Cast shadow modelling and detection

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/7848

Publisher: © Njad Al-Najdawi

Please cite the published version.
This item is held in Loughborough University’s Institutional Repository (https://dspace.lboro.ac.uk/) and was harvested from the British Library’s EThOS service (http://www.ethos.bl.uk/). It is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to: http://creativecommons.org/licenses/by-nc-nd/2.5/
Cast Shadow Modelling and Detection

by

Njad Al-Najdawi

A doctoral thesis submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science
Loughborough University
September 2006

© by Njad Al-Najdawi 2006
Supervisors: Dr. Helmut Bez and Dr. Eran Edirisinghe
Director of research: Professor Sameer Singh
BEST COPY

AVAILABLE
Cast Shadow Modelling and Detection

Njad Al-Najdawi, September 2006

Abstract

Computer vision applications are often confronted by the need to differentiate between objects and their shadows. A number of shadow detection algorithms have been proposed in literature, based on physical, geometrical, and other heuristic techniques. While most of these existing approaches are dependent on the scene environments and object types, the ones that are not, are classified as superior to others conceptually and in terms of accuracy. Despite these efforts, the design of a generic, accurate, simple, and efficient shadow detection algorithm still remains an open problem. In this thesis, based on a physically-derived hypothesis for shadow identification, novel, multi-domain shadow detection algorithms are proposed and tested in the spatial and transform domains.

A novel “Affine Shadow Test Hypothesis” has been proposed, derived, and validated across multiple environments. Based on that, several new shadow detection algorithms have been proposed and modelled for short-duration video sequences, where a background frame is available as a reliable reference, and for long duration video sequences, where the use of a dedicated background frame is unreliable. Finally, additional algorithms have been proposed to detect shadows in still images, where the use of a separate background frame is not possible. In this approach, the author shows that the proposed algorithms are capable of detecting cast, and self shadows simultaneously.

All proposed algorithms have been modelled, and tested to detect shadows in the spatial (pixel) and transform (frequency) domains and are compared against state-of-art approaches, using popular test and novel videos, covering a wide range of test conditions. It is shown that the proposed algorithms outperform most existing methods and effectively detect different types of shadows under various lighting and environmental conditions.
Contents

Title Page ........................................... i
Abstract ............................................. ii
Table of Contents .................................... iii
List of Figures ....................................... vi
List of Tables ....................................... xi
Citations to Published Work ....................... xii
Acknowledgments .................................... xiii
Dedication .......................................... xiv

1 An Overview ....................................... 1
  1.1 Introduction .................................... 1
    1.1.1 What is a Shadow and a Cast Shadow ............. 2
  1.2 Applications .................................... 3
  1.3 Motivations for the Research ...................... 3
  1.4 Research Objectives and Contributions of the Thesis ... 4
    1.4.1 Objectives .................................. 4
    1.4.2 Contributions of the research .................. 4
  1.5 Organization of the Thesis ....................... 6

2 Literature Review .................................. 8
  2.1 Overview ....................................... 8
  2.2 Introduction .................................... 9
  2.3 Moving Cast Shadow Detection Approaches .......... 11
    2.3.1 Object independent and environment dependent approaches . 11
    2.3.2 Object independent and environment independent approaches 19
    2.3.3 Object dependent and environment dependent approaches . 25
  2.4 Summary ....................................... 27

3 Shadow Hypothesis ............................... 29
  3.1 Overview ....................................... 29
  3.2 The Affine Shadow Hypothesis .................... 30
  3.3 The Derivation of the Affine Shadow Hypothesis ...... 32
### Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3.1 Shadows cast by a single point source</td>
<td>32</td>
</tr>
<tr>
<td>3.3.2 The neighbourhood approximation</td>
<td>37</td>
</tr>
<tr>
<td>3.3.3 Shadows cast by two point sources</td>
<td>37</td>
</tr>
<tr>
<td>3.4 Reflection Modelling</td>
<td>41</td>
</tr>
<tr>
<td>3.5 RGB Space</td>
<td>43</td>
</tr>
<tr>
<td>3.6 Summary</td>
<td>44</td>
</tr>
<tr>
<td>4 Illumination and Shadow Models - Theoretical Comparisons</td>
<td>45</td>
</tr>
<tr>
<td>4.1 Overview</td>
<td>45</td>
</tr>
<tr>
<td>4.2 Illumination and Shadow Modelling</td>
<td>46</td>
</tr>
<tr>
<td>4.2.1 Using a single point source</td>
<td>46</td>
</tr>
<tr>
<td>4.2.2 Two point sources</td>
<td>48</td>
</tr>
<tr>
<td>4.3 The Proposed Illumination and Shadow Model</td>
<td>50</td>
</tr>
<tr>
<td>4.4 The Affine Hypothesis - Verification</td>
<td>51</td>
</tr>
<tr>
<td>4.5 Summary</td>
<td>52</td>
</tr>
<tr>
<td>5 Spatial and Transform Domains Shadow Detection Algorithms</td>
<td>54</td>
</tr>
<tr>
<td>5.1 Overview</td>
<td>54</td>
</tr>
<tr>
<td>5.2 Pixel domain</td>
<td>55</td>
</tr>
<tr>
<td>5.2.1 Pixel domain shadow detection</td>
<td>57</td>
</tr>
<tr>
<td>5.3 The Transform Domains</td>
<td>62</td>
</tr>
<tr>
<td>5.3.1 The Discrete Cosine Transform (DCT)</td>
<td>62</td>
</tr>
<tr>
<td>5.3.2 Fourier domain shadow detection</td>
<td>64</td>
</tr>
<tr>
<td>5.3.3 The Discrete Wavelet Transform (DWT)</td>
<td>68</td>
</tr>
<tr>
<td>5.3.4 Wavelet domain shadow detection</td>
<td>70</td>
</tr>
<tr>
<td>5.4 Summary</td>
<td>73</td>
</tr>
<tr>
<td>6 Shadow Detection in Video Sequences</td>
<td>76</td>
</tr>
<tr>
<td>6.1 Overview</td>
<td>76</td>
</tr>
<tr>
<td>6.2 Introduction</td>
<td>77</td>
</tr>
<tr>
<td>6.3 Shadow Detection in Video Sequences with Dedicated Background Image</td>
<td>79</td>
</tr>
<tr>
<td>6.3.1 Performance evaluation metrics</td>
<td>85</td>
</tr>
<tr>
<td>6.4 Shadow Detection in Video Sequences with a Non-Dedicated Background Image</td>
<td>89</td>
</tr>
<tr>
<td>6.4.1 Initialisation</td>
<td>92</td>
</tr>
<tr>
<td>6.4.2 Locating and extracting objects</td>
<td>92</td>
</tr>
<tr>
<td>6.4.3 Calculating average object velocities</td>
<td>93</td>
</tr>
<tr>
<td>6.4.4 Evaluation and discussion</td>
<td>94</td>
</tr>
<tr>
<td>6.5 Summary</td>
<td>96</td>
</tr>
</tbody>
</table>
Contents

7 Shadow Boundary Detection in Still Images .......................... 98
  7.1 Overview ........................................................................ 98
  7.2 Shadow Edge Detection Algorithms .................................. 99
    7.2.1 Edge detection ....................................................... 101
    7.2.2 Spatial domain shadow edge detection algorithm ............. 102
    7.2.3 Transform domain shadow edge detection algorithms ...... 106
    7.2.4 Subjective evaluation .............................................. 109
  7.3 Summary ...................................................................... 111

8 Conclusion and Future Work ............................................... 113

Bibliography ......................................................................... 118

A Taxonomy of Moving Cast Shadow Detection Algorithms ........ 126

B Additional Experimental Results ......................................... 129
List of Figures

1.1 Self shadow, and cast shadow (umbra and penumbra).......................... 2
2.1 Proposed taxonomy for shadow detection algorithms.......................... 10
3.1 Cast shadow umbra and penumbra [27]........................................... 30
3.2 Illuminated ‘floor’ $F$, point $q$ receives a direct ray from $p$............. 33
3.3 Same environment as figure 3.2 with object in presence, point $q$ receives
no light from $p$, therefore is completely dark.................................. 34
3.4 A simple ‘background’ environment illuminated by a single point light
source............................................................................... 35
3.5 Light received at $q$ with object present........................................ 36
3.6 A simple ‘background’ environment illuminated by two point light
sources............................................................................... 38
3.7 Point $q$ in umbra region receives reflected light from both sources. No
direct light received at this point.................................................. 39
3.8 Point $q$ in penumbra region receives direct light from one point source,
and reflected light from both sources.............................................. 40
3.9 A neighbourhood $U_q$, and two subsets $V$ and $V^*$ assumed to determine
pixel values in the captured image.................................................. 42
3.10 Shadows can be identified in colour images by applying the affine test
in each colour layer and taking the intersection area.......................... 43
5.1 A digital image sampled using (from top to bottom and left to right)
128 $\times$ 128, 64 $\times$ 64, 32 $\times$ 32 and 16 $\times$ 16 pixels........................... 57
5.2 A 128 $\times$ 128 pixels image with luminance quantization using (from top
to bottom and left to right) 8, 4, 3 and 1 bits.................................. 58
5.3 Frames $F_{bg}$ (background frame) and $F_{obj}$ (object frame), are used for
the detection of shadows in the object frame $F_{obj}$.............................. 59
5.4 Three-levels DWT decomposition for an image.................................. 69
5.5 Three-level wavelet decomposition of a 8 $\times$ 8 pixel block..................... 70
6.1 The benchmark “Hall” video sequence, represents an indoor video sequence, with multiple combinations of light sources, spectrally equal and of equal intensities. This figure illustrates the use of the background and object frame to detect shadows. In the processed frame, shadows have been removed using the pixel domain shadow detection and removal algorithm.

