

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



FEUP

Video Object Tracking - Contributions to Object Description and Performance Assessment

Pedro Miguel Machado Soares Carvalho

Programa Doutoral em Engenharia Electrotécnica e de Computadores

Supervisor: Luís António Pereira de Meneses Côrte-Real (Prof. Dr.)

Second Supervisor: Jaime dos Santos Cardoso (Prof. Dr.)

2012, March



UNIÃO EUROPEIA
Fundo Social Europeu

FCT Fundação para a Ciência e a Tecnologia
MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR

© Pedro Carvalho, 2012

Resumo

O seguimento de objetos, e em particular de pessoas, em sequências de vídeo é o alvo de inúmeras atividades de investigação com um elevado número de algoritmos propostos, focando diferentes cenários ou contextos. A monitorização de ambientes como aeroportos, centros comerciais, edifícios governamentais e de escritórios tem adquirido uma importância crescente. No entanto, a tecnologia necessária à videovigilância automatizada de pessoas também pode ser utilizada em outras áreas como desporto, medicina, marketing ou engenharia civil.

Seguir múltiplos objetos em movimento é um problema de elevada dificuldade e que apresenta muitos desafios, especialmente quando realizado em ambientes não controlados. O seguimento de um objeto implica o seu correto reconhecimento ao longo da sequência. Assim, a descrição dos objetos é um aspecto essencial e, idealmente, deveria permitir a discriminação entre os objetos e o fundo. Todavia, este objetivo frequentemente não é atingido devido a fatores que introduzem ruído no processo de reconhecimento como é o caso de similaridades de aparência, cenários com grande densidade de objetos ou alterações de perspetiva.

Devido à complexidade do seguimento de objetos em vídeo, os investigadores tendem a focar num cenário específico e a introduzir um conjunto de simplificações para restringir os requisitos e tornar o problema mais tratável. Esta prática aumenta a importância da caracterização e avaliação objetiva dos algoritmos, não só para determinar a precisão e robustez de uma solução, mas também a sua adequabilidade a um determinado cenário operacional. A avaliação dos algoritmos é igualmente importante durante as várias fases de desenvolvimento para verificar a concordância com os requisitos subjacentes.

Esta tese apresenta uma nova abordagem ao problema da avaliação de algoritmos de seguimento de objetos em sequências de vídeo que procura estabelecer uma ponte entre diferentes estratégias. A solução complementa métricas de avaliação do estado-da-arte com diferentes requisitos relativamente a informação de referência de forma a unificar a sua utilização e ultrapassar fraquezas das abordagens individuais. Foi combinada informação de diferentes métricas de modo a obter uma medida de erro tão precisa quanto possível, mas com um decréscimo significativo da informação de referência necessária. Tal permitirá preparar soluções de avaliação mais flexíveis e capazes de lidar com uma maior gama de informação de entrada, resultando numa ferramenta poderosa para analisar e caracterizar o comportamento dos algoritmos.

Reconhecendo a importância da descrição dos objetos, esta tese contribuiu também para uma maior compreensão do impacto de diferentes técnicas de representação de aparência no processo global de seguimento visual, complementando análises individuais. Diferentes técnicas do estado-da-arte foram integradas e comparadas numa solução comum através da avaliação do algoritmo de seguimento. Esta é uma evolução lógica destes estudos para incluir o efeito das dependências entre os módulos que constituem um sistema de seguimento de objetos em vídeo.

Palavras-chave: visão por computador, seguimento de objetos, descrição de aparência, avaliação objetiva de algoritmos de seguimento.

Abstract

The tracking of objects, and in particular the tracking of people, in video sequences is the center of many research activities with a great number of algorithms being proposed, targeting different application scenarios or scopes. The automatic monitoring of environments such as airports, shopping centers, government buildings and offices has gained vital importance. Nevertheless, the technology used for automated visual surveillance of humans can also be used in other areas such as sports, medicine, marketing or civil engineering.

Tracking multiple moving objects is a difficult problem. It presents many challenges, especially if it occurs in non-controlled environments. To be tracked, an object must be properly recognized throughout the sequence. Hence, its description is a key aspect and, ideally, should enable the discrimination between objects and the background. Nonetheless, this objective is often foiled by factors that introduce noise in the matching process as is the case of appearance similarity, cluttered scenes or changes in perspective.

Due to the complexity of video object tracking, researchers typically focus on a specific applications scenario and introduce a set of simplifications to restrict the requirements and make the problem more tractable. This practice augments the importance of an objective characterisation and evaluation of the algorithms, not only to determine the accuracy and robustness of a solution, but also its suitability for a given operational scenario. Evaluation of the algorithms is equally important throughout their development to verify the compliance with the underlying requirements.

This thesis presents a novel approach to the problem of evaluating video object tracking algorithms bridging the gap between existing strategies. It complements existing state-of-the-art tracking evaluation metrics with diverse requirements in terms of reference information, aiming to unify its use and overcome weaknesses of individual approaches. Information from different types of metrics was combined to obtaining an error assessment as precise as possible, but with a significant decrease of required reference information. This will enable an evaluation framework more flexible and capable of dealing with a wider range of input information, providing a powerful tool to assess and characterize the behavior of the algorithms.

Aware of the importance of object representation, this thesis also contributed to a better understanding of the impact of different appearance representation techniques on the overall process of visual tracking complementing existing standalone analysis. Different state-of-the-art techniques were integrated and analysed in a common solution through the evaluation of the tracking algorithm. This is a logical evolution of such studies to include the effect of inter-dependencies of the modules of a video object tracking system.

Keywords: computer vision, object tracking, appearance description, objective evaluation of tracking algorithms.

Acknowledgments

This work became possible due to the support of some different people and organizations. In particular, I would like to thank my supervisors, Professor Luís Côrte-Real and Professor Jaime Cardoso, for their support, guidance and availability over the last four years. A very special thank you to my family. To Sandra for her support, love and faith in my ability to finish this thesis, and to Lara who was my source of strength and showed me that sleeping is overrated. I would also like to thank INESC Porto for providing the right environment for high-quality research and to the collaborators who supported me with their companionship. Finally, I thank Fundação para a Ciência e Tecnologia (FCT) and European Commission for financial support through the grant SFRH/BD/31259/2006 and Fundo Social Europeu (FSE).

Pedro Carvalho

To my wife and my daughter.

Contents

Resumo	i
Abstract	iii
Acknowledgments	v
1 Introduction	1
1.1 Contextualisation	1
1.2 A Sample of Application Scenarios	3
1.3 Motivation	6
1.4 Thesis' Structure	6
1.5 Contributions	7
2 Notions on Video Object Tracking	9
2.1 Introduction	9
2.2 An Object and Tracking Definition	9
2.3 Common Obstacles	11
2.4 Assumptions	14
2.5 A Tracking Strategy	15
2.6 Multi-Camera Scenarios	16
3 A Video Object Tracking Landscape	21
3.1 Introduction	21
3.2 Literature Reviews	22
3.3 Tracking Methods	23
3.4 Multi-Camera Scenarios	28
3.5 Object Description	29
3.6 Recent and Future Trends	32
3.7 Tracking Evaluation	33
4 Partition-Distance Methods for Assessing Video Object Tracking Algorithms	37
4.1 Introduction	37
4.2 Extending the Partition-Distance Metrics	38
4.3 Experimentation and Results	41
4.4 Discussion	45

5	Filling the Gap in Quality Assessment of Video Object Tracking	49
5.1	Introduction	49
5.2	A Ground Truth Hybrid Approach	51
5.3	A Hybrid Approach Combining Ground Truth and Non Ground Truth Metrics . .	53
5.4	Experiments	54
5.4.1	Dataset	55
5.4.2	GT and NGT-Based Metrics Correlation	55
5.4.3	Metrics Combinations	56
5.5	Results	59
5.5.1	PD Metrics with GT Combination	59
5.5.2	GT and NGT-Based Metrics Correlation	60
5.5.3	Metrics Combination	61
5.6	Discussion	62
6	Analysis of Object Description Methods in a Video Object Tracking Environment	69
6.1	Introduction	69
6.2	A Multi-Camera Tracking Experiment	70
6.3	Analysis of Object Description Methods	71
6.3.1	Dataset	72
6.3.2	Description Model Variations	72
6.3.3	Evaluation Methodology	75
6.4	Results	76
6.5	Discussion	83
7	Conclusion	87
7.1	Final Remarks	87
7.2	Future Research Lines	88
A	Partition-Distance Metrics	91
A.1	Introduction	91
A.2	The Intersection-Graph Between Two Segmentations	91
A.2.1	Definitions and Notation	92
A.2.2	The Intersection-Graph	93
A.3	Partition-Distance	94
A.4	Asymmetric Partition Distance	96
A.5	Mutual Partition-Distance	97
B	A Detection Based Approach For Tracking Vibrating Lines in Video Sequences	99
B.1	Introduction	99
B.2	Background	100
B.3	Line Detection Using a Shortest Path Approach	102
B.3.1	Shortest Path Method Description	102
B.3.2	Detecting Vibrating Lines in Different Positions	104
B.4	Line Tracking	104
B.4.1	A Windowed Optical Flow Approach	105
B.4.2	A Line Structure Based Approach	106
B.4.3	A Dynamic Time Warping Approach	107
B.5	Validation of the proposed approach	108
B.5.1	Dataset	108

B.5.2	Line Detection Assessment	109
B.5.3	Tracking Assessment	109
B.6	Results	110
B.6.1	Line Detection	110
B.6.2	Line Tracking	110
B.7	Discussion	113
References		117

List of Figures

2.1	Example of a scenario depicting different types of object of interest. Depending on the target application one may only be interest on a specific type (e.g., cars) or every moving object (image from the 2007 IEEE International Conference on Advanced Video Signal based Surveillance).	10
2.2	Example of typical errors in a background/foreground mask obtained with a state-of-the-art image segmentation algorithm.	12
2.3	Example of an object split error where the segmentation represents a unique object as two distinct ones.	12
2.4	Example of a merge error with the segmentation representing two distinct objects as a single one.	13
2.5	Examples of occlusion situations due to the background and due to another object in the scene.	13
2.6	Illustration of the <i>predict-match-update</i> strategy, commonly used in tracking algorithms.	17
2.7	Example of a multi-camera scenarios without overlapping field of view with visible differences in scale and colour.	18
2.8	Example of a multi-camera scenarios with overlapping fields of view.	18
2.9	Example of a multi-camera scenarios with overlapping fields of view, different camera deployments and visual obstacles.	19
3.1	Illustration of the relation between the camera positioning and head occlusion situations.	23
3.2	Problems posed by a high oblique camera positioning to the detection of people through the use of a vertical projection histogram.	24
3.3	Example of a head miss detection resulting from the superimposition of to objects due to the image perspective.	26
4.1	Video segmentation evaluation - setting corresponding to analysing a frame independently from the others (from [CCTCR09]).	39
4.2	Video segmentation evaluation - setting corresponding to analysing the whole sequence at once (from [CCTCR09]).	40
4.3	Video segmentation evaluation - setting corresponding to analysing pairs of consecutive frames (from [CCTCR09]).	40
4.4	Illustrative frames of the synthetic sequence used in the assessment of the framework (from [CCTCR09]).	41
4.5	Time evolution of the proposed video metric for a synthetic sequence. The core metric is the normalised $d_{sym_1}^c$, with linear costs. In the graphic are marked some of the key perturbations introduced in the sequence (from [CCTCR09]).	42

4.6	Representative frames of the SH and OD sequences.	43
4.7	Time evolution of the proposed video metric for the SH and OD sequences. The core metric is the $d_{sym_1}^c$, with linear costs (from [CCTCR09]).	44
4.8	Example of a frame from the SH sequence and the corresponding results (track labelled mask) from the algorithms being used.	46
4.9	Example of a frame from the OD sequence and the corresponding results (track label mask) from the algorithms being used.	47
4.10	Time evolution of the proposed video metric for results for two existing tracking methods. The core metric is the $d_{sym_1}^c$. (from [CCTCR09])	47
5.1	Illustration of some of the possible types of reference information.	50
5.2	Ground Truth combination over a video sequence for the GT-hybrid evaluation proposal (from [CCCR10]).	52
5.3	Excerpt of a graphic depicting evaluation results and illustrating the error transformation effect in the GT-hybrid approach (from [CCCR10]).	53
5.4	Illustration of the concept of using different types of reference information and metrics	54
5.5	Sample frames of a real and synthetic sequence of the dataset used for validation.	55
5.6	Illustration of the Primary Error (PE) and Secondary Error (SE) combination.	57
5.7	Error comparisons for the GT hybrid approach over a synthetic sequence, considering different levels of generated noise and distance between consecutive RS frames (from [CCCR11]).	64
5.8	Error comparisons for the GT hybrid approach over sequence OSOW1 and OSOW2 considering different distances between consecutive RS frames (from [CCCR11]).	65
5.9	Error comparisons for the hybrid approach over a synthetic sequence with 50% noise (from [CCCR11]).	66
5.10	Error comparisons for the hybrid approach over sequence OSOW1 and OSOW2 considering different distances between consecutive RS frames (from [CCCR11]).	67
6.1	Block diagram of the architecture for a wide area surveillance system (from [Tei09]).	70
6.2	Illustrative frames of the dataset used in the experiments.	73
6.3	Example of the height variation of the bounding box. In the left image, full bounding box is used; the bounding box is scaled down from left to right.	74
6.4	Tracking error, using the hybrid metric, for SIFT descriptors with both key points and a grid of points using the full object image (from [COC ⁺ 12]).	78
6.5	Tracking error, using the hybrid metric, for SURF descriptors with both key points and a grid of points using the full object image (from [COC ⁺ 12]).	79
6.6	Comparative analysis of using SIFT with interest point detection and SURF with a grid of points (number of points equal to 1% the object's image pixels) over the test sequences (from [COC ⁺ 12]).	80
6.7	Comparative analysis of colour histogram and HOG-based appearance models for the CAVIAR sequences (from [COC ⁺ 12]).	81
A.1	The right partition is a refinement of the left partition (from [CCTCR09]).	92
A.2	The middle partition is the intersection of the left and right partitions (from [CCTCR09]).	93
A.3	Examples of bipartite graphs (from [CCTCR09]).	93
A.4	Intersection-graph for two segmentations. The weights shown correspond to the number of pixels in the intersection (from [CCTCR09]).	94

A.5	On left and right, two different partitions of the same image—the middle image highlights the points to be removed when comparing both with the partition-distance (from [CCTCR09]).	95
A.6	The middle partition highlights the points to be removed for the asymmetric measures $d_{asy}(R, Q)$ and $d_{asy}(Q, R)$ (from [CCTCR09]).	97
A.7	Partitions A and B are a mutual refinement of each other. (from [CCTCR09]). . .	97
B.1	Image pre-processing (from [CPCCR12]).	104
B.2	Use of multiple displacement vectors in a neighborhood of the point of interest to minimize the tracking error (from [CPCCR12]).	105
B.3	Illustration of the relative positioning concept for tracking points over the line (from [CPCCR12]).	107
B.4	Examples of images from the dataset (from [CPCCR12]).	109
B.5	Various spaces of colours applied to a synthetic frame, followed by a gradient filter (from [CPCCR12]).	111
B.6	Image representations of the dissimilarity matrix for several combinations of the euclidean distances and normalized cross correlation distance as defined by Equation B.6. Darker colours represent lower dissimilarities. The red/yellow line represents the optimal assignment resulting from the DTW algorithm (from [CPCCR12]).	114

List of Tables

2.1	Typical assumptions in motion capture systems (from [MG01])	14
5.1	Correlation between NGT-based metrics and PD metrics with reference segmentations (from [CCCR11]).	60
6.1	Assessment of tracking performance using object metrics. It summarizes results for the use of a sparse (key points) or dense (grid) scan in the computation of SIFT and SURF descriptors, and for different heights of the object image (from [COC ⁺ 12]).	77
6.2	Assessment of tracking performance using object metrics. It summarizes tracking results for the use HOG and histogram based models (from [COC ⁺ 12]).	81
6.3	Overall comparison of the experimented models. For grid based representations, the best results were selected for each type of descriptor (from [COC ⁺ 12]). . . .	82
6.4	Comparison of the processing times for the solution and for individual components of the models: extraction and matching (from [COC ⁺ 12]).	84
B.1	Results for line detection normalized by the image diagonal (from [CPCCR12]). .	111
B.2	Comparison of optical flow methods (from [CPCCR12]).	112
B.3	Results for line tracking considering a single point and the window-based approach (from [CPCCR12]).	112
B.4	Frequency outputs (in rad/s) from sequences 3, 4 and 5 (from [CPCCR11]). . . .	113
B.5	RMSE for tracking results (from [CPCCR12]).	113

Abbreviations and Symbols

ADVISOR	Annotated Digital Video for Intelligent Surveillance and Optimized Retrieval
API	Application Programming Interface
BB	Bounding Box
BBE	Bounding Box Error
BoV	Bag of Visterms
BoW	Bag of Words
CAVIAR	Context Aware Vision using Image-based Active Recognition
CCTV	Closed Circuit Television
CV	Computer Vision
CVIU	Computer Vision and Image Understanding
CVML	Computer Vision Markup Language
CVPR	Computer Vision and Pattern Recognition
DBN	Dynamic Bayesian Network
DET	Descriptor Extraction Time
DMT	Descriptor Matching Time
DR	Detection Rate
DSC	Distributed Smart Cameras
DTW	Dynamic Time Warping
ECCV	European Conference on Computer Vision
EM	Expectation-Maximization
FAR	False Alarm Rate
FNR	False negative Rate
FER	Fragmentation Error Rate
FIW	Full Interval Weighting
FIRST	Fast Invariant Robust Scale feature Transform
FN	False Negative
FOV	Field Of View
FP	False Positive
GLOH	Gradient location-orientation histogram
GMM	Gaussian Mixture Model
GPS	Global Positioning System
GT	Ground Truth
HCI	Human-Computer Interface
HIW	Half Interval Weighting
HMM	Hidden Markov Models
HOG	Histogram of Oriented Gradients
HSV	Hue Saturation Value
IEEE	Institute of Electrical and Electronics Engineers
IJCV	International Journal of Computer Vision

IRFET	Illumination Robust Feature Extraction Transform
IVC	Image and Vision Computing
JPEG	Joint Photographic Experts Group
LI	Linear Interpolation
LT	Linear Transformation
LUL	Living Usability Lab
MAP	Maximum A Posteriori
MF	Multiplication Factor
MPEG	Motion Picture Expert Group
MOT	Multiple Object Tracking
NF	Non Fragmented
NGT	Non Ground Truth
NGTE	Non Ground Truth Error
NW	Normal Weighting
OSOW	OneShopeOneWait
PAL	Phase Alternating Line
PAMI	Pattern Analysis and Machine Intelligence
PCA	Principal Component Analysis
PD	Partition-Distance
PDA	Personal Digital Assistant
PE	Primary Error
PETS	Performance Evaluation of Tracking and Surveillance
PF	Particle Filter
PIDS	Perimeter Intruder Detection Systems
PR	Pattern Recognition
PTZ	Pan-Tilt-Zoom
QoS	Quality of Service
QREN	Quadro de Referência Estratégico Nacional
R&D	Research and Development
RMSE	Root Mean Square Error
ROI	Region Of Interest
RS	Reference Silhouette
RSE	Reference Silhouette Error
SE	Secondary Error
SIFT	Scale Invariant Feature Transform
SOT	Single Object Tracking
SPT	Sequence Processing Time
SURF	Speeded-Up Robust Features
TG	Total Ground truth
TP	True Positive
TRDR	Tracker Detection Rate
TSR	Tracking Success Rate
ViPER	Video Performance Evaluation Resource
ViSOR	Video Surveillance Online Repository
VMD	Video Motion Detection
VOT	Video Object Tracking
WACV	Workshop on Applications of Computer Vision
XML	Extensible Markup Language

Chapter 1

Introduction

"In the middle of difficulty lies opportunity."

Albert Einstein

1.1 Contextualisation

Computer vision is coming down from fiction and making its way into everyday life. Over the last decades many steps have been given towards the goal of automatically interpreting an image, or sequence of images, but this goal has not yet been reached.

Everyday, humans solve many visual problems effortlessly, without conscious awareness (and usually without knowledge) of the visual cognition process. Studies have demonstrated that human observers are capable of recognizing complex real-world scenes and understand a variety of semantic and perceptual information with a mere glance [Oli05]. A common example is the rapid flipping through TV channels; such grasping of vast amounts of information from possibly complex scenes has been referred to as *"gist of a scene"* [Fri79]. Alas, with machines such level of perception has proved difficult to achieve.

Scientists have long been fascinated by the possibility of building truly intelligent machines. To achieve this objective, the capability of perceiving and understanding the surrounding visual world is commonly identified as a prerequisite. Many researchers have been trying to determine how humans understand visual information and studying image properties that contribute to a more effective recognition and categorization of a real-world scene [Mar82, Oli05]. It is expected that it can contribute to build machines capable of perceiving the environment in which they are embedded. Such level of awareness would enable machines to extract information from their environment, rather than being dependent of physical user inputs. Therefore, the interactions and communications between users and machines could be greatly enhanced with the possibility

of automatic identification of the humans present and recognition of their actions (e.g., pose or gestures).

Over the last three decades, the research on computer vision has increased significantly, with visual surveillance and human motion capture being active application domains. It benefited from a favorable technological context. Individual cameras and camera networks have become ubiquitous and pervasive. A report to the European Commission from 2002 [MN02] estimates that four million cameras were installed just in the United Kingdom. In this report, it was also analysed the distribution of systems and cameras installed in different types of locations such as stores, transport stations and streets. In [Vla08] it was estimated that more than 30 million cameras were installed in the United States alone, producing close to 4 billion hours of video footage on a weakly basis. Moreover, the access to high quality and inexpensive video cameras and their integration in personal devices such as mobile phones, laptops, game consoles or PDAs (Personal Digital Assistant) has increased their penetration and made them a part of everyday life. This evolution has been accompanied by a proliferation of powerful computers with ever reducing size and an increase of the speed and flexibility of communication networks.

The use of video cameras is appealing and has contributed to advances in different areas such as medicine or sports. In the context of surveillance, large-scale camera networks are highly desirable since they can provide permanent wide-area monitoring of private and national spaces and infrastructures. Unlike other types of sensors, video cameras can provide wide field of view and good space-time resolution [SKJ10], which favors the continuous proliferation of video surveillance systems. Despite the advances and efforts so far, practical surveillance systems currently deployed are still unable to provide the desired autonomous analysis. This limitation implies that the billions of hours of video captured worldwide are typically stored and, only in some cases, analysed a posteriori for forensic purposes [Vla08]. Furthermore, in the absence of real time analysis, the information cannot be effectively used for preemptive or alert purposes. The increasing demand for automated video processing and analysis, has lead to a growing interest in object detection and tracking algorithms.

Visual surveillance is a paradigmatic case that illustrates the penetration of video cameras and the derived interest for video processing techniques enabling the automatic detection and tracking of objects. Image processing began its impact on the security area in the early 1980s with the introduction of Video Motion Detection (VMD) systems intended to automate Perimeter Intruder Detection Systems (PIDS) [SY99]. However, the poor performance of the systems, which presented a high rate of false alarms, prevented their generalized adoption and more sophisticated solution were demanded. Video surveillance systems kept growing, with a corresponding proliferation of Closed Circuit Television (CCTV) cameras; the monitoring continued to be made by trained human operators, typically in well-staffed facilities, covering large and complex spaces such as airports, parking lots and warehouses. The effective protection of the areas covered by these systems requires the analysis of the video streams captured by an increasing number of cameras, which in turn is highly dependent on significant human supervision. However, the number of human observers is limited and harder to scale. This has resulted in an increasing need to assist

human operators and extend their capabilities. Such autonomous systems are required to operate in highly cluttered outdoor and indoor environments.

Automatically tracking multiple objects, or people, in video sequences is a difficult problem presenting many challenges, especially if it occurs in non-controlled environments as is the case of most everyday scenarios. In these situations, algorithms or tracking systems must deal with factors such as coverage of large areas, humans moving in a group, partial or total occlusion, fast changes in direction, changes in objects' shape, illumination changes and shadows, among others. Ordinary actions like people passing by each other, walking in a group or passing behind stationary objects present complex problems that must be efficiently addressed.

Due to the complexity of video object tracking (VOT) researchers typically focus on specific application scenarios and introduce a set of simplifications to restrict the requirements and make the problem more tractable. For example, two aspects commonly considered in the development of a tracking algorithm are accuracy, also referred to as precision, and computational performance. The former aims to measure how close the captured motion is to the real motion of the object and changes in its shape are captured; the latter is intended to measure how fast incoming information can be processed. Precision is a key aspect for a control application, but can be relaxed in a surveillance scenario. Likewise, for a subway surveillance system the detection of events such as someone falling on the tracks must be immediately signaled, while in video annotation the process can be performed off-line, without time constraints.

Focusing on a specific application scenario has been the tendency so far and it is expected to continue. This practice augments the importance of an objective characterisation and evaluation of the algorithms, not only to determine the accuracy and robustness of a solution, but also its suitability for a given operational scenario. Evaluation of the algorithms is equally important throughout their development to verify the compliance with the underlying requirements. Nevertheless, tracking algorithm's evaluation is itself a research topic, with some existing proposal, but without a generally accepted and used method.

1.2 A Sample of Application Scenarios

Interest in the automatic detection and tracking of people using video cameras has been driven by the many important applications that derive from it. The tracking of objects, and in particular the tracking of people, is the center of many research activities with many algorithms being proposed, targeting different application scenarios or different scopes. The automatic monitoring of environments such as airports, shopping centres, government buildings and offices is an area that has gained vital importance and has received much attention from the research community. Nevertheless, the technology used for visual surveillance of humans can also be used in other areas such as marketing, sports or medicine. Among the many applicable tasks are automatic object detection, video indexing, automated visual surveillance, traffic monitoring, human-computer interfaces (HCI) and motion-based recognition. Recently, the use of gesture information captured by cameras has been highly divulged as a new type of controller for video games, as in the Kinect

system¹. Notably, applications of this system for purposes other than gaming are already being considered. In the Amisco system², tracking technologies have been successfully integrated in a semi-automated process, which enables the annotation and generation of information for videos of football games.

The remaining of this section consists of a brief description of some possible application scenarios for video object tracking, presented for illustrative purposes. The first two examples are directly related to security, whilst the third is focused on monitoring of humans for rehabilitation and health assistance. Finally, a less common application focusing on civil engineering scenarios is described.

Visual Surveillance

Visual surveillance systems typically consist of multiple cameras deployed in a way that maximizes the spaced covered or that has a main focus on critical areas. These cameras are typically placed at high positions to minimize blocking of their field of view (FOV).

Automated video surveillance is deemed to be one of the most resource-demanding applications of machine vision with a wide-spread field of applications [GPBM07]. The desire for a new generation of surveillance systems demands new technologies based on the acquisition and analysis of video sequences in real time. The already challenging task of robust object detection and tracking is further aggravated by the requirements imposed on such systems; in particular the need to operate at any time period and under varying environment conditions, possibly cluttered spaces, and the high expectations regarding performance and minimal error margin.

Examples of the application of automatic visual tracking in a surveillance context include: automatic trajectories recovery from stored videos; abandoned object detection; monitoring scene for unauthorized access; monitoring scene for suspicious activities; automatic person tracking using multiple cameras; pedestrian counting; traffic monitoring for unauthorized parking; traffic monitoring for speeding and law enforcement; monitoring for estimating traffic statistics. The use of these technologies is not intended to change the essence of the surveillance and tracking operations, but to extend it.

A Mobile Surveillance Platform

With the advances in the field of robotics, the use of robots to extend surveillance and security systems endowing it with mobility is closer to become reality. In areas like fire fighting and bomb deactivation, manually controlled robots are already being used. While traditional surveillance systems are limited by the number of installed cameras and their positioning, solutions with mobile units are endowed with higher flexibility. These units can be directed to augment the overall field of view, achieve stereo vision and follow identified moving objects. Moreover, the surveillance robots complement the human security by being able to operate in life-threatening

¹<http://www.xbox.com/en-GB/kinect>

²<http://www.sport-universal.com>

environments such as the presence of gases, fires or smokes. The research project Robot Vigilante (ROBVIGIL) [ROB09] is an example of the use of mobile surveillance units.

This application scenario is closely related to the previous one, sharing many of the difficulties and requirements. Nevertheless, it increases the complexity and places additional requirements to the tracking algorithms. The dimensions of the robots typically imply the positioning of the cameras closer to the ground (approximately at the level of a person), which increases the probability of reduction of the field of view due to nearby objects. Moreover, it is required a trade-off between autonomy and processing capabilities of the robot. Consequently, either more lightweight tracking algorithms are required or the images must be transmitted for remote processing, augmenting the burden on the communication infrastructure.

Assisted Living

The assistance to people with illnesses or the elderly is receiving increasing attention due to their limitations and increasingly isolation in contemporary society. Video cameras can be used to monitor people in their home or in an intermediary habitation to help in their rehabilitation, or used for preventive or alert purposes, e.g., to alert medical emergency teams in the case of falls. The Living Usability Lab for Next Generation Networks (LUL) [Lab09] is a Portuguese industry-academia collaborative R&D (Research and Development) project, active in the field of live usability testing, focusing on the development of technologies and services to support healthy, productive and active citizens.

The monitoring systems in this context typically consist of a camera deployment similar to a surveillance scenario, but with significant differences in the operation of the system. Less people are expected to move in the monitored space, but typically there is a higher probability of visual obstacles (e.g., furniture). Moreover, tracking must be assured under different illumination conditions. While for a surveillance system, the approximate position of a person in the physical space may suffice, in a assisted living context, more detailed information may be deemed. As an example, it is also important to determine possible falls and if people are inanimate for long periods. For rehabilitation, the estimation of posture or tracking of specific body parts may also be important.

A close cooperation between the automated visual monitoring system and other sensors (e.g., presence, gases, illumination) can contribute to a better adaptation of the household environment to people's needs and activities.

Civil Engineering

A visual tracking application scenario that has received less attention from the computer vision research community is civil engineering; more precisely the tracking of deformations and oscillations of structures such as bridges or antennas. This type of analysis is relevant to determine the 'health' of the structure and potential risks. Different types of sensors have been used for this purpose, but the use of computer vision and tracking through video cameras is desired as a noninvasive solution to overcome physical impossibilities or prohibitive prices.

This type of application scenarios differ significantly from the previous ones. For example, the control of the camera positioning is more limited due to the physical environment and displacements of the structures are more subtle than people moving. These and other requirements imply the development and usage of different techniques.

1.3 Motivation

Our interest in video object tracking is derived from its importance in the current social context and the interdisciplinary of the application scenarios.

In the literature we encounter research activities in the field of object detection and tracking with automated visual surveillance as an application scenario. This typically places strict constraints on performance and robustness to varying conditions (e.g., illumination), making the problem more difficult and challenging. Nevertheless, the algorithms and techniques can also be used directly, or with minor changes, in other contexts bestowing this field with a wide range of possible applications. The challenges of automated video surveillance systems are not limited to image processing or computer vision, but also involve the overall system. Traditional solutions are based on a client-server architecture with the server receiving the multiple feeds for processing. Moreover, the information may be conveyed on an existing network also used for other purposes. Hence, in the design of such systems it is also necessary to consider issues such as bandwidth efficiency and security.

The characterisation of the objects to be tracked is a key aspect of tracking and should enable the discrimination between objects and the background. We aim to analyse the impact of different object description techniques on the overall process of visual tracking. Knowing that the evaluation of algorithms is by itself a complex problem and reference information for comparisons is a scarce asset, we also aspire to contribute to the improvement of evaluation frameworks. By analysing the feasibility of combining different types of reference information and metrics, we aim to provide tools to make the tracking assessment process more flexible and generic, increasing the use of available dataset and conducing to a greater adoption of objective evaluation solutions.

Despite a particular focus on visual surveillance, our research interests include the extension or application of tracking techniques to different contexts. To better understand the impact of the different scenarios' requirements in the tracking techniques employed, we have analysed the tracking of vibrating lines in civil engineering structures. Even though it is not the main topic of this thesis, we aim to propose a novel approach to this topic.

1.4 Thesis' Structure

This thesis introduces in Chapter 2 a set of concepts related to visual object tracking intended to aid its reading. It starts by defining visual tracking and the concept of object of interest, followed by a description of common obstacles to algorithms and types of assumptions. Finally, it is presented

an overall tracking strategy common to many algorithms and is introduced the concept of multi-camera tracking.

A landscape of past research work is provided in Chapter 3, describing several existing literature reviews. It is also presented a set of publications to contextualize the contributions of this thesis. Work on video object tracking as well as on object description techniques is reviewed; recent research trends and topics identified as relevant for the near future are also discussed.

A new framework for the evaluation of tracking algorithms is presented in Chapter 4. It consists of an extension of the partition-distance metrics initially proposed for assessing image segmentations. We describe the underlying concepts of this framework, the methodology to assess its applicability to tracking evaluation and the corresponding results. The framework is further extended in Chapter 5 by introducing the concept of using different types of reference information in the computation of the error measures; it is followed by the description of the combination of the partition-distance measures with metrics without reference information.

In Chapter 6 we analyse different description techniques integrated in a common tracking solution to assess their impact in the overall result, characterizing the dataset and the evaluation methodology used in the experiments.

Final comments and conclusions are presented in Chapter 7 together with a description of a set of research topics than can be pursued in future work.

A summary of the theoretic concepts about the partition-distance metrics are provided in Appendix A to assist the reading of this thesis.

Appendix B describes a contribution parallel to the main objectives of this thesis, consisting of a proposal for automatic detection and tracking of vibrating lines in a civil engineering scenario. Different detection and tracking strategies are described and compared.

1.5 Contributions

We summarize below the contributions of this thesis toward more robust tracking solutions and more flexible evaluation techniques. In this thesis we have:

1. extended the partition-distance metrics' framework, initially conceived for the evaluation of image segmentation, to the assessment of video segmentation and tracking algorithms;
2. introduced the concept of a ground truth hybrid framework and proposed a strategy to reduce the effort associated with the generation of exact pixel based reference information required for the assessment of tracking algorithms using the partition-distance metrics; it is based on the fusion of the error measure obtained with different types of ground truth;
3. introduced the concept of metric fusion in tracking assessment, thus increasing the flexibility of the partition-distance evaluation framework, enabling its use over test or evaluation sequences with sparse reference information. The goal was accomplished through the integration of metrics without ground truth into the framework;

4. performed the first integration and test of the Fast Invariant Robust Scale feature Transform (FIRST) descriptors in a video object tracking scenario; resulting from the integration efforts and tests, feedback was provided to help improve the ongoing implementation of these descriptors and the corresponding Application Programming Interface (API);
5. analysed a set of state-of-the-art appearance description techniques in a common tracking solution to assess their impact in the overall tracking, providing a more complete characterization of their capabilities;
6. described a new proposal for the automatic detection and tracking of vibrating lines in a civil engineering scenario; the method makes use of the shortest path method and Dynamic Time Warping to track an arbitrary number of points of the lines.

Publications related to this thesis

- Luís F. Teixeira, Pedro Carvalho, Jaime S. Cardoso, and Luís Côrte-Real. Automatic Description of Object Appearances in a Wide-Area Surveillance Scenario. *IEEE International Conference on Image Processing (ICIP) 2012*. (submitted).
- Pedro Carvalho, Telmo Oliveira, Lucian Ciobanu, Filipe Gaspar, Luís F. Teixeira, Rafael Bastos, Miguel Sales Dias, Jaime S. Cardoso, and Luís Côrte-Real. Analysis of Object Description Methods in a Video Object Tracking Environment. *Machine Vision and Applications*. (submitted).
- Pedro Carvalho, Miguel Pinheiro, Jaime S. Cardoso, and Luís Corte-Real. A detection based approach for tracking vibrating lines in video sequences. *Pattern Analysis and Applications*. (submitted).
- Pedro Carvalho, Miguel Pinheiro, Jaime S. Cardoso, and Luis Corte-Real. A shortest path approach for vibrating line detection and tracking. In *Proceedings of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2011)*, pages 9-16, 2011.
- Pedro Carvalho, Jaime S. Cardoso, and Luís Corte-Real. Filling the gap in quality assessment of video object tracking. *Image and Vision Computing*. (submitted).
- Pedro Carvalho, Jaime S. Cardoso, and Luís Corte-Real. Hybrid framework for evaluating video object tracking algorithms. *Electronics Letters*, 46(6):411-412, 2010.
- Jaime S. Cardoso, Pedro Carvalho, Luís F. Teixeira, and Luís Corte-Real. Partition-distance methods for assessing spatial segmentations of images and videos. *Computer Vision and Image Understanding*, 113(7): 811-823, 2009.

Chapter 2

Notions on Video Object Tracking

“Everything should be made as simple as possible, but no simpler.”

Albert Einstein

2.1 Introduction

The tracking of objects in video sequences has been an important research topic over the last two decades and has promoted a vast number of published work, as presented in Chapter 3. Work on this topic has targeted different application scenarios and the proposed algorithms made use of techniques from areas such as image processing and machine learning.

This chapter aims to introduce a set of concepts intended to facilitate the reading of the remaining of this thesis. Section 2.2 presents a definition for video object tracking and characterizes the concept of object of interest. Section 2.3 characterises common obstacles affecting tracking algorithms. The role of assumptions is discussed in Section 2.4. In Section 2.5 it is described the principles of a commonly employed tracking strategy. Finally, Section 2.6 introduces the concept of multi-camera tracking.

2.2 An Object and Tracking Definition

In the context of video processing, tracking involves the detection of objects and the identification of correspondences in subsequent frames to achieve a consistent labeling. This can be understood as the identification of objects’ trajectories as they move through the observed space. However, in different contexts, the tracking algorithm can provide additional information such as the size, area, shape or orientation of the objects.

In visual tracking, an object is commonly defined as something that is of interest for analysis. From such a definition, one can conclude that the objects to track (i.e., the objects of interest) will depend on the application domain and intended objectives. For example, consider the scenario in Figure 2.1; if we are only interested in collecting traffic statistics, the trajectory and number of passing vehicles would suffice; however, in an application for traffic safety it would also be of interest to detect and track pedestrian to prevent dangerous crossings. For a given visual surveillance scenario determining the trajectories of objects may be sufficient, but in other cases, such as physiotherapy, more detail information (e.g., about limbs) is required. Nevertheless, a common factor is the coherent definition in time of relations between data associated to meaningful regions.



Figure 2.1: Example of a scenario depicting different types of object of interest. Depending on the target application one may only be interest on a specific type (e.g., cars) or every moving object (image from the 2007 IEEE International Conference on Advanced Video Signal based Surveillance).

