]

generation as image sequences are processed, robust handling of overexposure caused by, e.g., bright sparks or gas disturbances, and suppression of visible noise caused by, e.g., evaporating water, smoke wads or sparks and spilling. Due to the mentioned large overlap of consecutive templates a large amount of redundancy is available. We utilize this redundancy on the one hand for image noise removal and on the other hand for resolution enhancement of the resulting panorama image. Several consecutive templates that depict a common weld seam region do not exhibit the same amount of image noise. Thus, it is feasible to suppress noise and disturbances present in single templates. As the registration results in sub-pixel accuracy, the resolution of the panorama gets automatically increased. The final resolution of the panorama mainly depends on the amount of overlap as well as on the underlying welding speed.

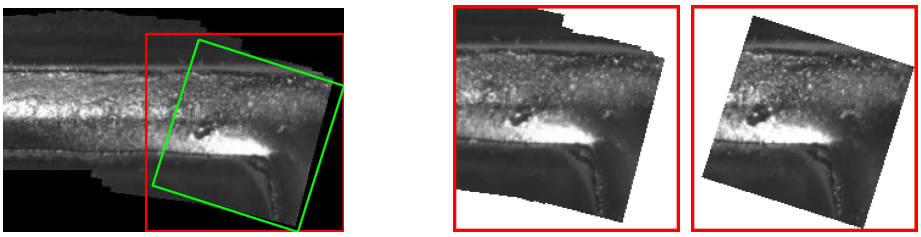


Figure 7.8: **Image Blending Update ROI:** For image blending only those parts of an existing partial panorama are considered, which are included in the warped version of a new template to be added. The left image illustrates such a region of interest (ROI). On the right the corresponding panorama image region and the warped template which should be combined are shown. [75]

In order to perform on-line image blending, only small parts of an existing partial panorama image are considered for image blending updates. These parts are given by image regions which are included in the warped version of a new template to be added. Figure 7.8 illustrates this issue. A standard incremental image blending approach that allows dynamic adoptions of the underlying image blending weights is given by the flexible image blending approach presented by Zhao [138]. Thereby image blending based on a per-pixel weighting function $\mathcal{W}(x, y)$ is defined as

$$\mathbf{I}_p^{1\cdots k} = \sum_{i=1}^{k} \mathcal{W}\, warp\left(\mathbf{I}_i\right), \tag{7.8}$$

where $\mathbf{I}_p^{1\cdots k}$ denotes the partial panorama image up to time k, $\mathcal{W}$ is the actual pixel-wise weighting function for the warped version of the actual template, and $warp()$ denotes

the rigid motion transformation warping that maps the template to the global mosaic coordinate frame. The incremental multi-band blending for M Laplacian pyramid levels at time k is then given by

$$\mathbf{I}_p^{1\cdots k} = \sum_{i=1}^{M} \frac{G_{\mathcal{W}_k} L_{I_{k_i}^W}^{\sigma_i} + G_{\mathcal{W}_{k-1}} L_{T_{k-1_i}}^{\sigma_i}}{G_{\mathcal{W}_k} + G_{\mathcal{W}_{k-1}}}, \tag{7.9}$$

where $G_{\mathcal{W}}$ denotes the Gaussian pyramid of weighting function $\mathcal{W}$, and σ_i are specific wave-length ranges in the Fourier domain. In this way overlapping templates are combined by weighted summation within a Laplacian pyramid [20]. The final blending result is then retrieved by collapsing the combined pyramid. In contrast to the standard multi-band blending approach presented in [138] we reformulate the pixel-wise weighting function by degenerating the incremental weighting update function in [138], and by additionally incorporating exposure fusion weights [90]. If we consider a ray to the final panorama image, the targeted pixel contains information from several overlapping sequential weld seam templates. Therefore, an averaging scheme that considers similar weights for each pixel is applied. In [138] a geometry based weighting function $g(u, v)$ which assumes that the center of the image has sharper focus and less distortions is additionally applied. Our proposed weld seam panorama image generation approach does not consider camera calibration or image undistortion operations. Hence, geometry based weighting seems to be not applicable for our purposes as we perform image blending of arbitrarily extracted templates where consequently the assumption of sharper focus in the center of the template does not hold. However, we can assume that the template is extracted at an approximately perpendicular camera view angle, but we cannot assume that the template is extracted at a position in the image where distortions and noise are not present. It might happen that the template gets extracted from an image region where, e.g., newly welded seam is still fluid. In such cases the incorporation of geometry based weighting would make sense, although the underlying motivation is different than in [138]. The resulting sequentially updated weighting function that includes averaging and geometry assumptions is then given as

$$\mathcal{W}^{flexible} = (1 - w_k) \cdot w_{k-1} + w_k, \tag{7.10}$$

where w_k denotes the combined flexible weighting at time k, proposed in [138]. Figure 7.9 illustrates the incremental image blending approach and the incorporated flexible blending weights.

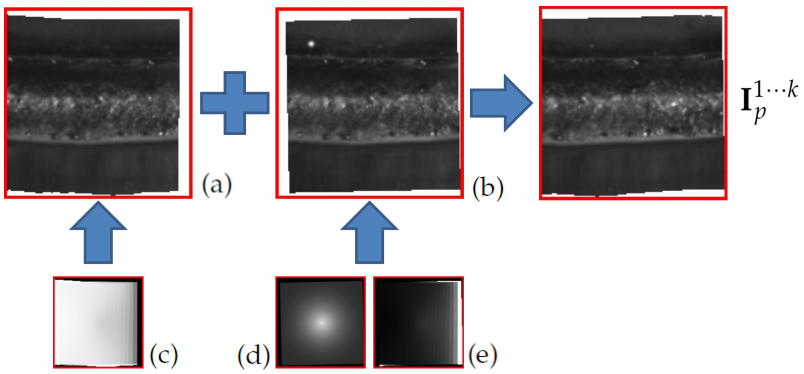


Figure 7.9: **Incremental Image Blending:** An existing partial panorama image (a) is combined with a warped weld seam template (b). Thereby sequential blending weights of the existing panorama (c), geometry based weights (d), and relative weights of the warped template (e) are incorporated in the computation of $\mathbf{I}_p^{1\cdots k}$ at time k. [75]

Exposure fusion presented in [90] describes an image fusion approach that allows for combining images that are acquired at varying exposures, resulting in a single high quality image that depicts only the best parts of a multi exposure image sequence. The underlying method considers three different quality measures that commonly contribute to the final fusion result. First, the contrast C is obtained via Laplacian filtering of gray scale versions of the input images, where the indicator C is given by the absolute value of the Laplacian filter responses, respectively. The second measure is given by the saturation S, which is computed as the standard deviations within the RGB channels at each pixel. The third quality measure is called well-exposedness E, which decreases intensities near zero (underexposed) and one (overexposed). Therefore, a Gaussian curve is used to weight each intensity i according to $e^{-\frac{(i-0.5)^2}{2\sigma^2}}$ with $\sigma = 0.2$. In this way the well-exposedness measure evaluates the distance of an intensity i to 0.5. The pixel-wise linear combination of all three quality measures gives the desired exposure fusion weights for a multi-exposure image sequence.

$$\mathcal{W}^{exposure}(x,y) = C(x,y)^{\alpha} \times S(x,y)^{\beta} \times E(x,y)^{\gamma} , \tag{7.11}$$

where α, β, and γ are weighting exponents that control the effect of the different quality measures. The final combined weighting function that considers both, incremental weighting and exposure fusion is given by multiplying both weighting maps.

$$\mathcal{W}(x,y) = \mathcal{W}^{flexible}(x,y) \times \mathcal{W}^{exposure}(x,y) \tag{7.12}$$

Figure 7.10 illustrates the advantages of considering exposure fusion weights and incremental flexible weights for incremental image blending. Overexposed image regions in the panorama image are reduced, resulting in better focused weld seam structures at the specific image regions.

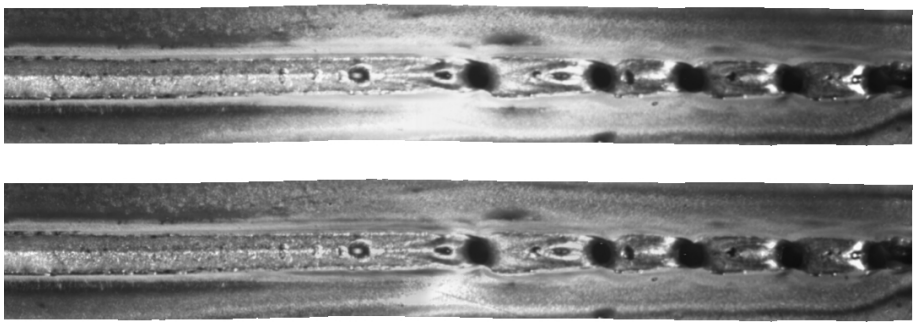


Figure 7.10: **Exposure Fusion:** Incorporation of exposure fusion weights into flexible image blending significantly increases the quality of the resulting panorama image (bottom) in overexposed regions compared to the panorama generated without exposure fusion weights (top). [75]

Table 7.2 presents a comparison of different image blending strategies and how well they perform in the presence of different types of noise, compared to noise-free ground truth data. We compare our proposed incremental blending with a feathering-based approach presented in [116], with binary weighted pyramid blending as presented in [19], with the gradient domain based blending presented in [78], and with median blending in four experiments on different welding tasks. Thereby similarities of individually generated panorama images with noise free ground truth data based on mean structural similarity $\mathcal{MSSIM}$, normalized cross correlation $\mathcal{NCC}$ and normalized sum of squared differences $\mathcal{NSSD}$ are utilized as measures of quality. It turned out that the proposed incremental image blending strategy achieved best results in most cases, emphasizing the robustness and accuracy of our presented approach. Except for the Salt & Pepper noise where we achieved second best results after median blending our panorama generation concept outperforms other existing blending strategies, thus demonstrating that our approach should be favored for welding panorama image generation.

	Gaussian			*Poisson*			*Salt & Pepper*		
	$\mathcal{MSSIM}$	$\mathcal{NCC}$	$\mathcal{NSSD}$	$\mathcal{MSSIM}$	$\mathcal{NCC}$	$\mathcal{NSSD}$	$\mathcal{MSSIM}$	$\mathcal{NCC}$	$\mathcal{NSSD}$
Straight Welding a									
Incremental	0.6546	0.9717	0.0086	0.9156	0.9968	0.0010	0.5821	0.9527	0.0153
Feathering	0.5353	0.9433	0.0180	0.8535	0.9934	0.0020	0.4649	0.9080	0.0299
Binary	0.2340	0.7746	0.0902	0.5996	0.9644	0.0110	0.2194	0.6823	0.1404
Gradient	0.5350	0.9433	0.0180	0.8534	0.9933	0.0020	0.4644	0.9080	0.0300
Median	0.5926	0.9615	0.0123	0.8882	0.9956	0.0013	0.9841	0.9906	0.0029
Straight Welding b									
Incremental	0.7716	0.9861	0.0039	0.9459	0.9976	0.0007	0.7078	0.9782	0.0060
Feathering	0.6483	0.9722	0.0079	0.9002	0.9952	0.0013	0.5781	0.9565	0.0119
Binary	0.3025	0.8786	0.0397	0.6659	0.9759	0.0069	0.2837	0.8249	0.0569
Gradient	0.6481	0.9721	0.0079	0.9002	0.9952	0.0013	0.5776	0.9563	0.0120
Median	0.7095	0.9807	0.0055	0.9260	0.9967	0.0009	0.9885	0.9960	0.0011
Curved Welding a									
Incremental	0.8727	0.9958	0.0024	0.8855	0.9963	0.0021	0.8471	0.9956	0.0026
Feathering	0.7795	0.9956	0.0025	0.8630	0.9959	0.0023	0.7515	0.9953	0.0028
Binary	0.7228	0.9942	0.0033	0.8123	0.9952	0.0028	0.7145	0.9930	0.0041
Gradient	0.7693	0.9948	0.0034	0.8363	0.9950	0.0034	0.7435	0.9942	0.0037
Median	0.8585	0.9956	0.0026	0.8809	0.9959	0.0023	0.8750	0.9955	0.0026
Curved Welding b									
Incremental	0.9874	0.9997	0.0001	0.9872	0.9997	0.0001	0.9672	0.9996	0.0002
Feathering	0.9471	0.9995	0.0002	0.9828	0.9997	0.0001	0.9285	0.9993	0.0004
Binary	0.8907	0.9984	0.0007	0.9662	0.9996	0.0002	0.8679	0.9973	0.0012
Gradient	0.9162	0.9975	0.0024	0.9416	0.9978	0.0023	0.9042	0.9972	0.0023
Median	0.9851	0.9997	0.0001	0.9882	0.9997	0.0001	0.9874	0.9997	0.0001

Table 7.2: **Blending Comparison:** Different image blending strategies are compared based on different similarity measurements of the individually generated mosaics from noisy input data with the mosaic generated from noise free data, serving as ground truth. The results clearly show that the proposed incremental blending strategy should be favored as it outperformed the other approaches except for Salt & Pepper noise, where the second best results have been achieved. [75]

Global Panorama Refinement

Up to now the panorama image generation pipeline is able to incrementally generate an overview image of an entire welding from start to end. Thereby, potential errors in the transformation estimation procedure accumulate over time as the global transformation is composed from individual local rigid motion transformation. In order to overcome this problem, we apply a global refinement consisting of a graph based optimization [114] and an iterative global alignment [33] once all templates are registered. Thereby, the incremental global transformations are first adjusted to an a priori known end position in the mosaic coordinate frame, followed by an iterative global alignment that eliminates visible artifacts generated due to the graph optimization. As welding sequences from a common welding process start and end at common coordinates, we assume template positions and their specific orientations in the mosaic coordinate frame as camera poses lying in a common plane. With a given camera pose at the end of the welding sequence we are able to perform the loop closure correction according to [114]. This allows for correcting potentially accumulated registration errors, resulting in a new set of camera poses and hence transformations where the final registration error is minimal. Figure 7.11 depicts sequential camera poses for an exemplary welding sequence before and after graph based optimization is applied.

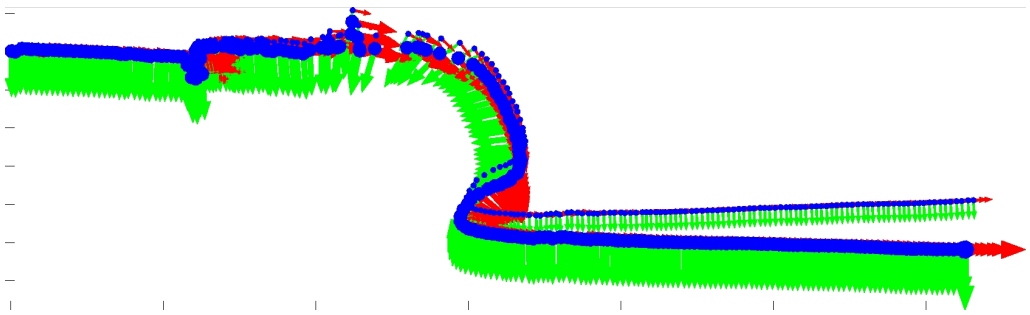


Figure 7.11: **Graph-based Pose Optimization:** Sequential camera poses (small coordinate frames) of a welding sequence in a common mosaic coordinate frame can be corrected (large coordinate frames) to a priori known start and end poses using a graph based optimization that allows for applying loop closure correction. [75]

Once the camera pose sequence is corrected, the iterative global alignment presented in [33] refines feature positions in the global mosaic coordinate frame and the global transformations from the images to the mosaic coordinate frame in an alternating manner. Thereby, two linear computations are iteratively performed. First, feature positions

${}^{m}\widehat{\mathbf{x}}_j$ in the mosaic coordinate frame are re-estimated according to

$$ {}^{m}\widehat{\mathbf{x}}_j = \frac{1}{N_j} \sum_{{}^{k}\mathbf{x}_i \in \eta_j}^{m} \mathbf{x}_i^k \,, \tag{7.13} $$

where N_j is the number of images that depict feature point ${}^{m}\widehat{\mathbf{x}}_j$, η_j is a set of image points that match with an equal feature point in the panorama image, and ${}^{m}\mathbf{x}_i^k$ denotes the projection of the i^{th} feature point from the k^{th} image to mosaic point ${}^{m}\mathbf{x}_j$ according to

$$ {}^{m}\mathbf{x}_i^k = {}^{m}\mathbf{H}_k \, {}^{k}\mathbf{x}_i \,. \tag{7.14} $$

In a second step the list of transformations ${}^{m}\mathbf{H}_k$ is re-estimated based on the new point sets consisting of ${}^{k}\mathbf{x}_i$ and ${}^{m}\widehat{\mathbf{x}}_j$. Thereby, the following cost function is minimized by iterating the two introduced linear steps until a threshold for the decrease rate of the cost function error E is exceeded.

$$ E = \sum_{j=1}^{M} \sum_{{}^{k}\mathbf{x}_i \in \eta_j} \| {}^{m}\mathbf{x}_j - {}^{m}\mathbf{H}_k \, {}^{k}\mathbf{x}_i \|_2 \tag{7.15} $$

Due to the linearity of the applied alignment procedure low computational costs and hence real-time capability are achieved. Figure 7.12 illustrates the effect of the two global panorama refinement steps. An incrementally generated panorama that exhibits a cumulative registration error is first corrected to graph optimized poses, resulting in partially blurry regions in the panorama. These are subsequently eliminated using the presented global alternating alignment.