6.2 “Side” video sequence, represents an indoor video sequence, with arbitrary combination of light sources, with different intensities. This figure illustrates the use of the DCT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence, frames (A', B', C') are the corresponding processed frames.

6.3 Example of using different block sizes in the shadow detection and removal algorithm. The algorithm starts with a 16 x 16 block size down to size 3 x 3 as required.

6.4 “Sara”, represents an indoor video sequence, with multiple combination of light sources, spectrally equal and of equal intensities. This figure illustrates the use the DWT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence, frames (A', B', C') are the corresponding processed frames.

6.5 “Holywell”, represents an outdoor video sequence with strong sun light, and strong shadows cast on the road. The left hand side image represents the original image and the right hand side image is the corresponding processed one. This figure illustrates the use the pixel domain shadow detection and removal algorithm.

6.6 The benchmarks “laboratory”, “campus”, and “Intelligent room” video sequences. (a) frame number 223 of Laboratory video sequence, (a’) is the corresponding processed frame in the pixel domain, (b) is frame number 125 of Campus video sequence, (b’) is the corresponding processed frame in the DCT domain, (c) is frame number 94 of Intelligent room video sequence, (c’) is the corresponding processed frame in the DWT domain.

6.7 “Farnham” video sequence, (a) the background frame, (b) the current frame, (c) the binary image resulting from the background subtraction process, (d) the shadow detection frame, detected shadow blocks are presented as white blocks, (e) the shadow removed frame, detected shadow blocks are replaced with corresponding non-shadow blocks from frame a, (f) the binary image of the processed frame.

6.8 An exemplar shadow detection algorithm diagram for non-dedicated background algorithms.
6.9 "Corridor", represents an indoor video sequence, with multiple combination of light sources, spectrally equal and of equal intensities. This figure illustrates the use the background non-dedicated approach based on the DCT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence; frames (A', B', C') are the corresponding processed frames.

6.10 Extracting objects into different layers, in this examples 4 objects are extracted into 4 new layers.

7.1 Shadow boundary detection algorithm diagram.

7.2 Vertical and Horizontal intervals used to detect edges, \( P_1, P_2 \) belongs to one side of the edge, \( P'_1, P'_2 \) belongs to the other side of the edge.

7.3 A combination of vertical, horizontal, and Diagonal intervals used in the 1-D shadow boundary detection algorithms.

7.4 Edge detection using different detectors.

7.5 Shadow edge detection in the spatial domain. Still images are extracted from the following video sequences: "Holywell", "Black", and "Sara", respectively. The presented shadow edge detection figures show that the algorithm succeeds to detect self and cast shadows in the images.

7.6 Shadow edge detection in the transform domain. Still images are extracted from the following video sequences: "Holywell", "Black", and "Sara" respectively. The presented shadow edge detection figures show that the algorithm succeeds to detect self and cast shadows in the images. Results are relatively similar to the pixel domain shadow edge detection algorithm, with minor differences.

B.1 Indoor environment with single point source and neutral walls, 'Ghost' video sequence, frames: 50, 52, 56, 85. The frames shown in this figure are captured using a high resolution camera. The distance between the camera and the object is relatively small. In this sequence, the shadow is cast vertically on the wall and horizontally on a wooden floor. The cast shadow covers a wide area of the frame. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.2 Indoor environment with single point source and textured walls, 'Legs' video sequence, frames: 9, 11, 13, 15. The frames shown in this figure are captured using a high resolution camera. The distance between the camera and the object is relatively small. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
B.3 Indoor environment with multiple arbitrary combination of light sources, ‘Black’ video sequence, Frames: 21-24. The frames shown in this figure are captured using a low resolution camera. Multiple shadows in this sequence are cast vertically onto the wall. In this particular scene, due to the use of static a background subtraction technique and a change in illumination, it is clear in the output video that the brightness of the replaced blocks is slightly different from the brightness in the current frame. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.4 Indoor environment with multiple combination of light sources spectrally equal with different intensities, ‘Side’ video sequence, frames: 63-66. The frames shown in this figure were captured using a low resolution camera. Multiple shadows in this sequence are cast vertically onto the wall. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.5 Indoor environment with multiple combination of light sources spectrally equal with equal intensities, ‘Sara’ video sequence, frames: 35-38. The frames shown in this figure were captured using a high resolution camera. In this sequence the distance between the objects and the camera varies from relatively high to relatively low. The shadows are cast vertically onto the walls and horizontally onto the carpet. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.6 Indoor environment with multiple combination of light sources spectrally equal with equal intensities. The well-known ‘Hall’ video, frames: 31, 35, 42, 132. In this sequence the distance between the objects and the camera varies from relatively high to relatively low. The shadows are cast vertically onto the walls and horizontally onto the carpet. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.7 Outdoor in overcast condition, ‘Asphalt’ video sequence, frames: 25-28. The frames shown in this figure were captured using a low resolution camera. In this sequence, shadows are cast horizontally onto the asphalt street. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
List of Figures

B.8 Outdoor environment in sunlight, ‘Holywell’ video sequence, frames: 122, 125, 128, 131. The frames shown in this figure were captured using the low resolution camera. This sequence was captured in bright sunlight, casting strong shadows with well-defined edges onto the asphalt street. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.9 Outdoor at Dusk time without any artificial lights, ‘Farnham’ video sequence, frames: 8-11. The frames shown in this figure were captured using the high resolution camera. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.10 Outdoor at Dusk time with artificial lights, ‘Door’ video sequence, frames: 54-57. The frames shown in this figure were captured using a high resolution camera. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.

B.11 Outdoor video sequence at night with multiple street lights, ‘Falkner’ video sequence, frames: 21, 23, 25, 27. The frames shown in this figure were captured using a high resolution camera. In this sequence, shadows create a great challenge for the algorithm because of their darkness and lack of texture information. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
List of Tables

6.1 Benchmark video sequences, conclusion drawn from sources. 84
6.2 The new video sequences description. 86
6.3 Quantitative evaluation of the proposed and the state-of-art algorithms, based on the benchmark video sequences. The results of the first four algorithms are obtained from Prati et al. work [61]. Siala et al [72] provided their own results as shown in the fifth algorithm. 88
6.4 Quantitative evaluation of the proposed algorithms, based on the new video sequences. 89

A.1 A Four-Layer taxonomy, Independent/Dependent of object type, Independent/Dependent of the environment, Deterministic/Statistical. 127
A.2 Taxonomy based on the Domain -pixel (spatial) or transform (frequency) and colour spaces $RGB$, $HSV$, and $YUV$. 128
Citations to Published Work

The work presented in chapters 2 and 3, and some sections of chapter 5 and 6, have been submitted for journal publication, and is under review as follows:


The content of chapters 3, 4, and 5, has been published in the following paper:


Chapter 5 and some sections of chapters 2, 3 and 6 have been published in the following papers:


The work presented in chapters 7 and 3, is to be submitted as:

Completing this doctoral work has been a marvellous and often overwhelming experience. I have been extremely privileged to have undoubtedly the most intuitive, smart, and supportive supervisors anyone could ask for, the beautiful mind, “Dr. Helmut E. Bez”, and the beautiful soul, “Dr. Eran A. Edirisinghe”. Their enthusiasm and mission for providing ‘only high-quality work and not less’, has made a deep impression on me. I owe them lots of gratitude for their guidance. Helmut and Eran are as close as relatives and good friends to me. I am also grateful for my director of research “Professor Sameer Singh” for his support and guidance. It was a real pleasure getting to know all of them.