In the literature there are references to the tracking of objects in video sequences and to the tracking of people. Although in the context of tracking, people are consider a specific type of object, the term ‘people’ is typically present when referring to algorithms customized to the tracking of humans. In this thesis, the terms ‘object’ and ‘people’ will be used interchangeably, except for specific topics where special attention will be given to the intended scope. Also in the literature, it is common to find the term ‘track’ to refer to an object that was detected and is being observed. ‘Track’ is also often used to encompass the set of information associated to an object which typically includes the object identity, position, size and appearance features, but may also include trajectory, posture or other application specific information.

2.3 Common Obstacles

The automated visual tracking of multiple objects in video sequences, in general, is a challenging problem, affected by difficulties related to the complexity of the scene and the complexity of the tracked objects [YJS06], such as:

- camera motion
- noise in images
- abrupt and complex object motion
- changing appearance patterns of both the object and the scene
- scene illumination changes
- nonrigid object structures
- loss of information caused by projection of the 3D world on a 2D image
- partial and full object occlusions
- real-time processing requirements

Typically, video object segmentation is used to detect the objects of interest, preceding the actual tracking. Hence, common errors in video segmentation produced by causes like illumination changes, shadows and reflections can propagate to tracking. If objects cannot be properly detected, then tracking will most probably fail. Moreover, if objects are wrongly detected, tracking will try to process them. In either case, tracking errors may result from the error propagation. Figure 2.2¹ depicts an image obtained from a surveillance camera and the corresponding background-foreground segmentation mask obtained with a state-of-the-art algorithm [TCCR07]. The segmentation mask depicted in Figure 2.2(b) illustrates common problems such as the influence of shadows and errors in the detection of object boundaries.

Two other common types of error typically induced by a faulty segmentation are referred to as split and merge. The former is illustrated in Figure 2.3 and occurs when an object being tracked is split into two distinct ones; they will receive distinct labels and will be tracked independently. In the latter, two objects are merged with the system being unable to recognize them as the objects already tracked; consequently the object resulting from the merge is identified as a new one (this is also referred to as object fusion). The object merge/fusion error may occur in the case of an isolated object, but is very frequent in the case of group movement. An example of a situation that may originate merge errors is depicted in Figure 2.4, where people close together are segmented as a single blob.

¹Throughout this thesis images of sequences of the CAVIAR project [Cav04] and PETS 2006 workshop [PET06] will be used for illustration purposes as well as in experiments.

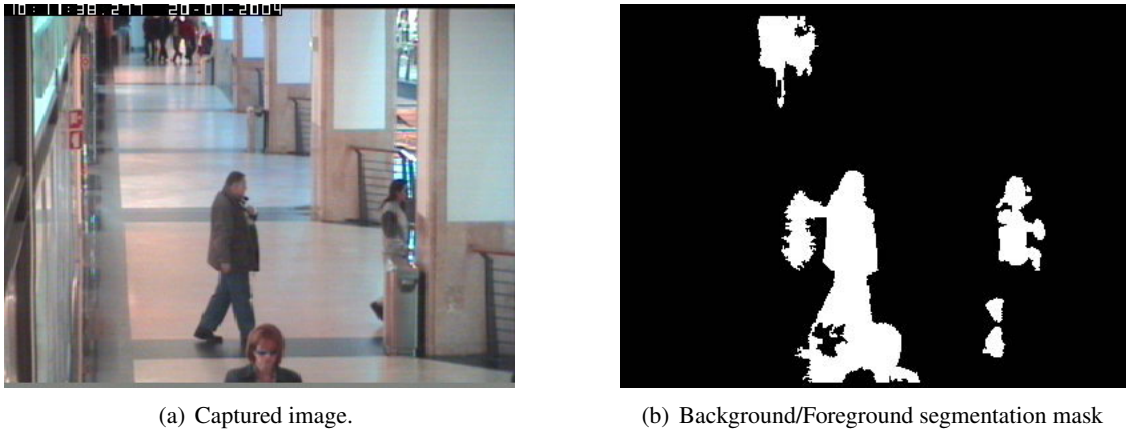


Figure 2.2: Example of typical errors in a background/foreground mask obtained with a state-of-the-art image segmentation algorithm.



Figure 2.3: Example of an object split error where the segmentation represents a unique object as two distinct ones.

Occlusion is another problem that has received much attention. This situation occurs when an object is occluded, or masked, by another one closer to the camera. Occlusion may consist of: mutual occlusion, in which an object is occluded by another; self occlusion, in which an object masks part of itself; occlusion by the background which happens, for example, when a person walks behind a stationary object. A tracking system should be capable of resolving the identities of objects before and after occlusion. Examples of occlusion situations are depicted in Figure 2.5.

Video Object Tracking algorithms can be classified into Single Object Tracking (SOT) or Multiple Object Tracking (MOT) algorithms, according to the number of targets. SOT has been the target of research over the last decades [WADP97, HYJ11]. However, when considering multiple targets, additional obstacles are added to this already challenging problem, specifically [CSWY09]: multiple targets; multiple motion conditions; visual ambiguity. Multiple moving objects mean that

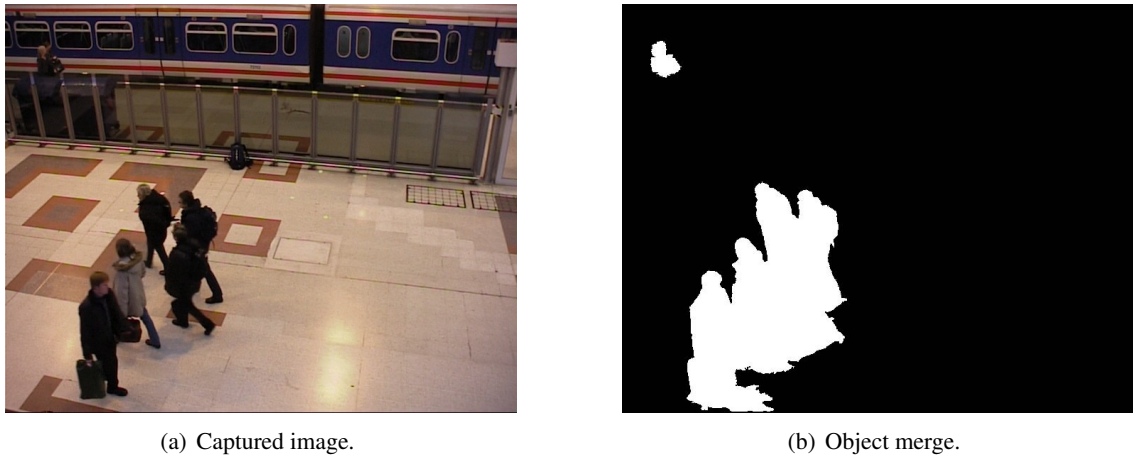


Figure 2.4: Example of a merge error with the segmentation representing two distinct objects as a single one.

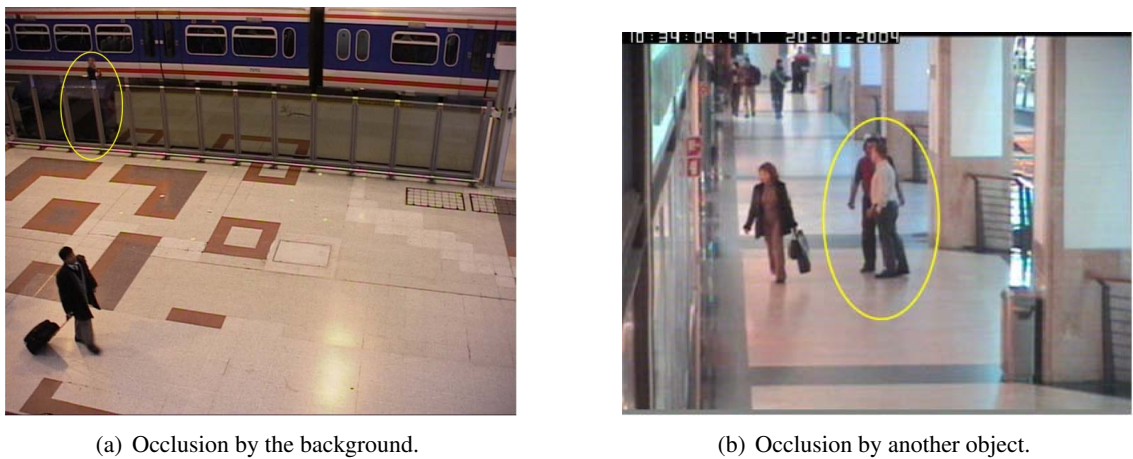


Figure 2.5: Examples of occlusion situations due to the background and due to another object in the scene.

it is not only necessary to distinguish objects from the background, but also between objects, with an augmented difficulty due to a possible similarity of shape, appearance and motion. The possibility of complex motions, subject to many conditions such as exiting, entering or splitting, which may interlace, make the position prediction process more complex and increase the probability of visual ambiguities between the objects and consequently the swap of identities. Also, the existence of multiple target objects means that it is much harder to determine the optimal configuration of the targets' temporal correspondences.

2.4 Assumptions

Tracking is seldom used by itself. Rather, tracking is typically used to convey information of objects' shape and/or location to be used in a higher-level application. For example, in [MG01] tracking is presented as an initial step of a human motion capture system.

The complexity of the object, and people, tracking process makes it very difficult for a single solution to cover every possible application scenario and to deal with the magnitude of problems that may rise. Hence, tracking algorithms generally target a specific domain and try to tackle the problem through the use of prior knowledge. This information may be incorporated in the form of models or translated into a set of assumptions. The latter are made to constraint the tracking problem to a particular application context resulting in its simplification [MG01, YJS06, SMC05].

Many assumptions, of several types, can be made in the context of a particular application scenario, from constant illumination to a fixed number of people in the scene. A possible classification and examples of assumptions in the context of human motion capture were provided by Moeslund [MG01] who divided them into two classes: appearance assumptions, which concern aspects of the subject to be tracked; movement assumptions, which concern the movements of both the camera and the subject. Typical assumptions are listed in Table 2.1.

Table 2.1: Typical assumptions in motion capture systems (from [MG01])

Related to movements	Related to appearance
subject remains inside the scene	Environment
none or constant camera motion	constant lighting
only one person in the scene	static background
the person faces the camera at all time	uniform background
movements parallel to the camera	known camera parameters
no occlusion	special hardware
slow and continuous movements	
only move one or a few limbs	Subject
motion pattern of the subject is known	known start pose
subject moves on a flat ground plane	known subject
	markers placed on the subject
	special coloured clothes
	tight-fitting clothes

Given the complexity of visual tracking and recognizing that the requirements, e.g., precision, computational performance, robustness and real-time constraints vary considerable with the application scenario, it is easy to understand the benefit of designing algorithms in accordance with the specific requirements of each one. Precision measures how closely the captured motion is to the real motion. Computational performance measures how fast incoming information can be processed. Robustness is directly related to the insensitivity to varying conditions such as illumination. Finally, real-time constraints entail that information (i.e., the frames) must be processed (1) at the rate it is captured or (2) before the end of an event's lifespan (e.g., before a thief breaks in) [MG01].

Which assumptions (their type and number) are chosen for a specific solution depend on the goal of the system and on the environment in which it will operate. Assumptions can serve as factor of evaluation in the sense that the more assumptions are made, the more sensitive systems tend to be to context changes, even if more processing efficient.

2.5 A Tracking Strategy

Video object tracking is essentially a similarity search across two neighboring frames of the video sequence and can be viewed from two different perspectives [Zha04]: detection-based and matching-based. In the first, the temporal correspondence is accomplished by recognizing the objects across different frames, whilst in the second there is an estimation of the trajectory of each object. The computational efficiency of this similarity search can be improved by taking into account the approximate object location in the previous frames and modelling motion and shape [IB98]. Once the position of the target in the current frame is estimated, as well as its uncertainty, one can then search for a confidence region for the target candidate that is most similar to the target model according to a given comparison measure [ZYS09].

As alluded, the manner in which a tracking algorithm operates and the type of information used is context dependent. Nevertheless, it is possible to identify three aspects common to nearly every tracking algorithm. The first consists in the identification of relevant objects in the scene. This is typically accomplished with figure-ground segmentation, commonly based on change detection [WADP97, HHD00, SM02], that may be followed by some additional filtering (for example, identify people to be tracked in a scene also containing moving vehicles). Once the relevant objects are segmented they are transformed into a different representation, more suitable for the algorithm. Examples of representations are points, primitive geometric shapes, silhouettes or contours and skeletal models [YJS06]. Finally, the objects are tracked from frame to frame according to given strategy. A common tracking approach is the *predict-match-update* illustrated in Figure 2.6: at time instant $t - 1$ a prediction of the position α of the object at time t is made based on existing information (Figure 2.6(a)); at time instant t , a search is conducted in the neighborhood of α for the object (Figure 2.6(b)); if the object is found the information is updated with the new observation (Figure 2.6(c)). Gravilla [Gav99] presented a general framework for model-based tracking based on the work described in [OB80], which encompasses these concepts.

This tracking strategy is well reflected in the use of Dynamic Bayesian Networks (DBNs) and recursive Bayesian state estimation. For a more detailed description of Bayesian networks and their role in tracking, the reader is referred to existent literature [IB98, Cha91, FP02, KF09]. In particular, it is urged the reading of the paper by Dore et al. [DSR10], which provides a good and concise description of these concepts and references for relevant related work.

The intended objective in tracking is to consistently assign labels to objects across a sequence, thus estimating its trajectory. This task is performed by observing, or measuring, the approximated state of the objects in each frame. In this process, the complexity of each frame due to the background and presence of other moving objects can be seen as noise. Dynamic Bayesian

Networks have been a widely used technique in video object tracking, providing the mechanisms to update the state of objects being tracked at each time step, thus acting as a filtering process that minimizes or eliminates the noise from the observations. Recursive Bayesian state estimation uses a two step method based on a prediction-correction strategy [DSR10]. At time instant t , information available up to instant $t - 1$ is used to predict the next state; a correction is performed when the observation at instant t becomes available. It is clear the overlap with the genetic tracking strategy described above. Two well known solutions for the use of DBNs are the Kalman filter [Kal60, WB95, May79] and particle filter [IB98]. The former assumes linearity of the state observation model and gaussian noise. The latter is a framework for recursive state estimation for problems where the underlying assumptions of the Kalman filter can not be guaranteed.

2.6 Multi-Camera Scenarios

A particularly relevant scenario in visual surveillance consists of tracking multiple objects (typically people) using multiple cameras. Multi-camera tracking is not within the scope of this thesis. Nevertheless, they are also characterised due to their relevance to the video object tracking topic. Multi-camera scenarios may occur if one wishes to observe a given area from multiple points of view (for example, different views of the same scene can help to solve occlusion situations), or monitor several areas (disjoint or not), each covered by one or more cameras, as is the case of public buildings or commercial areas. In these cases, the complexity of the system increases. While a single-camera tracker searches for correspondences only between frames (captured by a single camera), the task of a multi-camera tracker is also to establish correspondences between observations of objects across multiple cameras. The ultimate goal is to correctly tag all instances of the same visual object at any given location and at any given time instant [TCR09]. Figure 2.7 depicts an example of a multi-camera scenario where there is no overlapping between the covered areas. An example of a scenario with overlapping FOV is presented in Figure 2.8. It is possible to perceive differences in both the colour and scale between the captured images, which adds more complexity to the scenario. Figure 2.9 also consists of images of cameras with overlapping FOV, but with noticeable differences in the camera deployment.



(a) Time $t - 1$, state estimation (broken line).



(b) Time t , object matching (full line).



(c) Time t , update state.

Figure 2.6: Illustration of the *predict-match-update* strategy, commonly used in tracking algorithms.



Figure 2.7: Example of a multi-camera scenarios without overlapping field of view with visible differences in scale and colour.

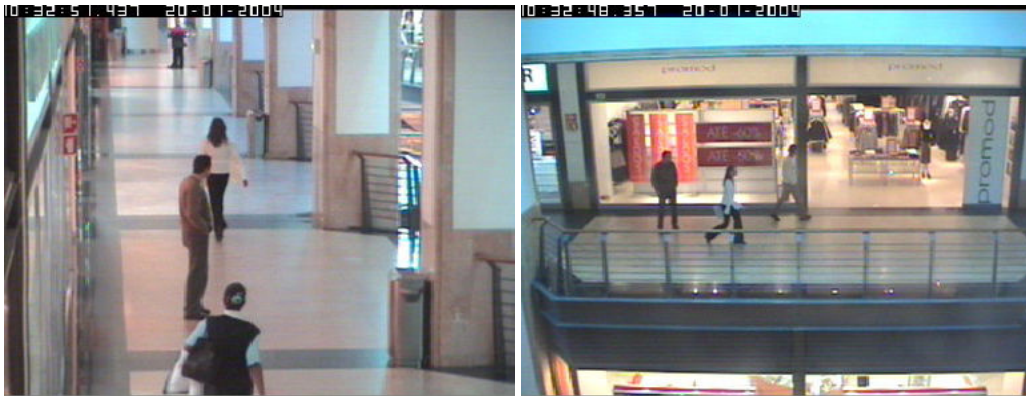


Figure 2.8: Example of a multi-camera scenarios with overlapping fields of view.

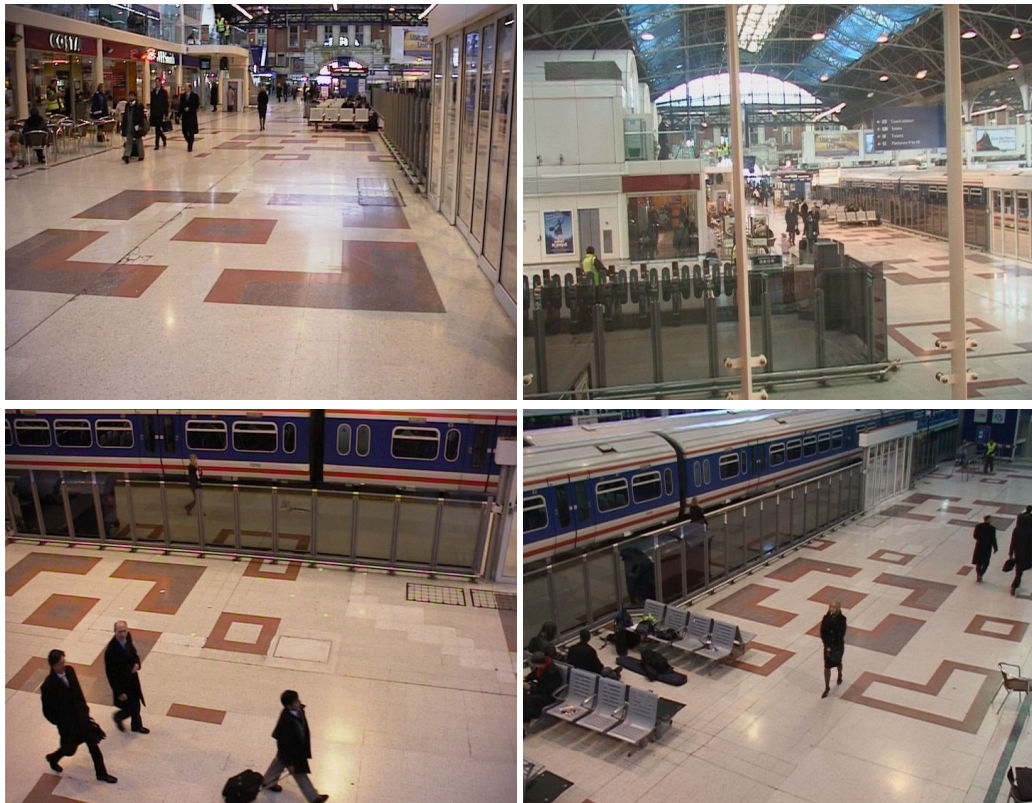


Figure 2.9: Example of a multi-camera scenarios with overlapping fields of view, different camera deployments and visual obstacles.

Chapter 3

A Video Object Tracking Landscape

“Science, in the very act of solving problems, creates more of them.”

Abraham Flexner

3.1 Introduction

Although the automatic tracking of objects, and in particular people, has gained vital importance in recent years, everyday situations still present complex problems within the scope of research activities. There are innumerable research efforts, targeting different aspects of associated problems and a vast number of published work in leading international journals such as IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Pattern Recognition (PR), Computer Vision and Image Understanding (CVIU), IEEE Transactions on Image Processing, Image and Vision computing (IVC) and International Journal of Computer Vision (IJCV), as well as international conferences and workshops such European Conference on Computer Vision (ECCV), IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), IEEE International Conference on Image Processing and the Workshop on Applications of Computer Vision (WACV).

Given the existence of several comprehensive surveys on object tracking, many with a special focus on visual surveillance and human motion analysis, it is not within the goals of this thesis to make an exhaustive description of existing research areas and techniques. Rather, a set of related work is presented in Sections 3.3 to 3.7 to contextualize this thesis and its contributions to ongoing and future research. These surveys also present different taxonomies to organize the many reviewed techniques. For a more in-depth study of underlying principles, techniques and algorithms related to video object tracking, the reader is referred to existing surveys. A brief description of these literature reviews is presented in Section 3.2.

3.2 Literature Reviews

One of the earliest relevant literature reviews is due to Aggarwal et al. [ACLS94]. The survey covered various methods used in articulated and elastic non-rigid motion prior to 1994. It also described approaches to articulated motion with and without prior shape models. Aggarwal and Cai provided two other surveys [AC97, AC99]. The first covered work prior to 1997, while the second focused on the interpretation of human motion and reviewed additional publications, thus extending the previous paper [AC97]. Aggarwal and Ryoo provided a recent update in [AR11]. Cédras and Shah [CS95] reviewed methods for motion extraction prior to 1995. Human motion analysis was decomposed into action recognition, recognition of body parts and body configuration estimation. Gavrilin [Gav99] described work prior to 1998 on face and hand gesture recognition, but also covered full body tracking. The survey was organized according to a taxonomy consisting of 2D approaches with and without explicit shape models and 3D approaches.

Work on human motion capture from 1980 to 2000 was revised by Moeslund [MG01] using a taxonomy based on major stages of such systems: initialization, tracking, pose estimation and recognition. Moeslund et al. [MHK06] provided an update to the previous survey, reviewing a large number of papers over the preceding five years using the same taxonomy. Poppe [Pop07] provided an overview of techniques for vision-based human motion analysis, restricted to large body parts, without the use of markers.

Wang et al. [WHT03] provided a comprehensive survey on human motion analysis through computer vision covering research from 1997 to 2001. They defined human motion analysis as encompassing the detection, tracking, recognition of people and, more generally, human behavior understanding. The survey followed an hierarchical organization from low to intermediate to high-level vision, arguing that such structure maps a general human motion analysis framework. A similar organization was followed by Hu et al. [HTWM04], but mapped to a visual surveillance framework, to provide a survey on object motion and behavior analysis in a visual surveillance context. The authors described a visual surveillance framework as being composed of the following stages: environment modeling; motion detection; moving object classification; tracking; behavior description and understanding; human identification; fusion of data from multiple cameras. Relevant developments in general strategies for all the stages were also reviewed.

A recent survey by Yilmaz [YJS06] provides a very good overview of video object tracking techniques and algorithms. It describes many important underlying issues such as appearance representations, motion models and object detection. The tracking methods covered in this survey are grouped into three major categories: methods establishing point correspondence; methods using primitive geometric models; methods using contour evaluation.

Tuytelaars and Mikolajczyk [TM08] provided an overview of invariant interest point detectors (as the authors initially explain, the term ‘detector’ is commonly used to refer to the tool that extracts features from an image), documenting their evolution. The authors characterised an ideal local feature detector and reviewed the corresponding literature of the preceding four decades. Overviews and performance assessment of local interest point detectors and local descriptors can

also be found in [MS05, MS03, MS04, MS05].

3.3 Tracking Methods

Although the Pfinder system [WADP97] dates from 1997, it is one of the most referenced works in the literature. This system was limited to a single person and fixed camera. It used a Maximum A Posteriori Probability (MAP) to detect and track people by employing statistical models for colour and shape. It used an adaptative filter to update the statistics of the visible pixels thus compensating illumination changes and a blob model to represent the global aspects of the person. For its initialization, Pfinder required the observation of the scene without people to build the scene model. The system was devised to expect a single person in the scene space and had difficulties dealing with large and sudden changes in illumination, making it more suitable for indoor environments.

Several people tracking algorithms in the literature have proposed the use of head position information to discriminate between objects and detect people for tracking [HHD00, SM02]. It has been argued that heads are less likely than other body parts to be occluded [ZN04]. However, the validity of this argument requires a camera positioning above the normal level of people's heads. With a camera positioning at the level of objects in the scene, there is a high probability of occlusion due to objects closer to the camera. Figure 3.1 presents two images captured with cameras place at two different positions: at the level of person's body and nearly parallel to the ground; in a high oblique position. In these images it is possible to observe the impact of the camera position in occlusion situations and, in particular, visibility of people's heads.



(a) Horizontal positioning (image from the dataset of the Network of Excellence VISNET II).



(b) High oblique positioning.

Figure 3.1: Illustration of the relation between the camera positioning and head occlusion situations.

The W^4 system [HHD00] was intended for operation in real-time and outdoor environments. It used a monocular gray-scale camera with the objective of detecting and tracking humans, and recognizing their interactions with other people or objects. The authors used shape analysis and

tracking to locate both people as a whole and their body parts, and object silhouettes resulting from the segmentation process. Silhouettes were classified into three predetermined classes, namely singular people, group of people and other objects; their vertical projection histogram was used to locate peoples heads. The location of tracked persons in subsequent frames was computed through the use of second order motion model applied to the median coordinate of the silhouette, and matching was accomplished through binary edge correlation. The system was also able to determine if a person was carrying an object, by analyzing the symmetry of the silhouette relative to the vertical axis passing through the median coordinate. The authors used an intensity-based statistical background model, which made the system susceptible to illumination changes and shadows. When people moved in groups, the system was unable to detect individual persons or their characteristics (postures, body parts or objects being carried) and assumed that people in the group were in the upright position to segment the group into individuals. High oblique camera positioning posed several problems as partial superimposition of objects closer to the camera can occur resulting in a vertical projection histogram through which it is not possible to detect the heads of people. An example of such situation is depicted in Figure 3.2. To avoid such problems the W^4 system required the camera to be placed nearly parallel to the ground plane. Such restriction increases the probability of occlusion situations (see Figure 3.1). Hydra [HHD99] was developed as an extension to the W^4 system to provide a more refined detection and tracking of people moving in small groups.

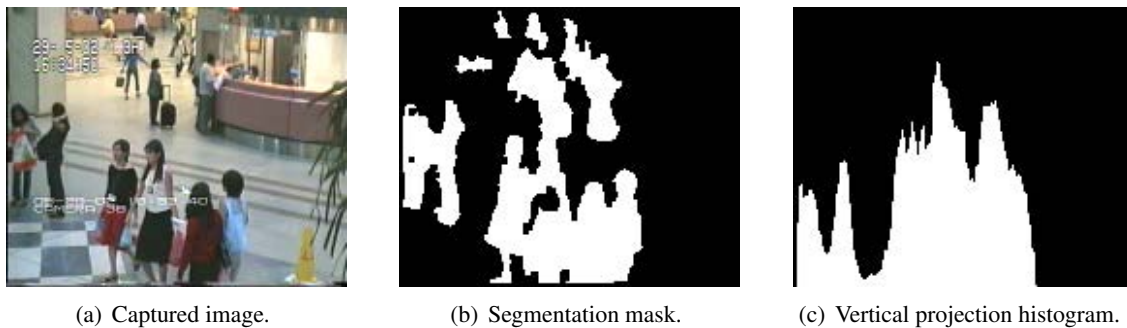


Figure 3.2: Problems posed by a high oblique camera positioning to the detection of people through the use of a vertical projection histogram.

In [Sie03], Siebel described the development of a new people tracking module, the Reading People Tracker, which was integrated in the ADVISOR system (Annotated Digital Video for Intelligent Surveillance and Optimized Retrieval) [SM02]. This work was based on the Leeds People Tracker [Bau95] and on the premise that tracking objects with a combination of two different detection algorithms makes that process more robust. The module used a monocular camera and targeted indoor environments. It was composed of three main parts: a Head Detector to aid in the initialisation of tracks; a Region Tracker, featuring region splitting and merging for multiple hypothesis matching; an Active Shape Tracker using a PCA (Principal Component Analysis) [Shl05] generated model of pedestrian outline shapes. An active shape model was used for the contour of

people in the images. The models were trained with images of walking pedestrians. For position estimation, the authors employed a Kalman filter with second order motion models. Although the tracker proved capable of dealing with some occlusion if a sufficient part of the object was visible, it faced difficulties in situations where people move close to each other and significant occlusion occurred. In more complex situations or when people are too close to each other the track was also lost. In these situations there were no means to identify a person as one being followed when the track was regained. Moreover, people moving as a group were wrongly identified because the system was unable to detect the individual silhouettes. If a segmented object was split due to partial occlusion, the separate regions were incorrectly classified as independent objects.

Head detection was also used by Zhao and Nevatia [ZN04] as a first step for person detection. The algorithm was designed for a single stationary camera placed above people's head in an inclined position looking down. The placement of cameras at this position helps to avoid some situations of occlusion and reduce the probability of people's heads being hidden. Nevertheless, situations as the one illustrated in Figure 3.3, where the upper torso of a person closer to the camera superimposes on another person further away, may prevent the head from being detected. The authors proposed to track human locomotion in complex situations by taking advantage of models that encompass prior knowledge. It was used: a camera model in conjunction with the assumption that humans move in the ground plane to perform transformations between the physical world and the image; ellipsoids as a coarse human shape model to constrain the shape of an upright human; a 3D human articulated locomotion model to assist the identification of locomotion modes and recognition of walking humans. The knowledge provided by the human shape and camera model was used in boundary and shape analysis to compute human hypotheses. Each hypothesis was then tracked in 3D in subsequent frames using a Kalman filter and the object appearance constrained by its shape. The use of these models and in particular of the human shape model augmented the ability to identify individual people and handle occlusion. In the second part of work of Zhao, the modes of locomotion (e.g., walking, running, standing) and 3D body postures were estimated by making inference in a prior locomotion model. The description of the algorithm states that shadows were removed by taking into account the camera model, the date and hour, and the position of the sun to calculate the shadow projection and remove it. However, this process cannot be so easily performed in non controlled indoor environments where modeling of the illumination sources is more complex.

Denman et al. [DCS05] used a stereo camera and targeted indoor scenarios with a high rate of occlusions. They used the depth map, motion models and a Kalman filter, a two step ellipse fitting process and a new fit metric to solve the ambiguities caused by occlusion. This system first used an adaptive background segmentation and a coarse people detection to identify the regions of interest, i.e., those that may contain people. Afterward, people were located by applying head detection and ellipse fitting algorithms as described in [HHD00, ZN04]. Occluded people were also located through the use of the depth map. The matching between located people and those being tracked was done by combining into a metric computationally simple object statistics: distance from the persons expected median to the median of the object; differences in shape and size; differences in

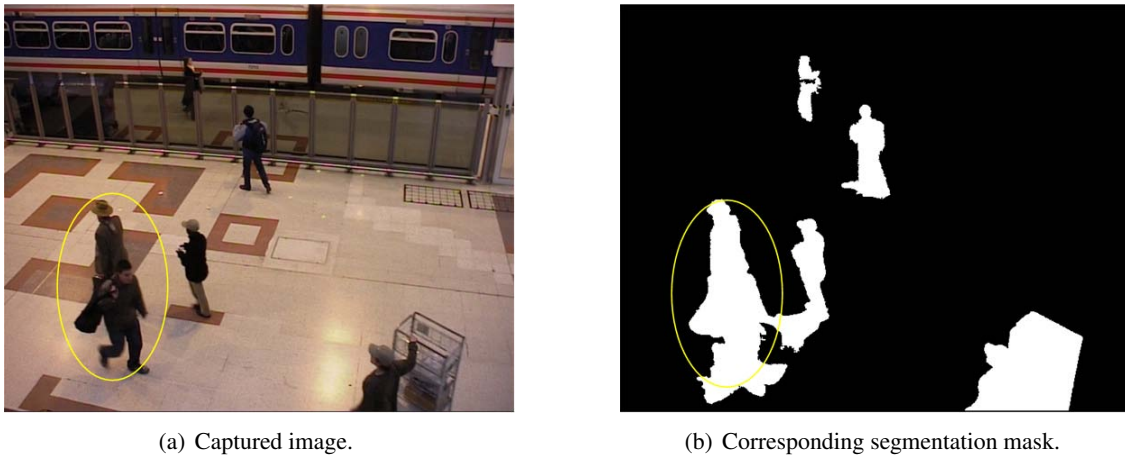


Figure 3.3: Example of a head miss detection resulting from the superimposition of to objects due to the image perspective.

the expected disparity of the person and that of the object; expected direction of travel. The system maintained a list of previous and currently tracked people, storing statistics relating to size, shape, appearance and motion models for each person. Nevertheless, the system had a limited reliability ranging from 5 to 15 meters away from the camera.

A different tracking approach was proposed by Cavallaro [CSE05] which used both object and region information to track generic objects in video sequences. The algorithm used the image representation of an object as a partition hierarchy with regions being described using spatial and temporal features. A correspondence between objects in successive frames was established through the correspondence of the objects' regions. The two major limitations of the proposed algorithm were the initialization of tracks when groups of objects enter the scene and the occurrence of occlusion of the objects. Moreover, proximity between objects could cause confusion in the assignment of regions to objects.

In the seminal work of Isard and Blake [IB98], particle filters were proposed for multi-object tracking in cluttered environments. Particle filters can be used in problems where the non-linearity and non-gaussianity are inadequate for the use of the Kalman filter [DSR10]. The higher complexity and computational cost of particle filters have incited researchers to develop methods to improve their applicability as in [Smi07].

Snidaro et al. [SMC05] have implemented a counting and tracking algorithm for the purpose of environment control in the context of an intelligent building. The algorithm was intended, for example, to count the number of people in a room to control the air conditioning, and consequently adjust the temperature to the room occupation. The authors used cameras placed on a top view to avoid occlusion situations and tracking was accomplished through the use of a Kalman filter. The authors took advantage of the fact that, from a top view, the area occupied by a person is nearly constant; a fact that can be exploited for counting. To eliminate objects that could be interpreted as person candidates, for example shopping cars, the authors used Canny's edge detection and Hough transform [SS01] to identify rigid bodies which can typically be associated

to straight lines. Groups of people were detected based on the corresponding area and tracked as a new entity. A hierarchical relationship between objects in situations of merge and split was maintained. In this type of scenario, switch of identities is not critical since the main interest is to estimate the number of objects present and their location. The vertical positioning of the camera and the absence of requirements concerning detailed shape information enables a simplification of the algorithm. However, it also means that less information can be extracted from the objects, preventing the implementation of other functionalities such as object identification, determining the body posture or tracking limbs. People counting, more precisely, the counting of pedestrians moving in groups, was also the objective of the work of Kilambi et al. [KRJ⁺08]. Accurate camera calibration information was used to determine the projection of the foreground blobs (resulting from a background/foreground segmentation step) in a ground and a head plane. Based on projection information concerning the average area and shape of a single individual, the number of people present in the group was estimated.

The mean-shift algorithm has been widely used for visual tracking [Bra98, LYZ⁺09, ZYS09]. colour-based mean-shift has been identified as a fast and effective algorithm for tracking colour blobs. It is robust to small “distracters”, but if these are larger than the object colour area and near, it may cause the lost of tracking. To try to overcome limitations of colour-based mean-shift regarding the complexity of the background and occlusion, Liu et al. [LYZ⁺09] proposed the integration of motion cues with the colour cues. Although experiments were performed with sequences containing multiple people, only a single object was tracked. Mean-shift was also used by Zhou et al. [ZYS09] in combination with the Scale Invariant Feature Transform (SIFT) technique [Low04] with the objective of improving tracking. The SIFT features were used to match a region of interest across frames and mean-shift for a similarity search through the use of colour histograms. The expectation-maximization (EM) algorithm [Del02] was used to combine information from the two measures and achieve the match of similar regions. The authors demonstrated that the results obtained with the proposed framework were better than when SIFT and mean-shift were used isolated. Experiments focused on tracking a single object and problems in the presence of severe image clutter and occlusion were reported.

The integration of multiple cues has been identified as a technique contributing to robust visual tracking [LYZ⁺09]. In [HYJ11] colour histograms and Histogram of Oriented Gradients (HOG) [DT05] were combined for object tracking. The choice of these features was justified with their ability to effectively represent an object and their computation efficiency. It was proposed the adaptative assignment of confidences to each feature to augment the weight of the most discriminative feature at each time. The authors experimented with both Kalman and particle filters for the feature evaluation and temporal consistency. In the paper it was stated that the proposal approximates real time and was robust to complex backgrounds. However, during the experiments only one object was tracked and the algorithm was unable to handle large scale changes to the object.

In [AVBK10] it was proposed a cascade of region descriptors where each grid segments an object into a different number of sub-rectangles of equal sizes. Grids of rectangles of finer blob size describe local information whereas grids of rectangles of coarser size describe a more global be-

havior; SIFT and SURF (Speeded-Up Robust Features) [BETVG08] were analysed as description techniques.

Avidan proposed an ensemble of weak classifiers combined through AdaBoost [FS97] into a strong classifier that decides for each pixel in the next frame if it belongs to the object or to the background [Avi07]. The classifier builds a confidence map and mean-shift was used to find the match. The algorithm required the marking of the region of interest at the beginning and HOG was used for the descriptors. The method failed in the presence of long term occlusion.

Despite the huge amount of research work published, a robust and effective solution for video object tracking remains challenging in many scenarios. Occlusion is a factor that greatly contributes to this difficulty; in particular full occlusion that can last for long periods of time. The majority of algorithms, such as the ones described above, assume that part of the objects are always visible, i.e., only partial occlusion occurs. When this assumption is not verified the algorithms fail and recovery of tracks is seldom possible. Papadourakis et al. [PA10] proposed a tracking strategy to deal with long term and full occlusion based on the concept of “object permanence”, which suggests that a fully occluded object will reappear near its occluder. Object position estimation in the next frame was accomplished through a simple linear model. An ellipse and a Gaussian Mixture Model (GMM) were used for spatial and colour distribution respectively. Although the authors obtained interesting results, their strategy depends on a good background-foreground segmentation with the guarantee that there is only one blob per object. In realistic scenarios these assumptions are easily violated. Hence, the algorithm is severely affected by segmentation errors. Moreover, it has difficulties coping with irregular motions and complex shapes.