7.2.3 Evaluations

For a meaningful evaluation of the panorama image generation pipeline overall 25 different datasets, consisting of overall 674 welding sequences, are processed. The evaluated datasets also include welding sequences where typical welding errors that are accompanied by large amounts of noise appear. The dimension of the extracted templates is chosen according to the weld seam width in pixels and according to the overlap of consecutive images in welding direction. The dimension of the individual weld seam templates is set to a height of about the double weld seam thickness (typically between 50 and 150 pixels) and a width that allows an overlap of approximately 80% between consecutive weld seam templates. Quantitative results on the panorama image genera-

Figure 7.12: **Global Panorama Refinement Results:** The incrementally generated weld seam panorama (top) exhibits cumulative registration errors, which are corrected using graph based pose optimization (middle). The resulting image artifacts are finally removed by applying a global iterative alignment that minimizes feature point reprojection errors (bottom). [75]

tion quality are presented in Table 7.3. The assignment as *OK* or *NOT OK* is done by visual inspection of the generated panoramas, where visible artifacts or geometric trajectory errors are the main classification criteria. Figure 7.13 exemplary shows panorama images that have been classified as *NOT OK* due to visible artifacts and geometric errors. Thereby, visible artifacts are mainly caused by several consecutive templates with severe amounts of smoke that occlude the welded seam. Consequently, this issue results in homogeneous grayish regions where the underlying registration results in large errors or even fails. The undesired effects in the final panorama are then given by visible seams, smoke edges, or geometric errors. Nevertheless, an overall performance of 99% on the three different welding test series has been achieved. The quite low performance on set 07 from test series α results from very fast robot rotations during welding, as illustrated

Series	Set-ID	# Seq	Img / Seq	OK	Percentage
α	01	20	252	20	100%
α	02	20	254	20	100%
α	03	20	254	20	100%
α	04	20	259	20	100%
α	05	20	259	19	95%
α	06	20	281	20	100%
α	07	20	289	10	50%
α	12	20	286	20	100%
α	13	20	275	20	100%
α	14	20	274	20	100%
α	15	20	282	20	100%
α	16	20	275	20	100%
α	17	20	273	20	100%
α	18	20	297	20	100%
α	19	20	289	20	100%
α	20	20	267	20	100%
α	21	20	277	20	100%
α	22	20	134	20	100%
α	23	20	135	20	100%
α	24	20	137	20	100%
α	25	20	292	20	100%
Average					**97%**
β	01	73	225	73	100%
β	02	99	236	99	100%
Average					**100%**
γ	01	21	573	21	100%
γ	02	21	325	21	100%
Average					**100%**

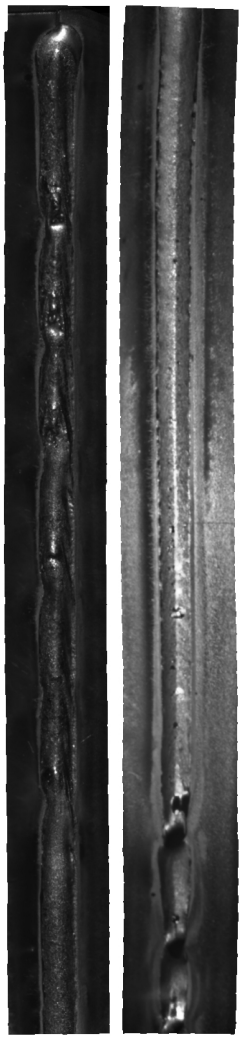


Table 7.3: **Panorama Image Generation Results:** Quantitative evaluation of 674 runs on welding image sequences from 25 different welding datasets, which include various types of visible noise, different welding defects and severe illumination changes. Evaluations are made by visual inspection of the final panorama, where a run is marked as *OK* if it results in a single panorama image without visible artifacts or geometric trajectory errors that further allows a visual rating of the weld seam at a glance. On the right two sample welding panorama images are shown.

in Figure 7.14. Tracking errors and unforeseen spline deformations result in large transformation estimation errors. However, this issue can be resolved by considering cameras with higher frame rates for the image acquisition system.

In the following several illustrative experimental results for the presented panorama

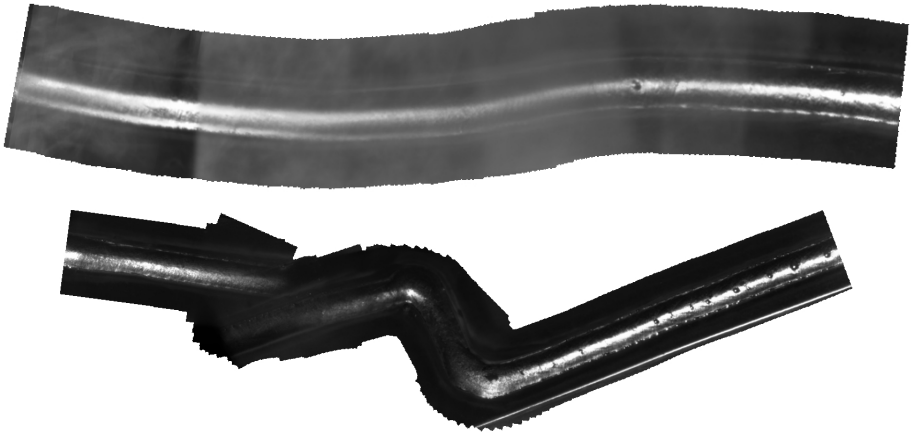


Figure 7.13: **Panorama Generation Errors:** Due to large amounts of image noise like, e.g., severe smoke in several consecutive templates or due to large registration errors the panorama images result in unsatisfactory visible artifacts and geometric errors that are consequently classified as *NOT OK* in the quantitative evaluations.

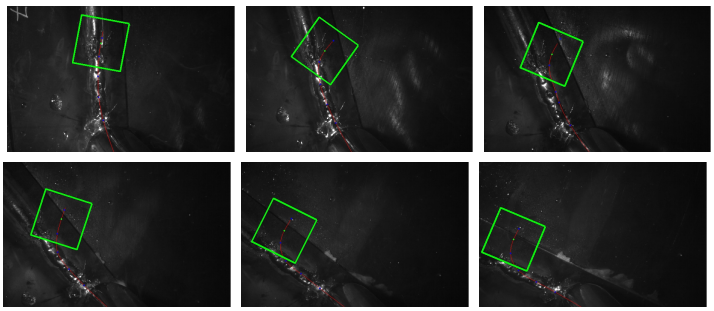


Figure 7.14: **Fast Robot Rotation:** Large angle rotations during welding cause tracking errors and unforeseen spline deformations which in turn result in registration errors as the overlap of consecutive templates is severely reduced. Higher frame rates during acquisition could easily resolve this issue.

image generation approach that demonstrate the robustness against typical industrial noise like sparks, spilling, plasma explosions, dense smoke wads or evaporating water are presented. Thereby, different welding trajectories, welding processes as well as environmental conditions are covered. For further evaluations considering different welding panorama image generation approaches, different feature concepts, and different image blending strategies we would refer to the experimental work presented by Lanner [75].

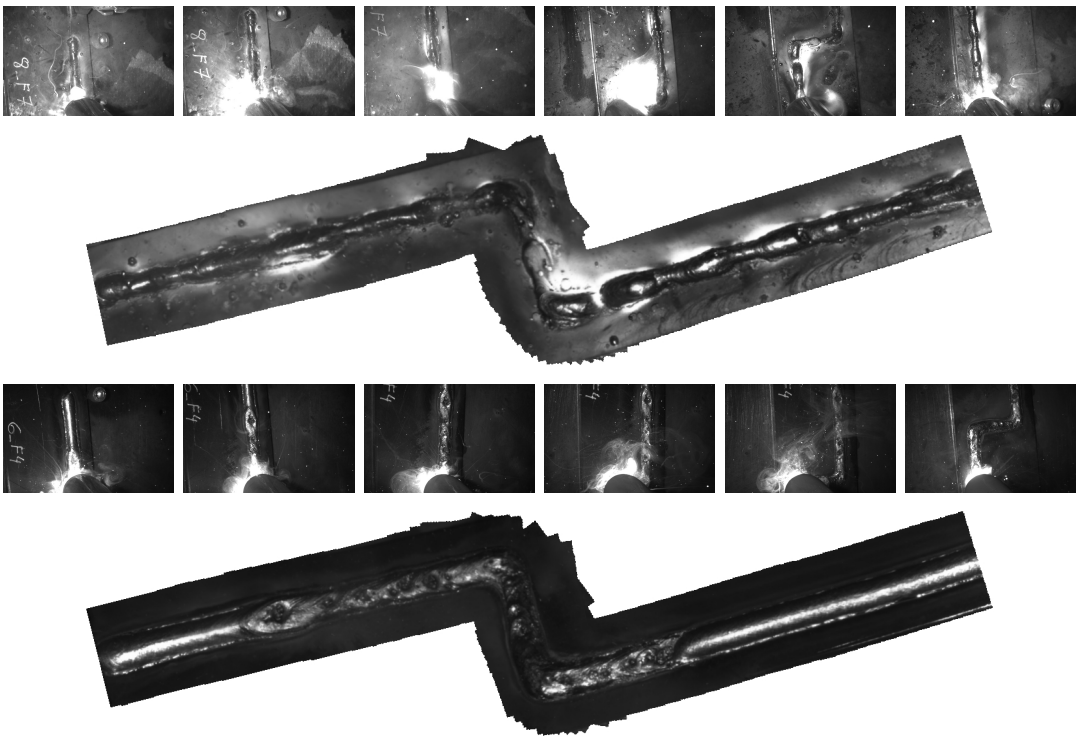


Figure 7.15: **Welding Panorama Images in Harsh Robotic Welding:** Examples for panorama images generated from narrow curve welding image sequences. Despite saturated regions and severe appearance changes in the curve regions tracking succeeds, resulting in high quality panorama images.

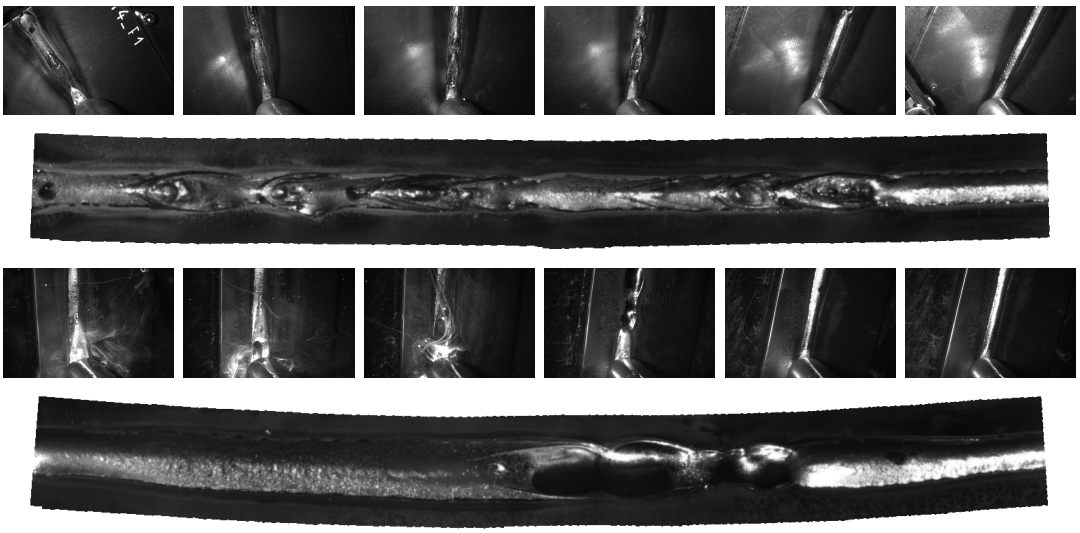


Figure 7.16: **Welding Panorama Images in Harsh Robotic Welding:** Examples for panorama images generated from straight welding image sequences with varying camera pose.

Figure 7.17: **Welding Panorama Images in Harsh Robotic Welding:** Examples for panorama images generated from curved welding image sequences. Again saturated image regions and noise do not affect the final quality of the incrementally generated panorama.

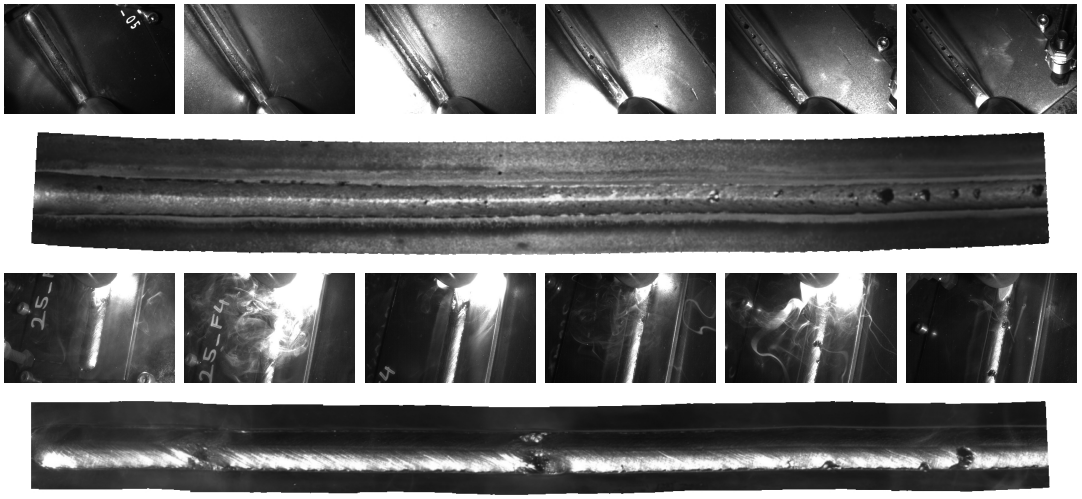


Figure 7.18: **Welding Panorama Images in Harsh Robotic Welding:** Examples for panorama images generated from straight welding image sequences with severe specular reflections and image noise.

7.3 Image-based Defect Detection in Robotic Arc Welding

The second template based quality assessment approach that we present is an automated vision-based welding quality analysis method that is able to automatically detect weld seam templates that deviate from typical behavior. Therefore, we rely on very few reference data that is error-free in order to compute a representative model of a corresponding welding process in an off-line training step. We hereby follow the strategy of acquiring as few as possible error-free reference samples and then build a reference database. During the welding task, weld seam templates of the newly welded seam are acquired and subsequently classified, according to their deviations from the reference. To cope with typical appearance changes during a welding task, we additionally incorporate temporal proximity that is typically given, e.g., by timestamps in conjunction with defined robot motion. In the error-case this also allows for exactly localizing detected welding defects on the specimen. The separation of the inspected weld seams into either error-free or defective ones is applied during the welding task in a real-time fashion. Although, there exist approaches that automatically assess the quality of robotic welding tasks, there are no approaches that work on-line and that consider an inspection of the newly welded seam during the welding process for quality inspection. Thus, our main contributions considering weld seam inspection are the introduction of a vision based method that allows for assessing the quality of a robotic welding task in real-time without the necessity of negative training data, that is furthermore automatically trained from very few non-defective training templates. Moreover, we present two welding quality assessment applications that are based on the proposed quality inspection method: **a)** an automatic classification of entire welding sequences, and **b)** an automatic localization of punctual welding defects.