I am very grateful for my wife, Sara, for her love, patience, and persistence, and for my brother, Ashraf, for his support and care. One of the best experiences that we lived in this period, was the fact that the three of us are studying for our PhD degrees; my happiness won’t be completed until Sara and Ashraf successfully finish their degrees.

I feel a deep sense of gratitude for my father Amin, and my mother Najwa, who formed part of my vision, and taught me the good things that really matter in life. I am grateful for my lovely young sisters, Hazar, Manar, and Saba, for rendering me the sense and the value of brotherhood. I am glad to have them in my life.

This research has been supported and funded by “Al-Balqa Applied University”, I thank them for giving me the opportunity to pursue my higher degrees. I am also grateful for the Department of Computer Science at Loughborough University for providing me an excellent work environment during the past years. My colleagues from the Digital Imaging research group all gave me the feeling of being at home. Dhammike, Murali, Ken, Moi, Iffat, Alvin, Rupesh, many thanks for being my colleagues. Last but not least, the happy memory of my director of research late “Professor Ondrej Sykora” will be in my mind forever.

Njad
Dedicated to my father Amin,  
my mother Najwa,  
and my beautiful wife Sara.
Chapter 1

An Overview

1.1 Introduction

Computer vision can be considered as an immature and a diverse field of science. Although earlier work exists, it was not until the late 1970's that a more purposeful study of the field commenced. These studies originated from various other fields, and as a consequence, no standard formulation of the "computer vision problem", nor of the "computer vision solution" exists. However, there exists a profusion of methods for solving various definite computer vision tasks. These methods are often task specific and can hardly be generalised over a wide range of applications. Sub-domains of computer vision include object tracking, object recognition, scene reconstruction, event detection, learning, indexing, ego-motion, image restoration, et cetera [25]. The research carried out in this thesis is intended to serve the object tracking and recognition sub-domains, by proposing a solution for one of the major problems in these fields, "The existence of cast shadows".

In object tracking and recognition applications, cast shadows are considered as a major problem associated with segmenting and extracting moving objects, as cast shadows cause object merging, object loses, and alter object shape characteristics [61].
The following section introduces and illustrates the context of shadows and cast shadows before further applications are discussed.

1.1.1 What is a Shadow and a Cast Shadow

A shadow is a region of darkness where light is blocked. A shadow occurs when an object totally or partially occludes direct light from the light source. Generally, a shadow is divided into two parts: a self shadow, and a cast shadow. A self shadow is the part of the shadow on the main object, that is not illuminated by light. A cast shadow is the part of a shadow on the background of the scene. A cast shadow is further sub-divided into umbra and penumbra regions [61], see Figure 1.1.

![Figure 1.1: Self shadow, and cast shadow (umbra and penumbra).](image)

The umbra (Latin: "shadow") is the darkest part of a shadow. Within the umbra, the light source is completely blocked by the object casting the shadow [27]. The penumbra (Latin: "almost-shadow") is that part of the shadow where the light source is partially blocked. Penumbras occur only when the light source is not a point-source [28].
1.2 Applications

Shadow detection can be useful in many applications, including: video coding, image processing and computer graphics. The following are some examples of applications where shadow detection is of high importance.

Object Recognition

The term recognition is used to refer to many different visual abilities, including identification, categorisation, and discrimination. Recognising an object means that the object has been successfully categorised as an instance of a particular object class [52]. Shadows are a major problem associated with object recognition, for the reason that shadows have the same motion as the objects casting them, and shadow points are detectable as foreground points since they typically differ significantly from the background. For these reasons, in object recognition, shadow identification is critical for both image sequences (video) and still images [61].

Object Tracking

Automated video surveillance systems require some mechanism to track interesting objects in the field of view of the sensor. In object tracking, cast shadows might be classified as other objects due to the fact that objects and their shadows have similar visual characteristics. Shadows may result in object merging, and shape alteration, which may cause significant confusion to the tracking system [61].

1.3 Motivations for the Research

Computer vision applications are challenged by the need for a generic, simple and accurate model that is capable of detecting shadows' umbra and penumbra in indoor and outdoor scene environments with domain independence.

Motivated by these facts, several spatial and transform domain shadow detection algorithms are proposed in this thesis. These algorithms are supported by a novel physically-derived shadow test hypothesis, i.e. "The Affine Shadow Test Hypothesis",
that is capable of detecting several types of shadows in different environments.

1.4 Research Objectives and Contributions of the Thesis

The general research aim is to "Design and implement a novel, generic, simple, and accurate model for cast shadow detection".

1.4.1 Objectives

Specific research objectives are as follows:

- to investigate the available illumination and reflection model hypotheses and propose modifications, where applicable,
- to investigate the relationship between shadow and non-shadow points and to extract a relationship (the Affine relationship),
- to investigate the best possible methods for calculating the affine parameters in various spatial and transform domains,
- to design and implement shadow detection algorithms for video sequences,
- to design and implement shadow detection algorithms for video sequences with a non-dedicated background image, and
- to design and implement shadow detection algorithms for still images that are capable of detecting self and cast shadows.

1.4.2 Contributions of the research

In this research, the following original contributions have been made. The resulting conference and journal papers are included in the list of citations.
Chapter 1: An Overview

1. Introducing a novel illumination and reflection model hypothesis

In this thesis, the author proposes a novel illumination and shadow model, which models the ambient light more accurately than existing models, by assuming less ambient light is received at a point on a surface when a shadow is cast.

2. Introducing a novel shadow test hypothesis- affine relationship between shadow and non-shadow regions

This proposed hypothesis represents the core of this thesis. The hypothesis presents a novel method for detecting moving cast shadows using a novel physically-derived shadow test condition as follows: let \( q \) be a point on the surface of an object in an illuminated three-dimensional scene, and \( n_q \) be a neighbourhood of \( q \) in the surface. Using a simple geometric representation of light rays and a simple reflection model, it is possible to show that the light energy received at points \( r \in n_q \) in the absence of an object casting a shadow over \( n_q \) is affinely related, to a high degree of approximation, to the energy received when a shadow is cast over \( n_q \) by an object. The same affine parameters are applicable to the entire neighbourhood \( n_q \).

The above shadow hypothesis is used as the core of all the following contributions.

3. Developing shadow detection algorithms for video sequences in the spatial and transform domains

The algorithms presented in this work, use background and object frame pairs, for the detection of moving shadows in the object frames. The proposed shadow detection algorithms are modelled in the following domains: the pixel domain, the Fourier transform domain (Discrete Cosine Transform (DCT)), and the Discrete Wavelet Transform (DWT) domain.

4. Developing a shadow detection algorithm for video sequences with a
non-dedicated background image

In the proposed algorithms, the use of a fixed background frame as a reference is not required if any of the previous frames in the video sequence contain the corresponding non-shadow area, and therefore can be used as a reference frame. An automated approach is proposed and used to determine which previous frame is the best to be used as a reference.

5. Developing shadow edge detection algorithms for still images

In this part 1-D interval-based shadow edge detection algorithms for still images are proposed in the spatial and transform domains. The proposed algorithms in the pixel and transform domains use the Canny edge detector to locate edges in the scene as a pre-processing step. Based on the resulting edge detected image, a shadow boundary detection algorithm compares regions at each edge point in order to decide whether or not that edge point forms a part of the shadow boundary.

1.5 Organization of the Thesis

Remainder of the thesis is organised into seven chapters:

Chapter 2: This chapter starts with providing an overview of the existing hypotheses, and research work carried out in the field of shadow detection. Further, it includes a comprehensive survey on existing cast shadow modelling and detection techniques.

Chapter 3: This chapter represents the core of the thesis. It introduces the proposed novel shadow hypothesis, the shadow condition-affine relationship, and illustrates the derivation of the affine hypothesis.

Chapter 4: This chapter theoretically compares the existing illumination and shadow models with the new proposed model. It also details the differences between
Chapter 1: An Overview

the existing and proposed hypotheses.

Chapter 5: This chapter introduces the cast shadow modelling and detection algorithms in the spatial (pixel) and transform (Fourier and wavelet) domains. It discusses the mechanism of applying the proposed affine shadow test hypothesis to each of the domains.

Chapter 6: This chapter describes the applications of the proposed algorithms in video processing, with an extensive performance evaluation and comparison against other benchmark algorithms. The chapter also discusses a method of applying the proposed algorithms to video applications without the use of a dedicated background image.