Unlike most common bottom-up approaches that have as typical steps object detection and tracking, Kosmopoulos et al. [KC10] suggested the use of the raw data as a whole to recognize behaviors. The extraction of features directly from the image frames has advantages. Particularly, it avoids the need to segment and track the objects of interest, a process with a high degree of difficulty and prone to errors. The authors used Hidden Markov Models (HMM) [Rab89] with a t-student observation model to cope with the characteristics of the sequential data. Information from multiple cameras was fused to deal with confusions and occlusion. Promising results were obtained in scenarios like assembly lines where it is possible to take advantage of the statistical consistency and learned behavior. However, the inability to use object models in the holistic representations increases the probability of object ambiguity. Consequently the view point or the use of multiple cameras assumed greater importance.

3.4 Multi-Camera Scenarios

The algorithms and systems described in the previous section were essentially related to tracking in an area covered by a single camera. Nevertheless, typical surveillance systems involve the coverage of wide areas requiring the use of multiple cameras which may cover overlapping and non-overlapping areas. Hence, it becomes necessary to track objects as they move in an area

covered by a single camera, but also as they move across other areas and are captured by other cameras, while assuring the consistent system wide assigning of identity to the objects.

Although algorithms such as those described can be used in multi-camera systems for object tracking at an individual camera, i.e., intra-camera tracking, it is also necessary to deal with the differences between cameras and perform inter-camera tracking. The use of camera calibration information is a common strategy for detecting objects and defining correspondences across multiple cameras as in [CA99]. In [GB08] the authors described an approach to object tracking using multiple uncalibrated cameras with non-overlapping field of view. The system was intended to incrementally learn relationships between cameras to overcome differences in scale and colour. This approach did not require *a priori* calibration or explicit training periods.

In [TCR09] it was proposed a method for the recognition of objects across multiple views. This work was extended in [Tei09], testing the integration of the description model and recognition framework with a single camera tracking system to achieve a scalable multi-camera solution without information about camera deployment and dedicated tracking algorithms. It was proved that the propose solution could recognize objects and solve ambiguities and errors, but full integration was precluded due to the use of different appearance models.

Inspired by the vision system of a chameleon, Zhou et al. [ZWW10] proposed a framework using PTZ (Pan-Tilt-Zoom) cameras that could be used either independently or in cooperation. Such use of the cameras enabled the capture of images with different resolutions, since the parameters of these cameras are variable. Moreover, by working in cooperation, stereo vision can be achieved. Micheloni et al. [MRF10] also proposed the use of PTZ cameras in combination with static ones. The PTZ cameras were used to complement the static cameras enabling multiple overlapping fields of view (FOV), which is useful to solve problems such as occlusions or localization.

Alahi et al. [AVBK10] proposed a multi-camera framework for object detection with the cameras working in a master-slave mode. The authors proposed to capitalise on the increasing number of mobile cameras existing in devices such as PDA or mobile phones. It was assumed that the objects were properly detected in the the fixed cameras (the masters) with a corresponding model, based on a cascade of grid descriptors, being extracted and used for recognition in the mobile cameras (the slaves).

3.5 Object Description

A key issue in video object tracking is the representation of the objects and how effectively it discriminates between objects and from the background during the tracking process [HYJ11]. colour and colour histograms have been widely used in monocular tracking with interesting results, but leading to poor performance in a multi-view system. Hence, different description methods and corresponding representations have been researched, with edge or gradient based features such as SIFT [Low04], SURF [BETVG08] or HOG [DT05] receiving significant attention as an alternative or complement to colour based descriptors [AVBK10]. In this context, local descriptors

have been increasingly used. These are commonly computed at *key points* or *interest points*¹ of an image, which are salient patches that contain rich local information about the image [JYNH09]. The first step in the computation of these descriptors is the identification of the key or interest points. These are selected at distinctive locations in the image, such T-junctions or corners. High repeatability of the key points is desirable since it expresses the reliability of a detector for finding the same physical key points under different viewing conditions. The neighbourhood of the key point is then converted into a more compact and stable (invariant) descriptor that should be robust to noise, detection displacements and geometric and photometric deformations [Sze10]. Once the descriptors have been determined they can be matched between different images by measuring the distance between the description vectors. It has been recognized that the dimension of the descriptor has a direct impact on the computation time [BETVG08]. Although descriptors of lower dimensions are desirable for fast interest point matching, they are in general less distinctive than their high-dimensional counterparts.

Szeliski [Sze10] classified the approaches to identify interest points and establish correspondences into two classes: those that identify features in one image that can be accurately tracked using a local search technique (e.g., correlation); those that detect features in all the images under consideration and then match them based on their local appearance. The first class is more appropriate when using images taken in rapid succession (e.g., video sequences) or from nearby viewpoints, and the second when a large amount of motion or appearance change is expected, e.g., in wide baseline stereo or performing object recognition.

Gevrekci et al. [GG09] proposed a method to identify interest points robust to illumination changes, which they named Illumination Robust Feature Extraction Transform (IRFET). The proposed method determines the interest points in an image by analyzing the response of an interest point detector to a set of contrast stretching functions. The IRFET method can be used to improve interest point detectors, but at the expense of a higher computational weight.

SIFT (Scale Invariant Feature Transform) proposed by Lowe [Low04] is one of the most reference methods to determine local descriptors that are invariant to changes in scale, rotation, translation. The SIFT detector generates a high number of interest points compared to other interest point detectors. However, the high dimensionality of SIFT descriptors (vectors with 128 elements) has significant impact in the matching step. Ke and Sukthankar [KS04] applied PCA on the gradient image around the detected interest point to decrease the dimension of the vector. Although this vector is fast for matching, Mikolajczyk and Schmid [MS05] proved that the PCA slows down the feature computation. Gradient location-orientation histogram (GLOH) is another variant of SIFT using log-polar binning structure instead of four quadrants. In the study by Mikolajczyk and Schmid [MS05], GLOH slightly outperformed SIFT, but the use of PCA had an impact on the computational weight [KS04].

Tang and Tao [TT05] described the use of SIFT descriptors in a model represented as a graph. It was stated that the use of the graph increased the robustness to occlusion. Efficiency of the

¹Throughout this thesis the terms ‘key points’ and ‘interest point’ will be used interchangeably.

algorithm in the matching stage tended to decrease, especially when the number of objects or the complexity of model (number of key points) increased.

The Speeded-Up Robust Features (SURF) were proposed by Bay et al. [BETVG08] and described as a fast scale and rotation invariant interest point detector and descriptor. It shares many similarities with SIFT and has computational performance gains due to the approach followed on the detection of interest points and on the matching process. The dimensionality of these descriptor is smaller than SIFT and the detector typically outputs a smaller number of interest points.

Dalal et al. [DT05] proposed a new human detector based on a grid of Histograms of Oriented Gradient (HOG). The authors experimentally demonstrated that these descriptors outperformed state-of-the-art feature sets for human detection. Unlike other features such as SIFT descriptors, these are computed on a dense grid of uniformly spaced cells and it has been shown that HOG is insensitive to colour variation. The HOG human detector required the pedestrian to be in the upright position, with its performance deteriorating if they presented different poses. Moreover, pedestrian sufficiently far away from the camera may not be detected. Due to their high dimension (the descriptor vector has 3780 elements), the computation of the HOG descriptors requires a large amount of memory storage [JHM⁺10]. A disadvantage of HOG is its sensitivity to orientation [HYJ11]. In [Bas08] it was proposed a new feature detection and matching technique for for Augmented Reality and Computer Vision which they called Fast Invariant to Rotation and Scale feature Transform (FIRST). In these contexts, the FIRST algorithm is reported to be faster than most traditional approaches, such as SIFT, although yielding less distinctiveness in terms of rotation and scale. This common trade-off between quality and performance is also reflected in the number of key points extracted by FIRST, which is particularly smaller when compared with SIFT.

A popular key point-based representation is named bag-of-visual-words (BoW) [SZ03]. It consist of the grouping of similar key points descriptors into a large number of clusters; each cluster is treated as a visual word which is used to build a visual vocabulary. Several factors that influence the use of the BoW technique were described in the study by Jiang et al. [JYNH09]. In [NJT06], different sampling strategies were analysed to compare sparse (using key points) and dense (grid-base local image patches) representations. The authors state that the sample size plays a critical role in building the vocabulary. Hence, the random sampling of image patches can offer a more powerful representation than the use of sparse key points. Nister and Stewenius [NS06] proposed a tree vocabulary for object matching/recognition, having as base SIFT descriptors. The authors also described advantages and disadvantages of this method. A similar approach using a dense representations has been proposed by [TCR09].

The above mentioned techniques have shortcomings. Specifically, it has been recognized that these descriptors and representation techniques disregard shape or spatial information, which has been identified as an important feature to increase the object identification rate [OPZ06, TC02]. Wu et al. [WHL⁺09] also highlighted the shortcoming of disregarding spatial information in the use of BoW, which decreased their discriminative capabilities for objects with similar word frequencies, but different word spatial distributions. They proposed a modeling language com-

binning local appearance features and spatial context. Sivic et al. [SRE⁺05] augmented the visual vocabulary with “doublet” which are pairs of visual words co-occurring within a local spatial neighborhood. Jiang et al. [JHM⁺10] used the HOG people detector to initialize the bounding box (BB) of pedestrians and colour histograms to describe each object. Rather than extracting a single histogram, the BB was divided into a lower and upper part. A colour histogram, in the *Lab* colour space, was computed for each part, thus incorporating spatial information. During the tracking process, different weights were used for the upper and lower parts based on the argument that the lower part is more likely to become occluded. The authors reported that the HOG human detector failed to detect pedestrians when they walk to a position sufficiently far away from the camera.

Shahed et al. [SHY10] proposed to track an object with deformable shape by approximating the shape with small number of rectangles with a minimum overlap, with colour information extracted for each box. The authors demonstrated that the object could be tracked by adjusting the locations of the blocks. However, this approach required a stationary foreground appearance. In many applications, such as surveillance, this requirement can easily be violated.

3.6 Recent and Future Trends

Video analytics has been receiving increasing attention and is pointed as an important future research topic. It has been generally defined in [RCW⁺10] as the “*autonomous understanding of events occurring in a scene monitored by multiple video cameras*”.

Saligrama et al. [SKJ10] used a pixel-based analysis to detect anomalies or abnormal behaviors in video, avoiding high-level processing that involves the explicit tracking of objects. An anomaly can be defined as “*observations that do not conform to a well-defined notion of normal behavior*” [CBK09]. The underlying idea of the framework proposed in [SKJ10] is the detection of relevant activities and their locations before a higher-level analysis. Rather than focusing on object tracking and classification, the framework aims to model behavior and contextual attributes as they occur at different locations and time instants. It was stated that such approaches can be beneficial in highly cluttered scenarios. The framework was based on the notion of an anomaly as the presence or absence of usual or unusual objects at an unusual or usual location and time; it required background subtracted video sequences and assumed meaning motion labels can be computed, failing otherwise. Moreover, it required abnormalities to have a signature distinct from the background. The algorithm had restrictions with regards to the camera positioning and in multi-camera scenarios the occurrence of events needed to be uncorrelated.

Crowd analysis is of interest to many different areas with applications to visual surveillance, crowd management or intelligent environments, among others. Most of the work for detecting and identifying people, individually or in group have focused on non crowded situations [JSRM10]. In the survey on crowd analysis by Zhan et al. [ZMR⁺08], it was stated that conventional computer vision methods are not appropriate to analyse very crowded video sequences. In these scenarios, the number of severe occlusion increases significantly, augmenting the difficulty to segment and

track each individual object (or people). In such high-density video sequences, the precision of traditional methods for object tracking varies inversely to the density of people. Object-based approaches consider a crowd as a collection of individuals, while a “holistic” approach considers it as a single entity, without the need of segmenting each individual. Nevertheless, knowledge of individual trajectories can be used in the identification of abnormal behaviors and main flows of a crowd. Holistic approaches may be adequate for dense scenes, but usually the detected abnormality cannot be characterized with the same level of detail as in object-based methods.

Junior et al. [JSRM10] discussed the possibility of cooperation between computer graphics and computer vision applied to crowd analysis. In particular, computer graphics can contribute to the generation of reference information for evaluation or training of the algorithms. Such approach is also useful for tracking in general to aid the generation of sequences with reference information.

Surveillance and monitoring frameworks composed of smart cameras and embedded systems have been recognized as important research lines for the future [ZWW10]. Particularly, the capability of auto-configuration of the network is seen with great interest. Furthermore, as the computing power available in cameras increases, the computational intelligence is expected to move to the network edge, i.e., individual camera nodes, thus calling for distributed solutions [SKJ10]. In this line of research, Distributed Smart Cameras (DSC), which were defined in [RW08a] as “*real-time distributed embedded systems that perform computer vision using multiple cameras*”, are an active topic. The concept of DSC implies the execution of algorithms at the cameras instead of in a central workstation, but especially their operation in a distributed manner. The use of DSCs increases the complexity of the algorithms and the implementation of a surveillance infrastructure. Nevertheless, they are expected to reduce the impact on the communication networks and augment the system’s robustness and flexibility [RW08b].

3.7 Tracking Evaluation

Less work has been published on metrics and frameworks for the assessment of tracking algorithms than on the algorithms themselves. Currently, many researchers define their own test sequences and evaluation methodologies, making it extremely difficult to replicate results and compare algorithms. Proposals for evaluation frameworks and metrics already exist, but without a commonly accepted methodology.

In the current context of video object tracking (VOT) algorithm assessment, one can divide evaluation strategies into two major groups: those that use GT and those that do not, each with their strengths and weaknesses. The first type generates more precise results, but is strongly dependent on scarce information, whilst the second type is more easily applied, but produces noisier outputs.

The manual creation of ground truth data is typically a time consuming and tedious process, and is prone to errors. Ellis [Ell02] has described possible types of reference data as well as the difficulties associated with the generation of such information. Black et al. [BER03] also identified problems associated with tracking assessment and the generation of the required ground truth, and suggested the use of pseudo-synthetic video for the evaluation. Tracking data was

captured online, stored in a database and later used to generate reference video sequences with a controlled level of complexity. With this framework it was possible to generate a large variety of datasets representing different tracking scenarios with varied perceptual complexity. The problem was the bias towards the tracking algorithm used to capture the original data. In [SBWB04] semi and full-synthetic sequences were also used to overcome the problem of generating large amounts of GT. Moreover, the authors separated evaluation of motion segmentation and tracking. Other semi-automatic tools [DM00, JWSX02] have been proposed to help overcome the difficulty in GT generation. In [JSRM10] it was suggested the cooperation with the computer graphics area, in the context of crowd analysis, to facilitate the generation of reference information. An open source tool for GT generation - ViPER-GT - has been provided as part of the ViPER framework [ViP05].

Regarding the video surveillance application area, significant steps have been recently given with the creation of the Video Surveillance Online Repository (ViSOR) [VC10], intended to aggregate video sequences and ground truth data to be used by the research community. An ontology has been defined for annotation of the video sequences and modules for automatic annotation and evaluation can be integrated. Krinidis et al. [KST⁺05] have presented an audiovisual database intended for the assessment of people tracking algorithms, which includes GT in the form of 3D position in time. In [KK10], Karasulu and Korukoglu described a software tool for comparing people detection and tracking algorithms, which aggregates and implements metrics proposed in the literature.

Correia and Pereira [CP03] described a set of metrics for the objective evaluation of video segmentation quality. The proposed metrics targeted standalone (without GT) and relative (with GT) scenarios and are to be applied differently depending on the type of content class: stable content; moving content. As expected, the error in the evaluation is larger for the standalone metrics, which were said to be more adequate for a qualitative evaluation (e.g., ranking) rather than a quantitative one. In [GE09], a perceptually driven metric for the evaluation of video segmentation was proposed. The authors used sequences with synthetic artifacts to conduct psychophysical experiments and define a metric capable of predicting the quality perceived by the human viewers. Parameter optimization was conducted for a small set of application scenarios.

Erdem et al. [cEEST04] targeted video object segmentation and tracking evaluation without reference information (a non ground truth (NGT) approach) by performing intra and inter-frame evaluation. Intra-frame evaluation was accomplished by measuring colour differences along the boundary of an object while the inter-frame assessment consists of comparing an object's colour histogram in subsequent frames and measuring motion vector differences along the object's boundary.

Pingali and Segen [PS96] evaluated tracking algorithms by comparing the computed and reference trajectories. In [dNM06] is described a framework for evaluating object detection algorithms using GT information. Correspondences between detected and reference regions are established by measuring the overlap between them. A matrix of correspondences was defined based on the overlaps and used to measure correct detections, false detections, merges and splits and detection failures.

Recent research work by Bashir et al. [BP06] and McManus et al. [LMRMJ08] aimed to define a set of measures and frameworks for the evaluation of motion detection and tracking algorithms; it was proposed the use of a set of parameters such as the rate of false alarms and miss detection, and accuracy. In [BP06], bounding boxes were used as GT and frame-based and object-based paradigms for evaluation were defined. In [LMRMJ08], the F-measure was proposed to assess the performance of motion detection algorithm in the precision-recall space.

Unlike most tracking evaluation methods which are based on criteria derived from the processing of the images, Roth et al. [RKMR⁺08] proposed the detection of events as a basis for the evaluation. The process was based on a comparison with a ground truth consisting of a list of events.

In [NBTV07] the results of the ETISEO project are presented, which was oriented to the evaluation of video surveillance systems. It defined a methodology intended to address the related problems separately and study dependencies between the algorithms and video characteristics. Video sequences were collected for the purpose of illustrating specific problems. Denman et al. [DFSL09] proposed a set of metrics intended to dynamically assess tracking systems at various stages and enable real-time feedback about the system's performance. The proposed metrics were assessed by visual comparison and by using the ETISEO metrics.

Most proposals in the literature used bounding boxes as GT to compute a set of measures encompassing the entire sequence. Although some analysis about split/merge and fragmentation errors is sometimes made, typically there is no information about the temporal evolution of the error, causing a loss of temporal resolution and making it more difficult to identify failure points or events in the sequences. Moreover, as the dimension of the bounding box is typically greater than the enclosing object, usually there is also a loss of spatial resolution.

Chapter 4

Partition-Distance Methods for Assessing Video Object Tracking Algorithms

*“Do not worry about your difficulties in Mathematics.
I can assure you mine are still greater.”*

Albert Einstein

4.1 Introduction

Image segmentation is the first important process in innumerable applications, with subsequent operations relying heavily on its performance. The segmentation process partitions the image into different meaningful regions with homogeneous characteristics, using discontinuities or similarities of image components. The quality of obtained results is still improving and the segmentation algorithms are becoming more and more general. A major limitation in the design of image segmentation algorithms lies in the difficulty to measure their performance.

As with many other research areas involving image processing, many tracking algorithms rely heavily on the results of an image segmentation preprocessing step. Consequently, errors in the segmentation may be propagated to tracking causing failures as alluded in Chapter 2. Moreover, a common objective can be identified in the image segmentation and tracking underlying concepts: both aim at defining relevant regions in an image with an identifying label typically assigned. However, while in image segmentation labels are permutable between frames, in tracking the correct transportation of the labels to the next frame is desirable to achieve temporal consistency. Similarly to what happens in the image segmentation area, the evaluation of the performance

of tracking algorithms is also a difficult task with questions still without a consensual response. Which measures are more appropriate to characterize the performance of tracking algorithms? How can the ground truth for large video datasets be easily generated? How can the complexity of a dataset and quality of the associated ground truth be measured?

The following sections describe the extension of an existing framework for the evaluation of image segmentations [CCR05] to the evaluation of video segmentation and its applicability to tracking performance assessment [CCTCR09]. The framework compares results with reference information and is based on the principle of eliminating the minimum number of elements such that a specified condition is met. This rule translates directly into a global optimization procedure and the intersection-graph between two partitions emerges as the natural tool to solve it. To aid the reading of this chapter, Appendix A briefly describes the framework of partition-distance metrics developed by Cardoso and Corte-Real, and introduces some concepts and notations. Before continuing it is recommended the reading of Appendix A. However, for a more detailed understanding of the base framework, the reader should refer to [CCR05].

4.2 Extending the Partition-Distance Metrics

Video is essentially a series of two-dimensional images that are sequentially ordered in time. Considering this, the first intuitive approach to the evaluation of spatial segmentations of video would be to consider a frame isolated from the others and compare it with the corresponding reference segmentation. This analysis of the temporal evolution of the dissimilarity provides already valuable information regarding the performance of the segmentation algorithm and was successfully applied in [TCCR07] to compare algorithms that separate the scene into foreground and background regions. Nonetheless, there is more to spatial segmentation of video, and consequently to the tracking of objects, than a mere sequence of independent segmentations.

The output of a segmentation algorithm can be simply seen as a label assignment to each pixel of the image. For the following the labels were assumed to be non-semantic and permutable, and no assumptions were made about the underlying assignment procedure. Attend to Figure 4.1; if one evaluates the quality of the spatial segmentation at time t , ignoring the past frames, one gets a low distance value ($= 1$), as the segmentation under evaluation is almost identical to the reference segmentation. Still, when working with spatial segmentation of video, an additional characteristic plays a key role in the quality of the result: the correct tracking of regions in time.

Although in image segmentation labels are permutable inside a frame, in video segmentation one generally wants to assess the correct transportation of them to the next frame, thus preserving the identity of the region/object. The same region should have the same label in consecutive frames, whichever the label. When evaluating a frame independently from the others, we are in fact making unwarranted assumptions. More than assuming that labels are permutable, we are assuming that they are permutable in each frame independently from the other frames. Yet, if one wants to effectively assess if regions are being correctly tracked in time, one may only assume that labels are permutable in the set of all frames simultaneously.

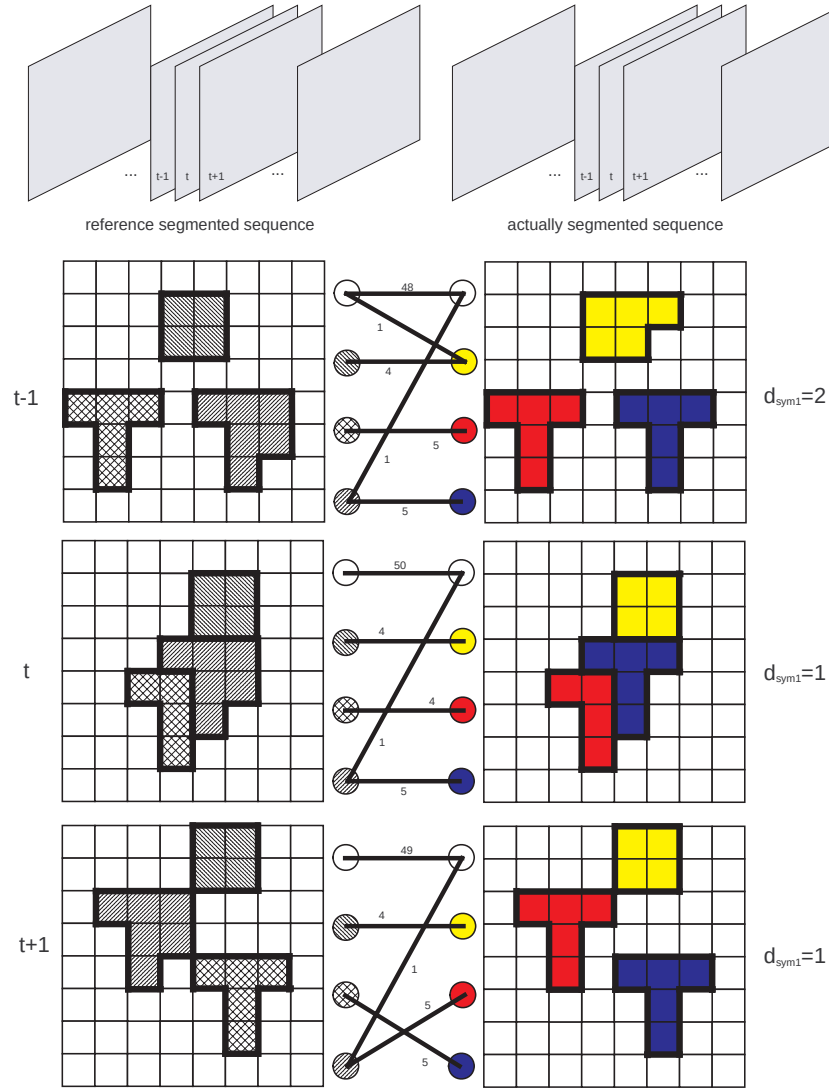


Figure 4.1: Video segmentation evaluation - setting corresponding to analysing a frame independently from the others (from [CCTCR09]).

At this moment, one could consider the set of all segmentations simultaneously (if each frame has N pixels, the set would have $N \times T$ elements, for a sequence of T frames), and compare the reference sequence with the sequence of segmentations under evaluation. Any swap of labels in consecutive frames would increase the distance between the two sequences. However, this would probably be a too drastic summary of the whole sequence of segmentations. By condensing the dissimilarity between two sequences of segmentations in a single real number, one loses any resolution on the temporal evolution of the quality of the sequence, as illustrated in Figure 4.2.

Therefore, a more balanced approach is to, at each time instant t , compare the set of the current and previous segmentations on the two sequences—illustrated in Figure 4.3. Any failure to correctly track labels on consecutive frames is penalized accordingly within this setting. The analysis of the time evolution of this distance provides information on the instants when the algorithm

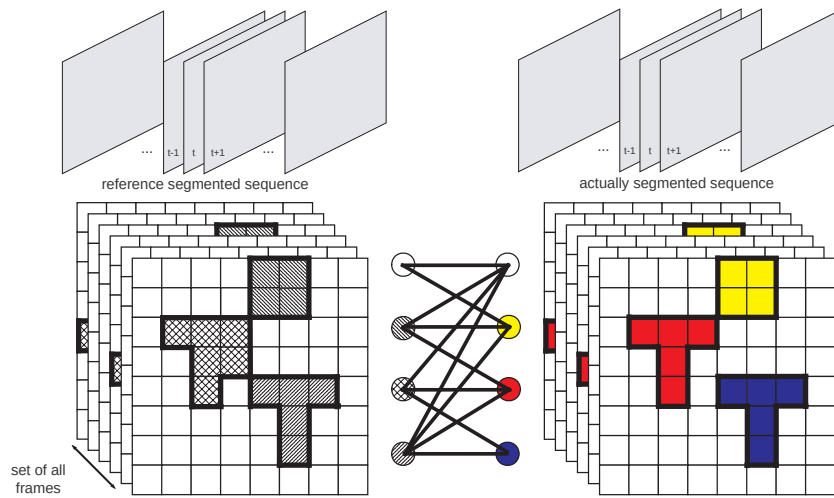


Figure 4.2: Video segmentation evaluation - setting corresponding to analysing the whole sequence at once (from [CCTCR09]).

faced difficulties.

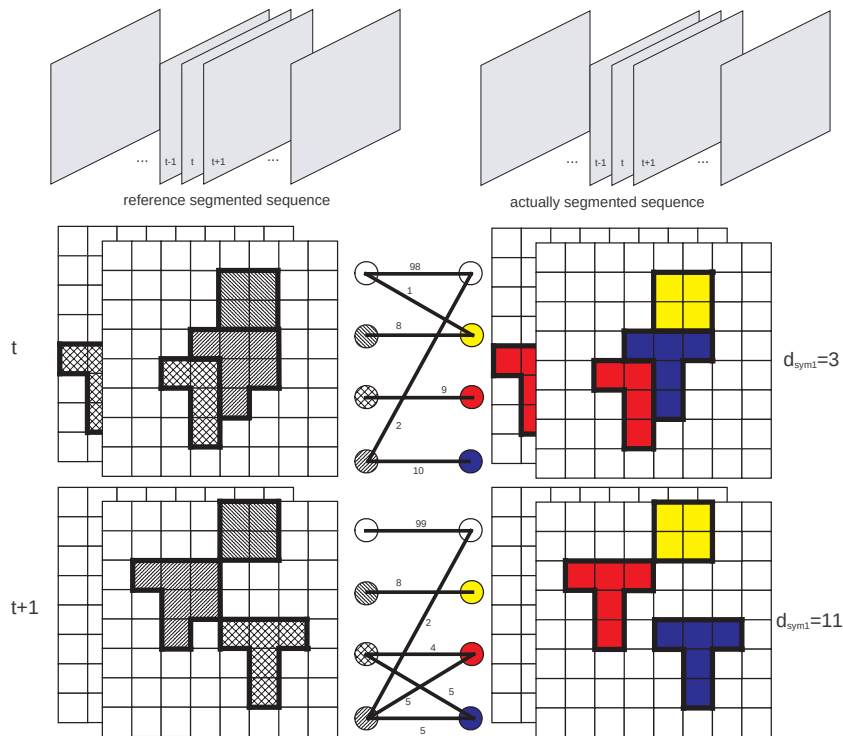


Figure 4.3: Video segmentation evaluation - setting corresponding to analysing pairs of consecutive frames (from [CCTCR09]).

In the sequence depicted in Figure 4.3, the identity of the two T-shaped objects was confused from time instant t to time $t + 1$. When matching the reference set formed by the elements of the

two reference segmentations on time instants t and $t + 1$ with the actually segmented set formed by the two actual segmentations, the object confusion is captured on the intersection-graph and reflected on a high d_{sym_1} ($= 11$) — see Appendix A. Note that in the intersection-graph at time instant $t + 1$, the edges represent the conjunction of the regions' intersections of the frames at time instants t and $t + 1$.

4.3 Experimentation and Results

In order to study the behaviour of the extension of the framework to spatial video segmentations, we first generated some synthetic segmentation results, in 200x200 pixel images, corresponding to different types of degradations of a ground truth we created. The used ground truth is composed of four components: 3 objects moving in different directions and the background. Figure 4.4 illustrates three frames of the sequence.

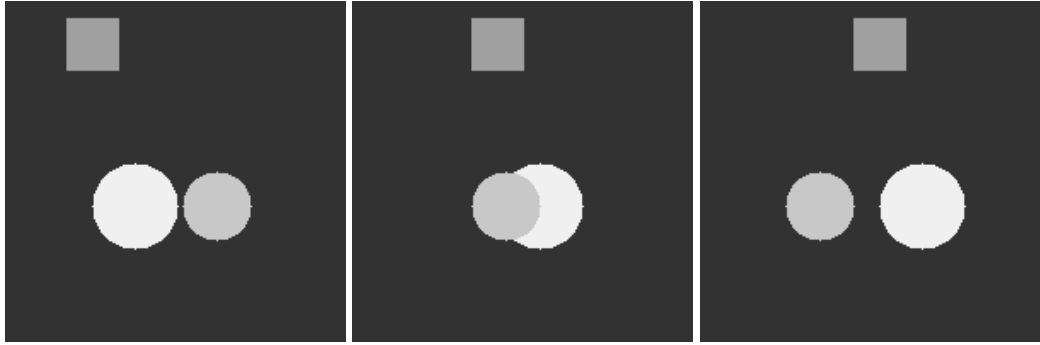


Figure 4.4: Illustrative frames of the synthetic sequence used in the assessment of the framework (from [CCTCR09]).

Similarly to [JMIB08], different kinds and amounts of artifacts were added to the reference:

- Boundary localisation error: obtained by randomly selecting a point p and finding the point q nearest to p which does not belong to the same region as p . Then, q is switched to the region of p provided that this step will not produce additional regions. This basic operation is repeated for $n\%$ (noise level) of all points.
- Under-segmentation: one or several objects of the ground truth are missing.
- Object fusion: two regions are fused from one frame to the next. This type of error typically occurs when two objects cross each other.
- Tracking errors: the identity of two objects is confused from one frame to the next. This type of error typically occurs when two objects cross each other.

Figure 4.5 presents the evolution of the proposed metric in face of different perturbations. The error measure consists of the normalised symmetric distance (*SymDist*) as described in Section A.3.

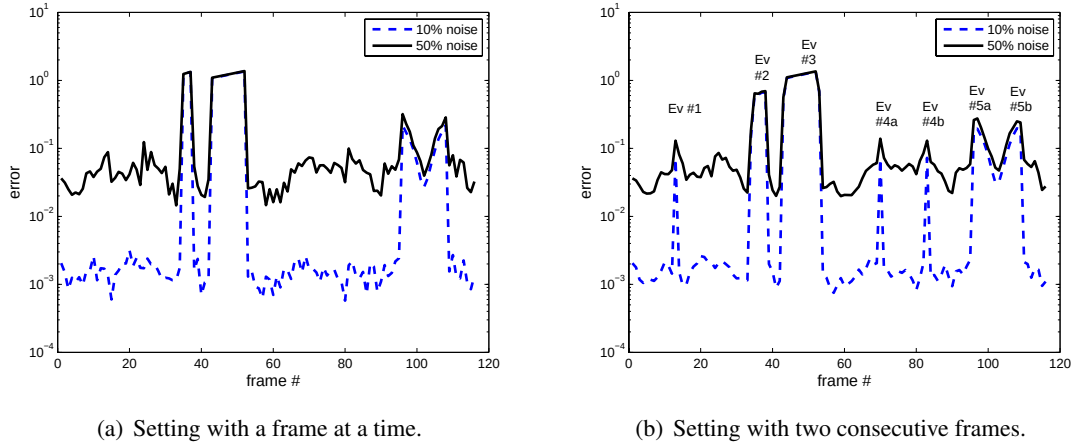


Figure 4.5: Time evolution of the proposed video metric for a synthetic sequence. The core metric is the normalised $d_{sym_1}^c$, with linear costs. In the graphic are marked some of the key perturbations introduced in the sequence (from [CCTCR09]).

We compare the simplest setting of using a frame at a time with the setting of using two consecutive frames. Two levels of localization error were gauged, 10% and 50%. We have marked on the plot the instants when major perturbations were simulated:

- Event #1 corresponds to label confusion on a single sphere, motivated by partial occlusion. As expected, this perturbation is captured only by the setting working with two consecutive frames.
- Events #2 and #3 corresponds to simulated under-segmentation, by missing to detect one of the objects. This perturbation is captured identically by both settings.
- Event #4 simulates label confusion, by mixing the identity of both spheres. Note this event is captured only by the second setting. On this setting only the beginning and ending of the label confusion are signaled; in-between, and because the ‘memory’ on this setting is only a single frame, the confusion is not signaled.
- On event #5 the two spheres are fused and segmented as a single object as they cross each other. It is interesting to note that, although the simulated error is captured through the whole duration, it has a valley-shape. This effect is due to increasing occlusion of the two spheres; in the middle of the event, one of the spheres is completely behind the other, therefore decreasing the error area on the image.

It is clear that major perturbations produce noticeable events on the error measure, well above the ‘continuous’ error due to simulated uncertainty on boundaries’ localization. Even with 50% noise all events are clearly detected. At this noise level, the objects have decreased their size and therefore the simulated events result in smaller impulses than at 10%. It is also suggested that the simplest setting captures all perturbations other than the tracking error. As reasoned before, this

is the expected behavior, and it is when the setting working with consecutive frames exhibit better performance, motivating its use.

In a second similar experiment we modified the ideally segmented reference of two real sequences, with the introduction of the type of perturbations previously defined. The first sequence, called shopping (SH) shows a view of a shopping corridor and is one of the test case scenarios made publicly available by the EC Funded CAVIAR project/IST 2001 37540. The scene consists of people walking, browsing the stores' displays or waiting for others. The second sequence, labelled outdoor (OD) shows an outdoor scene with several people passing along the camera field of view and is available from the MPEG-7 test set (results are presented for stream A). Figure 4.6 presents some illustrative frames.

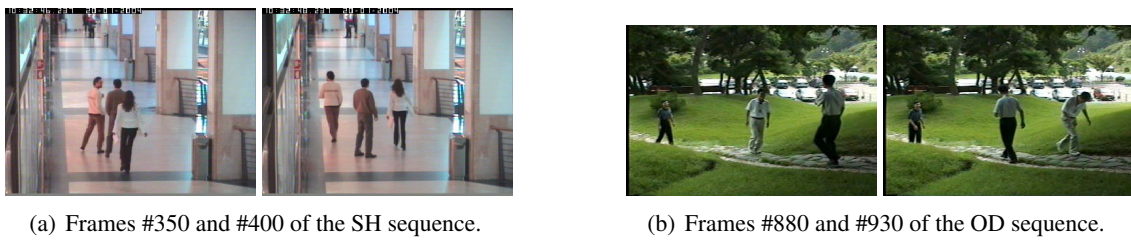


Figure 4.6: Representative frames of the SH and OD sequences.

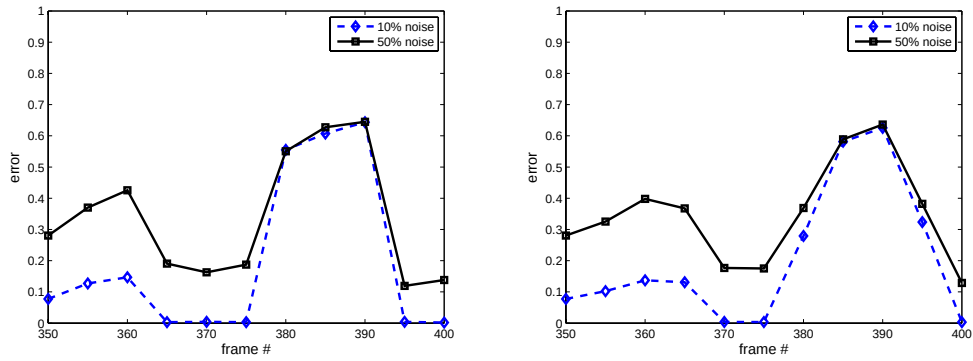
On the SH sequence two major perturbations were superimposed on the localization error. At the beginning of the sequence, object fusion was simulated, with two persons walking together segmented as a single object. At the end of the sequence, under-segmentation was assessed by failing to detect a motionless person. As observed on Figure 4.7(a), both events are clearly detected even by the simplest setting of processing a frame at a time.

For the OD sequence, besides the localization error simulated on all frames, a single tracking error was introduced, simulating the confusion of two objects; the confusion continues after the initial swap. The analysis of Figure 4.7(b) exposes the expected limitation of the setting measuring a frame at a time to detect such type of events. Ignoring the semantic meaning of labels on each frame, the segmentation is actually correct. However, when working with two consecutive frames simultaneously, the event is effectively captured as an impulse of the error. This object confusion is discernible even when in the presence of strong localization error.

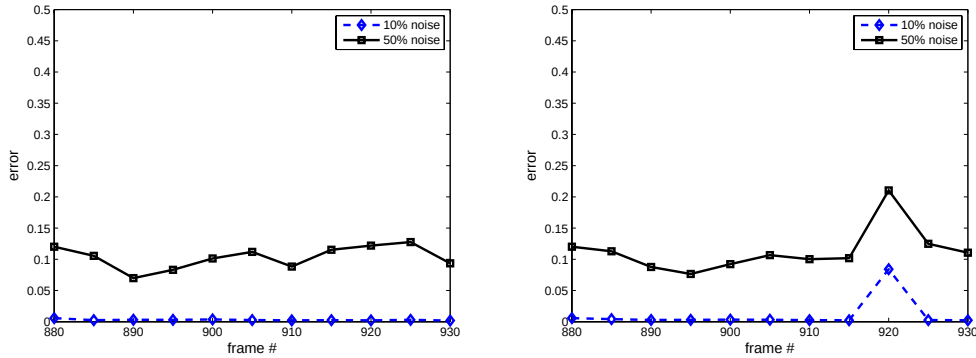
Note that, as the 'memory' on this setting is only a single frame, and although the confusion prevails (on a semantic level), it is not signaled after the initial swap.