7.3.1 Off-line Reference Database Generation

From a methodological point of view, we propose a real-time visual inspection of newly welded seam which consists of three steps: First, error-free reference data is generated in an off-line preparatory task. Second, we generate a small database of reference weld seam templates by reducing the present redundancy, and third, classification of new incoming templates during on-line welding is performed by evaluating similarities between new templates and the reference data entries. Thereby, a welding process specific detection threshold, that separates error-free templates from defective and unusual ones, is derived automatically from the available reference data. As mentioned

above we initially generate a reference database by applying an unsupervised clustering approach introduced by Frey and Dueck [44] to the error-free training data. This results in multiple clusters that are highly dissimilar, each describing several similar weld seam templates. For the reference data generation at least 2 complete error-free welding sequences denoted by Ref are acquired for each welding process. As illustrated in Figure 7.19 high weld seam template overlaps result in a large amount of redundancy considering the corresponding image data. Thereby, the overlap mainly depends on the underlying process specific welding speed as well as on the specific welding trajectories. Typical values for the overlap are $10 - 30$ frames for a single pixel. The clustering of these redundant welding images consequently results in a significant reduction of the redundancy, which further on allows for performing quality inspection with less computational effort.

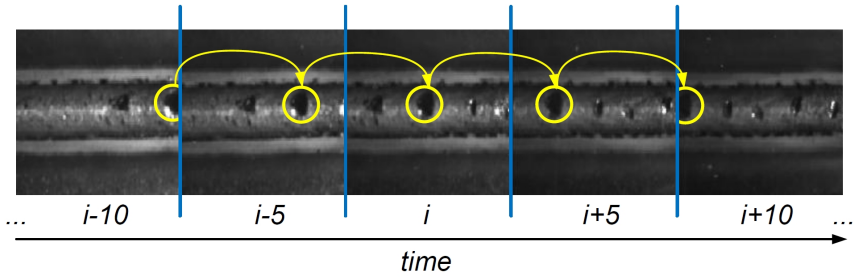


Figure 7.19: **Welding Image Data Redundancy:** A sample weld seam template sequence acquired at $20fps$. The marked structures on the individual templates illustrate the large amount of image data redundancy.

Template clustering reduces the available reference templates denoted by $\mathbf{f}_i, i \in [1, N]$ from all reference sequences Ref to specific prototypes $\mathbf{c}_j, j \in [1, M]$, where $M \ll N$. Although, the number of templates gets significantly reduced by the clustering the cluster centers still represent the entire welding process including possible systematic appearance changes. The cluster centers $\mathbf{c}_j$ are given as tuples, consisting of a representative template $\mathbf{c}$, and of a vector that contains timestamps $t_1 \cdots t_n$ from all templates that vote for $\mathbf{c}_j$.

$$\mathbf{c}_j = \{\mathbf{c}, [t_1 \cdots t_n]\} \ , \ \mathbf{c} \in \mathbf{f}_i \tag{7.16}$$

We use Affinity Propagation (AP) [44] for clustering, as the approach has shown state of the art performance for a variety of unsupervised clustering problems [80, 137] without complex parametrization. AP identifies prototypes out of a set of arbitrary data points by passing messages in between, with a pairwise similarity measure between

all data points given as input. Instead of exhaustively comparing new incoming templates against all reference data entries, we only keep M prototypes and corresponding timestamps for further on-line quality inspection. Thereby, typical values for N and M are ≈ 250 and $10 - 20$, respectively. The number of obtained cluster centers mainly relies on the intra process variability of a corresponding welding process. The usage of temporal proximity given by, e.g., timestamps or frame numbers further simplifies the classification problem, as incoming templates can be related to a subset of the cluster centers. Furthermore, potential welding defects can be exactly located on the welded seam, up to a synchronization gap Δ of a few ns ($\approxeq 0.5$px), caused by the underlying image acquisition system. Each obtained prototype or cluster center $\mathbf{c}_j$ is an exemplar of the initial reference data Ref and is accompanied by appropriate timestamps $[t_1 \cdots t_n]$ from all templates that vote for the prototype $\mathbf{c}_j$. In order to be robust against systematic appearance changes, and to be able to exactly locate potential welding defects, each new incoming template denoted by $\mathbf{p}$ with a corresponding time stamp t_p is compared to the cluster center $\mathbf{c}_j$ with minimal temporal gap. Thus, we first identify the cluster center with the minimal temporal gap denoted by the upper case letter $\mathbf{C}$, and then use the corresponding template $\mathbf{c}_j$ for a similarity evaluation denoted by $\mathcal{S}$ with the actual template $\mathbf{p}$.

$$\mathbf{C} = \mathbf{c}_j \quad \text{if} \quad t_p \pm \Delta \in [t_1 \cdots t_n] \tag{7.17}$$

$$s_p = \mathcal{S}(\mathbf{p}, \mathbf{C}) \ , \ \mathcal{S} : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R} \tag{7.18}$$

7.3.2 On-line Welding Defect Detection

In continuous industrial use, each new incoming template is compared to the available reference data entries by evaluating a similarity measure in between. Templates that exhibit high dissimilarities are then rated as unusual events or welding defects. Hence, the desired classification of incoming templates $\mathbf{p}$ into either error-free or defective is controlled by computed similarity values, respectively. If the similarity falls below a process specific threshold, the corresponding template $\mathbf{p}$ is classified as defective. Basically, any similarity metric $\mathcal{S} : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$, which allows a pairwise comparison of templates or images, can be applied. In order to evaluate the proposed quality inspection approach in a more wide-ranging manner, we chose two diverse similarity matching strategies for this task as follows.

Image Data Correlation

For the first similarity measure we chose the $\mathcal{NCC}$ measure, which is commonly used in computer vision for template matching. It is simple, fast, and more robust to lighting changes, compared to other approaches. Thereby, we perform robust matching by applying a rotating and sliding window approach as illustrated in Figure 7.20. Reference templates are rotated by a set of discrete angles α^i to overcome slight misalignments during image acquisition. The best correlation result amongst all rotated patch comparisons is finally chosen, resulting in an overall more reliable and robust matching. Scale variations could also be considered by a scale-space approach, but as the image acquisition unit and the welding robot are rigidly connected, resulting in a constant working distance for the entire welding process, this is not necessary.

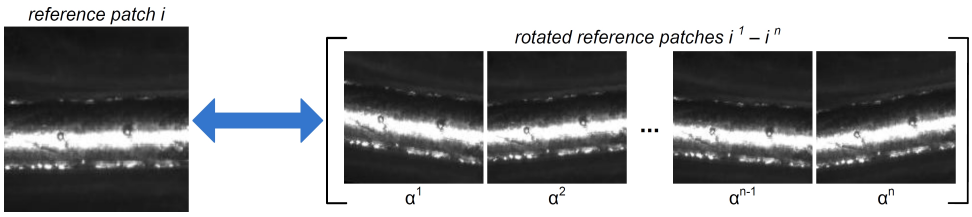


Figure 7.20: **Misalignment Aware Matching:** Similarity evaluations against a set of rotated reference templates allow for being robust to slight misalignments during image acquisition and give more reliable and robust matching results.

During welding each new incoming template $\mathbf{p}$ is assigned a similarity value s_p to the clustered reference data. All templates with a similarity value s_p smaller than a process-specific detection threshold θ_{corr} are reported as unusual events, and consequently as potential welding defects. We determine θ_{corr} from the available reference sequences Ref in a way, such that a welding sequence can be classified either into error-free or erroneous without any further information or learning effort. Although, a single threshold that separates error-free and defective templates could be used for the entire welding process, we utilize the available spatial information given by timestamps in order to obtain time specific thresholds for a given welding process. Thereby, the intra class similarity values s_{Ref} between reference templates are computed under consideration of the given spatial proximity. This results in reference similarity values for all available timestamps t_{Ref} in Ref, that also reflect possible systematic appearance changes. Online, we first determine the reference threshold from s_{Ref} with minimal spatial gap to the actual given timestamp, and then classify the specific template under an additional

consideration of a narrow band safety margin δ_{corr} to be robust against possible outliers.

$$s_{Ref} = \mathcal{S}\left(Ref_i, Ref_j\right),\ i \neq j \text{ and } i,j \in [1 \cdots |\text{Ref}|] \tag{7.19}$$

$$\mathbf{p} \text{ is } \begin{cases} \text{a welding defect} & \text{if } s_p < s_{Ref}\left(argmin\left(|t_p - t_{Ref}|\right)\right) - \delta_{corr} \\ \text{error-free} & \text{otherwise} \end{cases} \tag{7.20}$$

Gray Scale Cross Profile

Our second concept considering welding quality inspection is based on weld seam cross profile evaluations. Similar to the correlation based approach a reference cluster center is identified from the reference database by evaluating the minimal temporal gap between the reference timestamps and the actual one. In contrast to correlating templates, the proposed cross profile evaluation considers a one dimensional cross section signal that reflects the perpendicular weld seam cross profile as illustrated in Figure 7.21.

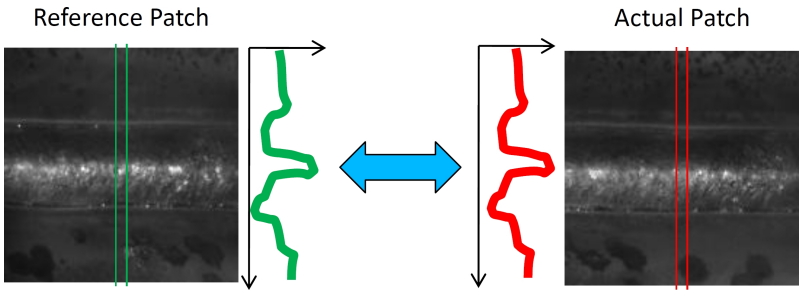


Figure 7.21: **Weld Seam Cross Profile:** Gray scale profiles extracted from the center of weld seam templates are used for welding quality relevant similarity measurements. The dimension reduction to 1D signals results in improved runtime performances.

In order to consider visible image noise like smoke wads or sparks, the cross profile denoted by $\mathbf{c}$ is robustly computed from columns in the center of a $m \times n$ template $\mathbf{p}$ according to

$$\mathbf{c} = median\left(\mathbf{p}\left(x_i, y_j\right)\right),\ i \in \left[\frac{m}{2} - k, \cdots, \frac{m}{2} + k\right], j \in [1, \cdots n]\ . \tag{7.21}$$

The necessary similarity measure for the desired quality inspection is finally obtained by comparing the cross profile $\mathbf{c}_ref$ obtained from the reference template with the minimal temporal gap with the cross profile $\mathbf{c}_p$ obtained from the actual template $\mathbf{p}$. Thereby, the Euclidean distance between both signals is computed in order to determine

the corresponding deviation.

$$d = \|\mathbf{c}_{ref} - \mathbf{c}_p\|_2 \tag{7.22}$$

The final classification is given by simple thresholding of computed deviations. Again, a welding process specific threshold θ_{cross} is derived from available reference data entries by evaluating time specific cross profile deviations d_{Ref}. This results in reference cross profile deviation values for each available timestamp t_{Ref} in Ref.

$$d_{Ref} = \mathcal{S}\left(Ref_i, Ref_j\right),\ i \neq j \text{ and } i, j \in [1 \cdots |\text{Ref}|] \tag{7.23}$$

Similar to the correlation based approach, for on-line processing the deviation at a minimal temporal gap is first identified from the reference, and subsequently compared to the deviation computed for the actual template under an additional consideration of a narrow band safety margin δ_{cross}. The inspected template gets classified as defective if the computed cross profile deviation exceeds the corresponding reference value.

$$\mathbf{p} \text{ is } \begin{cases} \text{a welding defect} & \text{if } d_p < d_{Ref}\left(argmin\left(|t_p - t_{Ref}|\right)\right) - \delta_{cross} \\ \text{error-free} & \text{otherwise} \end{cases} \tag{7.24}$$

7.3.3 Autonomous Quality Inspection Applications

Based on the presented autonomous image based quality inspection methods, we propose two different applications for the robotic arc welding quality assessment: Application A separates complete welding sequences into either error-free or defective ones. We show that a single unusual event suffices to classify the corresponding sequence as defective. Application B exactly locates punctual defects on the welded seam. This application is especially of interest for welding tasks, where small punctual defects might be tolerated considering the overall welding quality. We evaluate the proposed applications on the same welding datasets as used for the panorama image generation approach. As ground truth labels which have been generated by a welding expert are given only for the 21 datasets from test series α, evaluations on test series β and γ were not feasible. Each welding process dataset generally consists of 11 error-free welding sequences and 9 welding sequences that include typical welding defects like holes, narrow weld seam regions, blisters, deformations, or gaps. The evaluated sequences consist of 134 to 292 weld seam templates of size 161×161 pixels each, where 4 out of 10 error-free welding sequences are chosen for the off-line reference database generation. The remaining welding sequences are then used for on-line classification runs. In this way,

336 welding sequences consisting of a total of 85456 weld seam templates are processed and classified in our evaluation runs. The welding sequences are characterized by severe illumination changes, heavy smoke, spatter and spilling, curved weldings, different welding speeds, as well as different materials that are processed. In this way, we demonstrate that our inspection methods can cope with the high variability and with the large amount of noise which are typical for industrial welding processes. Considering the run-time performance of the proposed methods, our Matlab implementation reached an average processing rate of $51.20 fps$ for both similarity measures at a glance on an Intel Core i7 2.8 GHz processor. Hence, real-time processing at typical image acquisition rates of $10 - 20$fps is definitely feasible.

Application A: Sequence Classification

The aim of the proposed sequence classification is to classify entire welding sequences into either defective or error-free, regardless of the type or the amount of errors that occurred. Once a single similarity value exceeds the corresponding reference thresholds θ_{corr} or θ_{cross}, the entire sequence is classified as defective. In Figure 7.22 the amount of unusual events relative to the number of processed templates is shown over varying similarity threshold safety margins denoted by *delta* for the correlation based similarity measurement approach for a sample welding process. Thereby, the ideal separation between error-free and defective welding is marked, where ideal denotes that no detection responses are obtained for error-free weldings. The marked distance d illustrates, that a separation of error-free from defective weldings is definitely feasible.

Considering the quantitative evaluations for the sequence classification approach two different experiments are conducted for both similarity measurement concepts. First, the average classification error including false positives (FP) and false negatives (FN) is evaluated for each dataset from test series α. The second experiment incorporates the assumption that a welding defect must be visible in at least two consecutive weld seam templates due to the overlap in between. This assumption we refer to as two frame criterion. Again, the average classification error, false positives (FP), and false negatives (FN) are evaluated for the 21 welding datasets from test series α. Table 7.4 presents the numerical results for the conducted welding sequence classification experiments for the correlation based similarity concept, and Table 7.5 shows the results for the cross profile approach.

As the overall aim is to perform quality inspection of welding sequences at an overall minimal classification error and a coincidentally minimal amount of false negatives,

optimized parameter settings considering the specific safety margins δ_{corr} and δ_{cross} additionally presented in both tables are applied for the individual evaluations, respectively.