Chapter 7: This chapter presents an application of the affine shadow test hypothesis for still images to detect self and cast shadow boundaries. A new approach is proposed based on the use of shadow and non-shadow areas within an image to predict shadow edges.

Chapter 8: This chapter concludes this thesis, with an insight into the future directions of research.
Chapter 2

Literature Review

2.1 Overview

Several shadow detection approaches have been proposed in the literature. This chapter provides a comprehensive overview of some selected hypotheses and algorithms used in the literature for moving cast shadow detection. However, due to large volume of existing literature, the provision of a complete survey on general shadow detection algorithms is out of the scope of this thesis. For a complete survey, readers are advised to refer to Prati\textsuperscript{1} et al. [61] and their follow-up work.

Moving cast shadow detection algorithms can be classified in different ways. For the purpose of this research, these algorithms are categorised into a four-layer taxonomy \textsuperscript{2} (see Figure 2.1). The first layer of classification considers whether the method is independent/dependent on object types. The second layer considers whether the method is environment independent/dependent. The third layer considers whether the decision process introduces and exploits uncertainty. The third layer is subdivided into deterministic and statistical approaches, the former uses an on/off decision pro-

\textsuperscript{1}Prati et al. summarises the general shadow detection algorithms in the literature, and provides an extensive evaluation for the proposed methods.

\textsuperscript{2}In the literature, shadow detection algorithms have been classified into a two-layer taxonomy (see [61]), this taxonomy forms the bottom layers of the new taxonomy proposed in this thesis.
cess, and the later uses statistical measurements, and introduces uncertainty to reduce noise sensitivity. The deterministic class can be further subdivided based on whether the on/off decision can be supported by model based knowledge or not (this taxonomy is summarised in Appendix A -Table A.1) [61].

Another classification has been proposed in this thesis based on the shadow detection algorithm's domain. This classification, considers whether the method is applied in the spatial (pixel), or in the transform (frequency) domain, and whether it is based on the Hue-Saturation-Value (HSV), the Red-Green-Blue (RGB), or YUV colour spaces (this taxonomy is presented in Appendix A -Table A.2).

This chapter is organised as follows: section 2.2 introduces the existing cast shadow modelling hypothesis. Section 2.3 discusses the moving cast shadow detection approaches proposed in literature. This section is further subdivided based on the proposed taxonomy into: section 2.3.1, 2.3.2, and 2.3.3. Finally, section 2.4 summarises this chapter.

### 2.2 Introduction

This section gives an introduction to the illumination, shadow, and reflection models that form the basis for many shadow detection methods proposed in the literature. It also summarises the basic assumptions of many existing methods.

When digital cameras capture shadows in a scene, it is the light reflected from the surface that records that part of the scene. Therefore, the luminance (brightness) of a point \( q \), at the 2D image position \((x, y)\)\(^3\) and time instant \( t \), can be described by the following reflection\(^4\) model [73]:

\[
\psi_t(x, y) = \zeta_t(x, y) \rho_t(x, y)
\]

\(^3\) When considering the following model, \((x, y)\) corresponds to the 2D projection of the environment. Thus \( q \), which is \((x, y, z)\) projects to \((x, y)\).

\(^4\) Assuming the illumination spectrum is constant for each wavelength (white illumination) and matte surfaces
where $\rho_l(x, y)$ is the reflectance of the object surface, i.e. reflection coefficient, and $\zeta_l(x, y)$ is the irradiance (illumination), i.e. the amount of luminance energy (light power) received by $\psi_l(x, y)$, and $\zeta_l(x, y)$ is a function of the direction $L_{x,y}$ of the light source with respect to the object surface normal $N_{x,y}$, and the intensity of the direct light ($c_P$) and the ambient light ($c_A$) received at point $q$. The illumination $\zeta_l$ of a point $(x, y)$ when in or out of a shadow has been modelled as, [73]:

$$\zeta_l(x, y) = \begin{cases} 
  c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object (no shadow)} \\
  \lambda_{x,y} c_P N_{x,y} \cdot L_{x,y} + c_A & \text{penumbra} \\
  c_A & \text{umbra}
\end{cases}$$

The above model describes illumination, both before and after a shadow is cast, and therefore it is called the illumination\(^5\) and shadow model. The term $0 \leq$
\( \lambda_{x,y} \leq 1 \) describes the transition inside the penumbra, and depends on the light source and scene geometry (the model is based on Lambert's cosine law) [73].

In addition to the above model, the following general assumptions are made by many of the existing shadow detection methods [73]:

**Assumption 1:** light source intensity \( c_P \) is high. Consider a pixel at position \((x, y)\) that shows a part of the background. Assume that the pixel is outside a cast shadow at time instant \( t_1 \) and inside a cast shadow at time instant \( t_2 \). It follows that, if \( c_P \) is high at time \( t_1 \), then the difference \( \zeta(x, y) = \zeta_{t_2}(x, y) - \zeta_{t_1}(x, y) \) will be high. Note that, the reflectance of a static background does not change with time, thus \( \rho_{t_2}(x, y) = \rho_{t_1}(x, y) \) holds.

**Assumption 2:** Camera and background are static\(^6\). If both assumptions 1 and 2 hold, the results in the difference equation will be high in the presence of cast shadows covering a static background. This implies (as assumed in many other approaches) that shadow points can be obtained by thresholding the frame difference image.

**Assumption 3:** Background is plane, and light source position is at a distance from the background.

**Assumption 4:** The distance between the moving object and the background is not negligible compared to the distance between the light source and the object.

### 2.3 Moving Cast Shadow Detection Approaches

#### 2.3.1 Object independent and environment dependent approaches

The work of Stauder \(^7\) et al. [73] is based on the reflection model introduced in section 2.2. Given a video sequence, the authors exploit the local appearance change

---

\(^6\)Static camera requires a frame taken before the object enters the scene, and another frame taken after the object enters the same scene. Static background requires a scene illumination that does not change over time, otherwise dynamic background generation algorithms would be required.

\(^7\)The work forms the basis of many other algorithms proposed in literature.
due to shadows by computing the ratio $\xi_i(x, y)$ between the appearance of the pixel in the actual frame (frame with shadow), and the appearance in a reference frame (frame with no shadow) as:

$$
\xi_i(x, y) = \frac{\psi_i(x, y)}{\psi_{i_1}(x, y)} \leq 1.
$$

Using the reflection model $\psi_i(x, y) = \zeta_i(x, y)\rho_i(x, y)$ and assuming constant reflectance through time $t_1$ and $t_2$, i.e. $\rho_t(x, y) = \rho_t(x, y) = \rho_t(x, y)$, it follows that:

$$
\xi_i(x, y) = \frac{\zeta_{i_2}(x, y)}{\zeta_{i_1}(x, y)} \leq 1
$$

Therefore, by using the illumination and shadow model $\zeta_i(x, y)$ (see section 2.2), the ratio $\xi_i(x, y)$ can be written as:

$$
\xi_i(x, y) = \frac{c_A}{c_P N_{x,y} \cdot L_{x,y} + c_A} \leq 1 \quad \text{i.e.} \quad \frac{\text{Umbra(object)}}{\text{Illumination(background)}}
$$

Moreover, following assumption 3 -section 2.2, Stauder et al. assumed $N_{x,y}$ is spatially constant in a neighbourhood of the point. Thus, the pixel is marked as 'possible shadow'. The authors use various heuristic techniques in order to exploit all the four assumptions (such as edge detection and gradient calculation). Results show an excellent detection and removal of indoor shadows. However, the limitations come from the fact that the approach is not applicable for outdoor shadows (outdoor shadows are harder to detect), and the fact that it requires the background to be of a uniform colour.

Xu et al. [81, 82] assumed in their work that the work proposed by Stauder et al. [73] can generate many false negative edges, which denotes the moving edges considered as static edges. They proposed an alternative method of moving cast shadow detection and removal, in normal indoor scenes where the hypothesis and the general assumptions in section 2.2 hold. Their shadow detection and removal algorithm includes: the generation of initial Change Detection Masks (CDM), shadow region detection by multi-frame integration, edge matching and region growing, and finally
shadow region removal and post-processing for eliminating noise and tuning object boundaries. The results were compared with the related gradient filter approach proposed by Chien et al. [16]. Results have proved the high efficiency of the algorithm for shadow detection. However, the method is intended for insignificant indoor shadows, and is only applicable for indoor environments.