Finally, two tracking systems were compared using the proposed metric. The first step in a tracking pipeline aims to obtain the possible location of relevant visual objects. The object segmentation algorithm integrated in the two tracking methods under comparison is based on the cascaded detection of common types of changes [TCCR07]. The subsequent operations differentiate the two tracking systems.

The first chosen tracking method is the one presented by Zhao and Nevatia [ZN04] which is based on kernel tracking, approximating the human shape by a primitive geometric figure (ellipse)



(a) SH results for a frame at a time (left) and two consecutive frames (right).



(b) OD results for a frame at a time (left) and two consecutive frames (right).

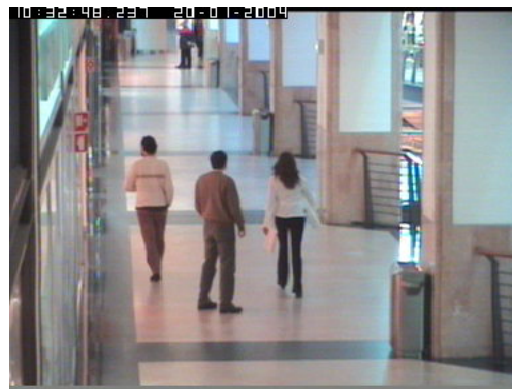
Figure 4.7: Time evolution of the proposed video metric for the SH and OD sequences. The core metric is the $d_{sym_1}^c$, with linear costs (from [CCTCR09]).

that is used for segmenting multiple people moving together or in occlusion situations and in the object matching process. A Kalman filter is used for position estimation. The algorithm aims at real time operation. The second tracking method is based on a hybrid strategy, using both object and region information to solve the correspondence problem [CSE05]. Low-level descriptors are exploited to track object's regions and to cope with track management issues. Appearance and disappearance of objects, splitting and partial occlusions are resolved through interactions between regions and objects. In Figure 4.8 and Figure 4.9 it is shown a frame from the SH and OD sequences respectively and the corresponding results from the two algorithms under consideration.

The results provided in Figure 4.10 confirm the expected superiority of the more complex tracking solution. The kernel method presents an inherent drawback resulting from the approximation of the human shape by an ellipse which may fail in fully detecting parts of the object, in particular the limbs (see Figure 4.8(b)). Both methods face difficulties with the SH sequence, particularly in the first frames, where objects are moving together with partial occlusion. At frame 360 the kernel method outperforms the hybrid one due to the strategy adopted in the identification of interest objects from the segmentation masks. While the kernel method uses a geometric shape as a model, the hybrid one aggregates regions based on their proximity and similarity. As a result, in the latter, when objects move close to each other, confusions such as the ones illustrated in Figure 4.8(c) occur. For the OD sequence, the partition-distance results consistently attribute better quality to the hybrid method. For this difference of performance contributed the inability of the kernel-based method to correctly track the three persons present in the sequence. Such behavior was already expected as the used kernel-based method was not designed for the position of the camera adopted in this sequence and there was poor camera calibration information. The results for these two sequences were found consistent with the subjective evaluation that a human observer would make by direct visualization of the segmentation partitions.

4.4 Discussion

The image segmentation evaluation framework developed by Cardoso [CCR05] was naturally extended to assess spatial segmentations of video. The use of consecutive pairs of frames allowed the detection of most of the typical errors introduced by any automatic segmenter of video. Nevertheless, the framework can be further extended for accommodating needs of specific domains. Labels were assumed non-semantic. However, the value of the label itself may be important, as one may be interested in evaluating spatial segmenters of video that keep track of the meaningful labels. Such an evaluation system could be easily constructed on top of the proposed framework, by keeping track of any label swap to update the correspondence between ground truth labels and automatically derived labels. Another possible extension is the assessment of the temporal stability of errors. An error may vary its characteristics through time. In some applications, a non smooth change of any spatial error deteriorates the quality of the segmentation itself. The temporal artifact caused by a variation of the spatial error may be called jitter or flickering. This perturbation may be assessed by comparing the evolution of the reference sequence with actually segmented



(a) SH sequence frame.



(b) Result from the kernel algorithm.



(c) Result from the hybrid algorithm.

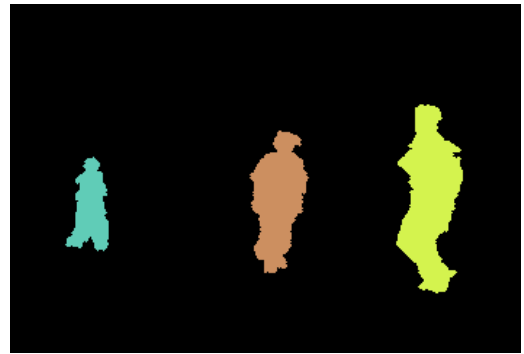
Figure 4.8: Example of a frame from the SH sequence and the corresponding results (track labeled mask) from the algorithms being used.



(a) OD sequence frame.

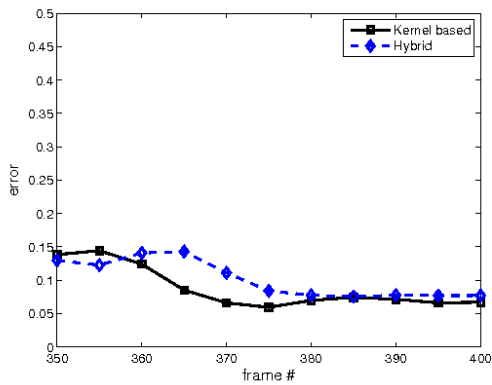


(b) Result from the kernel algorithm.

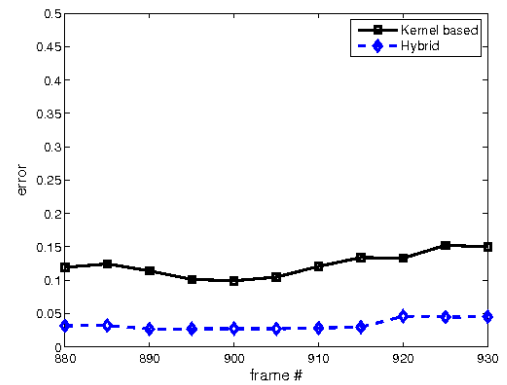


(c) Result from the hybrid algorithm.

Figure 4.9: Example of a frame from the OD sequence and the corresponding results (track label mask) from the algorithms being used.



(a) Results for the SH sequence.



(b) Results for the OD sequence.

Figure 4.10: Time evolution of the proposed video metric for results for two existing tracking methods. The core metric is the $d_{sym_1}^c$. (from [CCTCR09])

sequence. The ‘error’ difference between consecutive segmentations of the actually segmented sequence should evolve similarly to the ‘error’ difference between consecutive segmentations of the reference.

Most frameworks for measuring the performance of tracking algorithms calculate an overall measure for the complete sequence. The framework proposed for assessing the spatial segmentation and tracking performance operates at the pixel level outputting a set of metrics. However, unlike other frameworks, the calculation of the partition-distance metrics for consecutive pairs of frames also enables the observation of the temporal evolution of the error. With such information it is possible to detect when errors have occurred and consequently what caused the algorithm to fail.

Another aspect that must be taken into account in the application of the described framework, consists in the output format of the tracking algorithm. Algorithms found in the literature use different output formats (e.g., bounding box, ellipse, centroid, silhouette, etc.). Hence, it may be necessary to adapt the ground truth information to the output of the algorithm before evaluating it, or define a way to compare algorithms with different output formats. Consider the following: if the ground truth is prepared for silhouette comparison and the tracking algorithm outputs an ellipse, region outliers with regards to the ellipse (e.g., an arm) will contribute to an increase in the error, even though the overall region has been correctly identified and tracked. Nevertheless, this issue is not straightforward, as the inability to detect the actual shape may be considered, in some scenarios, a drawback of the algorithm and consequently should be penalised.

An obstacle to the easy application of this framework may consist of the requirement of ground truth for video segmentation and its nonexistence for most of the available sequences due to the effort required in its preparation. While in image segmentations a small set of ground truth images may be sufficient, for a video sequence the number will be significantly larger, resulting in a greater effort in the generation of the ground truth. Some strategies to overcome this obstacle include the generation of synthetic [VD10] or pseudo-synthetic [BER03] sequences. However, these tend to underestimate the complexity and problems of real scenarios, or be bias towards the tracking algorithm used for acquisition.

The work conducted on the extension of the partition-distance metrics to the assessment of video segmentation and tracking resulted in the following publication:

J. S. Cardoso, P. Carvalho, L. F. Teixeira and L. Corte-Real, *Partition-distance methods for assessing spatial segmentations of images and videos*, Computer Vision and Image Understanding, no. 7, vol. 113, pp. 811-823, 2009.

Chapter 5

Filling the Gap in Quality Assessment of Video Object Tracking

“The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them.”

William Lawrence Bragg

5.1 Introduction

Video object tracking (VOT), and in particular human motion capture and tracking, can be the objective of a system or a pre-processing step to achieve higher-level functionalities. Not minimizing the importance of the many components that make up a VOT system, the performance of the tracking algorithm will have a great influence on all posterior processing. Also, the specificities of the scenario may exert a strong conditioning on the choice of the algorithm. These factors, in conjunction with the large number of tracking algorithms proposed in the literature (see Chapter 3), tackling different application scenarios or scopes, augmented the need and importance of objectively comparing algorithms, characterizing their behavior and suitability for a given operational scenario.

Currently, many researchers define their own test sequences and evaluation methodologies, making it extremely difficult to replicate results and compare algorithms. Furthermore, the criteria used in the evaluation of the algorithms should be appropriate to the application scenario. This results in the use of different features (e.g., as objects' trajectory, silhouette or assigned identifier) and application of different metrics (e.g., trajectory root mean square error, detected and reference region overlap, identity consistency). Proposals for evaluation frameworks and metrics already exist, but they haven't been generally adopted by the research community. Some of these approaches

focus on evaluation without comparing with a reference (commonly known as ground truth (GT)), but the results provided typically lack sufficient discriminative information. Consequently, evaluations based on comparisons with GT are commonly favored. In this case, the problem lies in the difficulty to generate such information; it is a cumbersome job, especially when detailed pixel-based references (reference silhouettes) are needed.

The gap between the two types of approaches to the VOT assessment problem described above is mainly due to the required information. The use of reference information is desirable because it enables more precise results, but it is not reasonable to expect the existence of such information for every frame to be used for testing or evaluation purposes. We propose to bridge these approaches, complementing and unifying existing metrics, based on the concept of making the very best use of the information to our disposal. The research line pursued combines information from GT-based and standalone (without GT) metrics with the objective of obtaining an error assessment of the tracking algorithm that is as precise as possible, but with a significant decrease of the effort associated with the generation of reference information. The proposed approach aims to combine the flexibility of metrics without GT with the greater precision of metrics that compare algorithms' results with reference information. This will provide more flexible frameworks and means to deal with a wider range of input information. Such frameworks can be used over different video datasets, making use of the GT available. To the best of our knowledge no such an approach to combine information and metrics to obtain a measure for video object tracking evaluation has ever been attempted. The reference information and metrics used, and their proposed combination, are intended for tracking algorithms whose results can be mapped to an image region. Examples of such outputs are bounding boxes or labeled object silhouettes (also referred to as reference silhouettes) such as the ones illustrated in Figure 5.1; three different types of reference information are presented: on Figure 5.1(a) silhouette reference segmentation; on Figure 5.1(b) bounding box (BB) GT, shown represented over the image for illustrative purposes; on Figure 5.1(c) bounding box GT in the form of a segmentation mask.

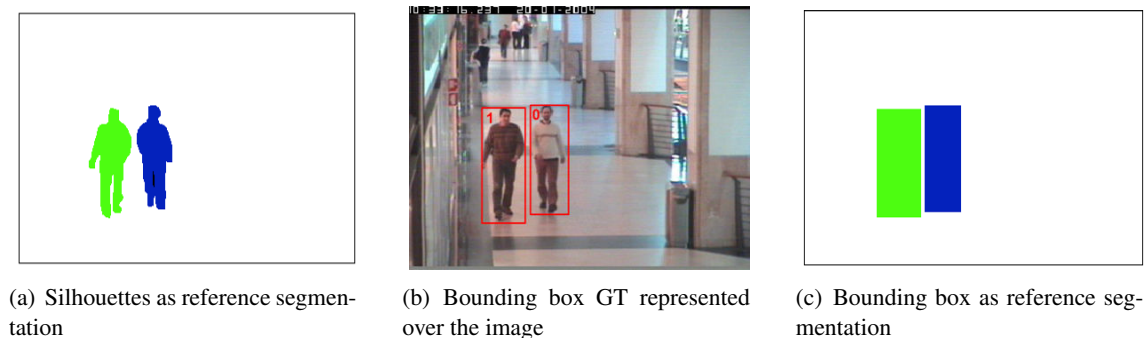


Figure 5.1: Illustration of some of the possible types of reference information.

In Section 5.2 it is described the use of different types of GT in the computation of tracking error metrics, specifically those proposed in the partition-distance framework (see Chapter 4), to

minimize the amount of reference silhouettes required. Section 5.3 introduces the concept of combining metrics that require GT and metrics that do not make use of such information. In Section 5.4 it is described the experiments conducted, including the different forms tested for combining information, and the analysis of the correlation between GT and NGT metrics. The results of the experiments are summarized in Section 5.5. A discussion on this topic is presented in Section 5.6.

5.2 A Ground Truth Hybrid Approach

As previously stated, most evaluation approaches compare the results of tracking algorithms with some form of reference information. Two widely used types of GT are reference silhouettes (RS) and bounding boxes [CZ05]. Although pixel-based reference silhouettes can provide more exact and complete information (other types of GT can be derived from it), they are very cumbersome to generate. To minimize this effort, the use of bounding boxes to locate and provide a coarse dimension of the objects has been the preferred strategy. As with image segmentation, the generation of GT is a process with a degree of subjectivity and different people may produce different reference information for a set of frames or even the full sequence.

In Chapter 4 it was described a novel framework for evaluating video segmentation and tracking algorithms that uses reference silhouettes as ground truth and partition-distance (PD) metrics [CCTCR09]. It was demonstrated that the proposed framework was capable of computing an overall error measure as well as exhibit its temporal evolution over the sequence. However, the requirement for RS implies an immense effort in the generation of this type of information. Hence, it is desirable to enable the computation of the PD metrics while minimizing the problem of generating a large number of frames with exact pixel-based GT. In this context it would be possible to use these metrics for a detailed analysis of the algorithm's behavior in challenging situations (e.g., group movement or occlusion), but with significantly less effort in the data preparation.

The approach proposed in this Chapter is based on the outcome and capabilities exhibited by the partition-distance (PD) metrics as described in Chapter 4 for the evaluation of video segmentation and tracking - the metrics of this framework using only reference silhouettes will be taken as the gold standard for the evaluation. Note that PD metrics can be used with both silhouette and bounding box segmentation GT.

The GT-based hybrid approach was intended to enable the computation of the PD metrics defined in [CCTCR09] while minimizing the problem of generating a large number of frames with exact pixel-based GT. The objective was achieved through the combination of different types of GT and their use in the computation of PD metrics (see Figure 5.2). Specifically, RS and BB, in the form of image masks, were considered.

The use of bounding boxes in the computation of PD metrics enables the capture of the most common types of problems in tracking. However, the obtained error is coarser due to the very nature of this type of GT (see Figure 5.1). To solve this, frames with reference silhouettes, when available, can be used to correct the previous error. Such a combination builds on the following:

GT information in the form of a bounding box is easier to generate and there are datasets available with this type of information; the availability of reference silhouettes is of added value particularly in challenging situations such as occlusions or group movement; reference silhouette information does not need to exist for the complete sequence and therefore can focus on the most relevant segments of it; there can be different frame intervals between consecutive frames with reference silhouettes.

First, the detected and reference bounding boxes are used as segmentation masks to compute the PD metrics. This error measure (referred to as bounding box error - BBE) is typically higher than what would be obtained through the use of reference silhouettes (reference silhouette error - RSE); also, the use of bounding boxes can mask errors only observable with silhouettes. Frames with reference silhouettes and the corresponding error (RSE) can be used to correct the BBE in these excerpts of the sequence. An example of such a combination is illustrated in Figure 5.3 and consists of the following: for two consecutive reference silhouette frames and the corresponding RSE (represented by the filled circles), a transformation (in this example a linear transformation) from BBE to RSE is computed; the transformation is then applied to the values of BBE in the interval being considered. The circles represent the RSE values: filled circles for the values used in the combination; empty circles for the values used for assessment. Note that the linear transformation is one of the possible forms of combining the information and is provided here as an example.

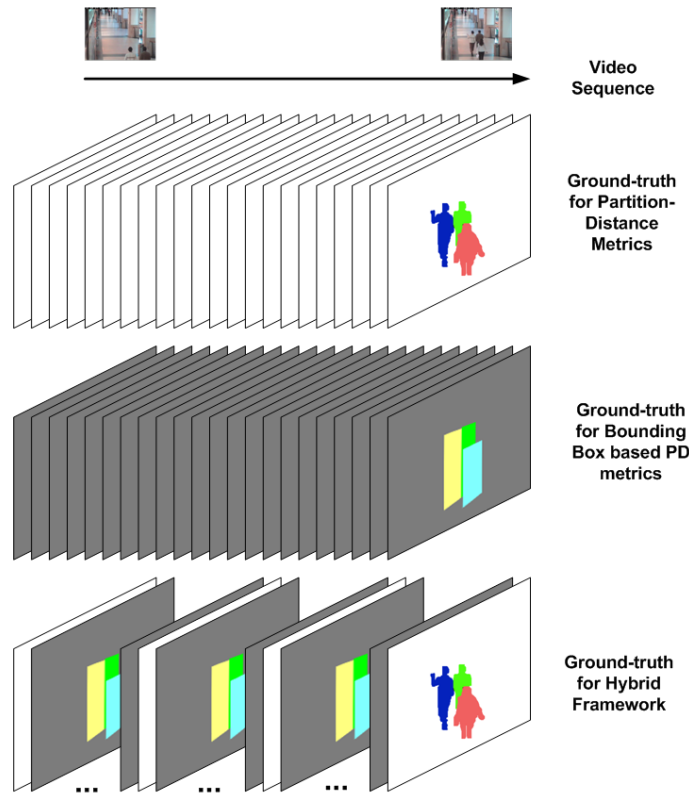


Figure 5.2: Ground Truth combination over a video sequence for the GT-hybrid evaluation proposal (from [CCCR10]).

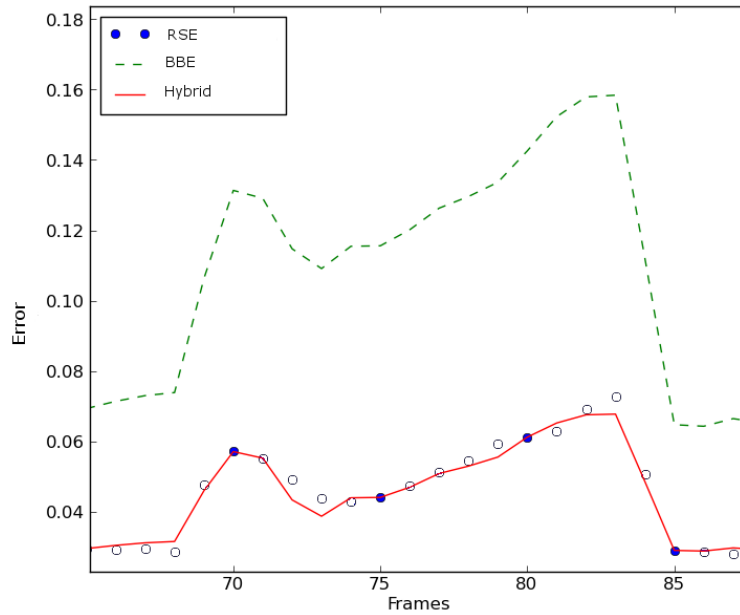


Figure 5.3: Excerpt of a graphic depicting evaluation results and illustrating the error transformation effect in the GT-hybrid approach (from [CCCR10]).

Through the experiments conducted it is empirically demonstrated that the tracking error using a coarser type of GT, with a spatial error associated, can be approximated to the ideal error using a smaller number of frames containing pixel-wise silhouette GT, thus leading to a significant decrease of the effort associated with the generation of this type of information. Although corrected, information of the error evolution conveyed by the BBE is still preserved, as can be observed in Figure 5.3.

5.3 A Hybrid Approach Combining Ground Truth and Non Ground Truth Metrics

The use of different types of GT in the computation of the tracking error enables the minimization of the effort involved in the generation of detailed reference silhouettes. Nevertheless, it still shares to some degree the shortcoming of the original framework: it requires GT information of some form. We argue that metrics without GT (Non Ground Truth (NGT)) can be used in conjunction with the previously defined approach to obtain an overall error measure while avoiding the need of reference information for every frame.

We assess whether the overall error can be approximated with a significant reduction of every type of GT and when a considerable number of frames have no corresponding reference information. The overall error is expected to provide a less exact value than in the previous approach.

However, we argue that even if the final measure has a slightly higher distance to the ideal error, the flexibility and less effort required can justify it.

With the proposed combination we aim to enable scenarios where test and validation datasets do not require full reference information. Instead, GT (of different types) can be generated only for critical or more relevant segments of the video sequence and used in the computation of the PD metrics. Information from the NGT metrics, referred to as NGT error (NGTE), is used to fill the gaps between the intervals of frames with GT (see Figure 5.4). As in the previous case, available GT is used to correct the errors associated with the NGT metrics. The process for this hybrid approach is similar to the GT-hybrid one, but now consists of two steps: first NGTE is combined with BBE; second, the result of the first operation is fused with RSE. It is out of the scope of this thesis to determine the very best form of combining the information for a given scenario or the distribution of the GT over a test/evaluation sequence.

In our experiments we have chosen the metrics proposed in [cEEST04] as they are well documented and the authors validated the results with GT information.

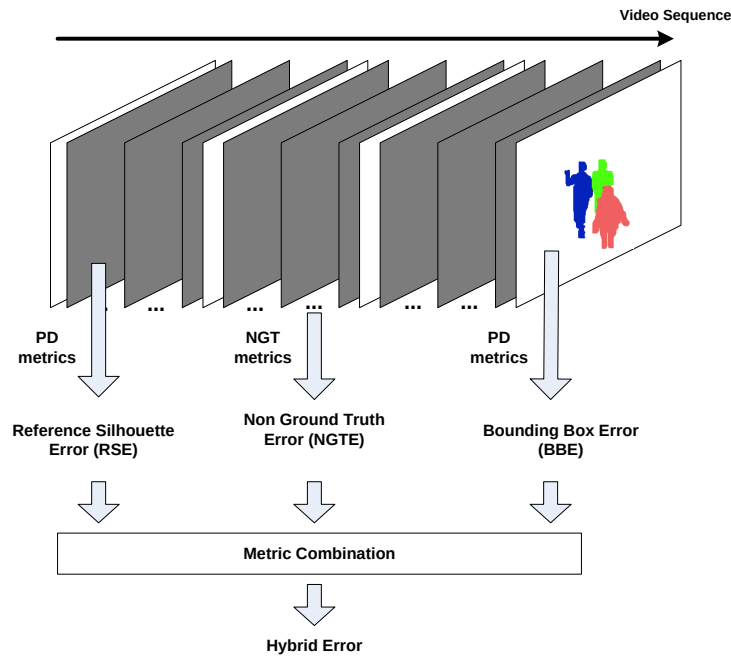


Figure 5.4: Illustration of the concept of using different types of reference information and metrics

5.4 Experiments

To verify the validity of our hypothesis and assess both the GT and metric hybrid frameworks, it was followed an experimental approach similar to the one described in Section 4.3.

5.4.1 Dataset

Synthetic sequences with over 2000 frames were generated, depicting several illustrative challenges for tracking algorithms. Group movements, occlusions, miss and over detections, similarity between objects and with the background, tracks fusion and switch were simulated. Moreover, different levels of noise for the boundary localisation error were used as proposed by Jiang et al. [JMIB08]: these errors were obtained by randomly selecting a point p and finding the point q nearest to p which does not belong to the same region as p ; then, q is switched to the region of p provided that this step will not produce additional regions; this basic operation is repeated for $n\%$ (noise level) of all points. The purpose of the boundary localisation error is to simulate common segmentation errors and make the detection of tracking errors harder; still it is only one of the type of errors simulated. To further increase the difficulty of these sequences, a real background was used. For object simulation, an elliptic form was chosen since this shape has been recognized as a good approximation of the form of a human in the upright position [YJS06, ZN04]. The elliptic shape varies according to its position and image perspective.

Although there are datasets with standard test sequences such as the PETS [PET06] and Caviar [Cav04] datasets, few possess bounding box ground truth and fewer still reference silhouettes, thus motivating the use of synthetic sequences. These enable the automatic generation of different types of GT. Specifically reference segmentations and bounding boxes were produced. A subset of reference segmentations was manually generated for two well known real sequences of the Caviar project [Cav04] which were also used in the assessment of the proposed approach. Illustrative frames of the dataset are depicted in Figure 5.5.



Figure 5.5: Sample frames of a real and synthetic sequence of the dataset used for validation.

5.4.2 GT and NGT-Based Metrics Correlation

Due to the differences between the GT and NGT-based metrics, an analysis of their correlation was conducted. The NGT-based metrics proposed by Erdem et al. [cEEST04] extract three measures

which are combined to produce the overall measure for each object in each frame. The individual components are intended to measure: color differences; motion difference; histogram difference. The first is measured along the border of the object while the others involve measures over two or more consecutive frames. For each object two possible combinations are proposed, a weighted average and a fuzzy combination. Nevertheless, the authors do not close this issue and other forms of combination can be pursued in future research. Concerning the frame measure, the authors suggest an average over the objects present or the use of the maximum error for an object, arguing this tends to capture most of the attention.

The correlation of individual components of the NGT metric and their combination, as suggested in [cEEST04], with the values obtained with the PD metrics using reference silhouettes - the gold standard - was analysed to assess the feasibility of combining the two types of metrics. The results of this study are presented in Section 5.5.

5.4.3 Metrics Combinations

For the combination of metrics, several possibilities were considered and compared. These are not exhaustive and others can be experimented with in future research work. Before proceeding to the description of the combinations tested, some underlying concepts are first presented.

For the following description and the remaining of the Chapter, the *primary error (PE)* is considered a more exact error and used to correct a noisier second measure, which will be referred to as *secondary error (SE)*; PE_i and SE_i are the errors at the i^{th} frame respectively. Moreover, M intervals are considered over the sequence. Each interval can be defined as $I_k = [a_k, b_k]$, $k = 1, \dots, M$. The extremities of the interval, a_k and b_k , represent two consecutive values of the PE . Note that the intervals do not need to cover the full sequence. These concepts are illustrated in Figure 5.6.

The following combinations of metrics were considered: linear transformation; linear interpolation; scaling factor; half-interval weighting; full interval weighting; normal weighting. These forms of combination are described with more detail below.

- **Linear Transformation (LT)**

For each interval, the values of the PE and SE at a_k and b_k are used to compute a linear transformation $f(x) = mx + b$; the overall error E value is obtained by applying the transformation to the values of SE .

$$E_i = f(SE_i) \quad , i \in [a_k, b_k] \quad (5.1)$$

The parameters m and b are determined by solving the solving the linear transformation $f(x)$ for the pairs (SE_{a_k}, PE_{a_k}) and (SE_{b_k}, PE_{b_k}) .

- **Linear Interpolation (LI)**

The error values E in each interval are determined through linear interpolation of the values

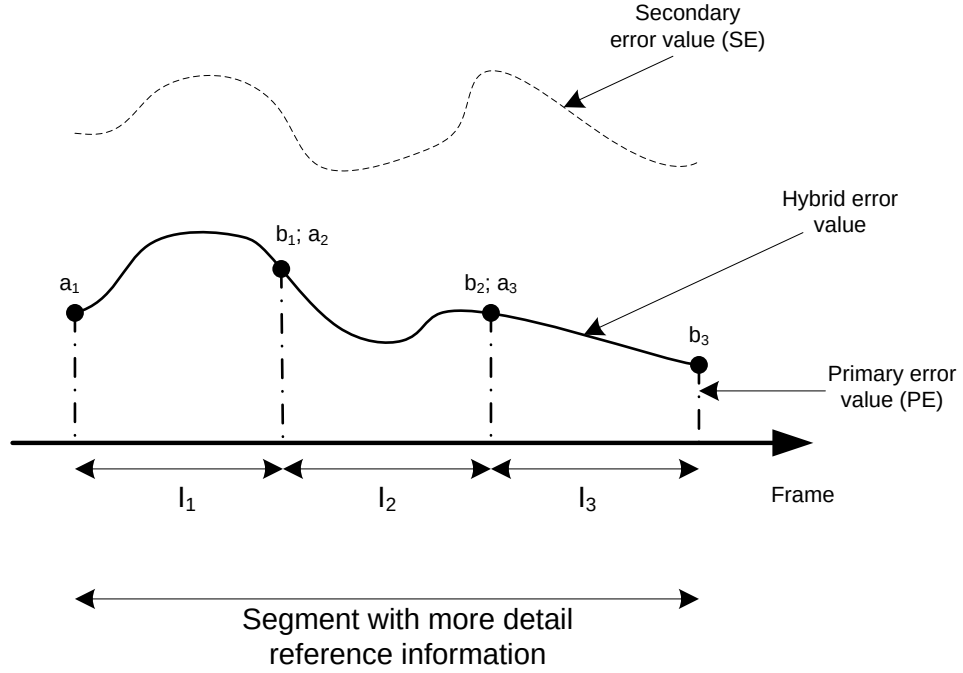


Figure 5.6: Illustration of the Primary Error (PE) and Secondary Error (SE) combination.

of PE at a_k and b_k . No SE values are used.

- **Multiplication Factor (MF)**

Known values of PE are used to compute the factor K that minimizes

$$\sum_{\gamma} (PE_{\gamma} - K \times SE_{\gamma})^2 \quad (5.2)$$

where γ represents frames for which PE exists. The error value in each interval is then obtained as

$$E_i = K \times SE_i, i \in [a_k, b_k] \quad (5.3)$$

- **Half-Interval Weighting (HIW)**

In each interval, the error value E is obtained as a weighted combination of SE and PE at the extremities.

$$E_i = \alpha_1 \times PE_{a_k} + \alpha_2 \times PE_{b_k} + \alpha_3 \times SE_i, i \in [a_k, b_k] \quad (5.4)$$

The values of α_1 , α_2 and α_3 are obtained as follows

$$\alpha_1 = \begin{cases} 1 - \frac{2}{D} \times d & , \quad d \leq \frac{D}{2} \\ 0 & , \quad d > \frac{D}{2} \end{cases} \quad (5.5)$$

$$\alpha_2 = \begin{cases} \frac{2}{D} \times d - 1 & , \quad d \geq \frac{D}{2} \\ 0 & , \quad d < \frac{D}{2} \end{cases} \quad (5.6)$$

$$\alpha_3 = 1 - \alpha_1 - \alpha_2 \quad (5.7)$$

where

$$D = b_k - a_k \quad (5.8)$$

$$d = i - a_k, i \in [a_k, b_k] \quad (5.9)$$

- **Full-Interval Weighting (FIW)**

In each interval, the error value is obtained through equation (5.4), but with the weights α_1 and α_2 computed as indicated bellow; the weights are normalized before being applied.

$$\alpha_1 = 1 - \frac{d}{D} \quad (5.10)$$

$$\alpha_2 = \frac{d}{D} \quad (5.11)$$

$$\alpha_3 = 1 - \frac{2}{D} \times \left| d - \frac{D}{2} \right| \quad (5.12)$$

The values of D and d are obtained according to equations (5.8) and (5.9) respectively. The final error value is given by equation (5.4).

- **Normal Weighting (NW)**

In this combination, the contribution of SE is weighted by a normal function centered at the middle of the interval I_k . The values of α_1 , α_2 and the final error value are computed as in the previous case.

In the experiments for the GT-hybrid approach, different interval widths were considered with each interval being delimited by consecutive frames with reference silhouette, and corresponding

PE. We define step as: $\Delta = b_k - a_k$. Specifically, steps ranging from 3 to 200 frames were used. Following the procedures described in [CCTCR09], the computed error was the symmetric distance (*SymDist*) of the PD metrics (see Appendix A).

For the combination of metrics using different types of GT, the error obtained through the use of reference silhouettes (*RSE*) is taken as *PE*, while the error obtained through the use of bounding boxes (*BBE*) assumes the role of *SE*. When combining GT and NGT-based metrics the process consists of two phases: first the error without GT (*NGTE*) is taken as *SE* and *BBE* as *PE*, which are fused through one of the previously described combinations; the result is an error measure SE' which is combined with the *RSE* (as *PE*) in the second phase, resulting in the final error value.

In the hybrid approach the interval widths described above were considered for the second phase, i.e., these intervals are delimited by consecutive frames with RS. Again, steps from 3 to 200 were considered for these intervals. For the combination of the *BBE* with the *NGTE*, subdivisions were performed over the interval. Specifically, divisions of the step by 2, 4, 8 and 16 were considered. When dividing the interval, a minimum width of 3 for the sub-intervals was guaranteed. For the real sequences only a division of the interval by 2 was considered due to a smaller number of frames with manually generated reference segmentation and to assure a considerable distance between consecutive frames with reference bounding boxes.

5.5 Results

This section summarizes the outcome of the experiments conducted. First, results from combining two different types of GT in the computation of the error with PD metrics are presented, conveying the information initially presented in [CCCR10] with more detail through new experiments. Next, the correlation analysis of PD and the chosen NGT metric are presented, followed by the results regarding the combination of the metrics.

For each experiment on GT and metrics combination, the root mean square error (RMSE) of each approach was computed, relative to the PD measure using only reference silhouettes - *RSE*.

5.5.1 PD Metrics with GT Combination

The graphics in Figure 5.7 depict the RMSE variation for the combination of metrics using reference bounding box segmentations and silhouettes in the computation of the PD metric considering different levels of noise in the boundary localisation error (in addition to the common tracking and detection simulated errors). For clarity, only the results for some of the combinations are shown. It was observed that the values for full and half interval weighting were very similar, with the same happening for the values for linear interpolation and normal weighting. Hence, we chose to only present the values for half interval and normal weighting (HIW and NW respectively). In the graphics of Figure 5.7 it can be observed that, up to a given interval size, the combination of the two types of GT produces better results than using only BBs, with the benefit becoming clearer with the increase of the boundary error noise level; this is mainly due to the increased error of using only BBs.

Correlation Factor With Regards to <i>PDSym</i>											
Sequence	Noise (%)	Objects Average					Objects Maximum				
		Color Diff.	Hist Diff.	Motion Diff.	Average	Fuzzy	Color Diff.	Hist Diff.	Motion Diff.	Average	Fuzzy
Synthetic	0	0.05	0.06	0.06	0.06	0.05	0.09	0.09	0.16	0.13	0.12
	10	0.07	-0.02	0.07	0.07	0.06	0.10	0.04	0.17	0.14	0.14
	50	0.32	-0.00	0.26	0.29	0.29	0.35	0.14	0.37	0.37	0.36
OneShopOneWait1	-	0.35	0.44	-0.45	0.79	0.96	0.02	0.45	-0.20	0.94	0.51
OneShopOneWait2	-	0.04	0.18	0.92	0.65	0.98	0.97	0.52	0.95	0.69	0.98

Table 5.1: Correlation between NGT-based metrics and PD metrics with reference segmentations (from [CCCR11]).

It can be observed that, although achieving better results in some of the experiments, the behavior of the linear transformation is more erratic. This is mainly caused by the magnification of the *SE* when the *PE* values delimiting the interval are very similar. The best results are obtained through the combination with the multiplication factor. This form of combination always outperforms the use of only BBs.

The results for the real sequences are depicted in Figure 5.8. As in the synthetic case, the combination of the GT information produces better results, with the exception of the linear transformation for the same reason stated above. For the real sequence, the combination MF is outperformed by other forms of fusion, which is not unexpected; when generating a synthetic sequence, the reference bounding boxes are derived from the corresponding silhouette. As the simulated form is an ellipse, there is a close relation between these two types of masks and their areas, thus favoring the combination through a multiplication factor.

5.5.2 GT and NGT-Based Metrics Correlation

As described, in the NGT metrics three measures are computed for each object and then combined to obtain an error measure. Two forms of combination were suggested by the authors: average and fuzzy. Two strategies for combining the error of the objects and obtain a frame level error are also proposed: average of the errors; maximum error. Table 5.1 summarises the correlation results through the computation of the normalised correlation coefficient of each individual component and their combination with regards to the value obtained with PD metrics using reference silhouettes. Moreover, the two frame-level combination strategies are also considered. In each frame strategy, the best results for each individual component and for the combination are highlighted.

From this analysis, one can observe that the correlations are weak for the synthetic case with better results being obtained for the real sequences. This is mainly due to the difficulty level associated with the synthetic sequences, specifically concerning similarity of movements and appearance of the objects which augment the error associated with the NGT measures. When analysing the results for the synthetic sequence, the best correlation at the frame level was obtained when the maximum error per object was considered; an expected behavior since in PD metrics errors in objects with higher dimensions tend to have a higher contribution. This is consistent with the argument that errors in larger objects tend to capture most of the attention [cEEST04]. For the real

sequences the correlation values are significantly higher which reveals that, even though the NGT measure is noisier than the PD metrics with reference silhouettes, their behaviours are closely related (i.e., they tend to capture the same errors).

The average of the maximum errors was taken for the metric fusion. Regarding the results of Table 5.1, note that only the suggestions of the authors regarding the combination of the components of the NGT metrics were implemented. Other forms of combination of measured components can be researched and may produce higher correlations.

5.5.3 Metrics Combination

This section summarises the results on GT and NGT-based metric combination over the dataset. From the several combinations tested, two have stand out: full interval and normal weighting (FIW and NW respectively). For clarity, only the RMSE, relative to the error obtained using PD metrics and only reference silhouettes, for the measurements resulting from these forms of combination are presented. Moreover, the RMSE values for BBE and for RSE with BBE interpolation (RS-BBI) are also presented for comparison purposes. The first uses bounding boxes as reference segmentations whilst the second results from the combination of the reference silhouette and bounding box associated errors (RSE and BBE respectively), with the values for frames between consecutive reference BBs given by linear interpolation. Thus, one can compare each fusion derived error with the: gold standard (RSE); use of only bounding boxes; combination of reference silhouettes and bounding boxes for a sparse existence of reference information.