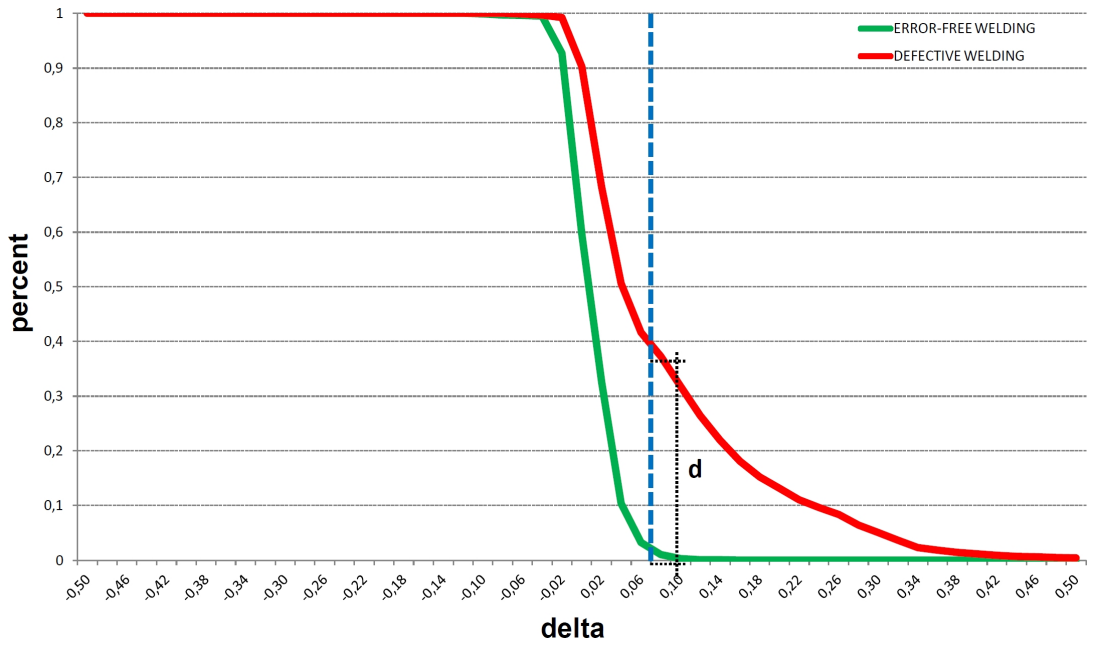


Figure 7.22: **Detection Response Curves:** The obtained detection response curves over varying reference threshold safety margins *delta* for error-free and defective welding sequences from a sample welding process clearly show that a separation in between is feasible. The optimal separation denoted by d does not include any responses for the error-free cases.

Application B: Defect Localization

Considering, e.g., weldings on the base plate of cars built in the automotive industry, long sequences ($\geq$ 1000 images) might be processed. Thereby the discarding of objects with small punctual welding defects could be unnecessary or produce undesired additional costs. Knowledge on the exact location of welding defects on the other hand provides the option for, e.g., an automatic repair operation or a targeted manual inspection afterwards. Due to the availability of ground truth labels provided by welding experts classification errors in terms of false positives (FP, wrong detection response) and false negatives (FN, wrong error-free response) are evaluated on welding images from 420 welding sequences out of 21 welding datasets, respectively.

Again, the overall aim is to perform quality inspection that results in a minimal overall error and a minimal amount of false negatives. In contrast to the sequence

	1 Frame Criterion				*2 Frame Criterion*			
Set	δ_{corr}	**E[%]**	**FN[%]**	**FP[%]**	δ_{corr}	**E[%]**	**FN[%]**	**FP[%]**
01	0.10	10	0	10	0.08	10	0	10
02	0.10	10	0	10	0.08	10	0	10
03	0.20	5	5	0	0.18	0	0	0
04	0.14	5	0	5	0.12	0	0	0
05	0.08	10	5	5	0.06	10	0	10
06	0.18	15	0	15	0.16	10	0	10
07	0.30	35	0	35	0.40	25	25	0
12	0.38	25	0	25	0.42	20	5	15
13	0.30	10	5	5	0.26	10	5	5
14	0.18	10	5	5	0.16	10	5	5
15	0.38	25	10	15	0.36	20	15	5
16	0.30	5	5	0	0.28	5	5	0
17	0.18	5	0	5	0.16	5	0	5
18	0.38	25	10	15	0.30	25	5	20
19	0.24	15	5	10	0.22	15	5	10
20	0.18	20	15	5	0.18	20	0	20
21	0.18	30	20	10	0.10	40	0	40
22	-0.02	10	0	10	-0.02	10	0	10
23	0.14	25	5	20	0.14	20	10	10
24	0.28	10	0	10	0.24	10	0	10
25	0.14	20	20	0	0.06	25	15	10
Average		**15.48**	**5.24**	**10.24**		**14.29**	**4.52**	**9.76**

Table 7.4: **Numerical Correlation-based Welding Sequence Classification Results:** Average classification errors **E**, and corresponding false negatives **FN** and false positives **FP** are presented. If all classification responses are considered (*1 Frame Criterion*), an average classification accuracy of **84.52%** is reached. If an additional visibility constraint which assumes welding defects being visible in at least two consecutive images (*2 Frame Criterion*) is considered, an improved classification accuracy of **85.71%** is achieved.

classification approach we present the results of five experiments for the image based defect detection again for both similarity measurement concepts. The experiments are an average classification error evaluation, an evaluation of the average error at a false negative rate of 5%, an evaluation of the average error at a false negative rate of 3%, an evaluation of the relative error at a false negative rate of 5%, and an evaluation of the relative error at a false negative rate of 3%. Figure 7.23 illustrates the working points and their relations to false negatives and false positives for a sample welding process. *Average* denotes that the obtained detection responses are related to the overall number of inspected templates, whereas *relative* states that the detection responses are related to

	1 Frame Criterion				*2 Frame Criterion*			
Set	δ_{cross}	**E[%]**	**FN[%]**	**FP[%]**	δ_{cross}	**E[%]**	**FN[%]**	**FP[%]**
01	0.21	15	0	15	0.22	10	0	10
02	0.30	15	10	5	0.29	15	15	0
03	0.24	0	0	0	0.24	0	0	0
04	0.24	5	5	0	0.17	5	0	5
05	0.20	5	0	5	0.14	10	5	5
06	0.50	35	0	35	0.45	20	0	20
07	0.39	60	0	60	0.39	55	0	55
12	0.47	35	0	35	0.46	20	5	15
13	0.47	35	0	35	0.47	25	5	20
14	0.42	5	0	5	0.38	5	5	0
15	0.40	35	5	30	0.49	20	10	10
16	0.49	0	0	0	0.35	0	0	0
17	0.44	5	0	5	0.32	5	0	5
18	0.49	25	0	25	0.41	30	0	30
19	0.45	10	10	0	0.37	10	10	0
20	0.37	5	0	5	0.34	10	10	0
21	0.35	40	25	15	0.49	40	40	0
22	0.26	10	0	10	0.26	10	0	10
23	0.41	20	0	20	0.41	25	5	20
24	0.44	55	0	55	0.49	15	0	15
25	0.29	20	15	5	0.23	15	15	0
Average		**20.71**	**3.33**	**17.29**		**16.42**	**5.90**	**10.48**

Table 7.5: **Numerical Cross Profile-based Welding Sequence Classification Results:** Average classification errors **E**, and corresponding false negatives **FN** and false positives **FP** are presented. If all classification responses are considered (*1 Frame Criterion*), an average classification accuracy of **79.29%** is reached. If an additional visibility constraint which assumes welding defects being visible in at least two consecutive images (*2 Frame Criterion*) is considered, an improved classification accuracy of **83.58%** is achieved.

the ground truth responses, respectively. For each experiment the overall classification error **E**, the false negatives **FN** and the false positives **FP** are evaluated as accuracy indicators for the corresponding quality inspection approach at optimized parameter settings for δ_{corr} and δ_{cross}. Figure 7.24 shows ROC curves of a sample welding process at the minimum error working points for both, the correlation and the cross profile approach.

Summarized numerical results for the 21 evaluated welding process datasets from test series α are presented in Table 7.6 for the correlation based inspection approach and in Table 7.7 for the cross profile similarity measurement concept.

	Average							Relative			
	MinErr[%]			FN5[%]		FN3[%]		FN5[%]		FN3[%]	
Set	**E**	**FN**	**FP**	**E**	**FP**	**E**	**FP**	**E**	**FP**	**E**	**FP**
01	5	3	2	6	1	5	2	6	1	21	18
02	9	7	2	10	5	11	8	11	6	16	13
03	5	2	3	5	0	5	2	5	0	8	5
04	8	5	3	8	3	9	6	10	5	10	7
05	5	3	2	6	1	5	2	32	27	17	14
06	6	2	4	7	2	6	3	19	14	19	16
07	8	5	3	9	4	10	7	16	11	24	21
12	17	11	6	20	15	20	17	39	34	39	36
13	10	6	4	11	6	13	10	20	15	23	20
14	7	3	4	7	2	7	4	20	15	40	37
15	13	7	6	13	8	21	18	28	23	39	36
16	12	6	6	13	8	18	15	24	19	24	21
17	8	2	6	8	3	8	5	8	3	14	11
18	16	16	0	30	25	40	37	40	35	46	43
19	16	13	3	20	15	29	26	29	24	35	32
20	10	9	1	13	8	21	18	28	23	39	36
21	17	13	4	45	40	58	55	66	61	66	63
22	13	9	4	30	25	52	49	76	71	79	76
23	12	8	4	16	11	20	17	54	49	74	71
24	13	11	2	16	11	23	20	23	18	26	23
25	3	2	1	5	0	3	0	25	20	25	22
Avg	**10.14**	**6.81**	**3.33**	**14.19**	**9.19**	**18.29**	**15.29**	**27.57**	**22.57**	**32.57**	**29.57**

Table 7.6: **Numerical Correlation-based Welding Image Classification Results:** Average classification errors **E**, and corresponding false negative **FN** and false positives **FP** rates related to the number of processed images as well as relative results related to ground truth values are presented for working points with a minimal overall error **MinErr**, with a false negative rate of 5% **FN5** and with a false negative rate of 3% **FN3**. The overall best performance with an average classification accuracy of **89.86%** is achieved for the minimal error working point **MinErr**.

Summarizing the achieved experimental results, we have empirically shown that a separation of welding sequences into defective and error-free is feasible. Thereby we achieved an average classification accuracy of **84.52%** by applying the correlation based similarity measure concept. The additional consideration of a visibility constraint even slightly increases the accuracy to **85.71%**. The cross profile based approach also achieved quite good classification accuracies, namely **79.29%** and **83.58%**, respectively. Considering the classification of individual images, different working points have been evaluated. It turned out that both similarity measurement concepts perform best in the minimal

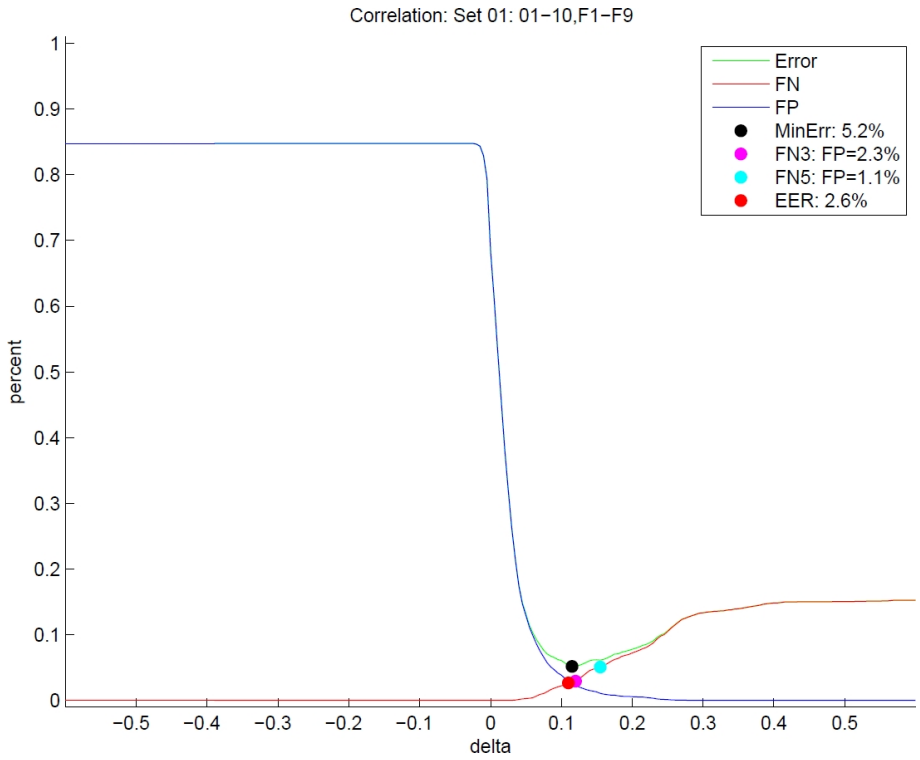


Figure 7.23: **Welding Defect Localization Working Points:** The false negative rate (*FN*), false positive rate (*FP*) and the total error (*Error*) over varying safety margin parameter values denoted by *delta*. *MinErr* denotes the minimum error working point, *FN3* is the 3% false negative working point, *FN5* is the 5% false negative working point, and *EER* denotes the equal error rate.

error working point, which is defined by a minimal amount of wrong classifications coincidentally with a minimal amount of false negatives. The corresponding achieved classification accuracies are **89.86%** and **86.00%** for the correlation and the cross profile based approaches.

In order to compare our quality inspection approach with other machine learning techniques, we applied Principal Component Analysis (PCA) presented in [7], a standard Support Vector Machine (SVM) approach presented in [36], a standard K-Nearest Neighbor approach based on the Euclidean L^2 norm (KNN L2) presented in [41], and a K-Nearest Neighbor approach based on the Mahalanobis distance (KNN M) presented in [130] to our welding template classification problem. Thereby, our presented correlation based approach achieved its best average test performances of 84% on the 21 welding process datasets from test series α with a training data set consisting of 4 error-free welding sequences per dataset. The best PCA test performance on test series α is

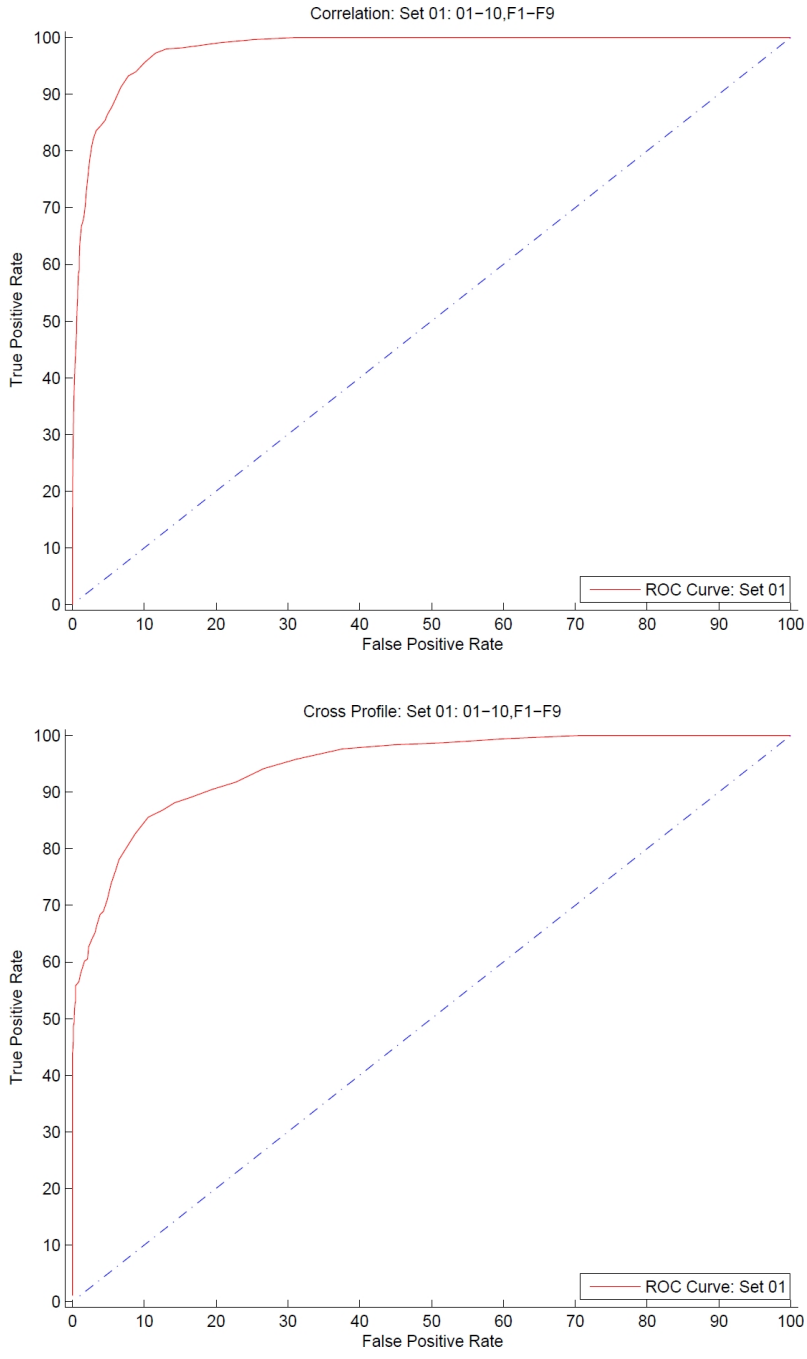


Figure 7.24: **Correlation and Cross Profile ROC Curves:** True positive rates (*Sensitivity*) over false positive rates (1 - *Specificity*) of a sample welding process for the correlation (a) and the cross profile (b) approaches at the minimum error (*MinErr*) working points, respectively.