Chien et al. [16] proposed a moving object segmentation algorithm for real-time applications. A background registration technique is used to construct a background image from the accumulated frame difference information. The moving object region is separated from the background region by comparing the current frame with the constructed background image. Finally, a post-processing step is applied on the resulting object mask to remove noise regions and to smooth the object boundary. A morphological gradient operations are used to filter out the shadow area while preserving the object shape. However, the use of morphological gradient operations can reduce the effects of insignificant indoor shadows only, and not the effects of outdoor shadows, or significant indoor shadows.

By ignoring the shadow penumbra, and only assuming illuminated and shadow points, Toth et al. [75] applied a simplified version of the illumination and shadow model described in section 2.2 as follows:

\[
\zeta_t(x, y) = \begin{cases} 
    c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object} \\
    c_A & \text{umbra}
\end{cases}
\]

Toth et al. applied the ratio between the shadow region in frame \(I_t\), and its corresponding region in the background frame \(I_b\), where it is illuminated as follows:

\[
k(x, y) = \frac{I_t(x, y)}{I_b(x, y)} = \frac{c_A}{c_P N_{x,y} \cdot L_{x,y} + c_A} \leq 1 \quad \text{i.e. shadow} \quad \text{Illuminate(background)}
\]

Since background reflectance does not change with time, the authors assumed that \(k(x, y)\) represents the ideal case without any noise. They also assumed that images are corrupted with Gaussian white noise. Accordingly, their background images were modelled as follows:
\[ \hat{I}_b = I_b + \epsilon(x, y) \text{ where } \epsilon(x, y) \sim N(0, \sigma^2) \]

with \( \sigma^2 \) being the camera noise, which is either known or can be estimated, and therefore, for a shadow point in the foreground image, it can be assumed that:

\[ I_t(x, y) = k(x, y)I_b(x, y) + \epsilon(x, y) \]

The authors then apply thresholds to classify a point as a non-shadow pixel or as a shadow point.

Siala et al. [72], applied the illumination and shadow model, described in section 2.2. The authors proposed that the distortion between a background image \( I_{bg} \) and a current image \( I_t \), where \( t \) denotes time, of a video-surveillance sequence expressed in the RGB colour space, can be approximated for shadow regions by:

\[ R_{sh} = d_R R_{bg}, \quad G_{sh} = d_G G_{bg}, \quad B_{sh} = d_B B_{bg}, \]

where \( R_{sh}, G_{sh}, B_{sh} \) and \( R_{bg}, G_{bg}, B_{bg} \) are respectively the RGB colour values of shadow pixel in \( I_t \) and non-shadow pixel in \( I_{bg} \), and the colour ratios \( d_R = \frac{B_{sh}}{R_{bg}} \leq 1, \quad d_G = \frac{G_{sh}}{G_{bg}} \leq 1, \quad d_B = \frac{B_{sh}}{B_{bg}} \leq 1 \). To detect shadows, the authors apply a learning stage, where a representative image containing the three classes: foreground, moving shadow, and background, is arbitrarily selected. The moving shadow regions are manually segmented. Colour ratios \( d_R, d_G, d_B \) are computed for pixels issued from a bootstrap sample. Although the use of learning methods is computationally exhaustive, it is supposed to give more accurate results. However, the results show a large number of misclassifications.

Elgammal et al [24] defines a local assumption on the ratio between shadow and non-shadow point luminance. The approach uses colour information to suppress non-shadow points from being detected, by separating colour information from lightness information. Given three colour variables, \( R, G, \) and \( B \), the chromaticity coordinates \( r, g, \) and \( b \) can be calculated as:

\[ r = \frac{R}{R+G+B}, \quad g = \frac{G}{R+G+B}, \quad b = \frac{B}{R+G+B}. \]

The work assumes that the use of chromaticity coordinates in shadow detection has the advantage of being more insensitive to small changes in illumination due to shadows. The
lightness at each pixel is measured using: \( s = R + G + B \). The method starts by assuming assumption 3 -section 2.2, and its basic idea is as follows: let the expected value for a pixel be \(< r, g, s >\) assume that this pixel is covered by a shadow in frame \( t \) and let \(< r_t, g_t, s_t >\) be the observed value for this pixel at this frame. Then, using: 
\[
\alpha \leq \frac{s_t}{s} \leq 1 \quad \text{i.e.} \quad \text{it is expected that the observed value } s_t, \text{ will be darker than the normal value } s \text{ up to a certain limit.}
\]
The work also assumes that a similar effect is expected for highlighted background, where the observed value is brighter than the expected value, up to a certain limit. 

Horprasert et al. [39] proposed a similar approach to Elgammal et al. [24]. Their work also assumes that shadows have similar chromaticity and lower brightness in comparison to the same pixel brightness in the background image. The algorithm is also based on the proposed computational colour model, which separates the brightness from the chromaticity component. Reported results show a good detection of shadows in indoor environments, and shadows in outdoor environments in overcast situations, where cast shadows are weak. It is observed that as the shadow gets stronger, pixels tend to be increasingly identified as foreground pixels. 

The work of KaewTrakulPong and Bowden [46] uses a computational model similar to that of Horprasert et al. [39]. If the difference in both chromatic and brightness components is within some thresholds, the pixel is considered as a shadow.

Friedman and Russell [30] proposed an unsupervised learning technique for shadow detection. The system classifies each pixel using a probabilistic model of how the pixel looks when it is part of different classes (background illuminated, background covered by a cast shadow, or part of foreground object). Assuming the appearance of the pixel in shadow is independent of the object that is casting it, then the shadow model for the pixel is assumed to be relatively constant. Furthermore, the probabilistic classification of the current pixel value is used to update the models, so that object pixels do not become mixed in with the background model when moving slowly. However, the results reported show a significant number of misclassified pixels.
Amamoto and Fujii \cite{2} described a method for tracking moving vehicles on a road. In the proposed method, the varying region in the monitoring image is derived from
the background difference, and is further classified into moving objects, stationary
objects, shadows and highlights. Their work is one of the very few attempts that have
been made to detect shadows in the frequency domain. The work uses the Discrete
Cosine Transform (DCT) to detect shadows of vehicles on roads. The authors suggest
that the shadow of an object varies the pixel values uniformly in comparison with the
background. In addition, within the DCT domain, the authors assume that shadows
of moving objects may be identified by their dc values, while moving objects maybe
identified by their ac values. Although it is a very simple method, highly applicable
for real-time applications, and can be used in the compressed domain, its limitations
come from the domain constrained assumptions, that if generalised will collapse.

Within the transform domain, Etemadnia and Alsharif \cite{29} proposed an approach,
which assumes that the illumination component of an image is generally characterised
by slow spatial variation, while the reflection component tends to vary abruptly, par-
ticularly at the junctions of dissimilar objects. These characteristics lead to associate
the low frequency components of the Fourier transform of an image with illumination,
and the high frequencies with reflection. A Low-Pass-Filter (LPF) and a High-Pass-
Filter (HPF) are defined to detect the shadows. Although the method is applicable
in the compressed domains, the associated results are obtained in very simple indoor
environments, with insignificant shadow.

Jacques et al. \cite{42} in their work, proposed that in shadow regions, it is expected
that a certain fraction of incoming light is blocked. The authors assume that the
observed intensity of shadow pixels is directly proportional to incident light; conse-
quently, shadow pixels are scaled versions (darker) of corresponding pixels in the back-
ground model. The Normalised Cross-Correlation (NCC) is used to detect shadow
pixel candidates. The NCC is used as an initial step for shadow detection, followed
\footnote{The work is considered as the first work to apply shadow detection in the transform domain.}
by a refinement process using local statistics of pixel ratios. Let $B$ be the background frame, and $I$ be the current frame of the video sequence. The method proposes that for each pixel $(x, y)$ belonging to the foreground, it considers a $(2N + 1) \times (2N + 1)$ template $T_{xy}$ such that $T_{xy}(n, m) = I(x + n, y + m)$, for $-N \leq n \leq N, -N \leq m \leq N$, i.e. $T_{xy}$ corresponds to a neighbourhood of pixel $(x, y)$. The NCC between $T_{xy}$ and image $B(x, y)$ is given by:

$$NCC(x, y) = \frac{E_R(x, y)}{E_B(x, y) E_{T_{xy}}}$$

where $E_R$, $E_B$, and $E_{T_{xy}}$ are defined as:

$$E_R(x, y) = \sum_{n=-N}^{N} \sum_{m=-N}^{N} B(x + n, y + m) T_{xy}(n, m)$$

$$E_B(x, y) = \sqrt{\sum_{n=-N}^{N} \sum_{m=-N}^{N} B(x + n, y + m)^2}$$

$$E_{T_{xy}} = \sqrt{\sum_{n=-N}^{N} \sum_{m=-N}^{N} T_{xy}(n, m)^2}$$

A pixel at position $(x, y)$ is pre-classified as shadow if:

$$NCC(x, y) = L_{noc} \quad \text{and} \quad E_{T_{xy}} < E_B(x, y)$$

where $L_{noc}$ is a fixed threshold. The proposed refinement stage holds if the ratio $I(x, y)/B(x, y)$ in a neighbourhood around each shadow pixel candidate is approximately constant, by computing the standard deviation of $I(x, y)/B(x, y)$ within the neighbourhood. Although the results show a good classification of shadow regions in indoor environments, it is shown in the results that there is a large number of misclassified pixels in weak shadow areas, where the authors apply a morphological operator to remove the misclassifications. The authors acknowledge that in outdoor environments, i.e. with strong shadows, the algorithm fails and shadows will be misclassified as foreground objects.