Figure 5.9 depicts the results over a synthetic sequence. For clarity, we chose to represent only the results for a boundary localisation error of 50% as this corresponds to the worst case scenario. One can observe that, with the exception of high intervals (above 100 frames), the hybrid approach error is closer to the gold standard than by simply using BBs. Even though the use of RSE with BBE interpolation presents, in some instances, an error distance similar or smaller than the full combination, its behavior is more erratic. This is due to the non-detection of errors occurring between the consecutive reference BBs. Up to 100 frame intervals, the full combination of metrics enables a smaller and/or more stable error distance.

For the real sequences, the advantage of using NGT metric information is more obvious, as shown by the experimental results summarised in Figure 5.10. With one exception, for sequence OneShopOneWait2 and interval between consecutive RS frames of 50, the full fusion derived error is significantly better than the two measures being used for comparison purposes.

The better results obtained through the FIW and NW combinations are justified by the contribution of the tree components (the two measures at the extremities of the interval and the noisier measure in between) for the complete interval. In HIW the weight of the extremities' values decreases towards the middle of the interval where only the noisier measure is considered.

For a better understanding of the benefits of this approach consider the following: for the real sequence OneShopOneWait1, the error obtained with GT-hybrid and hybrid approaches is approximately the same (≈ 0.05), but with significantly less reference information; for a sequence

of 2000 frames with 50 frames between RS frames and interval sub-division by 2, the hybrid approach requires at least 93% less reference information than the GT-hybrid one.

5.6 Discussion

There is still no consensus regarding the objective evaluation of tracking. The existence of multiple tracking algorithms with different objectives and input/output information makes it harder to establish a common methodology.

It was proposed a novel approach for the evaluation of video object tracking algorithms. It aims to complement existing metrics and unify the use of reference information making the assessment of these type of algorithms more flexible. The described approach is based on the combination of reference information and metrics to provide a rich description of the algorithm's behavior. It is not intend to replace frameworks and metrics previously proposed. Rather, it intends to complement them by unifying the use of reference information and benefit from the use of different metrics to overcome individual weaknesses. This proposal is intended for the assessment of visual tracking algorithms with outputs that can be mapped to an image region such as bounding boxes or labeled silhouettes.

It was demonstrated that the combination of different types of ground truth can be used to decrease the effort in the generation of reference information while still approximating the error measure obtained using only exact reference segmentations - here used as gold standard. The composite measure still maintains information about the temporal evolution of the tracking error. Despite the valuable minimization of the effort in GT generation, this type of information is still required, in one form or another, for every frame in the sequence.

To overcome the shortcoming of the approach combining different types of GT the fusion of different types of GT was proposed as well as the combination of information of metrics without GT. Based on this concept, it was successfully demonstrated that it is possible to significantly decrease the GT required and eliminate its need for every frame. An error measure is obtained from the NGT metric and corrected by the available GT information. With such a solution the final error approximates, up to a given interval between consecutive frames with detailed reference silhouettes, the ideal error better than, for example, the solution using only bounding box references for every frame. This solution can contribute to an even more flexible video object tracking algorithm assessment framework.

The results from the fusion of GT and NGT metrics are expected to improve with the augmented correlation between them, resulting from different combinations of the individual components of the NGT metric, as suggested by the better results obtained for the real sequences.

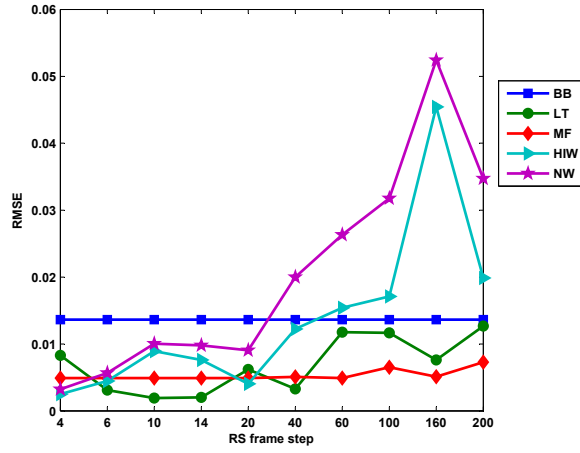
The experiments conducted were not exhaustive, covering every possible form of GT and metrics, since it was not the objective of this thesis to identify the very best combination for every scenario. Rather, the aim was to demonstrate that based on widely used types of GT and state-of-the-art metrics the ideal error can be approximated with sparse ground truth. As the number of frames with reference information, and in particular reference silhouettes, increases the error

measure can be made more accurate, but at the expense of greater effort; a trade-off must be decided according to a given assessment objective. As part of future work, it would be interesting to research, for a set of test/evaluation sequences, the impact of chosen start and end points of the intervals with reference information in order to avoid loss of information, minimizing the error uncertainty without a GT increase.

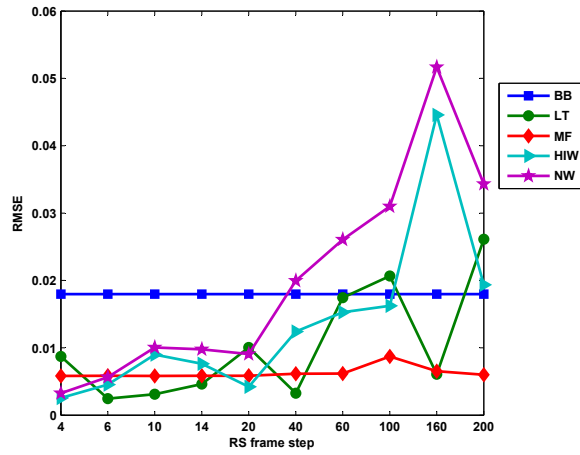
Other types of ground truth and metrics may be analyzed in future research work. Experiments can also be conducted to determine different forms of combination of the metrics and GT. Moreover, it would be interesting to research the impact of chosen start and end points of the intervals with reference information in order to avoid loss of information. Work on the hybrid evaluation framework resulted in the following publications:

P. Carvalho, J. S. Cardoso L. and Corte-Real, *Hybrid framework for evaluating video object tracking algorithms*, Electronics Letters, no. 6, vol. 46, pp. 411-412, 2010.

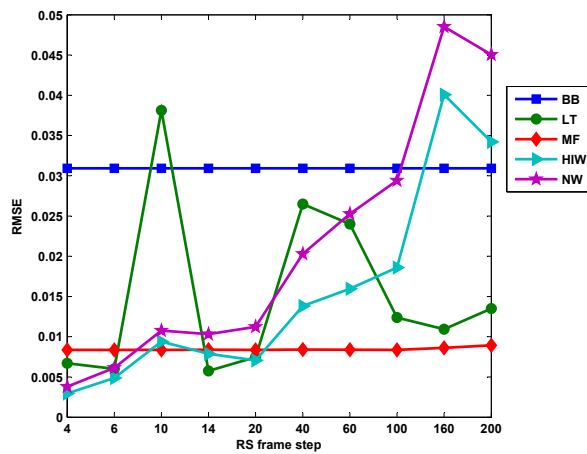
P. Carvalho, J. S. Cardoso L. and Corte-Real, *Filling The Gap in Quality Assessment of Video Object Tracking*, Image and Vision Computing. (submitted)



(a) Noise level = 0%

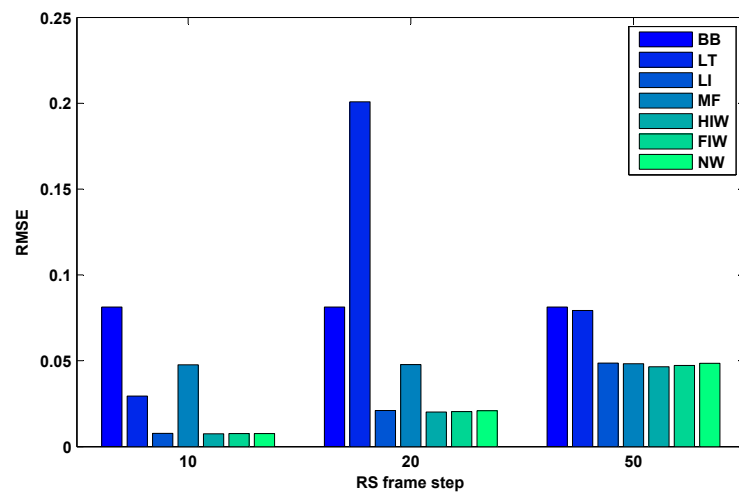


(b) Noise level = 10%

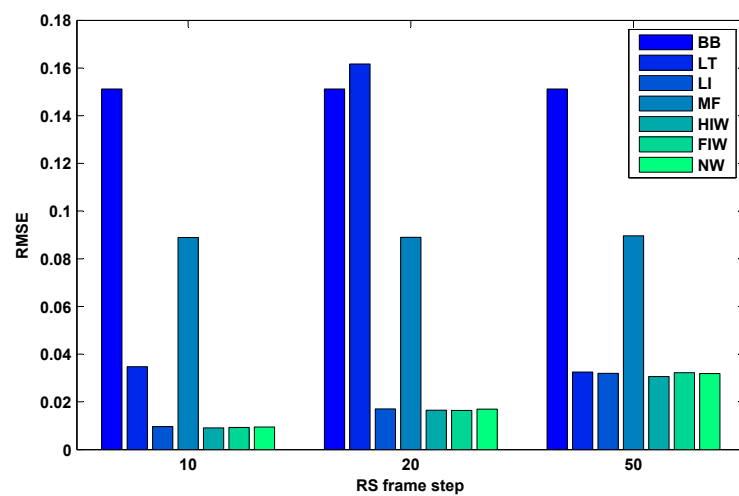


(c) Noise level = 50%

Figure 5.7: Error comparisons for the GT hybrid approach over a synthetic sequence, considering different levels of generated noise and distance between consecutive RS frames (from [CCCR11]).

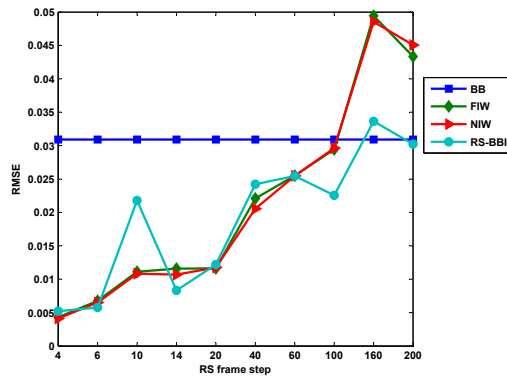


(a) Sequence OSOW1

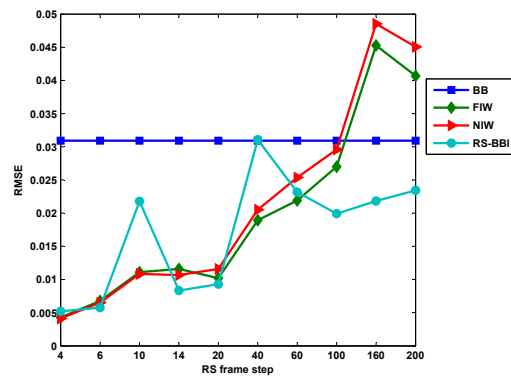


(b) Sequence OSOW2

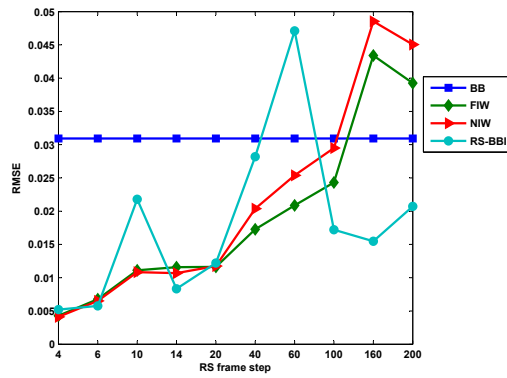
Figure 5.8: Error comparisons for the GT hybrid approach over sequence OSOW1 and OSOW2 considering different distances between consecutive RS frames (from [CCCR11]).



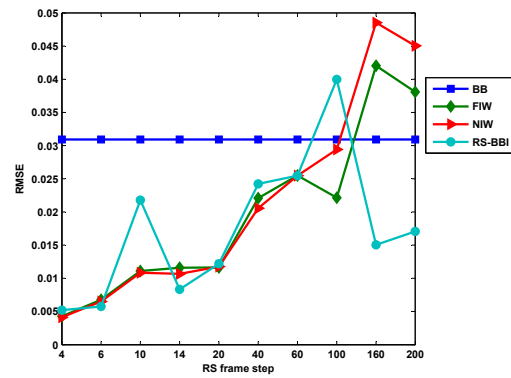
(a) Interval Subdivision - 2



(b) Interval Subdivision - 4

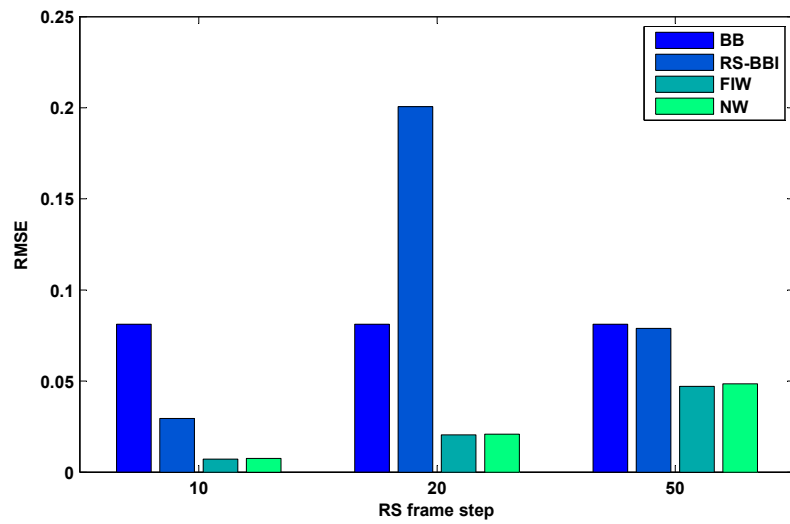


(c) Interval Subdivision - 8

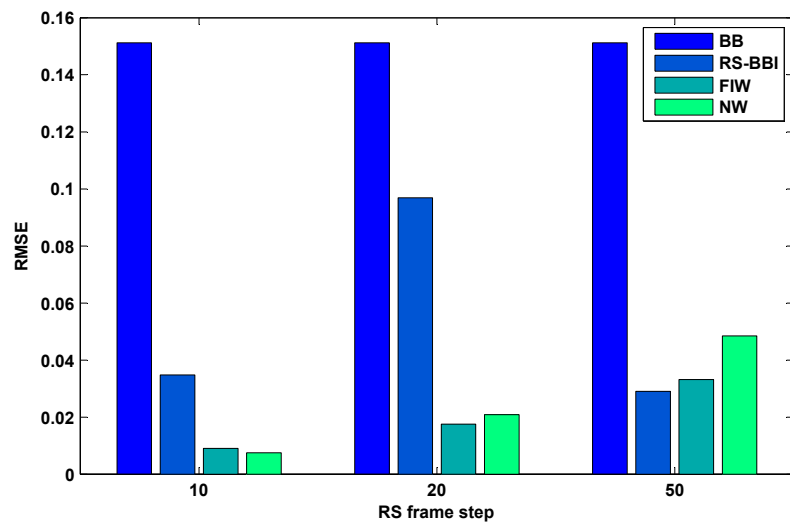


(d) Interval Subdivision - 16

Figure 5.9: Error comparisons for the hybrid approach over a synthetic sequence with 50% noise (from [CCCR11]).



(a) Sequence OneShopOneWait1



(b) Sequence OneShopOneWait2

Figure 5.10: Error comparisons for the hybrid approach over sequence OSOW1 and OSOW2 considering different distances between consecutive RS frames (from [CCCR11]).

Chapter 6

Analysis of Object Description Methods in a Video Object Tracking Environment

“Truth is what stands the test of experience.”

Albert Einstein

6.1 Introduction

A key issue in video object tracking is the representation of the objects and how effectively it discriminates between different objects and the background [HYJ11]. Several techniques have been proposed, each with its strengths and weaknesses, but without a generally accepted method. The robustness of some methods typically implies a higher computational cost that may prevent their use in application scenarios with real-time constraints; inversely, more lightweight methods may originate higher error rates, making them unsuitable for scenarios requiring little or no confusion of identities. The appearance model is a component of a video object tracking (VOT) framework; it receives and provides information from and to other modules. As a result, its performance will depend on previous processing stages and will affect those that succeed it. For example, by estimating the position of an object in the next frame it is possible to reduce the uncertainty that the appearance model must resolve; lower discriminative capabilities of the appearance model may cause track drift or even track loss.

The standalone comparison of descriptors has been the subject of several studies [MS04, BL02, TS04, SMB00]. Aware of the importance of these techniques to visual tracking we found necessary, and a logical evolution, to also perform this analysis from a tracking solution point of view.

This will enable a better understanding of the impact in the overall tracking. We proposed the comparison of a set of widely used description methods in a common object tracking solution. Furthermore, we performed the first integration and test of the Fast Invariant Robust Scale feature Transform (FIRST) descriptors [Bas08] in a people tracking algorithm; the goal was to assess its behavior, but also to provide feedback for improvement of, what is to our knowledge, the only existing FIRST implementation.

In Section 6.2 we describe an initial experiment to extend single camera tracking to a multi-camera scenario which partially motivated our work. In Section 6.3 we describe the set of experiments performed, including the description methods, dataset used and evaluation methodology. The results of the experiments are presented in Section 6.4 and final discussion is conducted in Section 6.5.

6.2 A Multi-Camera Tracking Experiment

An essential issue in multi-view tracking is the definition of correspondences involving the same object tracked by different cameras, either simultaneous or at different time instances. In [Tei09], it was proposed a system to perform multi-camera tracking using a state-of-the-art single camera algorithm and without any information about the deployment of the cameras. The single camera tracking algorithm was used to process the images captured by each camera; the information received from each individual tracker was processed and put in relation with all the others. To make possible a correct object matching, linking together different views of the same object while discriminating robustly views of different objects, a representative description of each object based on local descriptors was adopted. As output, the system would allow navigating through the different records of an object in all the system time tracks. The main modules composing the proposed system architecture are depicted in Figure. 6.1.

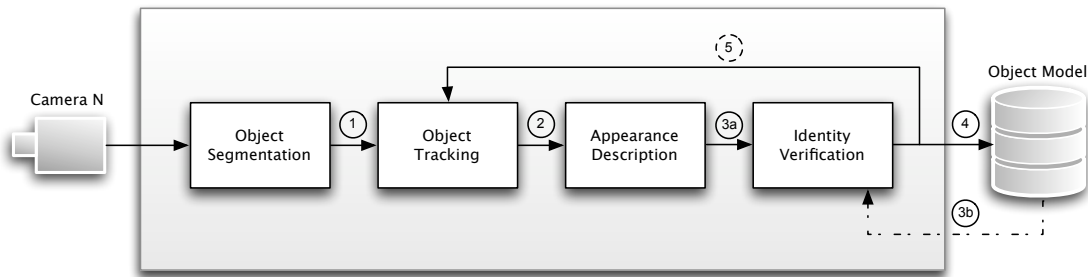


Figure 6.1: Block diagram of the architecture for a wide area surveillance system (from [Tei09]).

During the work of this thesis, there was a contribution to this experiment, particularly at the level of the single camera tracking and its integration. For the *Object Tracking* module we used an implementation of the tracking algorithm proposed by Zhao and Nevatia [ZN04]; it identified and tracked the people in a video sequence, outputting the position and dimensions of the bounding box that encompassed each object. This information was intended to spatially locate objects in

each frame and was then used by the *Appearance Description* module to extract descriptors for each object.

For each object frame comprising the full object track, a set of descriptors was extracted at specific points; these can be defined by an interest point detector (the number of points depends on the image content) or by a random point selection (the number of points is fixed and pre-defined). For object matching and classification, the number of descriptors extracted from the image is an important factor influencing performance [NJT06]. With the argument that objects' images are usually very small and interest point detectors tend to select an insufficient number of points to efficiently represent them, a random point selection was used. A 128-dimensional vector is extracted at each selected point using the SIFT descriptor developed by Lowe [Low04]; each descriptor was assigned a visual word, from a previously trained dictionary (bag-of-visual-words (BoW) [SZ03]), and a histogram representing the frequencies of occurring visual words was used as a signature to describe each object.

Using the additional knowledge provided by the appearance model that was constructed for each object, the system was able to identify and correct some misidentifications resulting from tracking; hence, it is expectable that it could be used to improve tracking reliability. Such possibility is represented in the architecture of Figure 6.1 by the feedback connection to the *Object Tracking* module. With such connection the appearance model could be used, for example, to recover a track, i.e., recognize an object as one already tracked when it reappears in the scene after exiting or occlusion. The use of two different appearance modules, for single and multi-camera tracking, prevented the full integration of the algorithm and development of the complete solution. A possible solution to this problem can be the use of a common appearance model.

6.3 Analysis of Object Description Methods

The descriptors and appearance representations analysed in this chapter were assessed through their integration in a common tracking solution. The testing environment consisted of an implementation of the tracking algorithm proposed by Zhao and Nevatia [ZN04]. The algorithm was intended for surveillance or human monitoring scenarios and was described as having: capability of real time operation; an acceptable detection and tracking rate in low to medium complexity scenarios. Furthermore, it has features that enable human detection and the separation of individuals in a group.

The test framework was prepared to accommodate the integration and test of different appearance models. Throughout these experiments only the extraction of the object appearance and the matching between two individual instances were model specific. All other stages of the algorithm were common. The appearance models were encompassed in modules with a common and generic API (application programming interface). Each module filtered and stored the required information and was responsible for performing all model specific information, namely model initialisation, comparison and update. Object matching was performed by searching for the most similar instance in an uncertainty region. In each iteration, the appearance model instance received

the images, shape and position information; it computed and compared the object descriptors. As a result, it outputted the corresponding dissimilarity measure that was used by the tracking algorithm to define the best match; once it was found, information was passed to the module for model update. Video segmentation was performed using the algorithm proposed in [TCCR07]. The experiments were performed in a computer with an Intel(R) Core(TM) i5CPU at 3.20GHz with 8GB of RAM.

6.3.1 Dataset

For the experiments we selected sequences of two widely used datasets. Specifically, we used the sequences: OneShopOneWait1 (OSOW1) and OneShopOneWait2 (OSOW2) from the Caviar project [Cav04]; ST1-C1/cam3 (ST1C3) and ST1-C1/cam4 (ST1C4) from the PETS 2006 [PET06] workshop. These sequences are representative of monitoring and surveillance scenarios depicting commonly observed problems: group movement; shape deformation; appearance similarity; occlusion; object crossing. Also, they were captured with dissimilar cameras and offer different perspectives over the scene. Figure 6.2 depicts illustrative frames of the sequences. The sequences of the CAVIAR project have half-resolution of the PAL (Phase Alternating Line) standard (384x288 pixels, 25 frames per second) and were compressed using MPEG-2; the resolution of the PETS sequences are PAL standard (768x576 pixels, 25 frames per second) and were compressed as JPEG image sequences (approx. 90% quality).

The sequences of the dataset have associated different types of ground truth (GT)). The Caviar project provided reference information, in the form of bounding boxes (BB), for every sequence of the dataset; this was provided in an XML file, following the CVML (Computer Vision Markup Language) syntax [LF04]. This BB reference information was also mapped to the corresponding image mask. Additionally, for sequences OSOW1 and OSOW2 reference segmentations were generated for a subset of the frames. Sequences ST1C3 and ST1C4 are not provided with reference information applicable to this context (provided GT consists of a list of events), thus reference segmentations were manually produced for a subset of frames.

6.3.2 Description Model Variations

For our experiments, we have selected the following state-of-the-art description methods: SIFT ¹; SURF ²; FIRST; HOG ³; Colour Histogram; BoW. Several variations for the appearance models were tested in order to emphasize particular contributions thereof to the performance of the tracking algorithm, e.g., precision and robustness of human detection, computational complexity of appearance extraction and matching, adequate discrimination of humans when occlusions occur among them.

For colour histogram, we used the proposal described in [JHM⁺10], where the bounding box is divided into a upper and lower part with a greater weight assigned to the upper part, thus adding

¹<http://www.cs.ubc.ca/~lowe/keypoints/>

²<http://www.chrisevansdev.com/computer-vision-opensurf.html>

³http://d.hatena.ne.jp/poor_code/20091111/1257944372



Figure 6.2: Illustrative frames of the dataset used in the experiments.

spatial information to the model. For both colour histogram and BoW-base models, the normalized χ^2 distance was used as dissimilarity measure.

For the key point based descriptors SIFT and SURF, the ensemble of descriptors was adopted as the model. The descriptors were computed in an object's image and in a possible instance of the object in the next frame, and a matching process was formulated. For two sets with N_1 and N_2 descriptors, and K matches, a dissimilarity measure was calculated. To this objective, different measures may be defined; we chose the formulation

$$D = \sum_{i=1}^K d_i + P_{max} [\max(N_1, N_2) - K] \quad (6.1)$$

where d_i is the distance for the i^{th} match; the second component of the measure was introduced to penalize non-matched descriptors, where P_{max} is a constant greater than the maximum possible value for the distance between two descriptors. The measure is normalized by the maximum possible value in each iteration. The best match corresponds to the pair that minimizes the

dissimilarity.

In the preliminary FIRST library available, we could only access the correlation value (*corr*) of the positive matches; thus, we set the dissimilarity measure to $1 - corr$.

A grid-based variation was considered for the SIFT, SURF and FIRST appearance models. It is composed of evenly spaced points and intends to provide an alternative to the use of interest point's detection by performing a dense scan. We tested grids with a number of points equal to 1% and 4% of the object's image.

Additional model variations were considered for the base descriptors, with the exception of dictionary-based models. Arguing that the top and bottom of the bounding box (for a human in the upright position these are associated with the head and feet) are more likely to be subject to noise (e.g., from the segmentation) and more background points may be included into the object's model, the size of the bounding box was altered. Specifically, the bounding box's height was scaled down around the center by predefined percentages (90%, 75% and 50%). This concept is illustrated in Figure 6.3. We will refer to these variations as object patch, or simply BB, accompanied by a percentage, e.g., 'for a 90% object patch' or '90% BB', where the percentage is related to the full BB height in each instance.

A different representation was also considered for the SIFT and SURF-based models, by encoding the descriptors into visual words following the approach described in [Tei09] and discussed in Section 6.2. Dictionaries were trained for both types of descriptors using generic images. This model was applied to the full object patch.



Figure 6.3: Example of the height variation of the bounding box. In the left image, full bounding box is used; the bounding box is scaled down from left to right.

Concluding, a total number of 47 appearance models were tested: 4 for Colour Histogram, 12 for SIFT, 12 for SURF, 4 for HOG, 12 for FIRST, 2 for BoW (with SIFT and SURF descriptors) and the baseline (texture model propose in [ZN04]). These were applied to 4 sequences, adding up to 188 experiments.

6.3.3 Evaluation Methodology

As stated, not all the sequences have reference information for every frame. Moreover, there are different types of reference information associated to the sequences. Hence, we decided to use two complementary metrics to objectively evaluate the impact of the different representation models in the tracking solution. We used the hybrid framework described in Chapter 5 which enables the computation of an error metric for every frame of the sequence. We will refer to the output of this framework as ‘hybrid metric’. This metric was complemented with measures proposed by Bashir et al. [BP06] to summarise evaluation results for a complete sequence. Specifically, the proposed object paradigm was used, which implies a previous alignment of reference and detected tracks. These will be referred to as ‘object metrics’. From the proposed set of metrics we selected:

$$\text{Tracker Detection Rate (TRDR)} = \frac{TP}{TG} \quad (6.2)$$

$$\text{Detection Rate (DR)} = \frac{TP}{TP + FN} \quad (6.3)$$

$$\text{False Alarm Rate (FAR)} = \frac{FP}{TP + FP} \quad (6.4)$$

$$\text{False negative Rate (FNR)} = \frac{FN}{FN + TP} \quad (6.5)$$

Moreover, in [BER03] the authors followed the same philosophy of [BP06] and proposed additional measures. Given a high superimposition, only one additional metric was selected:

$$\text{Tracking Success Rate (TSR)} = \frac{NF}{TG} \quad (6.6)$$

TG is the total number of ground truth tracks; true positives (TP) consists of the number of tracked objects mapped to a ground truth (GT) track; similarly, false positives (FP) are the number of tracked objects without a correspondence in the GT and false negatives (FN) the number of GT track without a tracked object mapped to it; NF is the number of non-fragmented tracks, i.e., with a 1:1 mapping with reference information. The tracker detection rate (TRDR), false alarm rate (FAR) and false negative rate (FNR) characterise the tracking performance of the algorithm, while the detection rate (DR) measures the sensitivity of the algorithm. The fragmentation error rate (FER) indicates how often a track label changes. Ideally, the FER value should be one, with larger values reflecting poor tracking and trajectory maintenance. The tracking success rate (TSR) summarizes the performance of the tracking algorithm with respect to track fragmentation. Moreover, several time measures were computed for each experiment to assess the impact of the descriptor and appearance representation in the algorithm execution and the corresponding computational weight. These consist of: the average frame processing time, which we will refer to as SPT (sequence processing time), in seconds per frame; the average descriptor extraction time (DET), in seconds per track; the average descriptor matching time (DMT), in seconds per

track. For each track, the values for DET and DMT were averaged over the number of iterations performed to define object correspondences.

It should be noted that, due to the limited reference information available for the PETS 2006 sequences it was not possible to compute the object metrics. In this case, evaluation was performed by employing the hybrid metric. This constitutes a concrete example of the benefits of the framework proposed in Chapter 5.

6.4 Results

Given the large number of experiments conducted, we will adopt a phased approach intended to make the presentation of the results clearer. We begin by comparing, in Table 6.1, the use of SIFT and SURF based models using both interest point detectors and grid points. The results presented correspond to the object metrics for sequences OSOW1 and OSOW2. Two grid densities were tested, specifically 1% and 4% of the number of points in the object's image. As described in Section 6.3.2, different heights for the patches (in percentage of the object's full bounding box height) were also considered. The best results are highlighted for each sequence and model. Note that for the first three metrics, TRDR, TSR and DR, we wish to obtain high values, while for metrics FAR and FNR low values are desirable. In the case of the FER the best result is 1, with increasing values traducing a performance degradation. These experiments were also assess with the hybrid metrics for the full dataset and the corresponding results are depicted in Figure 6.4 and Figure 6.5 for SIFT and SURF respectively.

In Table 6.1 it can be observed that a dense scan does not provide a significant contribute when using SIFT. In opposition, the performance of the tracking algorithm with SURF descriptors increased significantly by using a dense scan in alternative to interest point detection. These behaviors can also be observed in Figure 6.4 and, Figure 6.5. Given a superimposition of the values, for clarity of the graphs, only the values for full object's patch are used in this analysis. For SIFT, (Figure 6.4), little changes are visible between the use of interest and grid points; when applied to the PETS's sequences, slight differences are visible around frame 1500 (for Figure 6.4(c) and Figure 6.4(d)) and frame 900 (for Figure 6.4(d)) where the error value is zero due to the non-detection of objects. Note that only a reduced number of frames with reference information exists for these sequences; this may prevent the identification of these events if the corresponding frames are not encompassed in a sequence segment with frames containing reference information. Since the hybrid metric tends to be 'noisier' due to the sparse GT (about 3% of the full sequence) and given the superimposition of results, we chose to present only the hybrid metric for the PETS' sequences when justified by a relevant behavior difference between assessed methods. When using grid points, overall better results were obtained for smaller BB heights, indicating that grid points may introduce noise in the model.

It is well documented that the SURF interest point detector outputs less points than SIFT. In small image patches with possibly smooth regions this number may be very small or even null. By using a grid of specified points this problem can be minimized enabling better tracking results.

Table 6.1: Assessment of tracking performance using object metrics. It summarizes results for the use of a sparse (key points) or dense (grid) scan in the computation of SIFT and SURF descriptors, and for different heights of the object image (from [COC⁺12]).

Metric		OSOW1				OSOW2			
		Bounding Box Height (%)				Bounding Box Height (%)			
		100	90	75	50	100	90	75	50
SIFT	TRDR	0.68	0.58	0.61	0.44	0.64	0.64	0.58	0.59
	TSR	0.29	0.14	0.29	0.00	0.20	0.30	0.20	0.20
	DR	0.68	0.57	0.61	0.44	0.63	0.62	0.57	0.57
	FAR	0.10	0.12	0.12	0.26	0.23	0.25	0.32	0.24
	FNR	0.32	0.43	0.40	0.56	0.37	0.38	0.43	0.43
	FER	2.57	2.71	3.43	4.71	3.38	4.12	4.50	6.33
SIFT Grid 1%	TRDR	0.56	0.51	0.60	0.53	0.64	0.66	0.66	0.66
	TSR	0.29	0.43	0.43	0.43	0.30	0.40	0.30	0.20
	DR	0.55	0.51	0.60	0.53	0.63	0.65	0.64	0.65
	FAR	0.19	0.27	0.11	0.26	0.24	0.20	0.23	0.21
	FNR	0.45	0.49	0.40	0.47	0.37	0.35	0.36	0.35
	FER	2.00	1.86	1.57	1.86	2.78	2.88	3.56	3.13
SIFT Grid 4%	TRDR	0.45	0.47	0.63	0.43	0.56	0.56	0.62	0.66
	TSR	0.57	0.57	0.29	0.57	0.30	0.60	0.40	0.50
	DR	0.45	0.47	0.63	0.43	0.55	0.56	0.61	0.65
	FAR	0.34	0.36	0.15	0.38	0.39	0.36	0.34	0.29
	FNR	0.55	0.53	0.37	0.57	0.45	0.44	0.39	0.35
	FER	1.43	1.71	1.71	1.57	3.11	2.50	2.67	2.78
SURF	TRDR	0.14	0.14	0.14	0.09	0.20	0.17	0.17	0.10
	TSR	0.57	0.43	0.57	0.43	0.40	0.30	0.30	0.20
	DR	0.14	0.14	0.14	0.09	0.20	0.17	0.17	0.10
	FAR	0.08	0.04	0.12	0.52	0.21	0.23	0.22	0.50
	FNR	0.86	0.86	0.86	0.91	0.80	0.83	0.83	0.90
	FER	1.57	1.57	1.86	1.57	4.25	4.25	4.13	4.33
SURF GRID 1%	TRDR	0.71	0.72	0.70	0.71	0.68	0.70	0.72	0.65
	TSR	0.86	0.57	0.71	0.43	0.40	0.30	0.40	0.40
	DR	0.71	0.71	0.70	0.71	0.66	0.68	0.71	0.65
	FAR	0.02	0.09	0.10	0.03	0.21	0.18	0.16	0.19
	FNR	0.29	0.29	0.30	0.29	0.34	0.32	0.29	0.35
	FER	1.00	1.29	1.14	1.57	2.25	1.88	2.56	3.13
SURF GRID 4%	TRDR	0.78	0.54	0.72	0.75	0.64	0.42	0.63	0.64
	TSR	0.57	0.71	0.57	0.57	0.30	0.30	0.40	0.20
	DR	0.75	0.53	0.72	0.75	0.63	0.42	0.62	0.64
	FAR	0.12	0.36	0.09	0.05	0.37	0.59	0.38	0.36
	FNR	0.25	0.47	0.28	0.25	0.37	0.58	0.38	0.36
	FER	1.71	1.14	1.43	1.43	1.38	1.14	1.38	2.00

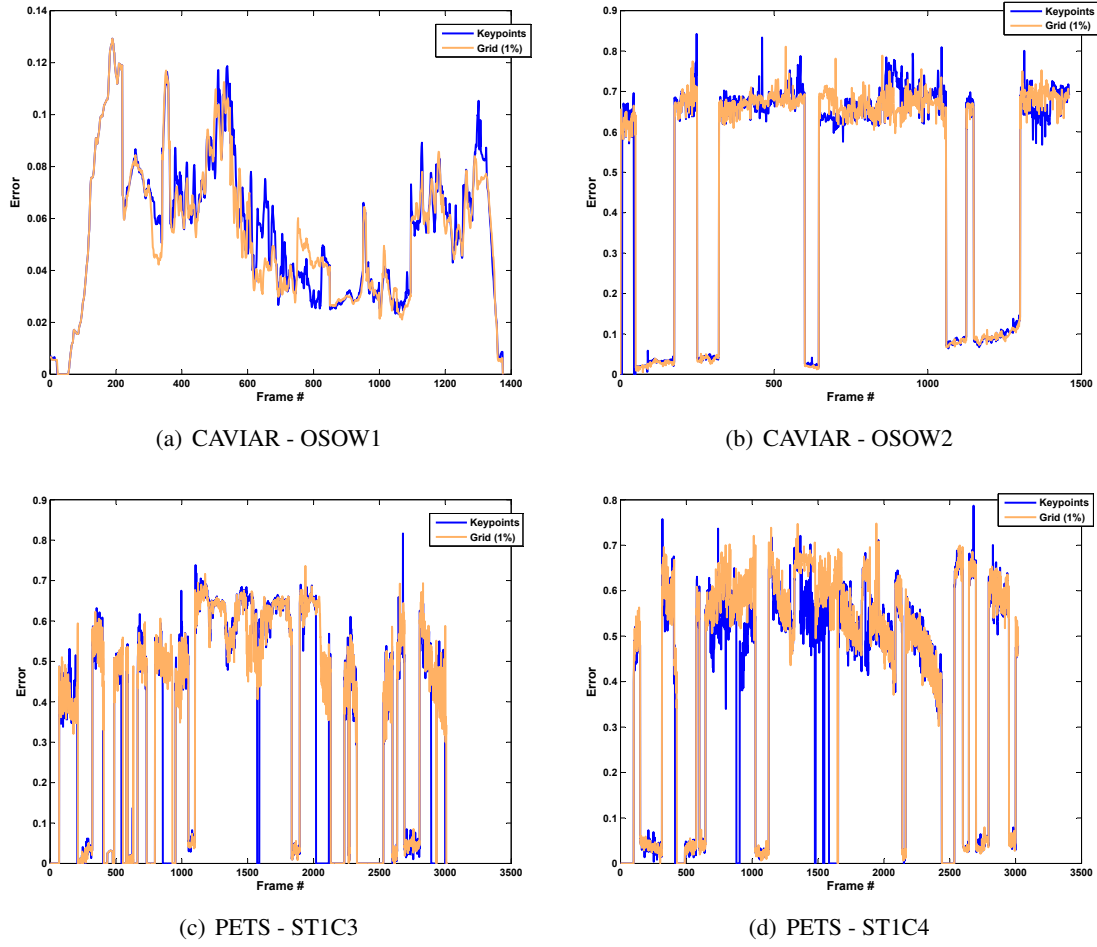


Figure 6.4: Tracking error, using the hybrid metric, for SIFT descriptors with both key points and a grid of points using the full object image (from [COC⁺12]).