	Average							Relative			
	MinErr [%]			FN5 [%]		FN3 [%]		FN5 [%]		FN3 [%]	
Set	**E**	**FN**	**FP**	**E**	**FP**	**E**	**FP**	**E**	**FP**	**E**	**FP**
01	8	6	2	9	4	12	9	24	19	30	27
02	13	11	2	19	14	24	21	32	27	38	35
03	6	4	2	6	1	8	5	14	9	24	21
04	12	10	2	17	12	23	20	39	34	39	36
05	6	3	3	6	1	6	3	22	17	29	26
06	12	5	7	13	8	17	14	32	27	42	39
07	13	8	5	18	13	26	23	43	28	51	48
12	17	14	3	28	23	34	31	49	44	52	49
13	12	10	4	17	12	24	21	38	33	44	41
14	7	6	1	8	3	14	11	41	36	57	54
15	14	11	3	24	19	36	33	45	40	53	50
16	12	7	5	15	10	22	19	29	24	36	33
17	10	7	3	14	9	27	24	36	31	49	46
18	17	15	2	41	36	47	44	50	45	57	54
19	17	15	2	37	32	45	42	50	45	55	52
20	12	11	1	27	22	42	39	57	52	63	60
21	18	16	4	54	49	67	64	71	66	76	73
22	14	5	9	14	9	19	16	66	61	59	56
23	15	10	5	20	15	26	23	40	35	40	37
24	22	15	7	40	35	46	43	46	41	54	51
25	4	3	1	5	0	4	1	63	58	75	72
Avg	**14.00**	**9.14**	**3.48**	**20.57**	**15.57**	**27.10**	**24.10**	**42.23**	**37.23**	**48.71**	**45.71**

Table 7.7: **Numerical Cross Profile-based Welding Image Classification Results:** Average classification errors **E**, and corresponding false negative **FN** and false positives **FP** rates related to the number of processed images as well as relative results related to ground truth values are presented for working points with a minimal overall error **MinErr**, with a false negative rate of 5% **FN5** and with a false negative rate of 3% **FN3**. The overall best performance with an average classification accuracy of **86.00%** is achieved for the minimal error working point **MinErr**.

achieved by using a training set of 10 error-free welding sequences per dataset, resulting in an average test performance of 71%. Overall, the SVM and KNN approaches achieved the best classification results, but at the cost of significantly more labeled training data required. The SVM approach achieves an average test performance of 86% on test series α, while being trained from 15 two class labeled sequences per dataset. The KNN approaches are both evaluated for $K = 1$ and $K = 3$ nearest neighbors, and achieve average test performances of 86% to 88% for a training dataset consisting of 15 two class

labeled welding sequences, respectively. Figures 7.25 and 7.26 illustrate the individual performances of the above mentioned approaches on the 21 datasets from test series α, as well as the individual average performances.

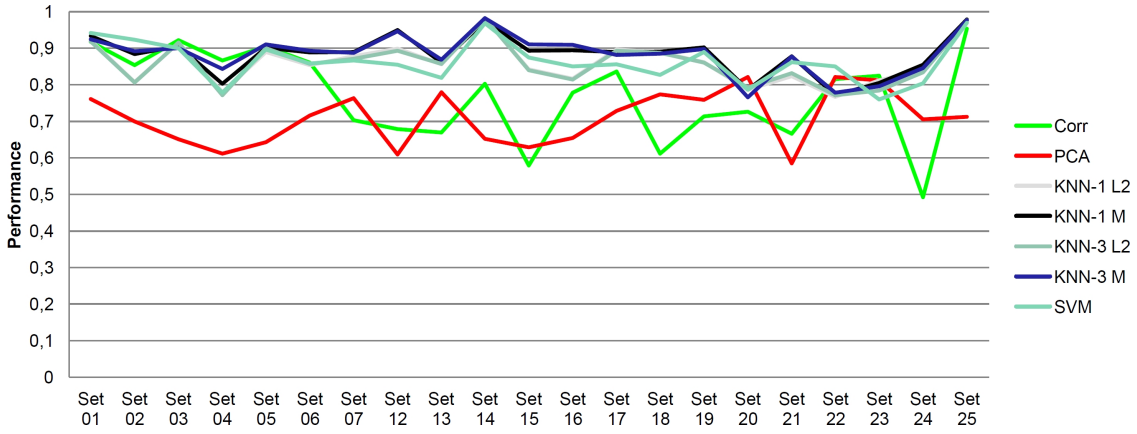


Figure 7.25: **Welding Template Classification Comparison per Dataset:** Average classification results on the welding process datasets from test series α. Thereby, all evaluated approaches used individual optimal amounts of either one class or two class labeled training data for each welding dataset.

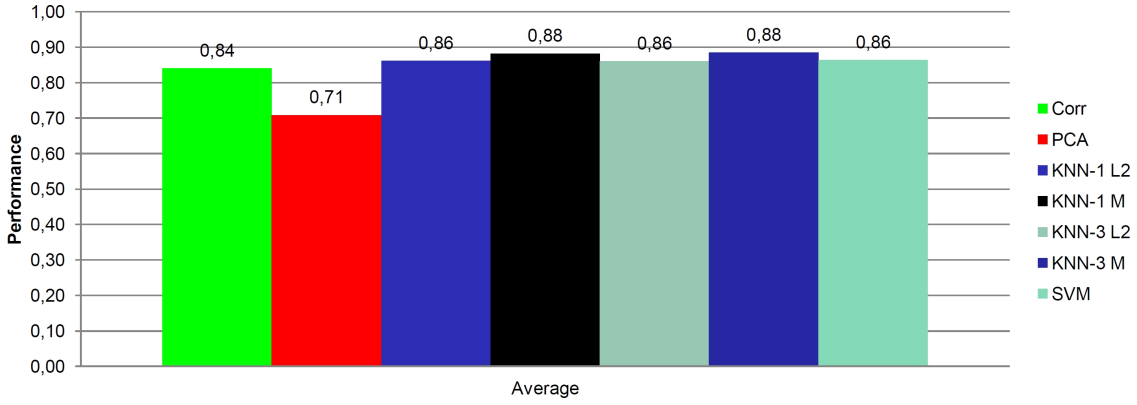


Figure 7.26: **Average Welding Template Classification Comparison:** Average classification results for our proposed correlation based approach (Corr) and other state of the art classification methods for the entire test series α. Again, all evaluated approaches used individual optimal amounts of training data. Although SVM and KNN perform best, our correlation based approach reaches nearly similar results with only fraction of necessary training data and coincidentally better runtime performances.

Although the machine learning approaches (SVM,KNN) achieved the best results, our approach requires the fewest training data while reaching nearly similar classification accuracies. Moreover, the nearest neighbor approaches require similarity evalua-

tions with all training data entries which consequently results in lower runtime performances. In this respect, our approach should be definitely favored as typical reference database sizes of $10 - 20$ cluster centers obviously outperform the others in terms of runtime or real-time capability. Thus, our proposed classification approach achieves test results that are comparable to other state of the art machine learning approaches, while clearly outperforming them in terms of required training data, off-line preparatory tasks in terms of data labeling, and runtime performance.

However, there are still many welding templates wrongly classified by either approach, leaving room for further improvement. Figures 7.27 and 7.28 show examples for welding templates that could not be classified correctly. Although, one could assume a certain amount of label noise present in the data, ambiguities or slight shape deformations as presented obviously require for further even better representations. We also observed that the correlation based similarity metric cannot robustly cope with dense smoke wads and severe illumination and intensity fluctuations caused by, e.g., gas disturbances, which again leaves room for improvement.

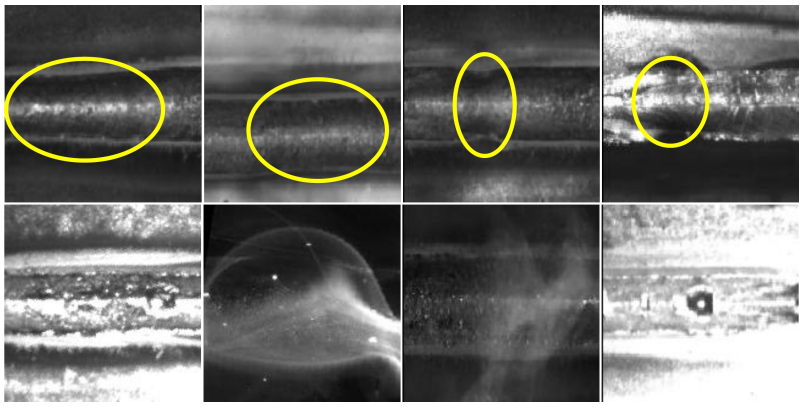


Figure 7.27: **Problematic Weld Seam Templates:** Severe illumination and intensity fluctuations, dense smoke, and slight contractions considering the weld seam width result in wrong classification results, which leaves room for further improvement.

7.4 Conclusion

In this Chapter, two image template based methods for the quality assessment of weld seams are presented and discussed. Both presented approaches rely on tracking related data in terms of axis aligned weld seam templates, extracted from consecutive weld seam images.

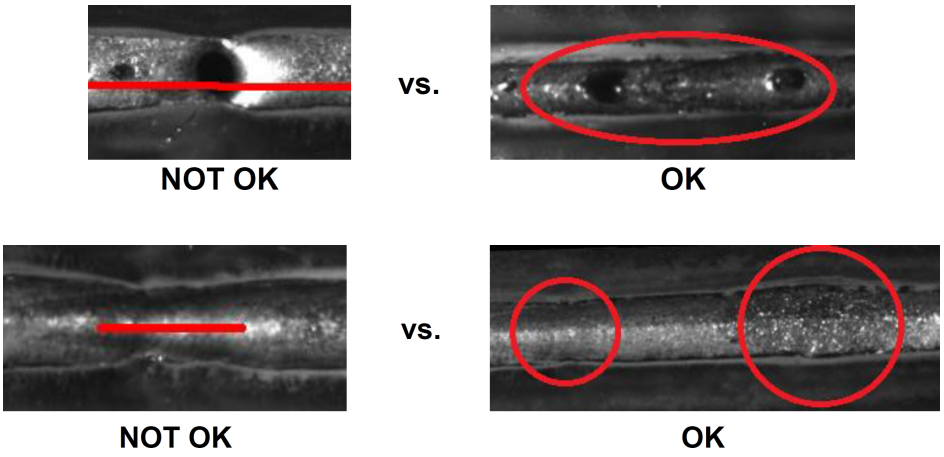


Figure 7.28: **Welding Template Ambiguities:** A remaining issue is given by ambiguities that could not be resolved by either of the evaluated classification approaches. Dark spots caused by slag cannot be distinguished from similar looking holes, and shape deformations that are classified as welding defect by visual inspection of the specimen can yet not be distinguished from very similar shape deformations that are acceptable according to welding experts.

The first approach represents a semi-supervised quality inspection method that robustly generates noise free high quality panorama images of the complete weld seam from start to end. Different image preprocessing algorithms as well as adaptive image blending allows for drastically reducing the typically large amount of industrial image noise like smoke, sparks, evaporating water, or gas disturbances such that a welding expert is able to classify the welding from the generated panorama into either defective or error-free. Thereby we perform adaptive histogram equalization, image dehazing, exposure fusion and graph based bundle adjustment. Our conducted experiments on a large welding dataset that includes diverse welding processes image data, different types of welding defects and errors as well as severe image noise demonstrate that on the one hand a separation of error-free and defective weldings by visual inspection of the generated panorama images is definitely feasible, and that on the other hand our proposed approach outperforms other state of the art methods in terms of repeatability and image quality. According to the visual inspection of the generated panorama images by a welding expert, an accuracy concerning geometric errors or image artifacts in the computed panorama image of $\approx 99\%$ has been achieved.

The second weld seam inspection approach is an unsupervised automatic quality inspection approach that relies on deviations of weld seam templates from an off-line

learned error-free weld seam model. The advantages of our presented approach are three-fold. First, the algorithm requires only few negative (error-free) training data compared to other state of the art approaches. Second, our approach reaches an average classification accuracy of $\approx$ 84%, which nearly equals the results achieved with other state of the art machine learning approaches. And third, due to the significant reduction of the on-line required similarity computations to only a few cluster center comparisons, our approach clearly outperforms the others in terms of runtime or real-time applicability. Moreover, in our experimental evaluations we have presented results based on two different similarity measures, namely the normalized cross-correlation (NCC) and the Euclidean distance between weld seam cross profiles. Both approaches achieved reasonable results for either the classification of entire welding sequences or for the spatial detection of welding defects.

8 Conclusion and Outlook

In this Thesis, the problems of image template based object tracking and quality inspection in harsh industrial environments have been addressed. Thereby, we set a special focus on vision based applications in industrial robotic welding.

Considering object tracking in harsh real-world environments, we have proposed two different template tracking techniques that found their appliance in an industrial robotic welding inspection framework and in an agricultural control system. Both tracking methods are especially designed to robustly perform in harsh industrial manufacturing and outdoor environments, while meeting the underlying requirements on real-time capability, a minimal amount on necessary parameterization, and robustness to various types of tracking complications, disturbing factors and noise.

Considering robust object tracking with a highest degree of invariants, a generic concept for the fusion of an arbitrary number of heterogeneous trackers has been proposed, motivated and evaluated. Thereby, we combined three diverse tracking approaches that in turn report different tracking outputs which are not directly combinable to demonstrate the high variability in both, trackers and reported output. In corresponding experiments we have shown that on the one hand the combination of different tracking approaches and of their specific strengths obviously results in significantly improved tracking performances. On the other hand, we showed that the usage of segmentations as tracking object representations allows for more precise state and model updates and coincidentally for significant noise and background structure reduction, which in turn again results in performance improvements for both, individual and fused trackers. Although, several existing tracking approaches are already able to cope with highly complex non-rigid objects and diverse challenging or highly dynamic scenarios, we come

to the conclusion that the entire range of object transformations, appearance changes, environmental changes, etc. cannot be robustly handled by a single approach. In contrary, the fusion of different cues definitely enables to simultaneously address and to successfully and robustly handle at least a large bulk of the mentioned challenges. To summarize, we proposed a general generic tracking fusion framework that is scalable in the number of contributing trackers, that is not restricted to a specific tracking output structure, and that reports appealing object segmentations, which are in turn used for improved individual updates of contributing trackers, respectively.

Considering future trends in image based tracking, we definitely see large potential in fusion concepts. Under consideration of the rapid development of imaging and processing hardware, the continuous increase of corresponding image quality, and the inevitable development of novel measurement cues, image features and descriptors, a successful handling of the entire range of linear and also non-linear transformations that an object can undergo in videos will definitely be feasible in the future. There exist numerous robust estimators, image features, measurement cues and methods for different problem formulations. It is obvious that a combination of their individual strengths is desired. In our opinion the best way to do so is to fuse the individual approaches in a smart way. Although we have presented such an approach, there is still a lot of room for further improvement. Thus, future work in terms of tracking fusion will address the combination of other trackers in order to, e.g., explicitly handle occlusions which the currently used tracking methods cannot cope with. On-line learning of the tracking states and the corresponding tracking fusion congruences will be another major issue. In this way, a dynamic selection of the most suitable trackers from a large pool of available approaches for a given tracking problems should be possible, instead of combining and iteratively refining tracking support sets.