Javed and Shah [44] assumed that pixels in the shadow regions are darker than those in the reference background, and that shadows retain some texture and colour
information of the underlying surface under general viewing conditions. In their work, all foreground regions in the image that are darker than the corresponding regions in the reference image are extracted. The algorithm then performs colour segmentation on the extracted regions. The algorithm uses the K-means approximation to perform colour segmentation. Each pixel value in a potential shadow region is checked against existing K Gaussian distributions until a match is found. Reported results show a good classification of indoor shadows.

Rosin and Ellis [67] developed an algorithm that works on grey level images taken by a stationary camera. Authors perform background subtraction, and segment foreground regions blobs. The intensity ratio between the current and the reference image is calculated for each pixel within the detected blobs. The authors also speculate on the photometric properties of the regions with shadows in the image division. The authors argue that the photometric gain with respect to the background image is roughly constant over the entire shadow region, except at the edges. A region-growing algorithm is used to build likely shadow regions. After that, shadow regions are selected on the basis that they should contain relatively homogeneous intensity ratio values. Reported results show good detection of indoor shadows. However, the total number of misclassified pixels is relatively high.

Fung et al. [31] proposed a shadow detection method, assuming that the luminance of the cast shadow is lower than that of the background, and the chrominance of the cast shadow is identical or slightly shifted when compared with that of the background, the difference in gradient density between the cast shadow and background is lower than the difference in gradient density between the object and background, and the cast shadow is at the boundary region of the moving foreground mask (i.e. the cast shadow can be formed in any direction of the object, but not inside the object). In order to extract the moving object without the cast shadow from the stationary background, their proposed methodology consists of the following stages: moving foreground extraction, shadow confidence score calculation and moving cast
shadow detection. In the first stage, the moving foreground mask is identified by subtracting the background image from the input image. In the second stage, the shadow confidence score is calculated by realising various characteristics of cast shadows in terms of luminance, chrominance, and gradient density from various mapping functions defined according to the cast shadow's characteristics stated above. Finally, based on the shadow confidence score calculated and the significant edge detected in the input image, the object and cast shadow are separated accordingly. Results show a relatively high number of misclassifications in the foreground. The method also comprises a large number of thresholds.

Nicolas [57] presented a scalable block-based video compression scheme for video-surveillance applications. In the approach, each block is classified according to its content, i.e. background, foreground, and cast shadow. Using assumption 3 in section 2.2, the author assumed that the shadow ratio between a shaded point in the image $I_t$ and the same illuminated point in the reference frame $I_{ref}$ can be expressed as:

$$R_t(q) = \frac{I_t^a(q)}{I_{ref}^a(q) + K}$$

where $K$ is a constant and $I_t^a(q)$ is the intensity of the ambient light at point $q$ and time $t$. $I_{ref}^a(q)$ is the intensity of the ambient light at a point $q$ in the reference frame. The method is not considered as a complete physical model for shadows and illumination, since the direct light received at a point $q$ is totally ignored. Even with a distant light source, direct light is still received at a point $q$ unless blocked by an object in the scene. Moreover, no results are reported.

2.3.2 Object independent and environment independent approaches

Mikic et al. [54] and Trivedi et al. [76] introduce an algorithm for moving cast shadow detection in traffic scenes. The method uses three sources of information to

\[\text{The proposed algorithm is the same in [54] and [76]. However, the applications are different.}\]
distinguish between moving cast shadows and their corresponding objects:

- **Local information**: based on the appearance of the individual pixels, a point covered by a shadow gets darker, its blue component increases and the red component decreases compared to the appearance when illuminated.

- **Spatial information**: objects and shadows inhabit compact regions in the image.

- **Temporal information**: object and shadow positions can be predicted from previous frames.

The mean and variance of all three-colour components for each background pixel is calculated. Given the statistics for a background pixel (Gaussian distributions are assumed for background and shadow pixels and a uniform distribution is assumed for foreground), the decisions are made for each pixel. The segmentation starts by comparing the feature vector for each pixel (a three-dimensional vector of \( R, G \) and \( B \) colour components) to the mean at that location in the background model. If not significantly different, the pixel is classified into the background class. Otherwise, prior probabilities are assigned to that location. Colour features, and the neighbourhood information are used to produce smoother classifications. Temporal information is used by modifying class prior probabilities, based on predictions from the previous frame. These two methods show excellent results and performance even in complex environments. However, they require information about the position of the sun with respect to the camera.

Cucchiara et al.\(^{10}\) [19, 21] exploit a similar concept to the work of Elgammal et al. [24] and Horprasert et al. [39]. The work aims to present a technique for shadow detection and suppression used in a system for moving visual object detection and tracking. The major novelty of the shadow detection technique is that the analysis is carried out in the Hue-Saturation-Value (HSV) colour space, to improve the accuracy in detecting shadows (the HSV colour space corresponds closely to the human

\(^{10}\)The authors applied the same shadow detection approach in four different applications.
perception of colour). The system tries to estimate how the occlusion due to shadow changes the values of \( H, S \) and \( V \).

\( HSV \) defines a colour space in terms of three constituent components [26]:

- **Hue**, the colour type (such as red, blue, or yellow): ranges from 0 – 360 (but normalised to 0 – 100% in some applications)

- **Saturation**, the vibrancy of the colour: ranges from 0 – 100%. Also sometimes called the purity by analogy to the colourimetric quantities excitation purity and colourimetric purity. The lower the saturation of a colour, the more greyness is present and the more faded the colour will appear.

- **Value**, the brightness of the colour: Ranges from 0 – 100%

Given a colour defined by \( RGB \) where \( R, G, \) and \( B \) are between 0.0 and 1.0, with 0.0 being the least amount and 1.0 being the greatest amount of that colour, an equivalent \((H, S, V)\) colour can be determined as follows [26]:

Let \( Max \) equal the maximum of the \((R, G, B)\) values, and \( Min \) equal the minimum of those values. The formula can then be written as:

\[
H = \begin{cases} 
60 \times \frac{G-B}{Max-Min} + 0 & \text{if } Max = R \text{ and } G \geq B \\
60 \times \frac{G-B}{Max-Min} + 360 & \text{if } Max = R \text{ and } G < B \\
60 \times \frac{B-R}{Max-Min} + 120 & \text{if } Max = G \\
60 \times \frac{R-G}{Max-Min} + 240 & \text{if } Max = B 
\end{cases}
\]

\[
S = \frac{Max - Min}{Max} \\
V = Max
\]

According to [19, 21], for each pixel belonging to the objects resulting from the segmentation step, the proposed method checks if it is a shadow according to the following considerations. First, if a shadow is cast on a background, the hue component changes, but within a certain limit. In addition, the saturation component is also
considered, which was proven experimentally to change within a certain limit. The
difference in saturation must be an absolute difference, while the difference in hue is
an angular difference. A shadow mask $SP^t$ is defined for each point $p$ resulting from
motion segmentation based on the following shadow model:

$$SP^t(p) = \begin{cases} 
1 & \text{if } \alpha \leq \frac{I^t(p).V}{B^t(p).V} \leq \beta \land |I^t(p).S - B^t(p).S| \leq \tau_S \\
\land D_H \leq \tau_H; & \alpha \in [0, 1], \ \beta \in [0, 1] \\
0 & \text{otherwise}
\end{cases}$$

Where $D_H = \min(|I^t(p).H - B^t(p).H|, 360 - |I^t(p).H - B^t(p).H|)$.

The $H$, $S$, and $V$ denotes the hue, saturation, and value components, respectively,
of vectors in the HSV space. $I^t(p).V$ is the intensity value for the component $V$ of
the $HSV$ pixel in the current frame at time $t$. $B^t(p).V$ is the intensity value for the
component $V$ of the $HSV$ pixel in the reference frame at time $t$ (reference frame
contains the scene with no shadows, i.e. no object is present).