With SIFT, the number of identified points is higher, hence grid points may not contribute with relevant information.

In Figure 6.6, the results for SIFT using key points and for SURF using grid points are compared using the hybrid metric. In both cases, full object patch was used. The results for the CAVIAR sequences (Figure 6.6(a) and Figure 6.6(b)) are very similar. For sequence ST1C3 (Figure 6.6(c)), the results are also alike, with the SIFT based solution failing to detect any object around frame 1700 and 2100. For sequence ST1C4 (Figure 6.6(d)), tracking with both methods does not detect objects around frame 1700, but with the SURF based solution failing for a longer period.

The object metrics' values for the tracking results using colour histogram and HOG based models are summarized in Table 6.2. These descriptors are computed over the object's image, thus a grid of points is not applicable in this case. It can be observed that the best results for the metrics are more disperse than for SIFT and SURF. Nevertheless, using small object patches

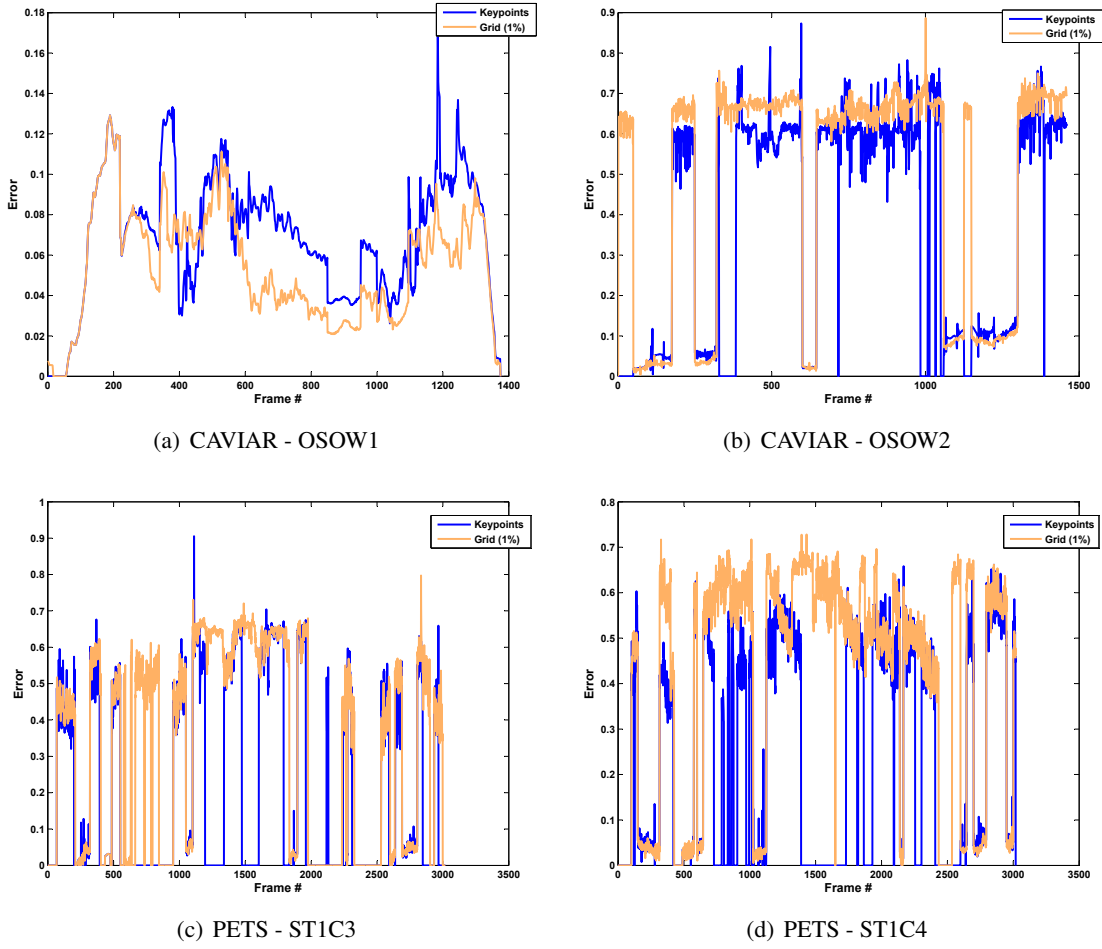


Figure 6.5: Tracking error, using the hybrid metric, for SURF descriptors with both key points and a grid of points using the full object image (from [COC⁺12]).

conducted to better results, which is coherent with the argument that smaller heights for the object bounding box (and consequently its image) can limit the background noise added to the models. The results conveyed by the hybrid metric are represented in Figure 6.7; it can be observed that no method clearly surpassed the other in terms tracking performance.

For the FIRST descriptors, we performed the same experiments as for SIFT and SURF. The overall tracking performance was low and the results for the object patch reduction did not provide improvements in comparison to the full patch; rather there was a slight performance degradation accompanying the reduction. Based on our observation we chose to present only the results for FIRST with key point detection and grid points for the full object patch, as the values for the other variations did not convey additional significant information. The corresponding object metrics are presented in Table 6.3. It should be noted that the comparison of the FIRST descriptors with the other methods is not a fair one. The obtained results did not met our expectations as they were impaired by implementation issues, which obliged a troublesome integration with consequent

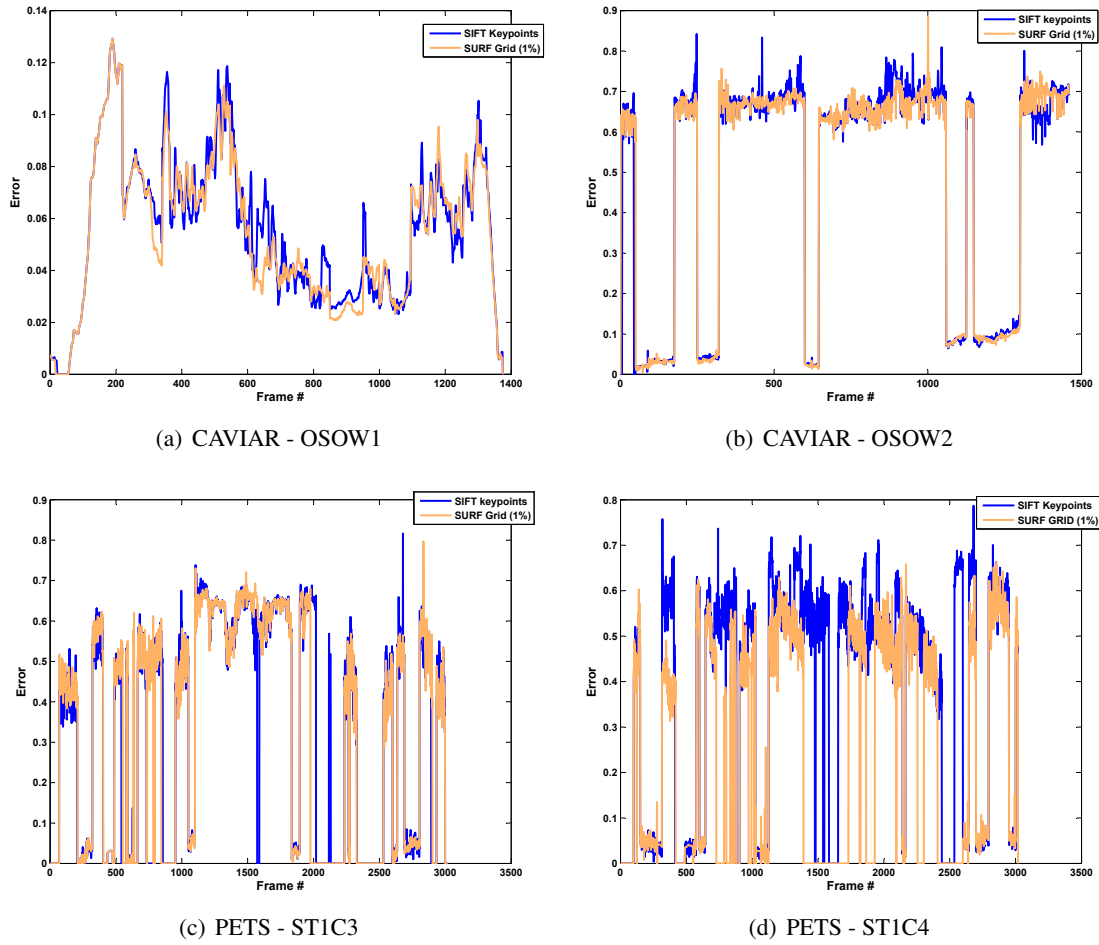


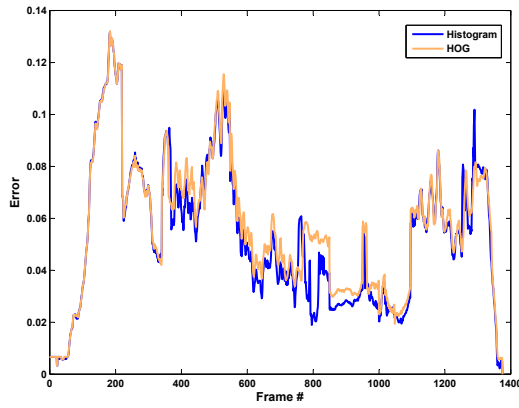
Figure 6.6: Comparative analysis of using SIFT with interest point detection and SURF with a grid of points (number of points equal to 1% the object's image pixels) over the test sequences (from [COC⁺12]).

computational performance degradations. Furthermore, the available information for computing a dissimilarity measure was limited and proved to be insufficient. As a result, there was high instability in establishing identities correspondences, thus preventing tracking.

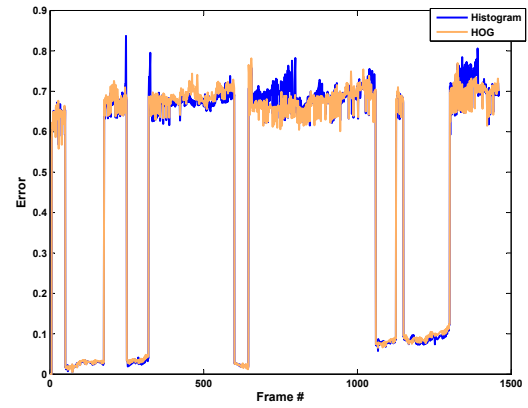
Table 6.3 summarises the results and enables the comparison of the tracking solution using the different methods and variations. For the experiments using grid points we selected those with the best performance for each model. These results show that SURF with grid points (1% density), HOG and colour histogram enabled the best performances. Furthermore, with these methods, the tracking solution performed better than with the originally proposed model (using texture [ZN04]). Between these three best approaches, it is difficult to highlight a particular one. Colour histogram enabled a good overall performance of tracking, accompanied closely by the solution using a HOG based model; the solution using SURF with grid points also presented a comparatively good performance as well as reliability concerning false alarms and misdetections. BoW-based models behaved poorly; they presented low values for FAR and FNR, but due to a very

Table 6.2: Assessment of tracking performance using object metrics. It summarizes tracking results for the use HOG and histogram based models (from [COC⁺12]).

	Metric	OSOW1				OSOW2			
		Bounding Box Height (%)				Bounding Box Height (%)			
		100	90	75	50	100	90	75	50
Histogram	TRDR	0.56	0.73	0.72	0.54	0.68	0.69	0.66	0.67
	TSR	0.71	0.57	0.43	0.57	0.20	0.60	0.70	0.60
	DR	0.56	0.73	0.72	0.54	0.67	0.69	0.66	0.67
	FAR	0.32	0.08	0.08	0.33	0.27	0.28	0.28	0.30
	FNR	0.44	0.27	0.28	0.46	0.33	0.31	0.34	0.33
	FER	1.00	1.29	1.29	1.29	1.29	1.25	1.22	1.11
HOG	TRDR	0.77	0.75	0.79	0.60	0.74	0.72	0.72	0.76
	TSR	0.57	0.43	0.57	0.71	0.40	0.40	0.40	0.40
	DR	0.76	0.74	0.78	0.59	0.73	0.72	0.70	0.73
	FAR	0.09	0.13	0.14	0.27	0.24	0.25	0.28	0.25
	FNR	0.24	0.26	0.22	0.41	0.27	0.28	0.30	0.27
	FER	1.29	1.57	1.29	1.14	1.87	1.62	1.50	1.56



(a) CAVIAR - OSOW1



(b) CAVIAR - OSOW2

Figure 6.7: Comparative analysis of colour histogram and HOG-based appearance models for the CAVIAR sequences (from [COC⁺12]).

low detection rate. Despite a worse performance with FIRST based models, due to the mentioned motives, it is noticeable an improvement when using a grid of points.

Table 6.4 presents the time measures presented in Section 6.3.3 for the most relevant experiments considering the tracking performance results. As expected, colour histogram and HOG exhibited a good computational performance. SIFT required a higher computational complexity, but the use of a grid of points enabled a significant decrease of the computational time. Higher descriptor extraction time is a known problem with SIFT, but it can be minimized through the use of grid points. SURF also enabled low computational time values, but the tracking results were poorer. However, it is noteworthy the values obtained for SURF with a grid of points; although they are higher than with the use of interest point detection, they are lower than with SIFT and it was demonstrated in Table 6.3 that they enable better tracking results. BoW with SURF descriptors presented low processing times for both descriptor extraction and matching, but the sequence processing time is higher than with the use of SURF or SURF grid. For BoW with SIFT it is

Table 6.3: Overall comparison of the experimented models. For grid based representations, the best results were selected for each type of descriptor (from [COC+12]).

Appearance Model Comparison												
	Metric	SIFT	SIFT Grid (1%)	SURF	SURF Grid (1%)	FIRST	FIRST Grid	Colour Histogram	HOG	SIFT Bow	SURF Bow	Texture
OSOW1	TRDR	0.68	0.60	0.14	0.71	0.00	0.03	0.73	0.79	0.12	0.10	0.50
	TSR	0.29	0.43	0.43	0.86	0.00	0.14	0.57	0.57	0.43	0.29	0.43
	DR	0.68	0.60	0.14	0.71	0.00	0.03	0.73	0.69	0.12	0.10	0.50
	FAR	0.10	0.11	0.04	0.02	0.00	0.14	0.08	0.14	0.00	0.33	0.35
	FNR	0.32	0.04	0.86	0.29	0.00	0.97	0.27	0.22	0.88	0.89	0.50
	FER	2.57	1.57	1.57	1.00	0.00	2.00	1.29	1.29	1.00	1.33	1.33
OSOW2	TRDR	0.64	0.66	0.20	0.72	0.01	0.01	0.69	0.76	0.15	0.22	0.58
	TSR	0.20	0.40	0.40	0.40	0.00	0.10	0.60	0.40	0.20	0.00	0.50
	DR	0.63	0.65	0.20	0.71	0.01	0.01	0.69	0.73	0.15	0.22	0.57
	FAR	0.23	0.20	0.21	0.16	0.68	0.61	0.28	0.25	0.05	0.05	0.37
	FNR	0.37	0.35	0.80	0.29	0.99	0.99	0.31	0.27	0.85	0.80	0.43
	FER	3.38	2.88	4.25	2.56	3.67	3.00	1.25	1.56	3.33	2.00	2.50

noticeable the very high description extraction times and consequently of the sequence processing times.

6.5 Discussion

Given the difference between tracking and standalone image or object matching, it was considered logical the analysis of different description models in a common tracking solution. A set of well known description techniques and different representations were analysed and assessed by measuring the output of the tracking solution in terms of computational performance, robustness and precision of the results. As part of this work, the foundations for a framework for the future test of other techniques within a visual tracking context were set.

Although the choice of the algorithm can be argued considering the vast number of published work, such discussion has no effect in the current context since all representations methods are assessed in a common framework and conditions; it is the impact of the individual models that we wish to evaluate.

SIFT achieved competitive tracking accuracy, but at the expense of a higher computational weight. It is noteworthy the negligible contributions of using a dense scan, in alternative to interest point detection. An opposite behavior was exhibited by SURF; its computational performance was significantly better than SIFT, but the very small number of detected key points caused a degradation of the tracking results. The use of a grid points in the computation of SURF descriptors improved the tracking results surpassing, in some cases, the other methods. This is indicative that using a dense scan of the object's patch can offer a more powerful representation than the sparse key points. Moreover, even using a grid of points, tracking with SURF still exhibited a better computational performance than with SIFT.

As expected, colour histograms and HOG exhibited good performances. In particular, the results for colour histograms were not surprising considering the scenario; the use of a single camera with a reasonable frame rate enabled smooth transitions from one frame to the next. Furthermore, object patches with small dimension and smooth regions are often an obstacle to the computation of edge or gradient based descriptors.

The results for FIRST were disappointing, but we maintain expectations of better results in future experiments. Feedback has been provided and changes to the FIRST descriptor algorithm are being analysed.

There was a decision to use sequences widely known to the research community, focusing on surveillance scenarios, and a solution designed for people tracking. Nevertheless the reasoning underlying this assessment proposal can and should be extended to other scenarios and description methods. It would be interesting to perform the analysis proposed in this Chapter in multi-camera scenarios thus obtaining more complete information regarding the description methods robustness and the feasibility their used for both single and multiple cameras.

The work described in this chapter resulted in the following paper submission:

Table 6.4: Comparison of the processing times for the solution and for individual components of the models: extraction and matching (from [COC⁺12]).

	OSOW1			OSOW2			STIC3			STIC4		
	SPT (sec/frame)	DET (sec/track)	DMT (sec/track)	SPT (sec/frame)	DET (sec/track)	DMT (sec/track)	SPT (sec/frame)	DET (sec/track)	DMT (sec/track)	SPT (sec/frame)	DET (sec/track)	DMT (sec/track)
SIFT	0.420	0.175	0.000	0.678	0.126	0.000	0.800	0.559	0.003	1.150	0.420	0.001
SIFT Grid (1%)	0.242	0.092	0.000	0.427	0.070	0.000	0.416	0.239	0.000	0.565	0.175	0.000
SURF	0.033	0.002	0.000	0.061	0.002	0.000	0.055	0.014	0.000	0.092	0.012	0.000
SURF Grid (1%)	0.194	0.074	0.002	0.316	0.036	0.001	0.354	0.202	0.010	0.509	0.151	0.007
FIRST	46.160	3.510	0.726	111.983	8.607	2.548	44.085	8.151	3.556	68.384	10.058	3.497
FIRST Grid (1%)	37.398	3.896	0.489	139.441	21.478	2.531	25.295	1.605	2.620	67.336	14.944	2.749
Histogram	0.213	0.000	0.000	0.351	0.000	0.000	0.190	0.000	0.000	0.292	0.000	0.000
HOG	0.052	0.007	0.001	0.090	0.008	0.001	0.058	0.010	0.002	0.090	0.009	0.001
SIFT BoW	12.693	11.221	0.005	12.113	1.132	0.001	35.587	34.492	0.004	35.415	17.210	0.005
SURF BoW	0.426	0.000	0.000	0.766	0.000	0.000	1.268	0.000	0.000	2.006	0.000	0.000
Texture	0.084	0.000	0.000	0.125	0.000	0.000	0.177	0.000	0.000	0.199	0.000	0.000

Pedro Carvalho, Telmo Oliveira, Lucian Ciobanu, Filipe Gaspar, Luís F. Teixeira, Rafael Bastos, Miguel Sales Dias, Jaime S. Cardoso, and Luís Côrte-Real. Analysis of Object Description Methods in a Video Object Tracking Environment. *Machine Vision and Applications*. (submitted).

This paper included an individual analysis of the descriptors using the framework described in [BD09], which simulates the complete pipeline of images 3D viewing and projection. It enables the generation and rendering of images under different viewing variations, such as scale and rotation, as well under different perturbations such as blurriness and random noise,; thus it is possible to test image descriptor's extraction and matching techniques between two images.

Chapter 7

Conclusion

“If I had only known, I would have been a locksmith.”

Albert Einstein

7.1 Final Remarks

This thesis focused on the evaluation of automated video object tracking algorithms and the analysis of object description schemes that can be applied to monocular tracking, but with potential to be extended to multi-camera scenarios.

Despite all the work that has been done, existing tracking systems still fail, especially when used in non-controlled environments. The dependency on other ongoing research areas, as is the case of image segmentation, makes the affirmation of tracking algorithms more difficult. Moreover, the rarity of available implementations of state-of-the-art algorithms and the lack of consistent use of open datasets are obstacles for research on this topic, commonly preventing the replication of results. Currently deployed solutions for tracking and monitoring are lacking a fully satisfactory response to many important problems faced by the contemporary society (e.g., thefts, murder, public safety, terrorism) reinforcing the need for new solutions. Furthermore, the focus on specific application scenarios, the testing of algorithms’ variations and the effort associated with the generation of reference information have highlighted and augmented the importance of objectively evaluating the algorithms, indicating the need for flexible solutions.

There is still no consensus regarding the objective evaluation of tracking, with the existence of multiple tracking algorithms with different objectives and input/output information making it hard to establish a common methodology. The image segmentation evaluation framework developed by Cardoso [CCR05] was naturally extended to assess spatial segmentations of video, also allowing the detection of the most typical errors introduced by tracking algorithms. Additional extensions

were engineered making the framework more flexible and generic by analysing its use with different types of reference information and fusing the results with the output of a metric without ground truth. The proposed approach was gauge through the use of synthetic sequences using a real background and depicting common tracking errors; different amounts of segmentation noise were simulated. Real sequences were also used in the validation process. This hybrid framework can be used in complement with other metrics of the state-of-the-art and can contribute to the unification of the use of ground truth. Notably, this solution enabled the evaluation of the experiments conducted in Chapter 6 with a dataset containing different types of ground truth; moreover, for two of the sequences only a small number of frames had corresponding reference information.

For the purpose of assessing different object description schemes, the implementation of a tracking solution was adapted to enable the integration of widely used implementations of state-of-the-art descriptors. This implied a considerable effort, but will facilitate future tests of other descriptors and appearance representation techniques. The experiments enabled an analysis of the behavior of the overall tracking solution, rather than focusing on the individual performance of the description method. Moreover, this approach enabled the comparison of different types of descriptors, e.g., colour histogram and SIFT. Results showed that for single camera scenarios HOG and colour histogram with spatial information enabled the best tracking results and computational performance. These description models were closely followed by SURF with a dense scan; it proved more robust and with better computational performance than SIFT.

A note regarding the work on vibrating line detection and tracking. This topic was proposed to us during the work on this thesis. The recognized relevance justified a parallel effort which resulted in contributions to the state-of-the-art. A strategy based on automatically line detection was proposed, enabling, not only the characterisation of marked points, but of all the line. The lack of appropriate sequences and corresponding GT motivated the generation of a set of synthetic sequences with different levels of complexity as well as the generation of the corresponding ground truth.

This thesis contextualized other research activities through the proposal and supervision of five Master dissertations. Furthermore, the work developed conducted to an active participation and contributions to European and national funding research projects, specifically:

- Network of Excellence VISNET II (IST - 1 - 038398) of the 6th Framework Programme;
- ROBVIGIL, a research project funded by the Quadro de Referência Estratégico Nacional (QREN), 15/SI/2009, Proj. n° 7905, 2009
- Living Usability Lab, a research project funded by the Quadro de Referência Estratégico Nacional (QREN), 15/SI/2009, Proj. n° 7900, 2009

7.2 Future Research Lines

The studies of this thesis, although providing tools to assist future developments and some conclusive results, constitute the starting point for possible larger projects, focusing on the preparation

of a flexible framework for the assessment of tracking algorithms and the development of such solutions targeting different scenarios.

Future research work will likely be closely related to multi-camera tracking. The proliferation of video cameras and the growth of camera networks potentiates and motivates work in this field. Given the complexity of these scenarios and the possible mutability of the networks, a multi-layer tracking solution with independent tracking for each camera and information being aggregated at higher layers can be an interesting solution. Flexibility and performance of the multi-camera tracking may benefit if the cameras and the corresponding intra-camera information can be ‘plugged’ in and out of the system in a transparent manner, with an automatic adaptation to those changes, for example, by incrementally learning the relations between cameras. Furthermore, the ability to cope with different types of algorithms, or even combine them, would be of great importance, augmenting the adaptability of the system. Research on this topic will undoubtedly required analysis of appearance models and definition of new solution as well as tools to enable the objective assessment of the developments.

A proposed line of evolution concerns the evaluation of tracking algorithms. The framework described in Chapter 5 proved capable of measuring the tracking error for every frame of the sequence with major types of errors being noticeable. The analysis of the error evolution should be studied to determine the feasibility of developing automatic methods to, not only identify the frame of occurrence of the errors, but also the type of error. Given the importance of common datasets open to the research community and the difficulty in generating reference information, the efficiency of the framework could be maximized by investigating methods to perform an analysis of the test sequences with the objective of determining the best distribution of the different types of ground truth according to a criteria: approximate the ideal error and/or minimize the required ground truth. Furthermore, accompanying the increasing interest for multi-camera tracking, it is of great relevance the research of a method to aggregate error information associated to multiple cameras, thus extending the framework to multi-camera assessment. Research initiatives are being launched with the purpose of building a generic evaluation platform and integrating different types of metrics. Hence, efforts should be devoted to establishing collaborations with such activities and promote the integration of the proposed framework in these platforms, as well as collect identified requirements for future developments.

A second line of research concerns the study of different description methods and representation techniques in a common tracking solution, extending more common assessment approaches. A first implementation of this strategy was performed. Despite the results obtained, the analysis conducted did not cover every possible descriptor or appearance model proposed in the literature and other techniques should be analysed. This study should also be extended to other datasets, types of images (e.g., infrared images) and to multi-camera tracking scenarios. This would allow a more complete characterisation of the capabilities of individual techniques and their tolerance to different viewing conditions, as well as the feasibility of building an appearance model horizontal to both single and multi-camera tracking. It would also be interesting to test the effect of lower frame rates, as this will typically imply larger differences in position, shape and even appearance

of an object in consecutive frames. This research line can provide valuable input for the development of research algorithms and, at the same time, benefit from advances in these solutions, as they would enable testing under different conditions.

In the field of signal processing and audio analysis, the Marsyas platform [TCJ⁺08] (an open source project) has been gaining momentum, providing a modular architecture that can facilitate prototyping, testing and replication of results. There already are efforts to extend such framework, enabling its use in an image and video processing context [TMLT08], but it is still an ongoing project. The possibility of using such platform to test and analyse different techniques and algorithms would benefit future developments in video object tracking.

The tracking of civil engineering structures presented an opportunity to contribute to scenarios often neglected by researchers, but with undeniable value for the application field. Our work focused on vibrating lines, but there is a need for similar analysis for other types of structures. Also, the same principles can be applied to other fields as is the case of mechanics. Preceding new developments or proposal of new methods, it is important to capture and prepare additional video sequences that should be accompanied by data, acquired by other types of sensors, to be used for validation purposes.

Appendix A

Partition-Distance Metrics

“Theory helps us bear our ignorance of facts.”

George Santayana

A.1 Introduction

The following sections briefly present the partition-distance metrics framework defined by Cardoso and Corte-Real [CCR05] as well as underlying concepts and notations. The metrics were initially defined to make up a framework for the independent and reliable measurement of the performance of image segmentation algorithms. As the contents of this appendix are not intended to make an exhaustive description of the partition-distance metrics, the reader is referred to [CCR05, CCTCR09] for further information .

In Section A.2 it is reviewed the intersection-graph between two segmentations and discussed its merits as a factory of similarity indices between segmentations. Section A.3 outlines the strict partition-distance, and illustrates its properties. It is also presented the general rationale underpinning all the measures presented. Section A.4 then describes an asymmetric measure encoding a measure of refinement in one direction only. It is widely agreed that human segmenters differ in the level of detail at which they perceive images. Therefore, in Section A.5 it is presented a symmetric measure, the mutual partition-distance, derived under the same general principle and that tolerates refinements in both directions.

A.2 The Intersection-Graph Between Two Segmentations

A *measure of the dissimilarity* between two segmentations, most often a real number, is nothing more than a sensible *summary* of that same dissimilarity. The reasons to aim for such a sum-

mary are well-known, ranging from objectiveness to easier interpretability and benchmarking. In [CCR05] the authors introduced the intersection-graph, to be presented next, as a powerful tool to derive measures of similarity between two segmentations, in the sense that, although retaining much less information than the two original segmentations (one cannot reverse the process and deduce the segmentations from the intersection-graph), it still retains the information that differentiates the two segmentations. Therefore, measures of the difference between two segmentations can be defined directly on the intersection-graph.

As the work on partition-distance metrics is based heavily on set and graph concepts, some helpful notions and notation on these topics are in order.

A.2.1 Definitions and Notation

Let S be a set of N elements. A cluster is a *non-empty* subset of S . A partition of S is a set of mutually exclusive clusters, whose union is S . Two partitions P and Q of S are *identical* if and only if every cluster in P is a cluster in Q . A partition P is a *refinement* of a partition R (or P is *finer* than R) if and only if each cluster in P is contained in some cluster of R — see Figure A.1. Consequently, by definition, any partition is a refinement of itself.

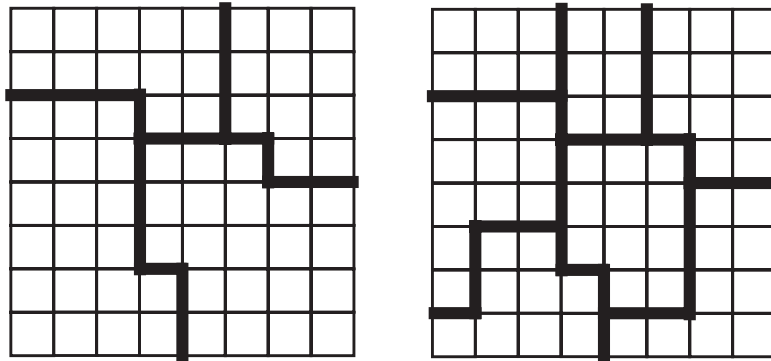


Figure A.1: The right partition is a refinement of the left partition (from [CCTCR09]).

The *intersection of two partitions* P and Q is a partition R so that every non-empty intersection of a cluster S_i from P and a cluster S_j from Q is an element of R — see Figure A.2. Note that R is a refinement of P and Q .

The *null partition*, P_ϕ , is the partition with only one cluster (the cluster has N elements); the *infinite partition*, P_∞ , is the partition with N clusters (each cluster has one element).

A *graph* $G = (V, A)$ is composed of two sets V and A . V is the set of nodes, and A the set of arcs (p, q) , $p, q \in V$. The graph is *weighted* if a weight $w(p, q)$ is assigned to each arc. Define a *path* in a graph as a chain of connected edges never turning back and the *diameter* of a graph as the length of its longest path. A bipartite graph $\mathcal{BG}$ is a graph whose set of vertices V can be split into two subsets V_Θ and V_Ψ in such a way that each edge of the graph joins a vertex in V_Θ and a vertex in V_Ψ —Figure A.3(a). A bipartite graph with θ vertices in V_Θ and ψ vertices in V_Ψ is denoted by $\mathcal{BG}_{\theta, \psi}$. A complete bipartite graph is a bipartite graph in which each vertex in V_Θ is joined to

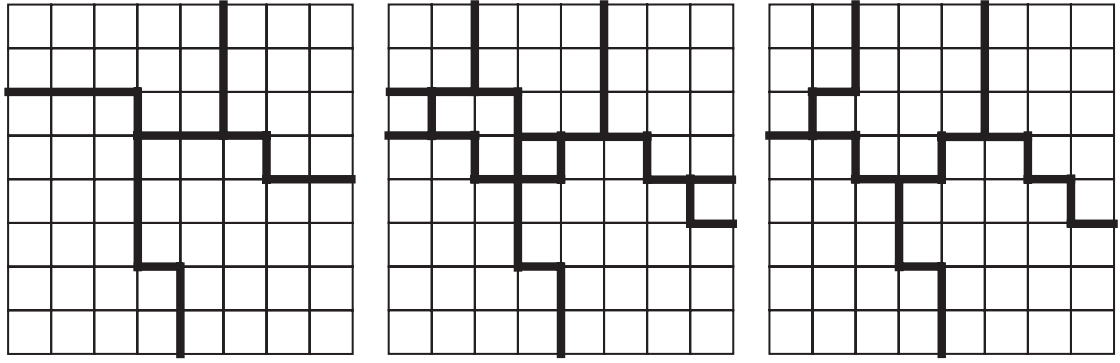


Figure A.2: The middle partition is the intersection of the left and right partitions (from [CCTCR09]).

each vertex in V_Ψ by an edge—Figure A.3(b). The complete bipartite graph with θ vertices in V_Θ and ψ vertices in V_Ψ is denoted by $K_{\theta,\psi}$. A tree graph is a simple, undirected, connected, acyclic graph—Figure A.3(c). A tree with n nodes has $n - 1$ edges. Conversely, a connected graph with n nodes and $n - 1$ edges is a tree. The n -star graph, S_n , is a tree on $n + 1$ nodes with one node having vertex degree n and the others having vertex degree 1. The complete bipartite graph $K_{1,n}$ is the star graph S_n —Figure A.3(d).

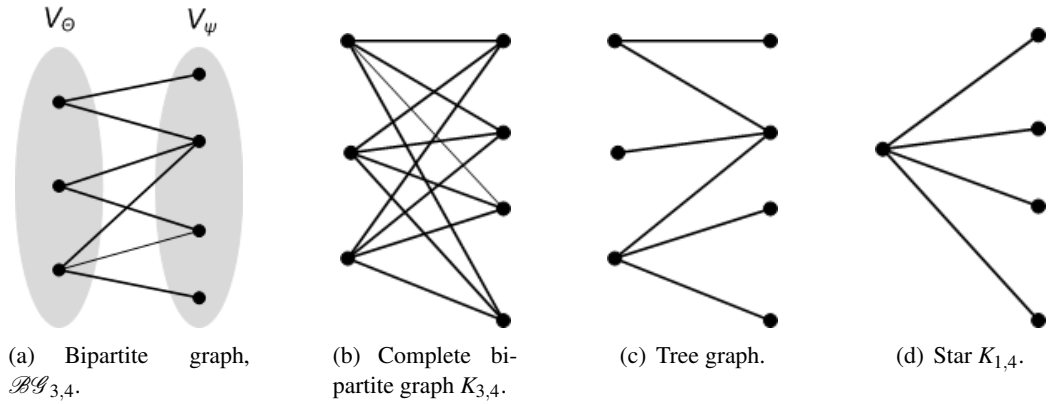


Figure A.3: Examples of bipartite graphs (from [CCTCR09]).

A.2.2 The Intersection-Graph

Given a set S of N elements and two partitions of S , P and Q , define the *intersection-graph* as the bipartite graph $\mathcal{BG}(P, Q)$ with one node for each region of the segmentations. Two nodes are connected by an undirected, weighted edge if and only if those two regions intersect each other. Figure A.4 exemplifies such setting.

Although not necessarily the best way to capture the perceived quality of a segmentation, the simplest way is to assign the area of the intersection to the weight of each edge (as chosen in

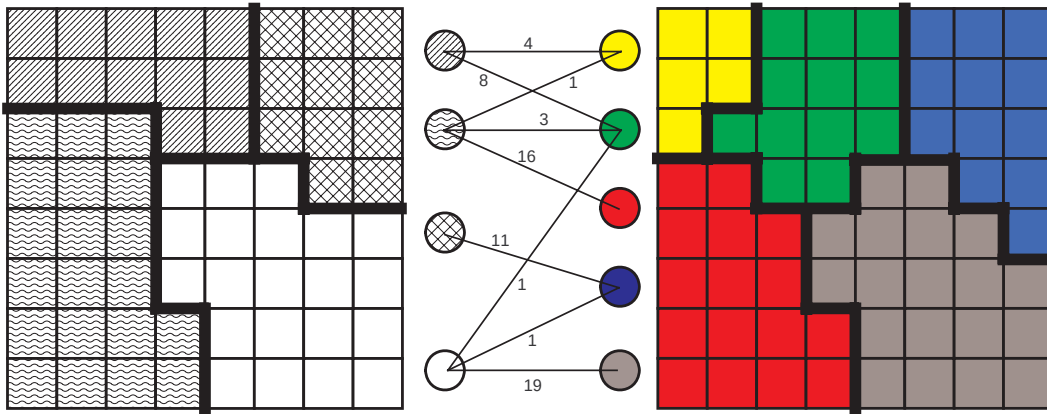


Figure A.4: Intersection-graph for two segmentations. The weights shown correspond to the number of pixels in the intersection (from [CCTCR09]).

Figure A.4, for illustrative purposes); however, shape characteristics on intersections can also be considered. The weight of an edge should express the importance of the corresponding intersection.

The intersection-graph associated with two image segmentations can now be used as a factory of indices of dissimilarity between partitions. Rules can be defined on the vertices and edges of the bigraph to create suitable measures.

A.3 Partition-Distance

The spatial segmentation of an image corresponds to the *partition* of the entire image into its constituent regions or objects, such that all regions are disjoint and every pixel belongs to a region. The segmentation evaluation problem is then to find an accurate description of the relation between two spatial partitions, the reference (or ground truth) and the actually segmented, by defining the dissimilarity between two partitions. The idea of *partition-distance* as first presented in [AF99] looks into the problem in exactly that form.¹

Given two partitions P and Q of S , the *partition-distance*, $d_{\text{sym}_1}(P, Q)$, is the minimum number of elements that must be deleted from S so that the two induced partitions (P and Q restricted to the remaining elements) are identical. Note here the general rationale underpinning the definition: define a measure as the minimum number of points to be removed from a set such that a certain condition is verified by the remaining elements. The condition to be verified with the partition-distance definition is the equality of the induced partitions. If an appropriate cost is assigned to each point, the rationale can be further generalized with the introduction of a cost-based distance, defined as the minimum sum of the costs of the points to be removed such that a certain condition is verified in the retained elements.

¹Throughout this appendix, the terms segmentation and partition will be used almost interchangeably.