Considering the task of vision based quality inspection, we have presented a weld seam inspection framework that relies on weld seam image patches provided by a robust weld seam tracking algorithm. Thereby, a semi-supervised approach in terms of high quality panorama image generation, and an unsupervised approach in terms of on-line welding defect detection have been proposed, motivated and extensively evaluated. We thereby especially focused on a considerable small amount of parameters, on real-time capability, and on a minimal amount of off-line preparatory work. Especially for industrial manufacturing the generation of large training databases for learning and classification algorithms is costly and sometimes even not feasible. Consider, e.g., the

intentional generation of welding defects that cover the entire spectrum of possible welding errors. Within our evaluations we came to the conclusion that for the specific task of welding quality assessment robust appearance based matching with an ideal weld seam model allows for a successful classification at reasonable performances, compared to other state of the art approaches that rely on significantly larger training datasets or that are definitely not real-time capable. However, the achieved average classification performances of both, our presented approach as well as of state of the art methods clearly show that a robust and reliable quality assessment based on machine learning concepts might not be ideal, as many welding defects could not be correctly classified by either methods due to remaining ambiguities that could not be resolved relying on 2D image data only.

The proposed welding quality inspection framework has been developed to the state of a research prototype. However, the research in the field of machine learning continuously results in novel and high performance algorithms, which consequently should also be applied for the task of welding quality assessment. Thus, future work in this respect will focus on learning algorithms that are on the one hand solely based on very few one-class training data, and that on the other hand allow for handling of a large variability of both, error-free and defective welding images. Nevertheless, the general fusion concept also finds its appliance in welding quality inspection, as combinations of different classification and measurement cues would allow for more accurate detection of defects. An example would be an additional consideration of 3D surface measures, which would e.g. allow for solving image related ambiguities. Thus investigations in this direction will also be an issue in future work.

A

Basic Mathematical Concepts

A.1 Statistical Moments

Statistical moments are mathematical approaches to analyze or to describe the texture of images or patches. An example is given by the statistical moment of an L-bin image histogram $p(z)$, which describes an n^{th} order statistical moment of image data z_i according to

$$\mu_n(z) = \sum_{i=0}^{L-1} (z_i - m)^n \; p(z_i) \quad with \quad m = \sum_{i=0}^{L-1} z_i \, p(z_i) \; . \tag{A.1}$$

The variance of image data z defines an example of a second order statistical image moment that represents a measure of contrast e.g. for smoothness descriptors according to

$$\mu_2(z) = \sigma^2(z) \; . \tag{A.2}$$

A third order moment is exemplary defined by the measure of skewness of a histogram, and a fourth order moment is exemplary given by the relative flatness of a histogram.

Other examples for statistical moments of an image are the measure of uniformity which is given by

$$U = \sum_{i=0}^{L-1} p^2(z_i) \; , \tag{A.3}$$

or the average entropy measure of an image which is given by

$$e = -\sum_{i=0}^{L-1} p\left(z_i\right)\ log_2 p\left(z_i\right)\ . \tag{A.4}$$

A.2 Bayes Rule

The Bayes Rule or Theorem describes the probability that a pattern x belongs to a specific class w_i according to $P\left(w_i|x\right)$.

$$P\left(A|B\right) = \frac{P\left(A\right)\ P\left(B|A\right)}{P\left(B\right)} \tag{A.5}$$

Consequently, a Bayes classifier allows for choosing the class with the minimum average loss defined by

$$r_j\left(x\right) = \sum_{k=1}^{W} L_{kj}\, P\left(w_k|x\right) \tag{A.6}$$

for a given pattern x from all available classes W.

A.3 Markov Random Fields

A Markov Random Field (MRF) is a statistical method that describes nodes i corresponding to, e.g., image pixels or agglomerations of pixels, hidden variables X_i that are associated with nodes i, e.g., describing the corresponding colors, joint probabilistic models build over nodes i and hidden variables X_i, and direct statistical dependences between hidden variables, e.g., by grouping them. Typically an MRF is represented by a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ consisting of vertices $\mathcal{V}$, e.g., represented by image pixels, and of undirected edges $\mathcal{E}$, e.g., representing image region connectivities. The joint probabilistic models thereby explicitly represent the associations between relatively few pairs of pixels. Examples for these models are Markov Chains, Hidden Markov Models or Hidden MRF Models.

In a Markov Chain the joint distribution $P\left(\cdot\right)$ for random variables $X = (x_1, x_2, \cdots, x_i)$ is given by $P\left(x_i|x_{i-1}, x_{i-2}, \cdots, x_1\right)$, hence modeling long range dependences.

Within a Hidden Markov Model (HMM) additional observations $Z = (z_1, z_2, \cdots, z_N)$ are considered. Thereby Markov models are used as prior models for state variables x_i inferred from independent observations z_i. The joint distribution of a HMM is given by

the product of the likelihood of observations and the prior distribution over the states according to

$$P\left(X = x|Z = z\right) \propto P\left(Z = z|X = x\right)\ P\left(X = x\right)\ . \tag{A.7}$$

Thereby the likelihood of observations can be seen as the measurement of quality of the measurements, and the prior distribution over the states can be seen as the knowledge on X without given observations. If additional model parameters w for states x_i or observations z_i or for both need to be considered, the joint distribution can be rewritten as

$$P\left(X = x|Z = z, w\right) \propto P\left(Z = z|X = x, w\right)\ P\left(X = x|w\right)\ . \tag{A.8}$$

For a Hidden MRF model the joint distribution of states x_i, observations z_i and model parameters w can be reformulated in terms of a Gibbs Energy for MRFs according to

$$P\left(x|z, w\right) = \frac{1}{Z(z, w)} e^{-E(x,z,w)}\ , \tag{A.9}$$

where the energy $E\left(\cdot\right)$ is given as the sum of the sum on unary terms and the sum of pairwise terms according to

$$E\left(x, z, y\right) = \sum_{i \in \mathcal{V}} \phi_i\left(x_i, z_i, w\right) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}\left(x_i, x_j, w\right)\ , \tag{A.10}$$

which can be solved by the Graph Cut algorithm [14], where x denotes an unknown segmentation (per pixel labels), z defines the image data, and w describes foreground and background distributions in terms of color histograms or Gaussian mixture models.

A.4 Cubic Spline Interpolation

A function $\mathcal{S}$ is a spline of degree k only if it conforms to three essential conditions. First, the domain of $\mathcal{S}$ is given by the interval $[a, b]$. Second, $\mathcal{S}$, $\mathcal{S}'$, $\mathcal{S}''$, ..., and $\mathcal{S}^{k-1}$ are continuous, and third, there exist data points x_i which conform $a = x_0 < x_1 < x_2 < \ldots < x_n = b$, and where each subinterval $[x_i, x_{i+1}]$ is defined by a polynomial of degree k. The fundamental idea of cubic spline interpolation is to find a smooth curve $\mathcal{S}\left(x\right)$ that passes through a given set of points, called spline knots. Thereby, cubic polynomials $\mathcal{S}_i\left(x\right)$ are created for each interval $[x_i, x_{i+1}]$, where spline weights

additionally bend the desired curve such that it continuously passes through the given data points. The desired smooth spline function $\mathcal{S}$ is finally given by the n connected polynomials under consideration of the three above mentioned conditions. In particular, a cubic spline interpolation is given by a piecewise function fitting of the form

$$\mathcal{S}(x) = \begin{cases} \mathcal{S}_0(x) & x_0 \leq x \leq x_1 \\ \mathcal{S}_1(x) & x_1 \leq x \leq x_2 \\ \vdots & \vdots \\ \mathcal{S}_{n-1}(x) & x_{n-1} \leq x \leq x_n \end{cases} \tag{A.11}$$

where $\mathcal{S}_i$ is a third degree polynomial with four coefficients, resulting in overall $4 \times n$ parameters. For cubic splines the third degree polynomial and its first and second order derivatives are given below.

$$\begin{aligned} \mathcal{S}_i(x) &= a_i (x - x_i)^3 + b_i (x - x_i)^2 + c_i (x - x_i) + d_i \\ \mathcal{S}_i'(x) &= 3a_i (x - x_i)^2 + 2b_i (x - x_i) + c_i \\ \mathcal{S}_i''(x) &= 6a_i (x - x_i) + 2b_i \end{aligned} \tag{A.12}$$

For a cubic spline interpolation $4 \times n - 2$ parameters are used to interpolate the given data points as well as to conform to the spline requirements of continuity in the function and its first and second derivatives. These requirements are summarized below:

$$\begin{aligned} &\mathcal{S}_i(x_i) = y_i, \quad i \in [0, n-1] \\ &\mathcal{S}_i(x_{i+1}) = y_{i+1}, \quad i \in [0, n-1] \\ &\mathcal{S}_i'(x_i) = \mathcal{S}_{i+1}'(x_i), \quad i \in [0, n-2] \\ &\mathcal{S}_i''(x_i) = \mathcal{S}_{i+1}''(x_i), \quad i \in [0, n-2] \end{aligned} \tag{A.13}$$

The remaining two parameters define the behavior of the spline at the end points, where three common choices should be mentioned here. First, fixed slopes at the end points can be defined:

$$\begin{aligned} \mathcal{S}'(x_0) &= C_0 \\ \mathcal{S}'(x_n) &= C_n \end{aligned} \tag{A.14}$$

A second common choice is called natural spline and is defined by the second derivative being zero at the end points, which results in the spline being extended by a line

outside the endpoints:

$$\mathcal{S}''(x_0) = \mathcal{S}''(x_n) = 0 \tag{A.15}$$

Finally, the so-called *not-a-knot* condition which prescribes that $\mathcal{S}'''$ should be continuous at x_1 and x_{n-1} can be applied. Algorithm 7 summarizes the natural cubic spline interpolation algorithm, where each polynomial $\mathcal{S}_i$ is computed from derived coefficients, and where $\mathcal{S}_i$ is finally described by the Taylor series $\mathcal{S}_i$ about x_i. [26]

Algorithm 7 Cubic Spline Interpolation:

Input: $n+1$ interpolation points $(x_0, y_0), \ldots, (x_n, y_n)$
Output: cubic interpolating spline $\mathcal{S}(x)$

(a) **for** $i = 0, 1, \ldots, n-1$
compute $h_i = x_{i+1} - x_i$ and $b_i = \frac{y_{i+1} - y_i}{h_i}$

(b) Set $u_1 = 2(h_0 + h_1)$ and $v_1 = 6(b_1 - b_0)$

(c) **for** $i = 2, 3, \ldots, n-1$
compute $u_i = 2(h_i + h_{i+1}) - \frac{h_{i-1}^2}{u_{i-1}}$ and $v_i = 6(b_i - b_{i-1}) - \frac{h_{i-1} v_{i-1}}{u_{i-1}}$

(d) Set $z_0 = 0$ and $z_n = 0$

(e) **for** $i = n-1, n-2, \ldots, 1$
compute $z_i = \frac{v_i - h_i z_{i+1}}{u_i}$

(f) Substitute the computed coefficients into $\mathcal{S}_i$:

$$\begin{aligned}\mathcal{S}_i(x) = {} & \frac{z_{i+1}}{6h_i}(x - x_i)^3 + \frac{z_i}{6h_i}(x_{i+1} - x)^3 \\ & + \left(\frac{y_{i+1}}{h_i} - \frac{z_{i+1}}{6}h_i\right)(x - x_i) + \left(\frac{y_i}{h_i} - \frac{z_i}{6}h_i\right)(x_{i+1} - x)\end{aligned}$$

which can be written as the Taylor series about x_i :

$$\begin{aligned}&\mathcal{S}_i(x) = A_i + B_i(x - x_i) + C_i(x - x_i)^2 + D_i(x - x_i)^2 \\ &\text{with} \quad A_i = \mathcal{S}_i(x_i) \quad B_i = \mathcal{S}_i'(x_i) \quad C_i = \frac{1}{2}\mathcal{S}_i''(x_i) \quad D_i = \frac{1}{6}\mathcal{S}_i(x_i)\end{aligned}$$

B

Acronyms and Symbols

List of Acronyms

AE	Angular Error
AP	Affinity Propagation
a.k.a.	also known as
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
DLT	Direct Linear Transformation
DOF	Degree of Freedom
DoG	Difference-of-Gaussians
e.g.	for example (exempli gratia)
EM	Expectation-Maximization
GPU	Graphics Processing Unit
i.e.	that is (id est)
i.i.d.	identically and independently distributed
MRF	Markov Random Field
MSER	Maximally Stable Extremal Region
NCC	Normalized Cross Correlation
OFC	Optical Flow Constraint
PCA	Principal Component Analysis
PDE	Partial Differential Equation
PDF	Probability Density Function
PROSAC	PROgressive SAmple Consensus

RANSAC	RANdom SAmple Consensus
RMSE	Root Mean Squared Error
SIFT	Scale-Invariant Feature Transform
SSD	Sum of Squared Differences
SVD	Singular Value Decomposition
SURF	Speeded-Up Robust Features
TPS	Thin Plate Spline
USB	Universal Serial Bus
FP	False Positives
FN	False Negatives
TP	True Positives
TN	True Negatives
GMM	Gaussian Mixture Model

List of Symbols

$\mathbb{R}^2$	2 dimensional Euclidean space
$\mathbb{R}^3$	3 dimensional Euclidean space
$\mathbb{P}^2$	2 dimensional projective space
$\times$	vector or cross product operator
$\otimes$	convolution operator
$\mathbf{0}$	zero-vector $(0,0,0)^T$ in $3D$ or $(0,0)^T$ in $2D$
$\mathbf{I}$	identity matrix
$\mathcal{N}$	normal distribution

C List of Publications

C.1 2008

Photogrammetric 3D Reconstruction of Lightning Discharges

Markus Heber, Matthias Rüther, Horst Bischof, and Stephan Pack
In: *Proceedings of 32nd Workshop of the Austrian Association for Pattern Recognition (AAPR/OAGM)*
May 2008, Linz, Austria
(Accepted for oral presentation)

Abstract: A computer vision setup is presented for the automated detection and multi-view 3D reconstruction of lightning discharges. The system is designed to operate autonomously in a continuous mode and provide accurate position information of the lightning discharge path and impact point relative to its surrounding area. Extensive tests in a medium scale laboratory environment show that a relative measurement accuracy of 1:200 is reachable with VGA resolution cameras.

Optical Detection and Evaluation of Lightning Discharges

Stephan Pack, Matthias Rüther, Markus Heber, and Horst Bischof
In: *Proceedings of the 29th International Conference on Lightning Protection (ICLP)*
June 2008, Uppsala, Sweden
(Accepted for poster presentation)

Abstract: The observation of lightning strikes by using various approved measurement and location systems is an important task in the field of the lightning research work. Still a lot of unknown phenomena and effects are under discussion belonging to the discharge itself and the striking point at ground. A new approach for the detection and evaluation of the lightning discharge can be presented in this paper by using an optical measurement system, which can give another type of detailed information about the lightning activities, the lightning behavior or the lightning location in an area under observation. The evaluation of the lightning channel in this research work is based on a photogrammetric measurement process. Using two or more camera systems positioned at different view points it is possible to reconstruct the three dimensional location of points in the observed volume. In order to use digital cameras in a measurement system their internal parameters need to be determined in a calibration process. Additional the relative orientation of the cameras is required to triangulate points in the observed volume. This is done by using algorithms of the field of multiple view geometry. A final registration makes it possible to bring the reconstructed points into a global coordinate system. The lightning channel itself will be represented by a number of calculated points in the observed volume. Once the system is installed measurements can be obtained in a fully automatic way. First results of artificial lightning strikes captured in the high voltage laboratory validate the feasibility of this optical measurement process with an achievable accuracy in the range of 1:200. Compared to conventional lightning observation systems this photogrammetric method gives a number of new information about the lightning strike within the limited observed area.

C.2 2010

Catadioptric Pose Estimation for Robotic Pick and Place

Markus Heber, Matthias Rüther, and Horst Bischof
In: *Proceedings of International Conference on Computer Vision Theory and Applications (VISAPP)*
May 2010, Angers, France
(Accepted for poster presentation)

Abstract: Robotic handling of objects requires exact knowledge of the object pose. In this work, we propose a novel vision system, allowing robust and accurate pose estimation of objects, which are grasped and held in unknown pose by an industrial ma-

nipulator. For superior robustness, we solely rely on object contour as a visual cue. We address the apparent problems of object symmetry and ambiguous perspective by acquiring multiple views of the object cheaply and accurately, through a mirror system. Self-calibration of the mirror setup allows us to model the mirror geometry and perform metric multiview contour matching with a known 3D model.