The first condition works on the luminance (the $V$ component), while the second
and third conditions account for both the saturation ($S$) and the hue ($H$) components.
On component $S$ a threshold on the saturation difference between the current and
the reference frame is performed $|I^t(p).S - B^t(p).S| \leq \tau_S$.

The lower bound $\alpha$ is used to define a maximum value for the darkening effect
of shadows on the background and is approximately proportional to the light source
intensity. It takes into account how strong the light source is, i.e. accounts for $c_P$,
$c_A$, and the angle defined in the ratio - according to Stauder et al. [73]:

$$\xi_k(x, y) = \frac{c_A}{c_P N_{x,y} \cdot L_{x,y} + c_A}$$

Thus, the stronger and higher the sun, the lower the value of $\alpha$ must be cho-

en. The upper bound $\beta$ prevents the system from identifying the points where the
background was darkened too little as shadow points. Approximated values for these
parameters are based on empirical dependence on scene luminance parameters such as
the average image luminance and gradient that can be measured directly. The choice
of the parameters $\tau_H$ and $\tau_S$ is calculated with the assumption that the chrominance of shadow and non-shadow points does not vary too much. Previously, they proposed a different version of their model as follows [18, 20]:

$$SP'(p) = \begin{cases} 
1 & \text{if } \alpha \leq \frac{l'(p).V}{B'(p).V} \leq \beta \\
& \land (l'(p).S - B'(p).S) \leq \tau_S \\
& \land | l'(p).H - B'(p).H | \leq \tau_H \\
0 & \text{otherwise}
\end{cases}$$

with the following assumptions: on component $S$ a threshold on the difference is performed, and shadows are assumed to lower saturation of points and, according to the authors' experimental tests, the difference in saturation between image and reference is usually negative for shadow points. On the component $H$ a threshold on the absolute difference gives better results. The parameters $\tau_S$ and $\tau_H$ are determined empirically with the assumption that the chrominance of shadow and non-shadow points does not vary too much.

Shastry and Ramakrishnan [71] applied the above model in their work, while Baisheng and Yunqi [5] used a slightly modified version of the above model to detect shadows. In their work, the shadow mask $SP^t$ is defined for each point $p$ as follows:

$$SP^t(p) = \begin{cases} 
1 & \text{if } (l'(p).V - B'(p).V) < 0 \\
& \land 1 < \frac{B'(p).V}{l'(p).V} < R \\
& \land (l'(p).S - B'(p).S) < \tau_S < 0 \\
& \land | l'(p).H - B'(p).H | < \tau_H \\
0 & \text{otherwise}
\end{cases}$$

Where $1 \leq R \leq 3$ is experimentally found.

Duque et al. [23] used a modified version of the shadow model to detect shadows and highlights. In their work, the shadow mask $SP^t$ and the highlight mask $LP^t$ are defined for each point $p$ as follows:
These approaches [18, 20, 19, 5, 21, 23, 71] are capable of detecting weak penumbra shadows on flat surfaces. However, they require all illumination sources to be white, and assume that both shadow and non-shadow points have similar chrominance. In addition, the results shown in these methods, show that dark shadows will tend to be classified as foreground objects.

Nadimi and Bhanu [56, 55] presented a method to detect moving cast shadows in outdoor environments. The method is based on a spatio-temporal test and accounts for both the sun and the sky illuminations. The method is a multistage approach where each stage of the algorithm removes moving object pixels, which cannot be shadow pixels. The method is independent of object types, models, background types and colours, and scene geometry. It is also capable of detecting umbra in outdoor scenes. Various experimental results are shown, however, the main draw-back is that the approach assumes the spectral power distribution of each illumination source to be equal.

Scanlan et al. [70] proposed a very simple method towards shadow detection and removal, in which a block-by-block mean of the image is computed and stored in an array, and the median is calculated accordingly. Thus, pixels belonging to the blocks
whose mean is less than the median value are considered as shadows and scaled to the median value using an iterative process.

2.3.3 Object dependent and environment dependent approaches

Jiang and Ward \(^{11}\) [45] in their deterministic non-model based approach, proposed a shadow identification and classification algorithm for grey scale images. Their work, extracts both self-shadows and cast shadows from a static image. Using three level processes: the low level process extracts dark regions by thresholding the input image; the middle level process detects features in dark regions, such as the vertices and the gradient of the outline of the dark regions, and uses them to further classify the region as self-shadow or cast shadow; the high level process integrates these features and confirms the consistency along the light directions estimated from the lower levels. The method is based on the analysis of shadow intensity and geometry in an environment with simple objects and a single light source. Only simple scenes, without occlusions between objects and shadows, are considered. The classification into cast and self shadows is based on the assumption that the intensity values of pixels in a self shadow region are larger than those in the corresponding cast shadow region. This represents a limitation of the method since it leads to misclassification if objects are significantly darker than the background or if a cast shadow receives light reflected from another object, which is the case in general.

Hsieh et al. [40] proposed a deterministic model based approach, that represents an algorithm for eliminating shadows of multiple pedestrians using Gaussian shadow modelling. First, a set of moving regions is segmented from the static background using a background subtraction technique. For moving cast shadow detection, a histogram-based method is proposed for isolating each pedestrian from the extracted moving region. Based on the results, a coarse-to-fine shadow modelling process is then applied for eliminating the shadow from the detected pedestrian. At the coarse stage,

\(^{11}\)The work detects self and cast shadows in still images.
a moment-based method is first used for obtaining the rough shadow boundaries. Then, the rough approximation of the shadow region can be further refined through Gaussian shadow modelling. The chosen shadow model is parameterised with several features including: the orientation, mean intensity, and centre position of a shadow region. The novelty of the method, comes from the fact that it uses vertical and horizontal image projection of binary silhouettes, and finds the points where feet and shadow intersect. However, the algorithm needs knowledge about the light source, works only for human objects, and requires the shadow to be on the ground.

Onoguchi [581] proposed a deterministic model based approach, similar to Hsieh et al’s work [40]. In his work, the author presented a method for eliminating pedestrian shadows. The proposed method removes the shadow areas using height information, since most of the shadow areas accompanying moving objects are assumed to be on the road plane. Two cameras are set at locations so that their shared visual fields include the surveillance area. The image obtained from one of the cameras is inverted and projected to the road plane and the projected image on the road plane is transformed to the view from the other camera. Shadows existing on the road plane occupy the same areas in the transformed image and in the image acquired from the other camera, whereas objects areas with different heights from the road plane occupy different areas in these images. Therefore, shadow areas can be removed by subtracting these images. Again the algorithm requires shadows to be on a flat road plane. Further, objects and shadows must be visible to both cameras, and the method requires manual registration and objects’ height.

Yoneyama et al. [83] proposed another deterministic model based approach, to eliminate moving cast shadows based on a simplified 2D vehicle/shadow model of six types projected to a 2D image plane. The parameters of vehicle and shadow models are estimated from input video without the light source and camera calibration information. Distinguishing the cast shadow region from the vehicle itself is done via the determination of parameters of the joint model.
Bevilacqua [7] proposed an algorithm to detect moving shadows in the context of an outdoor traffic scene, for visual surveillance purposes. The algorithm exploits some foreground photometric properties concerning shadows. The proposed method is based on multi-gradient operations applied on the division image (the division image between the current frame and the background of the scene) which aim to find the most likely shadow regions. Further, a binary edge matching is performed on each of the blob's boundaries to discard those regions inside the blob which are either too far from the boundary or too small. Reported results show that regions detected as shadows affect the objects' integrity and shadows with no clear boundaries fail to be recognised.

2.4 Summary

Distinguishing objects from their shadows is a challenging task for computer vision applications. Though, many shadow detection and removal algorithms have been proposed in the literature, the best of these methods are the ones that neither account for the object types nor the scene environment. Whilst these methods are intended to be independent, it is clear that they are not totally independent from the environment, as most of them have minor assumptions about scene geometry, or the spectral distribution of light sources.

In this chapter a comprehensive overview of selected hypotheses and algorithms for moving cast shadow detection and removal has been provided. Moving cast shadow detection algorithms have been classified into a four-layer taxonomy. The first layer of classification considers whether the approach is independent/dependent of object types. The second layer considers whether the approach is environment independent/dependent. The third layer considers whether the decision process introduces and exploits uncertainty, and is further subdivided into deterministic and statistical approaches. The deterministic class is further subdivided based on whether or not
the on/off decision can be supported by model based knowledge.