The definition of the partition-distance between two segmentations enjoys of a useful set of properties:

1. $d_{sym_1}(Q, P) \geq 0$
2. $d_{sym_1}(Q, P) = 0$ if and only if $Q = P$
3. $d_{sym_1}(Q, P) = d_{sym_1}(P, Q)$
4. $d_{sym_1}(Q, P) + d_{sym_1}(P, R) \geq d_{sym_1}(Q, R)$
5. $d_{sym_1}(P_\emptyset, P_\infty) = N - 1 = \text{maximal distance between any two partitions}$
6. the normalized distance, $d_{sym_1}/(N - 1)$, ranges from 0 to 1

The proof of the above properties can be found in [CCR05]. Any function with properties 1 to 4 is called a *metric*. The transitive property is especially significant in the context of comparing more than two algorithms. It conveys the desirable behaviour that if segmentation A is similar to segmentation B and segmentation B is similar to segmentation C, then segmentation A is similar to segmentation C. Property 6 tells us how to properly normalize the partition-distance in the $[0, 1]$ interval. This normalization enables the fair comparison of results among images of different sizes.

By now it should be natural to apply the partition-distance defined above to measure the discrepancy between the reference segmentation (nothing more than a partition of an image) and the segmentation under evaluation. Consider for instance the two partitions of the same 8×8 image, presented in Figure A.5.

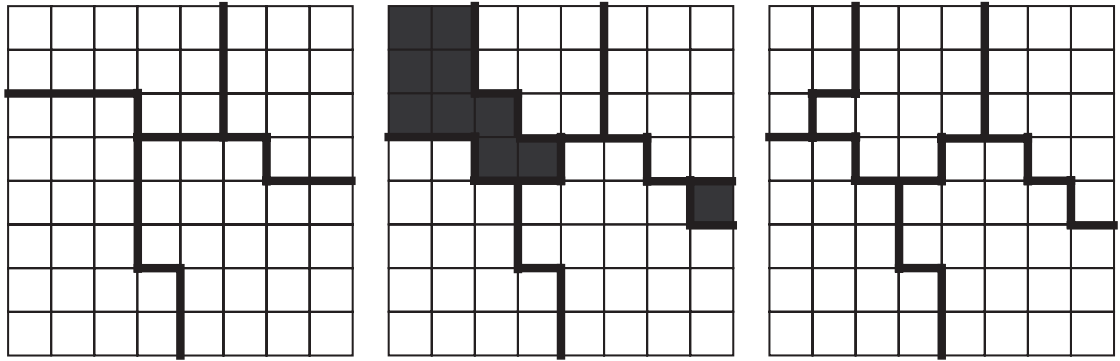


Figure A.5: On left and right, two different partitions of the same image—the middle image highlights the points to be removed when comparing both with the partition-distance (from [CCTCR09]).

According to the distance defined above, these partitions are 10 pixels away from each other. The pixels that had to be removed are highlighted in the middle image (unique solution in this particular case).

To be of any practical use, the proposed measures have to be efficiently computable. It is shown in [Gus02] that the partition-distance can be computed in polynomial time, formulating the

problem as an instance of the classical assignment problem, directly on the intersection-graph. The partition-distance equals the sum of the weights of the pruned edges. Up until now, two key ideas have been presented: the methodology of removing the minimum number of elements such that a condition is verified in the remaining elements and the intersection-graph as the tool to efficiently solve the resulting global optimization problem.

The weight assigned to a pixel should express the cost of erring in that pixel for the purpose of evaluating the segmentation. As alluded before, the rationale of removing the minimum number of elements such that the two partitions are equal can accommodate easily such generalization by minimizing the sum of the costs of the removed elements. Equally interesting, the intersection-graph continues to be the natural representation for this setting. By assigning to an edge's weight the sum of the costs of the corresponding pixels, the minimization of the cost-based partition-distance resumes to a matching problem.

The simplest cost selection is to assign unitary cost to each element, thus employing the area of the intersection as the weight of each edge. But this formulation is not necessarily the one better capturing the perceived quality of a segmentation. To accommodate human perception, the different error contributions should be weighted according to their visual relevance. Therefore, it may be, arguably, more sensible that the cost of erring a pixel increases with pixel distance to the object border.

A.4 Asymmetric Partition Distance

In many applications under-segmentation is considered as a much more serious problem than over-segmentation. This is so because it is easier to recover true segments through a merging process after over-segmentation rather than trying to split a heterogeneous region. For those environments, it would be sensible to define an asymmetric index between two partitions in such a way that the dissimilarity between a reference partition R and any partition under evaluation Q finer than R is zero. Note that this is no longer a symmetric setting.

Proceeding under the theoretical foundations presented so far, a proper measure can be defined for such applications. Given two partitions R and Q defined in a set S of N elements, define the *asymmetric partition-distance*, $d_{asy}(R, Q)$, as the minimum number of elements that must be deleted from S , such that the induced partition Q results finer than the induced partition R . Under this asymmetric distance, any partition finer than the R partition will be at zero distance from it. Notice also that, in general, $d_{asy}(R, Q) \neq d_{asy}(Q, R)$.

Working with the segmentation partitions already used to exemplify the symmetric partition distance, asymmetric distance attains the values (see Figure A.6):

$$d_{asy}(\text{left}, \text{right}) = d_{sym_1}(\text{intersection}, \text{right}) = 10$$

$$d_{asy}(\text{right}, \text{left}) = d_{sym_1}(\text{intersection}, \text{left}) = 6$$

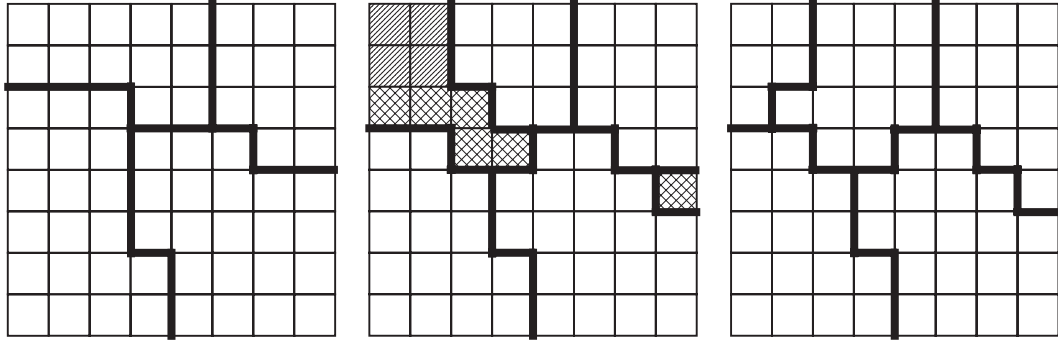


Figure A.6: The middle partition highlights the points to be removed for the asymmetric measures $d_{asy}(R, Q)$ and $d_{asy}(Q, R)$ (from [CCTCR09]).

A.5 Mutual Partition-Distance

It is known that humans may segment an image differently: the same scene may be distinctively perceived; different subjects may attend to different parts of the scene; subjects may segment an image at different granularities. Nevertheless, segmentations of the same image tend to be consistent in the sense that they are mutual refinements of each other [Mar03]. Therefore, it seems interesting to have measures encoding a tolerance to mutual refinements.

Formally, a partition P is said to be a mutual refinement of a partition Q if and only if every cluster in P contains or is contained in a cluster in Q —Figure A.7.

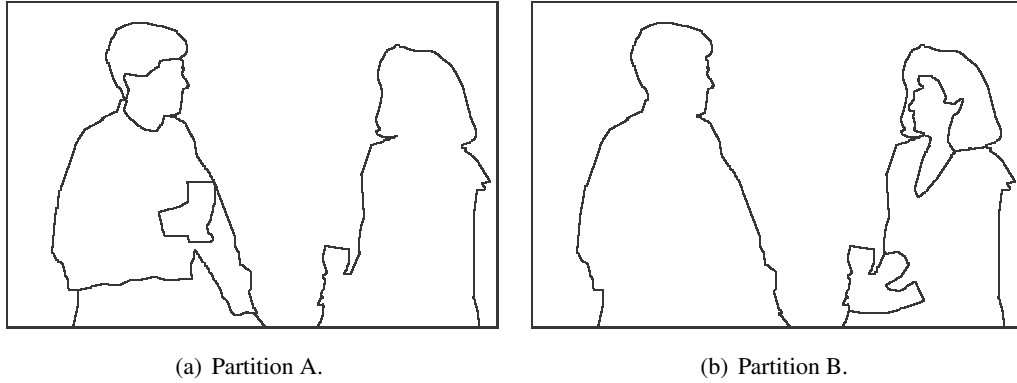


Figure A.7: Partitions A and B are a mutual refinement of each other. (from [CCTCR09]).

If partition Q is a mutual refinement of partition P , then P is a mutual refinement of partition Q .

The concept of mutual refinement is easily built-in in the proposed methodology: given two partitions P and Q defined in a set S of N elements, define the mutual partition-distance, $d_{sym_2}(P, Q)$, as the minimum number of elements that must be deleted from S , so that the induced partitions are mutual refinements of each other.

Once again, the problem of computing the mutual partition-distance can be casted naturally on the intersection-graph derived from the partitions. The partitions P and Q are a mutual refinement of each other if and only if the associated intersection-graph has only paths of length no greater than two [CCR05]. Hence, the mutual partition-distance can be formulated in the intersection-graph as the minimum sum of weights of pruned edges such that the induced (pruned) bigraph has paths of length at most two.

Appendix B

A Detection Based Approach For Tracking Vibrating Lines in Video Sequences

“Science is simply common sense at its best.”

Thomas Huxley

B.1 Introduction

The monitoring of vibrating structures is important in several areas including mechanics and civil engineering. The observation of structures like cable-stayed bridges, antennas or electricity distribution systems is vital for the extension of their life-span and for the reduction of system failures and consequent costs. Past approaches made use of devices, such as accelerometers or load cells (devices which translate force into an electrical signal), placed at the structures to be monitored. Although the use of such techniques has produced accurate results, the installation of this type of equipment may easily become very expensive. Moreover, in many cases such as antennas or even some cable-stayed bridges it may not be possible to install such devices. Hence, there is a desire to use Computer Vision (CV) techniques to provide a noninvasive alternative with a possibly lower cost. Despite the existence of past work, the natural difficulties of these scenarios pose serious problems. These include: cameras placed at large distances; illumination changes; adverse weather conditions; reduced thickness of the structures; minimal displacements in the image.

In the monitoring of vibrating lines (or cables), we are interested in detecting the line and establishing correspondences in consecutive frames. However, it is of greater importance to be able to track either specific points or every point of the line, thus establishing point correspondences

throughout the sequence and enabling full line characterisation. The characteristics of the lines or cables, namely their reduced thickness and lack of texture information, forced common CV approaches to this problem to introduce restrictions and assumptions. Difficulties in identifying the structures to be tracked typically require a user to mark a point or region of interest (ROI), for example in the line or cable themselves or in their image representation; situations like occlusion may require a new marking of the points. Also, scenarios where the projection of the cables onto the image plane causes the cables to be close together may originate problems as the vibrations may lead to occlusion between them causing the lost of tracking.

We propose a method for automated tracking of vibrating lines, and points thereof, in video sequences. It builds on line detection and uses the outcome of this step in conjunction with the natural physical constraints of the scenarios to resolve shortcomings of traditional approaches. Our focus is on scenarios of interest to civil engineering, for example, cable-stayed bridges or antenna's support cables. Nevertheless, the proposed techniques can be applied or extended to other applications. We argue that in such scenarios, by first detecting the line, correspondences between points in consecutive frames can be more reliably identified.

By enabling the automatic selection of an arbitrary number of points over the line to be tracked, it is possible to characterise any point and not only those manually marked during the initialisation. This allows the algorithm to keep track of the structure's overall displacement, contributing to a more complete knowledge.

Following the reasoning underlying this proposal, we divided its description into detection and tracking. In Section B.2 a brief review of the state-of-the-art in both line detection and line tracking, with a especial focus on applications to civil engineering, is conducted. The proposals for line detection and tracking are described in Section B.3 and Section B.4 respectively. In Section B.5 is described the methodology used to assess the detection and tracking. This section also includes a characterisation of the dataset used. Section B.6 summarizes the results of the assessment of the proposals. Finally, a discussion of the work conducted, results obtained and indications of future research work is given in Section B.7.

B.2 Background

The importance of observing the vibration of structures such as lines or cables has been well recognized. In areas such as civil engineering, efforts have been devoted to this topic. Magalhães et al. [MCC08] measured the vibration of a suspended roof of a football stadium which was connected by cables. Seismographs were placed on the roof in order to obtain readings of the roof's vibrating characteristics. Calçada et al. [CACD05] also proposed to measure semi-rigid line vibrations, obtaining the corresponding vibration parameters of the cables from a cable-stayed bridge, using load cells previously installed in the inferior anchorages of the stay cables. High-precision GPS has also been used to measure vibrations in structures by tracking its position over time and obtaining the displacements. In [ÇS02], the authors used GPS antennas, at a sample rate

of 10-20 Hz to track the position of a tall building. GPS devices were also used in [RMM⁺03] to obtain the displacement attributes of a bridge.

Although the above mentioned techniques have proved successful and can be used in different scenarios, they can easily become very expensive. Moreover, their application to structures like cables and power lines may not be feasible. The use of cameras and CV techniques to provide a noninvasive solution may help to overcome obstacles of such scenarios.

Line detection specifically targeted for video sequences was the subject of [CT05], which described an algorithm for field line detection in football videos. This method relied on the specifics of the scenario. Rodrigo et al. [RSS06] proposed an energy minimization scheme for line detection which makes use of the Hough transform [SS01].

More recently, a line detector was proposed by Cardoso et al. [CCRG08] with an application to staff line detection in handwritten music scores [CCR⁺09]. The image is treated as a graph, establishing the shortest path between left and right margins of the image resorting to the paths with less cost. This method was able to handle lines with curvature and overcome existing gaps in the line.

In [Gam89] and [DF90] the authors intended to obtain characteristics present in a space in order to develop an algorithm for a robot's path. To this end, they resorted to the edges on the walls, furniture and other objects present in the robot's visual space. Gamboto [Gam89] used information about the camera translation movement and the geometry of the lines to match them between frames, while also considering possible omissions in-between frames and splitting of some lines. The approach of Deriche and Faugeras [DF90] consisted of a prediction and a matching step after an initial identification of the edges. A Kalman filter was used to predict the position in the next frame followed by a matching, within the predicted region, using a normalized distance.

Common applications of CV techniques to obtain the characteristics of a vibrating line follow one of two approaches: those that require the placement of a marker on the structure with the algorithm detecting and tracking the marker throughout the sequence; those that require the user to choose one specific point, or a region of interest (ROI), in an image that will then be tracked throughout the sequence of frames. The tracking itself is typically accomplished through pattern matching or through the use of optical flow algorithms [HS81, LK81] and motion vector measures to determine displacements and vibration frequencies.

The most immediate solution for tracking a structure's displacement using video is to place a marker on the exact point desired to track. Providing a reasonable resolution for the output of the camera is available, a basic pattern recognition algorithm will then identify the marker throughout the video sequence [Ola99]. The knowledge of the marker's position allows the algorithm to compute its displacement over the sequence of frames. In [WCM03] several markers were placed on the structure to be tracked and were also used for calibrating the camera. Although this method was originally used to track only bridge decks, one can assume the premises can be applied to different types of structures like cables in cable-stayed bridges or general lines with vibrating characteristics. Nonetheless, these techniques raise similar problems as the previously discussed approaches - although they can present a less expensive approach when compared to the use of

accelerometers or load cells, it may not be possible to obtain an overall characterization of the line's vibration and the placement of markers on the structure can prove to be difficult or even impossible.

In [SBC07], the authors used the information of a ROI, defined by the user, to obtain a more complete overview of the line's displacement. It computes the optical flow method described in [HS81], for each pixel inside the ROI, which allows a more robust measurement of the vibration.

B.3 Line Detection Using a Shortest Path Approach

A vibrating line can be considered as a connected path between the two lateral margins of the image (for simplicity, the presentation will be oriented for horizontal lines only; the necessary adaptations for the detection of vertical lines should be clear at the end). As vibrating lines are almost the only extensive objects on the image, lines can then be found among the shortest paths between the two margins of the image if paths through high gradient positions are favored. Vibrating lines are then best modeled as paths between two regions Ω_1 and Ω_2 , the left and right margins of the image. These same ideas have been successfully applied to other applications [OC09, CCR⁺09, CSTC09, CCRG08].

B.3.1 Shortest Path Method Description

The method considers the image as a graph, where pixels are the nodes and a cost is associated with each edge connecting 8-neighborhood pixels.

One may assume that vibrating lines do not zigzag back and forth, left and right. Therefore, one may restrict the search among connected paths containing one, and only one, pixel in each column of the image¹. Formally, let I be an $N_1 \times N_2$ image and define an admissible line to be

$$\mathbf{s} = \{(x, y(x))\}_{x=1}^{N_1}, \text{ s.t. } \forall x |y(x) - y(x-1)| \leq 1, \quad (\text{B.1})$$

where y is a mapping $y: [1, \dots, N_1] \rightarrow [1, \dots, N_2]$. That is, a vibrating line is an 8-connected path of pixels in the image from left to right, containing one, and only one, pixel in each column of the image.

Given the weight function $w(p, q)$, defined over neighboring pixels p and q , the cost of a line can be defined as $C(\mathbf{s}) = \sum_{i=2}^{N_1} w(v_{i-1}, v_i)$, where v_{i-1} and v_i are consecutive pixels along the path as defined in Equation B.1. The optimal vibrating line that minimizes this cost can be found using dynamic programming. The first step is to traverse the image from the second column to the last column and compute the cumulative minimum cost C for all possible connected vibrating lines for each entry (i, j) :

¹These assumptions, 8-connectivity and one pixel per column, impose a maximum detectable 45 rotation degree.

$$C(i, j) = \min \begin{cases} C(i-1, j-1) + w(p_{i-1, j-1}; p_{i, j}) \\ C(i-1, j) + w(p_{i-1, j}; p_{i, j}) \\ C(i-1, j+1) + w(p_{i-1, j+1}; p_{i, j}) \end{cases}, \quad (\text{B.2})$$

where $w(p_{i, j}; p_{l, m})$ represents the weight of the edge incident with pixels at positions (i, j) and (l, m) . At the end of this process,

$$\min_{j \in \{1, \dots, N_2\}} C(N_1, j)$$

indicates the end of the minimal connected line. Hence, in the second step, one backtrack from this minimum entry on C to find the path of the optimal line.

Assume one wants to find all vibrating lines present in an image. This can be approached by successively finding and erasing the shortest path from the left to the right margin of the image. The erase operation is required to ensure that a line is not detected multiple times.

B.3.1.1 Stopping Rule

To stop the iterative vibrating lines search, a sequence of (arguably) sensible rules is used to validate the path found; if it does not pass the checking, the iterative search is stopped. The adopted validation rule assesses the quality of the detected path. A path is discarded and the iterative procedure is stopped if its overall cost is above a pre-specified threshold.

A semi-automated approach would be to receive from the user the number of lines to be detected and tracked. Since this could be done once for the entire sequence, it is a very reasonable alternative.

B.3.1.2 Design of the Weight Function

The weight function on the edges was defined so that the shortest path corresponds to a path that maximises the amount of edge strength in the image along the contour. An immediate approach is to support the design of the weight function solely on the values of the incident nodes, fixing the weight of an edge as a monotonically decreasing function of the average gradient value of the incident pixels. A straightforward solution is to use a linear dependency of the weight on the gradient. Previous similar works [CSTC09, CCR⁺09, CCRG08, SCdCC08] suggest that the introduction of a ‘saturation effect’ on the weight improves the performance.

Therefore, and although a more general setting could have been adopted, the weight of the edge connecting 4-neighbour pixels was expressed as an exponential law

$$f(g) = \alpha \exp(\beta (255 - g)) + \gamma, \quad (\text{B.3})$$

with $\alpha, \beta, \gamma \in \mathbb{R}$ and g is the average of the gradient computed on the two incident pixels. For 8-neighbour pixels the weight was set to $\sqrt{2}$ times that value [SCdCC08]. The values of α, β and γ were also set as suggested in [SCdCC08].

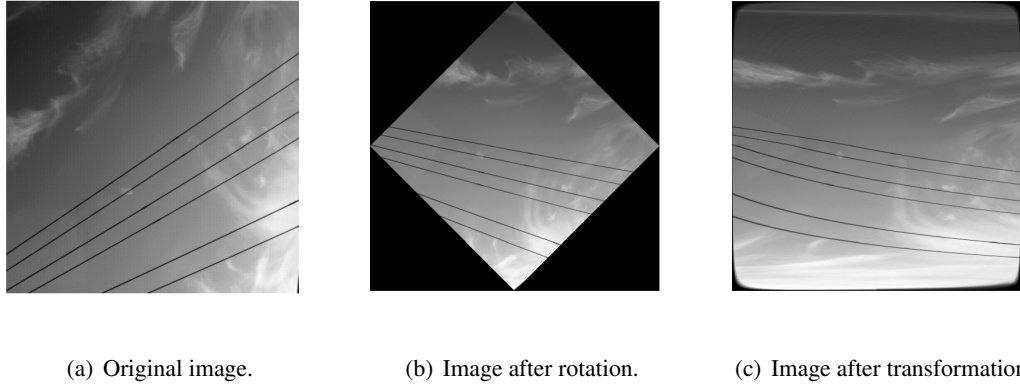


Figure B.1: Image pre-processing (from [CPCCR12]).

B.3.2 Detecting Vibrating Lines in Different Positions

It should be obvious now how to adapt the detection technique for lines between the upper and lower margins. It is also possible to apply the detection technique to adjacent margins. Suppose that the endpoints of the vibrating lines are in the left and bottom margins. A difficulty with searching for the shortest paths (shortest in the sense on minimizing the cost of the path) between the bottom row and left column is that small paths, near the bottom-left corner are naturally favoured. To overcome this challenge, we propose to pre-process the image, adopting the polar coordinates. The bottom row is mapped to the right column (corresponding to an angle of $\pi/2$ rads) and the left column stays in the same position. In this new coordinate-system, the path to search for is now between the left and right columns.

A different setting is when one of the endpoints of the vibrating lines is on the left OR the bottom margins and the other endpoint is in the top OR right margins (see Figure B.1). It is still possible to pre-process the image with an appropriate transform before the application of the proposed detection technique.

The goal of the transformation is to place the endpoints in opposing margins. Although different solutions exist, we started by rotating the image (Figure B.1(b)); then, we linearly scaled each row such that the 45° lines were transformed to horizontal lines margins (Figure B.1(c)).

Finally, it is worth mentioning that the vibrating line detection can be applied in a user defined region in the image (and in the video sequence). That can be important when other extensive objects are visible in the image.

B.4 Line Tracking

Many of the existing methods for vibrating line tracking make use of optical flow [HS81] to compute the displacement vector of one or more points of the line to be tracked. The typical lack of texture information of the line makes tracking methods that require this information less

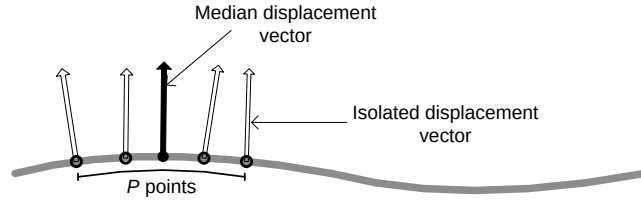


Figure B.2: Use of multiple displacement vectors in a neighborhood of the point of interest to minimize the tracking error (from [CPCCR12]).

suitable for the scenarios being considered. Optical flow has limitations concerning occlusion and deformation of the objects, but given the characteristics of these scenarios one can assume that such restrictions are respected. Nevertheless, the aperture problem associated with optical flow is present. The possible lack of information regarding the extremities and the colour homogeneity of the line makes it harder to reliably compute the tangential component of the displacement using optical flow. We propose the use of information from line detection to overcome this shortcoming.

We begin by describing an optical flow based method that uses information from line detection to minimize tracking errors (Subsection B.4.1). In Subsection B.4.2 it is proposed a structure based approach to track individual points without making use of optical flow or similar methods. Finally, in Subsection B.4.3 it is proposed the use of the Dynamic Time Warping algorithm to define point correspondences.

Throughout the remaining of this chapter we will use the term ‘point’ to refer to points of the line unless stated otherwise.

B.4.1 A Windowed Optical Flow Approach

This first approach makes use of optical flow, but we argue that using more points and their corresponding displacements vectors can minimize the tracking errors, while also contributing to a more complete characterisation of the line behavior.

The results obtained using optical flow to compute the displacement vector for points of the line are affected by several factors, but in particular by the aperture problem. Rather than considering each point isolated from the others, we take an interval of P points centered at a point of interest, i.e., a point to be tracked. The displacement vectors for every point in the interval are computed and the median vector assigned to the interest point (see Figure B.2), thus contributing to the canceling of errors in the tangential component. We argue that with such a formulation, the effect of the aperture problem can be minimized. This process can be repeated for an arbitrary number of points, which may be manually specified by a user or defined according to some criteria.

B.4.2 A Line Structure Based Approach

In scenarios as those being considered, the thickness of the lines in the image is very small and may be of a single pixel. When computing the optical flow for a point of the line, we are essentially looking at the surrounding information. Since neighbour points of the line will have similar colour information, the background will affect the matching process. The approach proposed in subsection B.4.1 intends to mitigate the effect of the aperture problem, but because it is based on the computation of optical flow it is still affected by it.

While the physical characteristics of the scenarios enable the identification of a line throughout the sequence, the definition of correspondences between points of the line in consecutive frames is a harder task.

We propose an alternative method to establish correspondences between line points without making use of optical flow or similar methods. It is assumed that the extremities of the line are static, i.e., fixed at physical support points. Moreover, it is important to note that points or parts of the line do not switch position.

For the following description it is assumed that the lines in the current frame have been detected and paired with the corresponding line in the previous frame - we are interested in tracking individual points of the line. Different approaches may be defined for pairing the lines; it is not within the scope of this thesis to propose a specific solution. Here, for simplicity, we followed the approach described in Subsection B.5.2 which is based on computing the distance between each possible pair of lines and applying the Hungarian algorithm [Kuh55, Kuh56] for assignment. Up to this point no assumptions were made about individual point correspondences.

We begin by considering that lines, or more precisely the image representation of the lines, are of constant size or uniformly extensible. Under this condition, the relative distance of a point to a fixed origin (e.g., the left extremity of the line) remains constant throughout the video sequence. Therefore, once the line is detected in each frame it is possible to establish point correspondences in consecutive frames (see Figure B.3).

To deal with the compression or extensibility of the line, we explore the relative positioning of the points in the identification of correspondences between frames: the distance of each point of the line to the origin, normalized by the length of the line in the image, is considered to be constant. This concept is illustrated in Figure B.3 and conveyed by Equation (B.4) where i is a point over the line L which has length l and $\Delta_i^{(t)}$ is the distance of i to the origin in frame t . Thus, the correspondence between each point i of the line in frame t and frame $t - 1$ is obtained through Equation (B.5). Note that if the length of the line remains constant the distance of each point to the extremity in different frames remains the same.

$$\frac{\Delta_i^{(t-1)}}{l^{(t-1)}} = \frac{\Delta_i^{(t)}}{l^{(t)}}, \forall i \in L, \quad (\text{B.4})$$

$$\Delta_i^{(t)} = \frac{l^{(t)}}{l^{(t-1)}} \cdot \Delta_i^{(t-1)}, \forall i \in L, \quad (\text{B.5})$$

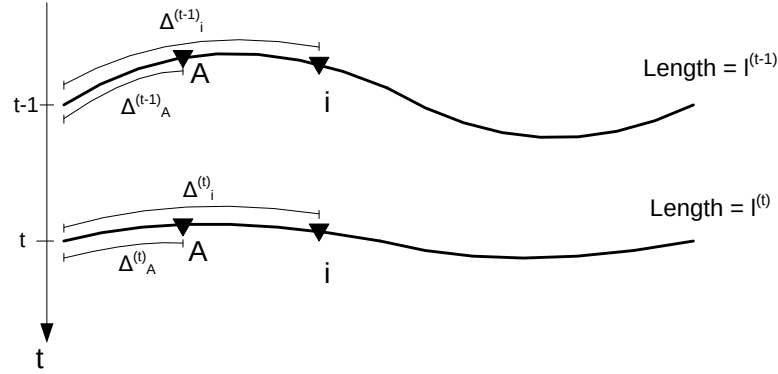


Figure B.3: Illustration of the relative positioning concept for tracking points over the line (from [CPCCR12]).

For the application of this method the line is considered inextensible or extending/compressing uniformly. However, this constraint may not always be satisfied; for example, due to the elasticity properties of the materials.

B.4.3 A Dynamic Time Warping Approach

The method described in this Subsection makes use of the same assumptions and reasoning of the structure based approach, but without the requirement of uniformity in the extension or compression of the line. For this purpose we discuss the use of DTW to define the point correspondences.

Dynamic Time Warping (DTW) is a well established algorithm for the comparison of time series and has been successfully applied to problems such as speech and handwriting recognition [SC90, NV05]. DTW computes the optimal match between two sequences where sections of the sequences can be stretched or compressed. It uses a matrix containing the distance of every point of a series to every point in the other. The minimum accumulated distance is computed over the matrix and backtracking allows the identification of the correspondences.

As before, we assume a fixed origin (e.g., the left most pixel). For the application of the DTW method we consider the line similar to a time series, but as a function of the position with regards to the origin rather than a function of time.

Similarly to the previous approach, we assume that the lines have been detected and paired. For each pair we apply the DTW algorithm to determine the optimal match between points of the line at time instants t and $t - 1$. To fill the dissimilarity matrix it is necessary to define how the distance between two points is computed. Although many different distance measures can be defined, we used the normalized cross correlation coefficient and the euclidean distance normalized by the image diagonal. The first measures the dissimilarity of the pattern in a 3×3 neighborhood of the points being matched, while the second measures the distance of a point to its possible previous position. The final distance is obtained through a weighted average of the individual distances as defined by Equation (B.6)

$$d = \alpha \times d_e + (1 - \alpha) \times d_c, \quad (\text{B.6})$$

were α is a weight factor; d_e and d_c are the normalized euclidean distance and normalized cross correlation distance respectively. The latter is defined has $1 - \text{correlation_coefficient}$. Different values of α were tested.

B.5 Validation of the proposed approach

This section describes the methodology used to assess the methods described in the previous Sections. The dataset used for validation is described in Subsection B.5.1. The evaluation is also two fold. First, the use of the shortest path method to detect the lines is assessed. Second, the tracking results are evaluated. The methodologies are described in Subsection B.5.2 and Subsection B.5.3 respectively.

B.5.1 Dataset

The difficulty in obtaining real sequences, depicting the characteristics of the scenarios, and the corresponding reference information motivated the generation of a set of synthetic sequences; it also enabled the exploration of a wide set of factors and conditions. The synthetic sequences consist of multiple lines, each vibrating with a given frequency and maximum amplitude. Different characteristics for the lines were simulated, namely: thickness; frequency; amplitude; curvature radius; grey level value. A real background was added to better approximate a real situation. Moreover, the use of synthetic sequences aided the generation of point reference information for validation.

Figure B.4 depicts representative frames of the dataset. Sequences 1 and 2 feature six curved lines each vibrating at a given frequency and with the lines in sequence 1 thinner than those in sequence 2. Sequences 3 and 4 feature six straight lines vibrating with different characteristics and with the lines of sequence 3 thinner than those of sequence 4. An example of a frame of these sequences is presented in Figure B.4(a). Sequence 5 also consists of straight lines, but a very ‘noisy’ background was added (see Figure B.4(b)) to augment the difficulty in detection and tracking. Sequences 6 and 7 consist of 3 curved lines with the extremities fixed at the margins of the image.

Sequences 1 through 7 provide a good set to test line detection. The background of sequence 5 is noisier and the colour of the lines less distinct from the background to augment the difficulty in the processing. Due to the formulation used in the generation of the lines, only sequences 3 through 7 possess ground truth for assessing tracking results, i.e., only these have reference information for every point of each line.

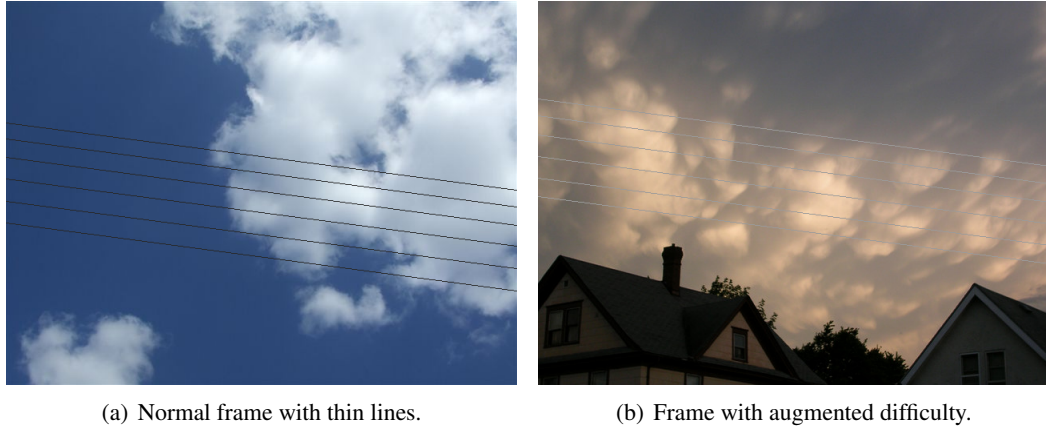


Figure B.4: Examples of images from the dataset (from [CPCCR12]).

B.5.2 Line Detection Assessment

The evaluation of the line detection approach was accomplished by comparing the detection results with the corresponding reference information, i.e., it uses the distance between reference and detected lines. Hence, we must first define these pairs. For each possible pair of reference and detected lines, the distance is computed and matches determined using the Hungarian algorithm, thus identifying the pairs. In the distance calculation, for each point in the reference line, the distance to each point in the detected line is computed and two overall measures are taken: the average and the Hausdorff distance. For line pairing the average distance was used. For the assessment of the results both the average and Hausdorff distances were considered. The measures for an image with multiple lines are the average of the displacements of the individual lines. Similarly, for a sequence the distances are averaged. Following this procedure we compared the method based on the shortest path algorithm with the use of the Hough transform [SS01].

B.5.3 Tracking Assessment

We divided the tracking evaluation in two parts. First, we compare the two methods that use optical flow: a common method that computes the displacement vector at a marked point; the proposed window based method. For the latter, different numbers of points of the line (window size) were considered. The results were compared with the reference information by measuring the euclidean distance. The distance measures obtained were averaged over the lines and subsequently the frames.

For the optical flow method we have considered the following algorithms: Lucas-Kanade [LK81]; Horn-Schunck [HS81]; pyramidal implementation of the Lucas-Kanade [Bou00]. Experiments were performed to assess the algorithm most adequate to the scenarios considered. Each algorithm was applied to sequences containing vibrating lines and compared to the corresponding

reference information. For each line, the average of the root mean square error (RMSE) for each point was computed followed by the average over the lines and subsequently over the frames.

The second set of experiments compares the windowed optical flow, structure based and DTW based methods. For each frame, the tracking result for each point of the line was compared with the expected position and the RMSE computed. For each line, the average and maximum error was calculated; these were averaged over the lines and frames.

For the windowed optical flow method we compared the estimated position of a point P in frame t with the corresponding reference information. In the two other methods, matches are established between points of detected lines; to separate the tracking and detection error, the points' estimated position is compared with the expected position over the detected line. Hence, the detection error is subtracted from the tracking error so that it is not accounted twice.

B.6 Results

B.6.1 Line Detection

As previously described, the shortest path method was applied over the gradient of the image. Experiments in the gradient calculation using different colour spaces for the input images were conducted. It was verified that the use of the grayscale space was not adequate causing large gaps in the edges corresponding to the lines, as depicted in Figure B.5(a). Such gaps can cause malfunctions in line detection methods and consequently prevent the proper detection of the lines present. The best results were obtained using the HSV colour space. In particular the Saturation channel provided robust information and was consequently used as input to the shortest path method. Figure B.5(b) and Figure B.5(c) depict results for the edge processing using the YCbCr and HSV colour spaces respectively.

Table B.1 presents the line detection results for the shortest path and Hough transform methods. As stated the average and Hausdorff distance were used to measure the distance between detected and reference line. The measures were normalized by the image diagonal. The results show that the shortest path method outperforms the Hough transform. Even though the average distance for sequence 4 is the same for the two methods, the maximum error is significantly higher for the Hough transform. For sequence 3 the higher error values for the Hough transform were caused by difficulties in edge detection. Moreover, it is noticeable an increase of the error for the Hough transform in sequences 6 and 7 due to the curvature of the lines. For these same two sequences, the error for the shortest path method varies slightly and the average distance is of the same order as for the other sequences.

B.6.2 Line Tracking

We begin by presenting the results of the comparison of different optical flow methods, namely: Lucas-Kanade; Horn-Schunck; Pyramidal Lucas-Kanade. The average of the RMSE for the results of each method applied to the dataset is summarized in Table B.2.

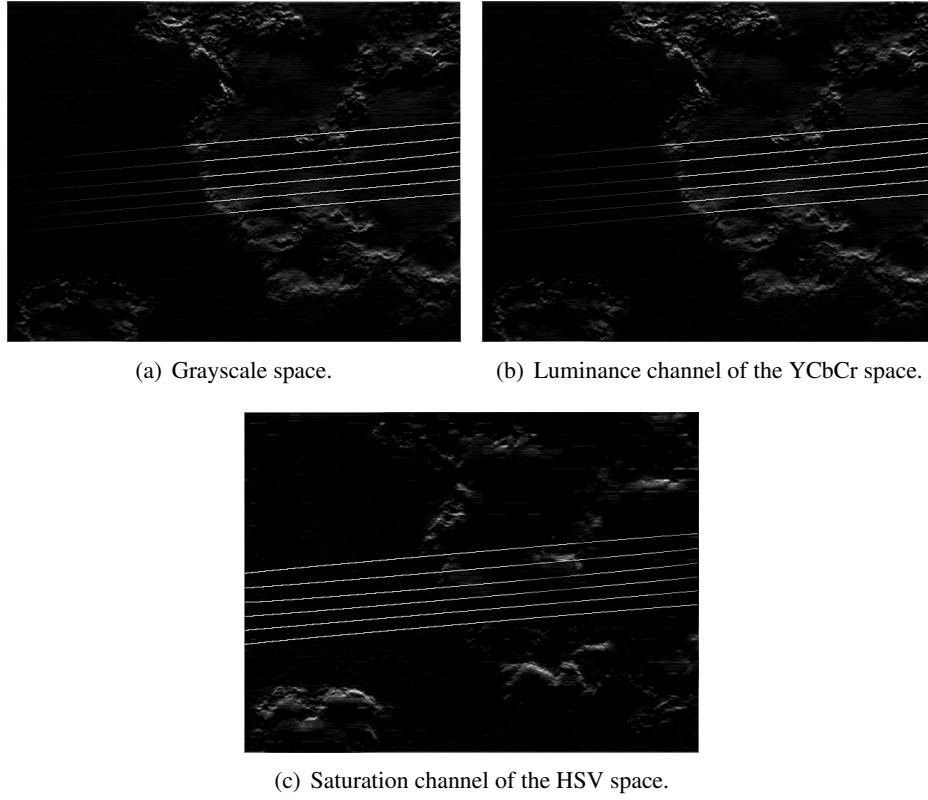


Figure B.5: Various spaces of colours applied to a synthetic frame, followed by a gradient filter (from [CPCCR12]).