An Online Quality Assessment Framework for Automated Welding Processes

Hartwig Fronthaler, Gerardus Croonen, Jürgen Biber, Markus Heber, and Matthias Rüther

In: *The International Journal of Advanced Manufacturing Technology (submitted)*
Submitted May 2010, revision submitted June 2012
ISSN: 0268-3768
(Impact factor 1.13)

Abstract: Despite the broad adoption of robotic welding, automated quality assessment of the resulting welds is still in its infancy and therefore quality checks are commonly performed by human experts. In this study, we conduct quality testing simultaneously with the welding process (online). For this purpose, a camera positioned behind the welding head provides images of the weld pool and parts of the solidified bead, which is tracked in successive frames. We suggest a weld bead segmentation algorithm, which links the outcome of several local hough transforms with two spline functions in each frame. The actual quality assessment is based on shape features, which can easily be extracted from the segmented weld bead. We propose several distance measures (e.g. Dynamic Time Warping) enabling a comparison between the currently observed weld bead and a reference. Additionally, we suggest variance based ad-hoc quality measures, making reference information expendable. A novel database featuring a large variety of welding tasks, materials and sources of error forms the basis of our experiments. An evaluation of the proposed online quality assessment framework yielding an equal error rate of less than 3%, based on more than 250 single welds, corroborates the robustness of the approach.

C.3 2011

Catadioptric Silhouette-Based Pose Estimation from Learned Models

Christian Reinbacher, Markus Heber, Matthias Rüther, and Horst Bischof
In: *Proceedings of 17th Scandinavian Conference on Image Analysis (IAPR/SCIA)*
May 2011, Ystad Saltsjöbad, Sweden
(Accepted for oral presentation)

Abstract: The automated handling of objects requires the estimation of object position and rotation with respect to an actuator. We propose a system for silhouette-based pose estimation, which can be applied to a variety of objects, including untextured and slightly transparent objects. Pose estimation inevitably relies on previous knowledge of the object's 3D geometry. In contrast to traditional view-based approaches our system creates the required 3D model solely from the object silhouettes and abandons the need to obtain a model beforehand. It is sufficient to rotate the object in front of the catadioptric camera system. Experimental results show that the pose estimation accuracy drops only slightly compared to a highly accurate input model. The whole system utilizes the parallel processing power of graphics cards, to deliver an auto calibration in 20 seconds and reconstructions and pose estimations in 200 milliseconds.

Vision-Based Quality Inspection in Robotic Welding

Markus Heber, Christian Reinbacher, Matthias Rüther, and Horst Bischof
In: *Proceedings of 35th Workshop of the Austrian Association*
for Pattern Recognition (AAPR/OAGM)
May 2011, Graz, Austria
(Accepted for oral presentation)

Abstract: In this work we present a novel method for assessing the quality of a robotic welding process. While most conventional automated approaches rely on non-visual information like sound or voltage, we introduce a vision-based approach. Although the weld seam appearance changes, we exploit only the information from error-free reference data, and assess the welding quality through the number of highly dissimilar frames. In our experiments we show, that this approach enables an efficient and accurate separation of defective from error-free weldings, as well as detection of welding defects in real-time by exploiting the spatial information provided by the welding robot.

C.4 2012

Weld Seam Tracking and Panorama Image Generation for Online Quality Assurance

Markus Heber, Martin Lenz, Matthias Rüther, Horst Bischof, Hartwig Fronthaler, and Gerardus Croonen
In: *The International Journal of Advanced Manufacturing Technology*
Submitted June 2010, accepted May 2012, published June 2012
June, 2012, DOI 10.1007/s00170-012-4263-4
(Impact factor 1.13)

Abstract: Traditionally, automated quality analysis of welding tasks relies on non-visual information, and is mainly done offline. In this work, we introduce an image acquisition system which is capable of monitoring the welding process online, resulting in high-quality image information during an ongoing welding process.We show how to further exploit the image information by automatically tracking the weld seam position in the image, even under heavy smoke and gas disturbances. We exploit the high information redundancy between subsequent frames to generate a seamless image of the entire weld seam, and effectively suppress adverse optical effects caused by smoke and sparks.

Segmentation-based Tracking by Support Fusion

Markus Heber, Martin Godec, Matthias Rẗher, Peter M. Roth, and Horst Bischof
In: *Computer Vision and Image Understanding (submitted)*
Submitted August 2012, revision submitted November 2012
ISSN: 1077-3142
(Impact factor 2.485)

Abstract: In this paper we present a novel fusion framework to combine the diverse outputs of arbitrary trackers, which are typically not directly combinable, allowing for significantly increasing the tracking quality. Our main idea is first to transform individual tracking outputs such as motion inliers, bounding boxes, or specific target image features to a shared pixel-based representation and then to run a fusion step on this representation. The fusion process additionally provides a segmentation, which, in turn,

further allows for a dynamic weighting of the specific trackers' contributions. In particular, we demonstrate our fusion concept by combining three diverse heterogeneous tracking approaches that significantly differ in methodology as well as in their reported outputs. In the experiments we show that the proposed fusion strategy can successfully handle highly complex non-rigid object scenarios where the individual trackers and state-of-the-art (non-rigid object and fusion based) trackers fail. We demonstrate high performance on a large number of challenging sequences, where we clearly outperform the individual trackers as well as state-of-the-art tracking approaches.

Bibliography

[1] D. Aboutajdine and E. Fedwa, *Fast block matching algorithms using frequency domain*, in Proc. International Conference on Multimedia Computing and Systems, IEEE Computer Society, 2011, pp. 1–6. (cited on page 15)

[2] A. Adam, E. Rivlin, and I. Shimshoni, *Robust fragments-based tracking using the integral histogram*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, IEEE Computer Society, 2006, pp. 798–805. (cited on page 92)

[3] S. Avidan, *Ensemble tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE Computer Society, 2005, pp. 494–501. (cited on page 33)

[4] B. Babenko, M.-H. Yang, and S. Belongie, *Visual tracking with online multiple instance learning*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 983–990. (cited on pages 33, 88, 92, and 95)

[5] V. Badrinarayanan, P. Perez, F. L. Clerc, L. O. *thomson R, and D. France, *Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2007, pp. 1–8. (cited on pages 38 and 96)

[6] H. Bay, T. Tuytelaars, and L. V. Gool, *Surf: Speeded up robust features*, in Proc. European Conference on Computer Vision, Springer, 2006, pp. 404–417. (cited on pages 17, 62, 112, and 114)

[7] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, *Eigenfaces vs. Fisherfaces: recognition using class specific linear projection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), pp. 711–720. (cited on page 139)

[8] S. Benhimane and E. Malis, *Homography-based 2d visual tracking and servoing*, Int. J. Rob. Res., 26 (2007), pp. 661–676. (cited on page 30)

[9] C. Bibby and I. Reid, *Robust real-time visual tracking using pixel-wise posteriors*, in Proc. European Conference on Computer Vision, vol. 5303, Springer, 2008, pp. 831–844. (cited on page 36)

[10] M. J. Black and A. D. Jepson, *Eigentracking: Robust matching and tracking of articulated objects using a view-based representation.*, International Journal of Computer Vision, 26 (1998), pp. 63–84. (cited on page 30)

[11] M. Bleyer, C. Rhemann, and M. Gelautz, *Segmentation-based motion with occlusions using graph-cut optimization*, in Proc. DAGM Annual Symposium of the German Association for Pattern Recognition, Springer, 2006, pp. 465–474. (cited on page 85)

[12] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, *Visual object tracking using adaptive correlation filters*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 2544–2550. (cited on page 31)

[13] J.-Y. Bouguet, *Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the algorithm*, 2002. (cited on page 114)

[14] Y. Y. Boykov and M.-P. Jolly, *Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images*, in Proc. IEEE International Conference on Computer Vision, vol. 1, IEEE Computer Society, 2001, pp. 105–112. (cited on pages 82 and 153)

[15] M. Breitenstein, H. Grabner, and L. V. Gool, *Hunting nessie: Real time abnormality detection from webcams*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2009, pp. 1243–1250. (cited on page 44)

[16] M. Brown and D. G. Lowe, *Automatic panoramic image stitching using invariant features*, International Journal of Computer Vision, 74 (2007), pp. 59–73. (cited on page 85)

[17] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*, John Wiley & Sons, 2009. (cited on page 16)

[18] R. Brunelli and T. Poggio, *Template matching: matched spatial filters and beyond.*, Pattern Recognition, 30 (1997), pp. 751–768. (cited on page 16)

[19] P. J. Burt and E. H. Adelson, *The laplacian pyramid as a compact image code*, IEEE Transactions on Communications, 31 (1983), pp. 532–540. (cited on page 119)

[20] P. J. Burt and E. H. Adelson, *A multiresolution spline with application to image mosaics*, ACM Transactions on Graphics, 2 (1983), pp. 217–236. (cited on pages 85 and 117)

[21] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, *Brief: Binary robust independent elementary features*, in Proc. European Conference on Computer Vision, vol. 6314, Springer, 2010, pp. 778–792. (cited on pages 17 and 112)

[22] A. B. Carsten Rother, V. Kolmogorov, *Grabcut: Interactive foreground extraction using iterated graph cuts*, ACM Transactions on Graphics, 23 (2004), pp. 309–314. (cited on pages 78, 81, and 82)

[23] A. Cavallaro, O. Steiger, and T. Ebrahimi, *Tracking video objects in cluttered background*, IEEE Transactions on Circuits and Systems for Video Technology, 15 (2005), pp. 575–584. (cited on page 44)

[24] L. Cehovin, M. Kristan, and A. Leonardis, *An adaptive coupled-layer visual model for robust visual tracking*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2011, pp. 1363–1370. (cited on pages 37 and 38)

[25] V. Chandola, A. Banerjee, and V. Kumar, *Anomaly detection: A survey*, ACM Computing Surveys, 41 (2009), pp. 1–58. (cited on page 43)

[26] E. Cheney and D. Kincaid, *Numerical mathematics and computing*, Brooks/Cole, 2007. (cited on page 155)

[27] P. Chockalingam, N. Pradeep, and S. T. Birchfield, *Adaptive fragments-based tracking of nonrigid objects using level sets*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2009, pp. 1530–1537. (cited on page 93)

[28] D. Comaniciu, V. Ramesh, and P. Meer, *Kernel-based object tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 (2003), pp. 564–577. (cited on pages 45, 46, 57, 67, 68, 77, 84, 87, and 92)

[29] J. W. Cooley and J. W. Tukey, *An algorithm for the machine calculation of complex Fourier series*, Mathematics of Computation, 19 (1965), pp. 297–301. (cited on page 15)

[30] G. S. Cox, *Template matching and measures of match in image processing*, 1995. (cited on page 16)

[31] D. Cremers and G. Funka-lea, *Dynamical statistical shape priors for level set based tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 28 (2006), pp. 1262–1273. (cited on page 35)

[32] N. Dalal and B. Triggs, *Histograms of oriented gradients for human detection.*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2005, pp. 886–893. (cited on page 14)

[33] A. Elibol, R. Garcia, O. Delaunoy, and N. Gracias, *A new global alignment method for feature based image mosaicing*, in Proceedings of the 4th International Symposium on Advances in Visual Computing, Part II, Springer, 2008, pp. 257–266. (cited on page 121)

[34] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, *The pascal visual object classes challenge 2007*. (cited on pages 67, 92, and 95)

[35] J. Fan, X. Shen, and Y. Wu, *Scribble Tracker: A Matting-based Approach for Robust Tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 34 (2012), pp. 1633–1644. (cited on page 36)

[36] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, *Liblinear: A library for large linear classification*, Journal of Machine Learning Research, 9 (2008), pp. 1871–1874. (cited on page 139)

[37] O. Faugeras, Q.-T. Luong, and T. Papadopoulou, *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*, MIT Press, 2001. (cited on page 17)

[38] D. Fecker, V. Märgner, and T. Fingscheidt, *Training of classifiers for quality control of online laser brazing processes with highly imbalanced datasets*, in DAGM/OAGM Symposium, vol. 7476, Springer, 2012, pp. 367–376. (cited on page 42)

[39] H. Fennander, V. Kyrki, A. Fellman, A. Salminen, and H. Kaelviaeinen, *Visual measurement and tracking in laser hybrid welding*, Machine Vision and Applications, 20 (2007), pp. 103–118. (cited on page 42)

[40] M. A. Fischler and R. C. Bolles, *Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography*, ACM, 24 (1981), pp. 381–395. (cited on pages 23, 27, 61, 62, 112, and 113)

[41] E. Fix and J. L. Hodges, *Discriminatory analysis, nonparametric discrimination: Consistency properties*, US Air Force School of Aviation Medicine, Technical Report 4 (1951), pp. 477+. (cited on page 139)

[42] K. Fredriksson, G. Navarro, , and E. Ukkonen, *FASTER THAN FFT: ROTATION INVARIANT COMBINATORIAL TEMPLATE MATCHING*, Transworld Research Network, 2002, pp. 75–112. To appear. (cited on page 16)

[43] Y. Freund and R. E. Schapire, *A decision-theoretic generalization of on-line learning and an application to boosting*, Journal of Computer and System Sciences, 55 (1997), pp. 119–139. (cited on page 44)

[44] B. J. Frey and D. Dueck, *Clustering by passing messages between data points*, Science, 315 (2007), pp. 972–977. (cited on page 129)

[45] M. Godec, P. M. Roth, and H. Bischof, *Hough-based tracking of non-rigid objects*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2011, pp. 81–88. (cited on pages 36, 45, 46, 57, 59, 60, 67, 68, 77, 84, 86, 88, 92, 93, and 95)

[46] J. Gower, *Generalized procrustes analysis*, Psychometrika, 40 (1975), pp. 33–51. (cited on pages 23, 24, and 113)

[47] H. Grabner, M. Grabner, and H. Bischof, *Real-time tracking via on-line boosting*, in Proc. British Machine Vision Conference, BMVA Press, 2006, pp. 61.–6.10. (cited on page 33)

[48] H. Grabner, C. Leistner, and H. Bischof, *Semi-supervised on-line boosting for robust tracking*, in Proc. European Conference on Computer Vision, vol. 5302, Springer, 2008, pp. 234–247. (cited on page 33)

[49] M. Grundmann, V. Kwatra, M. Han, and I. Essa, *Efficient hierarchical graph based video segmentation*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 2141–2148. (cited on page 35)

[50] C. Harris and M. Stephens, *A Combined Corner and Edge Detection*, in Proceedings of The 4th Alvey Vision Conference, 1988, pp. 147–151. (cited on page 17)

[51] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003. (cited on pages 17, 18, 23, 25, and 62)

[52] R. I. Hartley, *In defense of the eight-point algorithm*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), pp. 580–593. (cited on page 26)

[53] K. He, J. Sun, and X. Tang, *Single image haze removal using dark channel prior*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2009, pp. 1956–1963. (cited on pages 110 and 111)

[54] K. He, J. Sun, and X. Tang, *Guided image filtering*, in Proc. European Conference on Computer Vision, vol. 6311, Springer, 2010, pp. 1–14. (cited on page 111)

[55] S. Hinterstoisser, V. Lepetit, S. Benhimane, P. Fua, and N. Navab, *Learning real-time perspective patch rectification*, International Journal of Computer Vision, 91 (2011), pp. 107–130. (cited on pages 31 and 32)

[56] V. J. Hodge and J. Austin, *A survey of outlier detection methodologies*, Artificial Intelligence Review, 22 (2004), pp. 85–126. (cited on page 43)

[57] J. Hoey, *Tracking using flocks of features, with application to assisted handwashing.*, in Proc. British Machine Vision Conference, BMVA Press, 2006, pp. 367–376. (cited on page 39)

[58] S. Holzer, S. Ilic, and N. Navab, *Adaptive linear predictors for real-time tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 1807–1814. (cited on page 62)

[59] S. Holzer, M. Pollefeys, S. Ilic, D. Tan, and N. Navab, *Online learning of linear predictors for real-time tracking*, in Proc. European Conference on Computer Vision, vol. 7572, Springer, 2012, pp. 470–483. (cited on page 33)

[60] L. Ido, L. Michael, and R. Ehud, *A general framework for combining visual trackers – the "black boxes" approach*, International Journal of Computer Vision, 67 (2006), pp. 343–363. (cited on page 38)

[61] I. Ivanov, F. Dufaux, T. Ha, and T. Ebrahimi, *Towards generic detection of unusual events in video surveillance*, in Proc. Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, IEEE Computer Society, 2009, pp. 61–66. (cited on page 44)

[62] M. Jäger, C. Knoll, and F. Hamprecht, *Weakly supervised learning of a classifier for unusual event detection*, IEEE Transactions on Image Processing, 17 (2008), pp. 1700–1708. (cited on page 44)

[63] O. Javed, S. Ali, and M. Shah, *Online detection and classification of moving objects using progressively improving detectors*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, IEEE Computer Society, 2005, pp. 696–701. (cited on page 33)

[64] N. Jiang, W. Liu, and Y. Wu, *Order determination and sparsity-regularized metric learning adaptive visual tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2012, pp. 1956–1963. (cited on page 34)

[65] N. Johnson and D. Hogg, *Learning the distribution of object trajectories for event recognition*, Image and Vision Computing, 14 (1996), pp. 609–615. (cited on page 44)

[66] J.-H. Jung, H.-S. Lee, J. H. Lee, and D.-J. Park, *A novel template matching scheme for fast full-search boosted by an integral image*, IEEE Signal Processing Letters, 17 (2010), pp. 107–110. (cited on page 15)

[67] F. Jurie and M. Dhome, *Hyperplane approximation for template matching*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 24 (2002), pp. 996–1000. (cited on pages 31, 32, and 33)

[68] Z. Kalal, J. Matas, and K. Mikolajczyk, *P-N learning: Bootstrapping binary classifiers by structural constraints*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 49–56. (cited on pages 37 and 38)

[69] Z. Kalal, K. Mikolajczyk, and J. Matas, *Forward-backward error: Automatic detection of tracking failures.*, in Proc. International Conference on Pattern Recognition, IEEE Computer Society, 2010, pp. 2756–2759. (cited on page 40)

[70] T. Kenner, *Fehlererkennung mittels one-class boosting*, master's thesis, Institute for Computer Graphics and Vision, Graz University of Technology, 2007. (cited on page 44)

[71] Z. Khan, T. Balch, and F. Dellaert, *Mcmc-based particle filtering for tracking a variable number of interacting targets*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 27 (2005), pp. 1805–1918. (cited on page 92)

[72] J. Kwon and K. Lee, *Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2009, pp. 1208–1215. (cited on pages 37 and 92)

[73] ——, *Visual tracking decomposition*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 1269–1276. (cited on pages 38, 45, 46, 57, 59, 60, 67, 68, 92, 93, and 96)

[74] J. Kwon, K. Lee, and F. Park, *Visual tracking via geometric particle filtering on the affine group with optimal importance functions*, Proc. IEEE Conference on Computer Vision and Pattern Recognition, (2009), pp. 991–998. (cited on pages 37, 38, 88, and 95)

[75] R. Lanner, *Realtime incremental image stitching for industrial quality inspection*, master's thesis, Graz University of Technology, 2011. (cited on pages 110, 112, 113, 115, 116, 118, 119, 120, 121, 123, and 125)

[76] C. Leistner, A. Saffari, P. M. Roth, and H. Bischof, *On robustness of on-line boosting – a competitive study*, in Proc. On-line Learning for Computer Vision Workshop, IEEE Computer Society, 2009, pp. 1362–1369. (cited on page 33)

[77] A. Levin, D. Lischinski, and Y. Weiss, *A closed form solution to natural image matting*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2006, pp. 61–68. (cited on page 111)

[78] A. Levin, A. Zomet, S. Peleg, and Y. Weiss, *Seamless image stitching in the gradient domain*, in Proc. European Conference on Computer Vision, vol. 4, Springer, May 2004, pp. 377–389. (cited on page 119)

[79] J. P. Lewis, *Fast normalized cross-correlation*. `http://scribblethink.org/Work/nvisionInterface/nip.pdf`, 1995. Last checked on February 23, 2011. (cited on pages 11 and 50)

[80] X. Li, H. Su, and J. Chu, *Multiple model soft sensor based on affinity propagation, gaussian process and bayesian committee machine*, Chinese Journal of Chemical Engineering, 17 (2009), pp. 95–99. (cited on page 129)

[81] D. G. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal on Computer Vision, 60 (2004), pp. 91–110. (cited on pages 17, 62, 112, and 114)

[82] L. Lu and G. Hager, *A nonparametric treatment for location/segmentation based visual tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2007, pp. 1–8. (cited on page 75)

[83] B. D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, in International Joint Conference on Artificial Intelligence, 1981, pp. 674–679. (cited on pages 30, 37, 38, and 114)

[84] H. Ma, S. Wei, Z. Sheng, T. Lin, and S. Chen, *Robot welding seam tracking method based on passive vision for thin plate closed-gap butt welding*, The International Journal of Advanced Manufacturing Technology, 48 (2009), pp. 945–953. (cited on page 41)

[85] Y. Ma, S. Soatto, J. Košecká, and S. S. Sastry, *An Invitation to 3-D Vision, From Images to Geometric Models*, Springer-Verlag New York, Inc., 2004. (cited on page 17)

[86] E. T. Makris D, *Learning semantic scene models from observing activity in visual surveillance*, in IEEE Transactions System Manufacturing of Cybernetic - Part B Cybernetics, vol. 35, IEEE Computer Society, 2005, pp. 397–408. (cited on page 44)

[87] D. Markovic and M. Gelautz, *Video object segmentation using stereo-derived depth maps*, in Vision in a Dynamic World, C. Beleznai and T. Schlögl, eds., Österreichische Computer Gesellschaft, Wien, 2003, pp. 197–204. (cited on page 39)

[88] G. Martin, L. Christian, S. Amir, and B. Horst, *On-line random naive bayes for tracking*, in Proc. International Conference on Pattern Recognition, IEEE Computer Society, 2010, pp. 3545–3548. (cited on page 34)

[89] I. Matthews, T. Ishikawa, and S. Baker, *The template update problem*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 26 (2004), pp. 810–815. (cited on pages 2, 30, 49, 52, and 63)

[90] T. Mertens, J. Kautz, and F. Van Reeth, *Exposure fusion*, in Pacific Graphics, 2007, pp. 1–9. (cited on pages 63, 85, 117, and 118)

[91] F. Moreno-Noguer, A. Sanfeliu, and D. Samaras, *Dependent multiple cue integration for robust tracking*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 30 (2008), pp. 670–685. (cited on page 38)

[92] H. Murase and S. K. Nayar, *Visual learning and recognition of 3-d objects from appearance.*, International Journal of Computer Vision, 14 (1995), pp. 5–24. (cited on page 16)

[93] V.-T. Nguyen, K.-D. Le, M.-T. Tran, and A.-D. Duong, *Fast template matching using pruning strategy with haar-like features*, in Proc. International Conference on Intelligent Human-Machine Systems and Cybernetics, IEEE Computer Society, 2012, pp. 246–251. (cited on page 15)

[94] M. Niethammer and A. Tannenbaum, *Dynamic geodesic snakes for visual tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2004, pp. 660–667. (cited on page 35)

[95] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, *Locally orderless tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2012, pp. 1940–1947. (cited on page 36)

[96] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua, *Fast keypoint recognition using random ferns*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 32 (2010), pp. 448–461. (cited on page 31)

[97] D. W. Park, J. Kwon, and K. M. Lee, *Robust visual tracking using autoregressive hidden markov model*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2012, pp. 1964–1971. (cited on page 35)

[98] P. Perez, J. Vermaak, and A. Blake, *Data fusion for visual tracking with particles*, Proc. of the IEEE, 92 (2004), pp. 495–513. (cited on page 38)

[99] Y. Pritch, A. Rav-acha, A. Gutman, and S. Peleg, *Webcam synopsis: Peeking around the world*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2007, pp. 1–8. (cited on page 44)

[100] X. Ren and J. Malik, *Tracking as repeated figure/ground segmentation*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2007, pp. 1–8. (cited on page 75)

[101] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, *Incremental learning for robust visual tracking*, International Journal of Computer Vision, 77 (2008), pp. 125–141. (cited on page 92)

[102] E. Rosten and T. Drummond, *Fusing points and lines for high performance tracking.*, in Proc. IEEE International Conference on Computer Vision, vol. 2, IEEE Computer Society, 2005, pp. 1508–1515. (cited on page 17)

[103] E. Rosten, R. Porter, and T. Drummond, *Faster and better: A machine learning approach to corner detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 32 (2010), pp. 105–119. (cited on page 112)

[104] S. Roth, L. Sigal, and M. J. Black, *Gibbs likelihoods for bayesian tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2004, pp. 886–893. (cited on page 35)

[105] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof, *On-line random forests*, in Proc. On-line Learning for Computer Vision Workshop, IEEE Computer Society, 2009, pp. 1393–1400. (cited on pages 34, 88, 92, and 95)

[106] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof, *PROST Parallel Robust Online Simple Tracking*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2010, pp. 723–730. (cited on pages 37, 38, and 40)

[107] G. G. Scandaroli, M. Meilland, and R. Richa, *Improving ncc-based direct visual tracking*, in Proc. European Conference on Computer Vision, vol. 7577, Springer, 2012, pp. 442–455. (cited on page 32)

[108] D. Schreiber, L. Cambrini, J. Biber, and B. Sardy, *Online visual quality inspection for weld seams*, The International Journal of Advanced Manufacturing Technology, 42 (2008), pp. 497–504. (cited on page 42)

[109] S. Shahed Nejhum, J. Ho, and M.-H. Yang, *Visual tracking with histograms and articulating blocks*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2008, pp. 1–8. (cited on pages 35, 37, and 38)

[110] Y.-H. Shi, J.-T. Kim, and S.-J. Na, *Signal patterns of high speed rotational arc sensor for gas metal arc welding*, in Proceedings on Sensors for Industry Conference, IEEE Computer Society, 2005, pp. 9–14. (cited on page 41)

[111] B. G. Shin, S.-Y. Park, and J. J. Lee, *Fast and robust template matching algorithm in noisy image*, in Proc. International Conference on Control Automation and Systems, IEEE Computer Society, 2007, pp. 6–9. (cited on page 14)

[112] A. Sibiryakov, *Fast and high-performance template matching method.*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2011, pp. 1417–1424. (cited on page 14)

[113] B. Stenger, T. Woodley, and R. Cipolla, *Learning to track with multiple observers*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2009, pp. 2647–2654. (cited on page 38)

[114] H. Strasdat, J. M. M. Montiel, and A. J. Davison, *Scale drift-aware large scale monocular slam*, in Proceedings of Robotics: Science and Systems Conference, MIT Press, 2010, pp. 1–8. (cited on page 121)

[115] R. Szeliski, *Image alignment and stitching: A tutorial*, Tech. Report MSR-TR-2004-92, Microsoft Research, 2004. (cited on page 22)

[116] R. Szeliski and H.-Y. Shum, *Creating full view panoramic image mosaics and environment maps*, in Special Interest Group on Graphics and Interactive Techniques, 1997, pp. 251–258. (cited on page 119)

[117] S. Taylor and T. Drummond, *Multiple target localisation at over 100 fps*, in Proc. British Machine Vision Conference, BMVA Press, 2009, pp. 58.1–58.11. (cited on page 17)

[118] S. Taylor, E. Rosten, and T. Drummond, *Robust feature matching in 2.3µs*, in IEEE CVPR Workshop on Feature Detectors and Descriptors: The State Of The Art and Beyond, IEEE Computer Society, June 2009, pp. 15–22. (cited on page 17)

[119] L. Torresani and K. C. Lee, *Large Margin Component Analysis*, Advances in Neural Information Processing Systems, 19 (2007), pp. 1385–1392. (cited on page 34)

[120] D. Tsai, M. Flagg, and J. M. Rehg, *Motion coherent tracking with multi-label MRF optimization*, in Proc. British Machine Vision Conference, BMVA Press, 2010, pp. 56.1–56.11. (cited on pages 35, 93, and 94)

[121] M. Uenohara and T. Kanade, *Use of fourier and karhunen-loeve decomposition for fast pattern-matching with a large set of templates*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 (1997), pp. 891–898. (cited on page 15)

[122] M. Unger, T. Pock, W. Trobin, D. Cremers, and H. Bischof, *TVSeg - Interactive Total Variation Based Image Segmentation*, in Proc. British Machine Vision Conference, BMVA Press, 2008, pp. 40.1–40.10. (cited on page 81)

[123] S. D. Utthara Gosa Mangai, Suranjana Samanta and P. R. Chowdhury, *A survey of decision fusion and feature fusion strategies for pattern classification*, IETE Technical Review, 27 (2010), pp. 293–307. (cited on page 37)

[124] P. Viola and M. Jones, *Rapid object detection using a boosted cascade of simple features*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, IEEE Computer Society, 2001, pp. 511–518. (cited on page 33)

[125] T. Vojíř and J. Matas, *Robustifying the flock of trackers*, in Proc. Computer Vision Winter Workshop, Graz University of Technology, February 2011, pp. 91–97. (cited on page 39)

[126] J. F. Wang, B. Chen, H. B. Chen, and S. B. Chen, *Analysis of arc sound characteristics for gas tungsten argon welding*, Journal on Sensor Review, 29 (2009), pp. 240–249. (cited on page 41)

[127] S. Wang, H. Lu, F. Yang, and M. Yang, *Superpixel tracking*, in Proc. IEEE International Conference on Computer Vision, IEEE Computer Society, 2011, pp. 1323–1330. (cited on page 39)

[128] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, *Image quality assessment: from error visibility to structural similarity*, IEEE Transactions on Image Processing, 13 (2004), pp. 600–612. (cited on page 12)

[129] D. Wei and P. Justus, *A probabilistic approach to integrating multiple cues in visual tracking*, in Proc. European Conference on Computer Vision, Springer, 2008, pp. 225–238. (cited on page 38)

[130] K. Q. Weinberger and L. K. Saul, *Distance metric learning for large margin nearest neighbor classification*, Journal of Machine Learning Research, 10 (2009), pp. 207–244. (cited on pages 34 and 139)

[131] G. Welch and G. Bishop, *An introduction to the kalman filter*, tech. report, Department of Computer Science, University of North Carolina at Chapel Hill, 1995. (cited on page 42)

[132] M. Werlberger, W. Trobin, T. Pock, A. Wedel, D. Cremers, and H. Bischof, *Anisotropic huber-l1 optical flow*, in Proc. British Machine Vision Conference, BMVA Press, 2009, pp. 108.1–108.11. (cited on page 85)

[133] P. Xu, G. Xu, X. Tang, and S. Yao, *A visual seam tracking system for robotic arc welding*, The International Journal of Advanced Manufacturing Technology, 37 (2007), pp. 70–75. (cited on page 41)

[134] Z. Yan and D. Xu, *Visual tracking system for the welding of narrow butt seams in container manufacture*, in Proc. UKACC International Conference on Control, Control Systems Centre, 2008, pp. 1–6. (cited on page 41)

[135] A. Yilmaz, O. Javed, and M. Shah, *Object tracking: A survey*, ACM Computing, 38 (2006), pp. 13+. (cited on page 84)

[136] T. Zhang, K. Li, S. Dai, S. Xiao, and H. Huang, *Research on seam tracking controller of mobile welding robot*, in Proc. on Automation and Logistics, IEEE Computer Society, 2009, pp. 2011–2014. (cited on page 41)

[137] X. Zhang, P. Lu, H. Suo, Q. Zhao, and Y. Yan, *Robust speaker clustering using affinity propagation*, IEICE Transactions on Information and Systems, 91 (2008), pp. 2739–2741. (cited on page 129)

[138] W. Zhao, *Flexible image blending for image mosaicing with reduced artifacts*, International Journal on Pattern Recognition and Artificial Intelligence, 20 (2006), pp. 609–628. (cited on pages 63, 64, 85, 116, and 117)

[139] H. Zhong, J. Shi, and M. Visontai, *Detecting unusual activity in video*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, IEEE Computer Society, 2004, pp. 819–826. (cited on page 44)

[140] W. Zhong, H. Lu, and M.-H. Yang, *Robust object tracking via sparsity-based collaborative model*, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2012, pp. 1838–1845. (cited on page 40)

[141] K. Zuiderveld, *Contrast limited adaptive histogram equalization*, in Graphics Gems IV, P. S. Heckbert, ed., Academic Press Professional, Inc., San Diego, CA, USA, 1994, ch. Contrast limited adaptive histogram equalization, pp. 474–485. (cited on page 110)