Table B.1: Results for line detection normalized by the image diagonal (from [CPCCR12]).

	Hough Transf.		Shortest Path	
	Average Distance	Hausdorff Distance	Average Distance	Hausdorff Distance
Seq. 1	0.40%	10.90%	0.13%	0.35%
Seq. 2	0.81%	13.25%	0.10%	0.38%
Seq. 3	7.23%	35.20%	0.13%	0.28%
Seq. 4	0.20%	6.71%	0.20%	0.40%
Seq. 5	1.16%	11.07%	0.33%	2.15%
Seq. 6	1.87%	24.37%	0.13%	0.28%
Seq. 7	4.49%	25.52%	0.16%	0.51%

Table B.2: Comparison of optical flow methods (from [CPCCR12]).

Method	Average RMSE
Lucas-Kanade	2.196
Horn-Schunck	2.162
Pyramidal Lucas-Kanade	0.390

In our experiments the Pyramidal Lucas-Kanade method outperformed the other two. Hence, this was the method used in the subsequent experiments.

The results of the comparison of the optical flow based methods, using a window over the line or a single point, are summarized in Table B.3. Three sequences with 6 lines each were considered. For each line in a sequence a subset of points was randomly selected; in each frame the root mean square error relatively to the reference point at time instant t was calculated. The average RMSE over the lines is indicated for the individual components and overall displacement. Tracking using 1 point corresponds to using a state-of-the-art tracking algorithm with the user marking the interest point in the image. For the approach described in this thesis, windows with 3 and 11 points in length were considered. Moreover, the error gain in percentage relatively to the 1 pixel approach was calculated. In these experiments the RMSE decreased with the increase of the window size. Nevertheless, care must be taken in these situations: when the lines present a high degree of curvature, taking information from a large number of points may induce errors. A considerable part of the improvements in the line tracking is due to a better approximation of the horizontal component. In the scenarios considered, due to the characteristics of the lines, the aperture problem will exert a stronger effect on the horizontal component. For different line positions the changes are straightforward. By considering information from more points over the line, a decrease in the RMSE is visible. Moreover, the frequencies of vibrations were also computed and compared to the values used in the generation of synthetic sequences. The reference and obtained frequency for each line in the considered sequences are presented in Table B.4. The difference between the values is very small and can be considered negligible.

The results for the second set of experiments are summarized in Table B.5. For each point's position estimation, the RMSE was calculated and the average and maximum error per line deter-

Table B.3: Results for line tracking considering a single point and the window-based approach (from [CPCCR12]).

		Window Size		
		1 points	3 points	11 points
	Direction	RMSE	Error gain relatively to using 1 point (%)	
Seq. 3	horizontal	1.22	-2.45	-13.72
	vertical	0.79	0.03	4.32
	total	1.73	-1.39	-7.46
Seq. 4	horizontal	1.85	-6.34	1.05
	vertical	0.46	-2.65	-12.98
	total	1.94	-5.97	-0.42
Seq. 5	horizontal	0.89	-0.60	-4.45
	vertical	1.16	-0.13	0.93
	total	1.80	0.22	-0.79

Table B.4: Frequency outputs (in rad/s) from sequences 3, 4 and 5 (from [CPCCR11]).

Seq. 3		Seq. 4		Seq. 5	
Reference	Obtained	Reference	Obtained	Reference	Obtained
0.800	0.800	0.800	0.801	0.800	0.800
0.850	0.848	0.850	0.848	0.300	0.300
0.900	0.901	0.900	0.902	0.200	0.200
0.950	0.951	0.950	0.950	0.950	0.950
1.000	1.000	0.050	0.050	0.100	0.100
1.050	1.051	1.500	1.502	1.500	1.501

mined. These were averaged over the lines and frames. The table presents results for the windowed optical flow, structure based and DTW based methods. We tested several values for the weight of Equation B.6, but the final tracking error was the same.

From the data in Table B.5 we can conclude that the structure and DTW based methods outperform the optical flow based one. Moreover, the best results were obtained by the DTW based method, which enabled a correct identification of point correspondences in consecutive frames and consequently the tracking of the individual points. The computational cost of the DTW based method is higher than the structure based one, but it provides more reliable results.

Although the tracking error of the DTW based method appears independent of the weights of Equation B.6, it is important to understand the impact of the two individual measures considered in the experiments. To assist this analysis, Figure B.6 depicts examples of an image representation of the dissimilarity matrix for some of the combinations. In this representation, white represents the maximum distance and black a zero distance. The optimal correspondence determined by the DTW algorithm is represented by the yellow/red line that crosses the image from the upper left to the lower right corner.

Note that even though the tracking error for the DTW based method is approximately zero, the overall error must include the line detection error. Nevertheless, the detection results show that the corresponding values are small.

B.7 Discussion

The use of the shortest path method enabled the automatic detection of vibrating lines, straight or curved, to be tracked with small errors. It was demonstrated that the shortest path outperformed a state-of-the-art method and that it can be applied to detect civil engineering structures, specifically lines or cables, in images or video sequences. Errors in the extraction of gradient information

Table B.5: RMSE for tracking results (from [CPCCR12]).

[illegible]

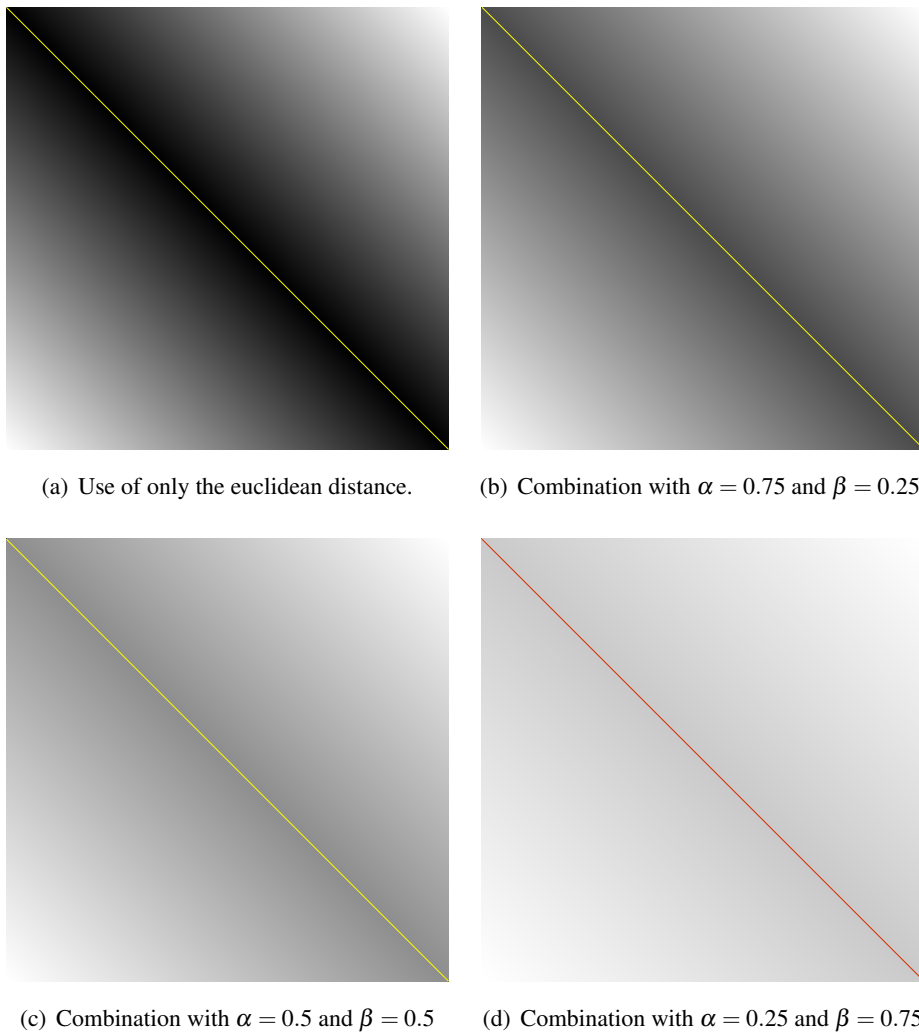


Figure B.6: Image representations of the dissimilarity matrix for several combinations of the euclidean distances and normalized cross correlation distance as defined by Equation B.6. Darker colours represent lower dissimilarities. The red/yellow line represents the optimal assignment resulting from the DTW algorithm (from [CPCCR12]).

influenced the detection of the lines. Hence, it is expectable that a better computation of the gradient, possibly in conjunction with other pre-processing operations, will contribute to a decrease of the error.

The ability to automatically detect the lines eliminates the requirement of manual initialisation of the points to be tracked. Moreover, it provides valuable information that can be used to improve line/point tracking. The results obtained show that by using a set of points of the line centered at a point to be tracked it is possible to reduce the tracking errors due to the optical flow computation; in particular, it helps to minimize the effect of the aperture problem in these scenarios.

The use of optical flow to compute displacements implies that information from the background will be used due to the reduced thickness of the lines. Hence, it is arguable the use of such information. The proposed strategies for the structure and DTW based methods to establish point correspondences, and subsequently track the line, differ from the strategy of the windowed optical flow. The latter takes information of a point of the line and its neighbors, and searches for a match in subsequent frames. The former methods take full advantage of line detection. The lines are first detected in consecutive frames and point correspondences identified over the detected lines. It was demonstrated that these methods, under the constraints of the target scenarios, can provide more accurate results than those using optical flow. In particular, the results show that once instances of the line in consecutive frames have been detected, the DTW based method correctly identifies point correspondences. Consequently, optical flow associated errors can be avoided.

Although it was not considered in this work, the application of the proposed method to the monitoring of civil engineering structures can take advantage of the knowledge of the structure dimensions to perform camera calibration and thus obtain information about the 3D measures of the displacement.

In many of the scenarios considered, the camera must be placed far from the structure to capture a typically large area of interest, if not the complete structure. This constraint usually implies that the lines or cables appear in the image with very small dimensions and their movements may be negligible (e.g., a pixel at most). It would be interesting to consider the use of more than one camera, installed closer to the structure and focusing on different parts of interest, and combine the images to achieve the overall tracking.

Future work will necessarily include the use of real sequences and the validation of the method with data captured from other devices such as accelerometers. Furthermore, attention must be given to the computation and analysis of the lines' vibration frequencies.

The work on vibrating line detection and tracking resulted in the following contributions:

- P. Carvalho, M. Pinheiro, J. S. Cardoso, L. Corte Real, "A Detection Based Approach For Tracking Vibrating Lines in Video Sequences", *Pattern Analysis and Applications*. (submitted)
- P. Carvalho, M. Pinheiro, J. S. Cardoso, L. Corte Real, "A Shortest Path Approach for Vibrating Line Detection and Tracking", *Proceedings of Iberian Conference on Pattern Recognition and Image Analysis 2011 (IbPRIA 2011)*, pp. 9-16, 2011.

References

- [AC97] Jake K. Aggarwal and Qin Cai. Human motion analysis: a review. In *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*, pages 90–102, June 1997.
- [AC99] Jake K. Aggarwal and Qin Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428–440, 1999.
- [ACLS94] Jake K. Aggarwal, Qin Cai, Wen Liao, and Bikash Sabata. Articulated and elastic non-rigid motion: a review. In *Motion of Non-Rigid and Articulated Objects, 1994., Proceedings of the 1994 IEEE Workshop on*, pages 2–14, November 1994.
- [AF99] Anthony Almudevar and Chris Field. Estimation of single generation sibling relationships based on DNA markers. *Journal Agricultural, Biological and environment statistics*, 4:136–165, 1999.
- [AR11] Jake R. Aggarwal and Michael S. Ryoo. Human activity analysis: A review. *ACM Comput. Surv.*, 43:16:1–16:43, April 2011.
- [AVBK10] Alexandre Alahi, Pierre Vanderghenst, Michel Bierlaire, and Murat Kunt. Cascade of descriptors to detect and track objects across any network of cameras. *Computer Vision and Image Understanding*, 114:624–640, June 2010.
- [Avi07] Shai Avidan. Ensemble tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:261–271, February 2007.
- [Bas08] Rafael Bastos. *FIRST, Invariant Image Features For Augmented Reality And Computer Vision*. PhD thesis, Instituto Superior de Ciências do Trabalho e da Empresa, 2008.
- [Bau95] Adam Baumberg. *Learning Deformable Models for Tracking Human Motion*. PhD thesis, School of Computer Studies, University of Leeds, October 1995.
- [BD09] R. Bastos and M. S. Dias. FIRST - Fast Invariant to Rotation and Scale Transform: Invariant image features for augmented reality and computer vision. In *VDM Verlag*, 2009.
- [BER03] James Black, Tim Ellis, and Paul Rosin. A novel method for video tracking performance evaluation. In *In Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, pages 125–132, 2003.
- [BETVG08] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110:346–359, June 2008.

- [BL02] Matthew Brown and David Lowe. Invariant Features from Interest Point Groups. In *In British Machine Vision Conference*, pages 656–665, 2002.
- [Bou00] Jean Y. Bouguet. *Pyramidal Implementation of the Lucas Kanade Feature Tracker: Description of the algorithm*. Microprocessor Research Labs, Intel Corporation, 2000.
- [BP06] Faisal Bashir and Fatih Porikli. Performance evaluation of object detection and tracking systems. In *Proceedings of IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS), PETS 2006.*, June 2006.
- [Bra98] Gary R. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, (Q2), 1998.
- [CA99] Qin Cai and Jake K. Aggarwal. Tracking human motion in structured environments using a distributed-camera system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11):1241–1247, November 1999.
- [CACD05] Rui Calçada, Álvaro Cunha, and Rosário Delgado. Analysis of traffic induced vibrations in a cable-stayed bridge. *Journal of Bridge Engineering, ASCE*, 10(4):370–385, 2005.
- [Cav04] Caviar. EC-Funded-CAVIAR-project, I. 2001-37540, 2004.
- [CBK09] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41:15:1–15:58, July 2009.
- [CCCR10] Pedro Carvalho, Jaime S. Cardoso, and Luís Corte-Real. Hybrid framework for evaluating video object tracking algorithms. *Electronics Letters*, 46(6):411–412, 2010.
- [CCCR11] Pedro Carvalho, Jaime S. Cardoso, and Luís Corte-Real. Filling the gap in quality assessment of video object tracking. *Image and Vision Computing*, 2011. (submitted).
- [CCR05] Jaime S. Cardoso and Luís Corte-Real. Toward a generic evaluation of image segmentation. *IEEE Transactions on Image Processing*, 14(11):1773–1782, November 2005.
- [CCR⁺09] Jaime S. Cardoso, Artur Capela, Ana Rebelo, Carlos Guedes, and Joaquim F. Pinto da Costa. Staff detection with stable paths. *IEEE Transactions Pattern Analysis Machine Intelligence*, 31:1134–1139, 2009.
- [CCRG08] Jaime S. Cardoso, Artur Capela, Ana Rebelo, and Carlos Guedes. A connected path approach for staff detection on a music score. In *Proceedings of the International Conference on Image Processing (ICIP 2008)*, pages 1005–1008, 2008.
- [CCTCR09] Jaime S. Cardoso, Pedro Carvalho, Luís F. Teixeira, and Luís Corte-Real. Partition-distance methods for assessing spatial segmentations of images and videos. *Computer Vision and Image Understanding*, 113(7):811–823, 2009.
- [cEEST04] Çiğdem Eroğlu Erdem, B. Sankur, and A. Murat Tekalp. Performance measures for video object segmentation and tracking. *IEEE Transactions on Image Processing*, 13(7):937–951, 2004.

- [Cha91] Eugene Charniak. Bayesian networks without tears: making bayesian networks more accessible to the probabilistically unsophisticated. *Artificial Intelligence Magazine*, 12(4):50–63, 1991.
- [COC⁺12] Pedro Carvalho, Telmo Oliveira, Lucian Ciobanu, Filipe Gaspar, Luís F. Teixeira, Rafael Bastos, Miguel Sales Dias, Jaime S. Cardoso, and Luís Côte-Real. Analysis of object description methods in a video object tracking environment. *Machine Vision and Applications*, 2012. (submitted).
- [CP03] Paulo L. Correia and Fernando M. Pereira. Objective evaluation of video segmentation quality. *IEEE Transactions on Image Processing*, 12(2):186–200, 2003.
- [CPCCR11] Pedro Carvalho, Miguel Pinheiro, Jaime S. Cardoso, and Luís Corte-Real. A shortest path approach for vibrating line detection and tracking. In *Proceedings of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2011)*, pages 9–16, 2011.
- [CPCCR12] Pedro Carvalho, Miguel Pinheiro, Jaime S. Cardoso, and Luís Corte-Real. A detection based approach for tracking vibrating lines in video sequences. *Pattern Analysis and Applications*, 2012. (submitted).
- [CS95] Claudette Cédras and Mubarak Shah. Motion-based recognition: a survey. *Image and Vision Computing*, 13(2):129–155, 1995.
- [ÇS02] Mehmet Çelebi and Ahmet Sanli. GPS in pioneering dynamic monitoring of long-period structures. *Earthquake Spectra*, 18(1):47–61, 2002.
- [CSE05] Andrea Cavallaro, Olivier Steiger, and Touradj Ebrahimi. Tracking video objects in cluttered background. *IEEE Transactions on Circuits and Systems for Video Technology*, 15:575–584, 2005.
- [CSTC09] Jaime S. Cardoso, Ricardo Sousa, Luis F. Teixeira, and Maria J. Cardoso. Breast contour detection with stable paths. In A. Fred, J. Filipe, and H. Gamboa, editors, *Biomedical Engineering Systems and Technologies*, volume 25 of *Communications in Computer and Information Science*, pages 439–452, 2009.
- [CSWY09] Peng Cui, Li-Feng Sun, Fei Wang, and Shi-Qiang Yang. Contextual mixture tracking. *IEEE Transactions on Multimedia*, 11:333–341, February 2009.
- [CT05] Z.Q. Cai and J. Tai. Line detection in soccer video. In *Fifth International Conference on Information, Communications and Signal Processing*, pages 538–541, 2005.
- [CZ05] Andrea Cavallaro and Francesco Ziliani. Characterisation of tracking performance. In *Proceedings of Workshop on Image Analysis For Multimedia Interactive Services (WIAMIS)*, Montreux, Switzerland, 13–15 April 2005.
- [DCS05] Simon Denman, Vinod Chandran, and Sridha Sridharan. Tracking people in 3D using position, size and shape. In *IEEE Proceedings of 8th International Symposium on Signal Processing and Its Applications*, volume 2, pages 611–614, August 2005.
- [Del02] Frank Dellaert. The expectation maximization algorithm. Technical Report GIT-GVU-02-20, College of Computing, Georgia Institute of Technology, 2002.

- [DF90] Rachid Deriche and Olivier D. Faugeras. Tracking line segments. In *In ECCV 90: Proceedings of the first european conference on Computer Vision*, pages 259–268, 1990.
- [DFSL09] Simon Denman, Clinton Fookes, Sridha Sridharan, and Ruan Lakemond. Dynamic performance measures for object tracking systems. In *Proceedings of the 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS '09*, pages 541–546, Washington, DC, USA, 2009. IEEE Computer Society.
- [DM00] David Doennann and David Mihalcik. Tools and techniques for video performance evaluation. In *Proceedings of the International Conference on Pattern Recognition (ICPR'00)*, pages 4167–4170, 2000.
- [dNM06] Jacinto Carlos Marques Peixoto do Nascimento and Jorge Salvador Marques. Performance evaluation of object detection algorithms for video surveillance. *IEEE Transactions on Multimedia*, 8(4):761–774, April 2006.
- [DSR10] Alessio Dore, Mauricio Soto, and Carlo S. Regazzoni. Bayesian tracking for video analytics. *IEEE Signal Processing Magazine*, 27(5):46–55, 2010.
- [DT05] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society.
- [Ell02] Tim Ellis. Performance metrics and methods for tracking in surveillance. In *3rd IEEE International Workshop on Performance Evaluation of Tracking and Surveillance PETS'2002*, Copenhagen, Denmark, June 2002.
- [FP02] David A. Forsyth and Jean Ponce. *Computer Vision: A Modern Approach*. Prentice Hall, August 2002.
- [Fri79] Alinda Friedman. Framing pictures: The role of knowledge in automatized encoding and memory for gist. *Journal of Experimental Psychology: General*, 108:316–355, 1979.
- [FS97] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.
- [Gam89] Jean Pierre Gambotto. Tracking points and line segments in image sequences. *Workshop on Visual Motion*, pages 38–45, March 1989.
- [Gav99] Darius M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding: CVIU*, 73(1):82–98, 1999.
- [GB08] Andrew Gilbert and Richard Bowden. Incremental, scalable tracking of objects inter camera. *Computer Vision and Image Understanding*, 111(1):43–58, July 2008.
- [GE09] Elisa Drelie Gelasca and Touradj Ebrahimi. On evaluating video object segmentation quality: A perceptually driven objective metric. *Selected Topics in Signal Processing, IEEE Journal of*, 3(2):319–335, April 2009.

- [GG09] Murat Gevrekci and Bahadır K. Gunturk. Illumination robust interest point detection. *Computer Vision and Image Understanding*, 113(4):565–571, April 2009.
- [GPBM07] Jesús García, Oscar Pérez, Antonio Berlanga, and José M. Molina. Video tracking system optimization using evolution strategies. *International Journal of Imaging Systems and Technology*, 17(2):75–90, 2007.
- [Gus02] Dan Gusfield. Partition distance: a problem and class of perfect graphs arising in clustering. *Information Processing Letters*, 82:159–164, May 2002.
- [HHD99] Ismail Haritaoglu, David Harwood, and Larry S. Davis. Hydra: multiple people detection and tracking using silhouettes. In *Proceedings of International Conference on Image Analysis and Processing*, 1999, pages 280–285, 1999.
- [HHD00] Ismail Haritaoglu, David Harwood, and Larry S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, August 2000.
- [HS81] Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [HTWM04] Weiming Hu, Tieniu Tan, Liang Wang, and S. Maybank. A survey on visual surveillance of object motion and behaviors. *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, 34(3):334–352, 2004.
- [HYJ11] Zhenjun Han, Qixiang Ye, and Jianbin Jiao. Combined feature evaluation for adaptive visual object tracking. *Computer Vision and Image Understanding*, 115:69–80, January 2011.
- [IB98] Michael Isard and Andrew Blake. Condensation – conditional density propagation for visual tracking. *International Journal of computer Vision*, 29(1), 1998.
- [JHM⁺10] Zhengqiang Jiang, Du Q. Huynh, William Moran, Subhash Challa, and Nick Spadaccini. Multiple pedestrian tracking using colour and motion models. In *Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications*, DICTA '10, pages 328–334, Washington, DC, USA, 2010. IEEE Computer Society.
- [JMIB08] Xiaoyi Jiang, Cyril Marti, Christophe Irniger, and Horst Bunke. Distance measures for image segmentation evaluation. *EURASIP Journal on Applied Signal Processing*, 2006(1):1–10, 2008.
- [JSRM10] Julio Cezar Silveira Jacques Junior and Cláudio Rosito Jung Soraia Raupp Musse. Crowd analysis using computer vision techniques. *IEEE Signal Processing Magazine*, 27(5):66–77, 2010.
- [JWSX02] Christopher Jaynes, Stephen Webb, R. Matt Steele, and Quanren Xiong. An open development environment for evaluation of video surveillance systems. In *Proceedings of the Third International Workshop on Performance Evaluation of Tracking and Surveillance (PETS'2002)*, pages 32–39, 2002.
- [JYNH09] Yu-Gang Jiang, Jun Yang, Chong-Wah Ngo, and Alexander G. Hauptmann. Representations of Keypoint-Based semantic concept detection: A comprehensive study. *Multimedia, IEEE Transactions on*, 12(1):42–53, November 2009.

- [Kal60] Rudolf E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, (82 (Series D)):35–45, 1960.
- [KC10] Dimitrios Kosmopoulos and Sotirios P. Chatzis. Robust visual behavior recognition. *IEEE Signal Processing Magazine*, 27(5):34–45, 2010.
- [KF09] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.
- [KK10] Bahadır Karasulu and Serdar Korukoglu. A software for performance evaluation and comparison of people detection and tracking methods in video processing. *Multi-media Tools and Applications*, pages 1–47, September 2010.
- [KRJ⁺08] Prahlad Kilambi, Evan Ribnick, Ajay J. Joshi, Osama Masoud, and Nikolaos Papanikolopoulos. Estimating pedestrian counts in groups. *Computer Vision and Image Understanding*, 110(1):43–59, April 2008.
- [KS04] Yan Ke and Rahul Sukthankar. PCA-SIFT: a more distinctive representation for local image descriptors. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR’04*, pages 506–513, Washington, DC, USA, 2004. IEEE Computer Society.
- [KST⁺05] M. Krinidis, G. Stamou, H. Teutsch, S. Spors, N. Nikolaidis, R. Rabenstein, and I. Pitas. An audio-visual database for evaluating person tracking algorithms. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP ’05). IEEE International Conference on*, pages 237–240, 2005.
- [Kuh55] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*, 2:83–97, 1955.
- [Kuh56] H. W. Kuhn. Variants of the Hungarian method for assignment problems. *Naval Research Logistics Quarterly*, 3(4):253–258, 1956.
- [Lab09] Living Usability Lab. QREN funded project, 15/SI/2009, n° 7900, 2009.
- [LF04] Thor List and Robert B. Fisher. CVML - an XML-based computer vision markup language. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR’04) Volume 1 - Volume 01*, ICPR ’04, pages 789–792, Washington, DC, USA, 2004. IEEE Computer Society.
- [LK81] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.
- [LMRMJ08] N. Lazarevic-McManus, J. R. Renno, D. Makris, and G. A. Jones. An object-based comparative methodology for motion detection based on the F-Measure. *Computer Vision and Image Understanding*, 111(1):74–85, Jul 2008.
- [Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

- [LYZ⁺09] Hong Liu, Ze Yu, Hongbin Zha, Yuexian Zou, and Lin Zhang. Robust human tracking based on multi-cue integration and mean-shift. *Pattern Recognition Letters*, 30, July 2009.
- [Mar82] David Marr. *Vision: a computational investigation into the human representation and processing of visual information*. W. H. Freeman, San Francisco, 1982.
- [Mar03] David Martin. *An Empirical Approach to Grouping and Segmentation*. PhD thesis, UC Berkeley, 2003.
- [May79] Peter S. Maybeck. *Stochastic models, estimation, and control*, volume 141 of *Mathematics in Science and Engineering*. 1979.
- [MCC08] F. Magalhães, E. Caetano, and A. Cunha. Operational model analysis and finite element correlation of the Braga stadium suspended roof. *Engineering Structures*, 30:1688–1698, 2008.
- [MG01] Thomas B. Moeslund and Erik Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81:231–268, 2001.
- [MHK06] Thomas B. Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2):90–126, 2006.
- [MN02] Michael McCahill and Clive Norris. CCTV in london. *Report to the European Commission Fifth Framework RTD as part of UrbanEye: on the threshold of the urban panopticon*, 2002.
- [MRF10] C. Micheloni, B. Rinner, and G.L. Foresti. Video analysis in pan-tilt-zoom camera networks. *Signal Processing Magazine, IEEE*, 27(5):78–90, Sep. 2010.
- [MS03] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–257 – II–263, June 2003.
- [MS04] Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60:63–86, October 2004.
- [MS05] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(10):1615–1630, 2005.
- [NBTv07] A. T. Nghiem, F. Bremond, M. Thonnat, and V. Valentin. Etiseo, performance evaluation for video surveillance systems. In *Proceedings of the 2007 IEEE Conference on Advanced Video and Signal Based Surveillance*, pages 476–481, Washington, DC, USA, 2007. IEEE Computer Society.
- [NJT06] Eric Nowak, Frédéric Jurie, and Bill Triggs. Sampling strategies for bag-of-features image classification. In *European Conference on Computer Vision*. Springer, 2006.
- [NS06] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2161–2168, Washington, DC, USA, 2006. IEEE Computer Society.

- [NV05] Ralph Niels and Louis Vuurpij. Dynamic time warping applied to Tamil character recognition. *Document Analysis and Recognition, International Conference on*, 0:730–734, 2005.
- [OB80] J. O’Rourke and N.I. Badler. Model-based image analysis of human motion using constraint propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(6):522–536, November 1980.
- [OC09] Hélder Oliveira and Jaime S. Cardoso. Image retargeting using stable paths. In *Proceedings of the Fourth International Conference on Computer Vision Theory and Applications*, volume 2, pages 40–47, 2009.
- [Ola99] P. Olaszek. Investigation of the dynamic characteristic of bridge structures using a computer vision method. *Measurement*, 25:227–236, 1999.
- [Oli05] Aude Oliva. Gist of the scene. In L. Itti, G. Rees, and J. K. Tsotsos, editors, *The Encyclopedia of Neurobiology of Attention*, pages 251–256. Elsevier, San Diego, CA, 2005.
- [OPZ06] Andreas Opelt, Axel Pinz, and Andrew Zisserman. Incremental learning of object detectors using a visual shape alphabet. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 1:3–10, 2006.
- [PA10] Vasilis Papadourakis and Antonis Argyros. Multiple objects tracking in the presence of long-term occlusions. *Computer Vision and Image Understanding*, 114:835–846, July 2010.
- [PET06] PETS. IEEE international workshop on performance evaluation of tracking and surveillance 2006, 2006.
- [Pop07] Ronald Poppe. Vision-based human motion analysis: An overview. *Computer Vision and Image Understanding*, 108(1-2):4–18, 2007. Special Issue on Vision for Human-Computer Interaction.
- [PS96] Sarma (Gopal) Pingali and Jakub Segen. Performance evaluation of people tracking systems. In *WACV ’96: Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV ’96)*, page 33, Washington, DC, USA, 1996. IEEE Computer Society.
- [Rab89] Lawrence R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.
- [RCW⁺10] Carlo S. Regazzoni, Andre Cavallaro, Ying Wu, Janusz Konrad, and Arun Hampapur. Video analytics for surveillance: Theory and practice. *Signal Processing Magazine, IEEE*, 27(5):16–17, September 2010.
- [RKMR⁺08] Daniel Roth, Esther Koller-Meier, Daniel Rowe, Thomas B. Moeslund, and Luc Van Gool. Event-based tracking evaluation metric. In *Proceedings of IEEE Workshop on Motion and video Computing, WMVC 2008.*, pages 1–8, January 2008.
- [RMM⁺03] G. Roberts, X. Meng, M. Meo, A. Dodson, E. Cosser, E. Iuliano, and A. Morris. A remote bridge health monitoring system using computational simulation and GPS

- sensor data. In *Proceedings, 11th FIG Symposium on Deformation Measurements*, 2003.
- [ROB09] ROBVIGIL. QREN funded project, 15/SI/2009, n° 7950, 2009.
- [RSS06] Ranga Rodrigo, Wenxia Shi, and Jagath Samarabandu. Energy based line detection. In *Canadian Conference on Electrical and Computer Engineering*, pages 2061–2064, 2006.
- [RW08a] Bernhard Rinner and Wayne Wolf. A bright future for distributed smart cameras. *Proceedings of the IEEE*, 96(10):1562–1564, 2008.
- [RW08b] Bernhard Rinner and Wayne Wolf. An introduction to distributed smart cameras. *Proceedings of the IEEE*, 96(10):1565–1575, 2008.
- [SBC07] S. Silva, J. Bateira, and E. Caetano. Development of a vision system for vibration analysis. In *2nd Int. Conf. on Experimental Vibration Analysis for Civil Engineering Structures*, 2007.
- [SBWB04] Thomas Schlogl, Csaba Beleznai, Martin Winter, and Horst Bischof. Performance evaluation metrics for motion detection and tracking. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4*, pages 519–522, Washington, DC, USA, 2004. IEEE Computer Society.
- [SC90] Hiroaki Sakoe and Seibi Chiba. Readings in speech recognition. chapter Dynamic programming algorithm optimization for spoken word recognition, pages 159–165. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990.
- [SCdCC08] Ricardo Sousa, Jaime S. Cardoso, Joaquim F. Pinto da Costa, and Maria J. Cardoso. Breast contour detection with shape priors. In *2008 15th IEEE International Conference on Image Processing, Vols 1-5*, IEEE International Conference on Image Processing (ICIP), pages 1440–1443, 2008.
- [Shl05] Jonathon Shlens. A tutorial on Principal Component Analysis, December 2005.
- [SHY10] S. M. Nejhum Shahed, Jeffrey Ho, and Ming-Hsuan Yang. Online visual tracking with histograms and articulating blocks. *Computer Vision and Image Understanding*, 114(8):901–914, 2010.
- [Sie03] Nils T. Siebel. *Design and Implementation of People Tracking algorithms for Visual Surveillance Applications*. PhD thesis, University of Reading, Mar 2003.
- [SKJ10] Venkatesh Saligrama, Janusz Konrad, and Pierre-Marc Jodoin. Video anomaly identification: A statistical approach. *IEEE Signal Processing Magazine*, 27(5):18–33, September 2010.
- [SM02] Nils T. Siebel and Steve J. Maybank. Fusion of multiple tracking algorithm for robust people tracking. In *Proceedings of European Conference on Computer Vision*, pages 373–387, 2002.
- [SMB00] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of Computer Vision*, 37:151–172, June 2000.

- [SMC05] Lauro Snidaro, Christian Micheloni, and Cristian Chiavedale. Video security for ambient intelligence. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 35(1):133–144, Jan 2005.
- [Smi07] Kevin Charles Smith. *Bayesian Methods for Visual Multi-Object Tracking with Applications to Human Activity Recognition*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2007.
- [SRE⁺05] Josef Sivic, Bryan C. Russell, Alexei A. Efros, Andrew Zisserman, and William T. Freeman. Discovering objects and their localization in images. In *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 - Volume 01*, pages 370–377, Washington, DC, USA, 2005. IEEE Computer Society.
- [SS01] Linda G. Shapiro and George C. Stockman. *Computer Vision*. Prentice Hall, 2001.
- [SY99] Kingsley Sage and Stewart Young. Security applications of computer vision. *IEEE Aerospace and Electronic System Magazine*, 14(4):19–29, Apr 1999.
- [SZ03] Josef Sivic and Andrew Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, October 2003.
- [Sze10] Richard Szeliski. *Computer Vision : Algorithms and Applications*. Springer-Verlag New York Inc, 2010.
- [TC02] Dennis Tell and Stefan Carlsson. Combining appearance and topology for wide baseline matching. In *Proceedings of the 7th European Conference on Computer Vision-Part I, ECCV '02*, pages 68–81, London, UK, 2002. Springer-Verlag.
- [TCCR07] Luís F. Teixeira, Jaime S. Cardoso, and Luís Corte-Real. Object segmentation using background modelling and cascaded change detection. *Journal of Multimedia (JMM)*, 2:55–65, 2007.
- [TCJ⁺08] George Tzanetakis, Carlos Castillo, Randy Jones, Luís G. Martins, Luís F. Teixeira, and Mathieu Lagrange. Interoperability and the MARSYAS 0.2 runtime. In *Proc. International Computer Music Conference (ICMC)*, Belfast, Northern Ireland, August 2008.
- [TCR09] Luis F. Teixeira and Luís Corte-Real. Video object matching across multiple independent views using local descriptors and adaptative learning. *Pattern Recognition Letters*, 30:157–167, January 2009.
- [Tei09] Luís F. Teixeira. *Contributions for the automatic description of multimodal scenes*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2009.
- [TM08] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Foundation and Trends in Computer Graphics and Vision*, 3:177–280, July 2008.
- [TMLT08] Luis F. Teixeira, Luis G. Martins, Mathieu Lagrange, and George Tzanetakis. MarsyasX: multimedia dataflow processing with implicit patching. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pages 873–876. ACM, 2008.

- [TS04] Prithiraj Tissainayagam and David Suter. Assessing the performance of corner detectors for point feature tracking applications. *Image and Vision Computing*, 22:663–679, 2004.
- [TT05] Feng Tang and Hai Tao. Object tracking with dynamic feature graph. In *ICCCN '05: Proceedings of the 14th International Conference on Computer Communications and Networks*, pages 25–32, 2005.
- [VC10] Roberto Vezzani and Rita Cucchiara. Video surveillance online repository (ViSOR): an integrated framework. *Multimedia Tools and Applications*, 50(2):359–380, 2010.
- [VD10] Péter L. Venetianer and Hongli Deng. Performance evaluation of an intelligent video surveillance system - a case study. *Computer Vision and Image Understanding*, 114:1292–1302, November 2010.
- [ViP05] ViPER. ViPER toolkit at sourceforge, 2005.
- [Vla08] James Vlahos. Surveillance society: New high-tech cameras are watching you. *Popular Mechanics*, pages 64–69, January 2008.
- [WADP97] Cristopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.
- [WB95] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, 1995.
- [WCM03] A.M. Wahbeh, J.P. Caffrey, and S.F. Masri. A vision-based approach for the direct measurement of displacements in vibrating systems. *Smart Materials and Structures*, 12(5):785–794, 2003.
- [WHL⁺09] Lei Wu, Yang Hu, Mingjing Li, Nenghai Yu, and Xian-Sheng Hua. Scale-invariant visual language modeling for object categorization. *IEEE Transactions on Multimedia*, 11:286–294, February 2009.
- [WHT03] Liang Wang, Weiming Hu, and Tieniu Tan. Recent developments in human motion analysis. *Pattern Recognition*, 36(3):585–601, 2003.
- [YJS06] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object Tracking: A Survey. *ACM Computing Surveys (CSUR)*, 38(4):13, Dec 2006.
- [Zha04] Tao Zhao. *Model-Based Segmentation and Tracking of Multiple Humans in Complex Situations*. PhD thesis, Faculty of the Graduate School of the University of Southern California, 2004.
- [ZMR⁺08] Beibei Zhan, Dorothy N. Monekosso, Paolo Remagnino, Sergio A. Velastin, and Li-Qun Xu. Crowd analysis: a survey. *Machine Vision and Applications*, 19:345–357, September 2008.
- [ZN04] Tao Zhao and Ram Nevatia. Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1208–1211, Sep 2004.

- [ZWW10] Jie Zhou, Dingrui Wan, and Ying Wu. The chameleon-like vision system. *IEEE Signal Processing Magazine*, 27(5):91–101, 2010.
- [ZYS09] Huiyu Zhou, Yuan Yuan, and Chunmei Shi. Object tracking using sift features and mean shift. *Computer Vision and Image Understanding*, 113:345–352, March 2009.