Another classification has been created based on the domain of the shadow detection algorithms. This classification considers whether the methods are applied in the spatial (pixel) domain, or are applied in the transform (frequency) domain, and whether they are based on the Hue-Saturation-Value ($HSV$), the Red-Green-Blue ($RGB$), or $YUV$ colour spaces.

The design of a simple, accurate, and efficient shadow detection algorithm still remains an open problem. The methods proposed in the literature have either used the illumination and shadow hypothesis- presented in section 2.2, or the shadow model presented in section 2.3.2. Other methods have used colour constancy, histogram, knowledge-based, heuristic, geometric, or projection models. The next chapter introduces a novel cast shadow detection approach, based on a physically-derived model that addresses shortcomings of the existing methods.
Chapter 3

Shadow Hypothesis

3.1 Overview

As discussed in chapter 1, the occlusion of a light source by an object in a scene creates a shadow. The part of the object that is not illuminated is called the self-shadow, while the part projected on the scene by the object is called the cast-shadow, which can be further classified into umbra and penumbra (see Figure 3.1). If the object is moving, the cast shadow is more appropriately referred to as a moving cast shadow, otherwise it is referred to as a still shadow.

This chapter introduces a new physical-based cast shadow detection model [8, 9] and a physical-based illumination and reflection model. Section 3.2 proposes a novel affine shadow test hypothesis, which is derived in section 3.3. Section 3.4 verifies the proposed affine condition using a reflection model. Section 3.5 introduces the principles of applying the affine shadow condition in RGB colour space. Finally, section 3.6 summarises this chapter.
Chapter 3: Shadow Hypothesis

3.2 The Affine Shadow Hypothesis

This section represents the core of this thesis; it introduces a novel method for detecting moving cast shadows using a physically-derived shadow test condition. If $q \in \mathbb{R}^3$ is a point on the surface of an object in an illuminated three-dimensional scene, and $n_q$ is a neighbourhood of $q$ on the surface. Using a simple geometric representation of light rays, and a simple reflection model, it is possible to show that the light energy received at points $r \in n_q$ in the absence of an object casting a shadow over $n_q$ is affinely related, to a high degree of approximation, to the energy received when a shadow is cast over $n_q$ by an object. The same affine parameters are applicable to the entire neighbourhood $n_q$ [8, 9]. It is of course clear that when a shadow is cast over a neighbourhood, less light is received as compared to the fully illuminated state. Therefore, this condition should also be included in the shadow model.

It follows that reflected energies behave similarly and hence the luminance function $L : n_q \rightarrow \mathbb{R}$ when no shadow is cast over $n_q$ is affinely related to the luminance function $L^* : n_q \rightarrow \mathbb{R}$ when a shadow is cast; i.e. for $n_q$ to be in shadow we have $L^*(r) \approx \lambda L(r) + \mu$ and $L^*(r) < L(r)$, for some constants $\lambda$ and $\mu$, for all $r \in n_q$. These
neighbourhood relationships are fundamental to the remainder of the thesis, and constitute the basis of the shadow detection algorithms described and subsequently evaluated. It should be noted that the affine condition \( L^*(r) \approx \lambda L(r) + \mu \) together with the condition \( L^*(r) > L(r) \) can characterise a region of highlight.

Imaging devices generate 2D representations of 3D scenes. Hence a two-dimensional version of the affine relationship for shadows will hold for images. The fact that the affine neighbourhood relationship holds for images suggests a block-wise approach to the detection of moving shadows in video sequences. It is assumed that the reference frame, i.e. the scene before the cast shadow covers the corresponding area is available. This being the case, the existence of an affine relationship can be checked for corresponding block pairs across the frames.

Potential affine parameters for a block pair are not required to be determined a priori, they may be computed from the data in the block pair to which they relate. The affine relationship for shadows is local, i.e. it applies in a neighbourhood. Hence, when processing advances to the next block pair new affine parameters are computed. The physical modelling predicts a scaling relationship under some illumination conditions corresponding to the affine case with \( \mu = 0 \).

To summarise, the hypothesis proposes that 'locally' i.e. in a neighbourhood, the relationship between the darkness levels is affine and the affine relation varies from point to point. There are parallels here with many other physical phenomena that may be highly non-linear when considered globally (or over a long time period). Nevertheless, those physical phenomena are linear or affine, to a high degree of approximation, when considered in a sufficiently small spatial neighbourhood (or over a sufficiently small time interval).
3.3 The Derivation of the Affine Shadow Hypothesis

This section represents the derivation from which the affine shadow hypothesis was derived. The derivation using single and multiple light sources shows the validity of the proposed hypothesis for both of the illumination and reflection models.

3.3.1 Shadows cast by a single point source

In a very simple environment comprising a ‘floor’ $F$ and a light source $p$, as shown in Figure 3.2, point $q$ receives a direct ray of intensity $i$ from $p$. If an object is placed on $F$ then the source $p$ casts a shadow of the object onto $F$, as shown in Figure 3.3. The incident ray onto $q$ is now blocked by the object and, as the only reflected light (i.e. ignoring secondary reflections) is off the part of the object facing $p$, the shadow is completely black.

Consider a more complex environment comprising a ‘floor’ $F$, a ‘wall’ $W$ and a single point white-light source at $p$ - as shown in Figure 3.4. A light ray of intensity $i$ from $p$ strikes the point $q$. Other light rays from $p$ strike the wall $W$, reflect, and are incident on $q$. If an object is added - as shown in Figure 3.5 and the following simplifying assumptions are made:

1. the reduction of flux with distance is ignored i.e. total energy radiated through all closed surfaces containing the source is equal - this is strictly only true in a vacuum,

2. all reflecting surfaces are diffusers,

---

1This follows what is done in computer graphics and is not strictly physical. However, it is possible to model light sources using continuous spectral distributions; reflected light from these sources enters the imaging device and is filtered to $R, G, B$ form at pixel locations for capture in digital form - the same conclusions, regarding the identification of shadow regions in the images, may then be drawn.
3. light sources are modelled as mixtures, \((i_R, i_G, i_B)\), of \(R, G, B\) primaries, rather than as having continuous spectral distributions,

4. reflections from surfaces are modelled using reflection coefficients \(\mu_R, \mu_G\) and \(\mu_B\),

5. secondary reflections can be ignored - i.e. each point of \(F\) is a diffuse source for \(W\); the reflections from which illuminate \(F\) as 'secondary' reflections.

A reflected ray from each point of \(W\) is also incident on the point \(q\) - 'primary' reflections. If \(W\) is a diffuser, it reflects light equally in all directions. The strength,

\[ i \mu(w) N_w \cdot L_w, \]

of the ray reflected to \(q\) from a position \(w = (x_w, y_w)\) on \(W\) depends on \(w\).

Modelling the light source at \(p\) by \((i_R, i_G, i_B)\), the light arriving at \(q\) in the absence of an object, denoted \((i_{bg,q,R}, i_{bg,q,G}, i_{bg,q,B})\), may be written as:
Figure 3.3: Same environment as figure 3.2 with object in presence, point q receives no light from p, therefore is completely dark.

\[(i_{bg,q,R}, i_{bg,q,G}, i_{bg,q,B}) = (i_R, i_G, i_B)N_q \cdot L_q + \int_W (i_R \mu_R(w), i_G \mu_G(w), i_B \mu_B(w))N_w \cdot L_w \, dw\]

\[= (i_R \left[ N_q \cdot L_q + \int_W \mu_R(w)N_w \cdot L_w \, dw \right], \]
\[i_G \left[ N_q \cdot L_q + \int_W \mu_G(w)N_w \cdot L_w \, dw \right], \]
\[i_B \left[ N_q \cdot L_q + \int_W \mu_B(w)N_w \cdot L_w \, dw \right];\]

i.e.

\[i_{bg,q,C} = i_C \left[ N_q \cdot L_q + \int_W \mu_C(w)N_w \cdot L_w \, dw \right] \text{ for } C \in \{R, G, B\}\]

With the object present we have:

\[i_{o,q,C} = i_C \left[ \int_{W \setminus S} \mu_C(w)N_w \cdot L_w \, dw \right]\]
Figure 3.4: A simple 'background' environment illuminated by a single point light source.

where $S$ is the intersection of the shadow with $W$.

We therefore have:

$$i_{o,q,C} = \frac{\int_{W \setminus S} \mu_C(w) N_w \cdot L_w \, dw}{N_q \cdot L_q + \int_W \mu_C(w) N_w \cdot L_w \, dw} i_{bg,q,C}$$

$$= \frac{\Omega_C}{N_q \cdot L_q + \Lambda_C} i_{bg,q,C}$$

$$= \gamma_C(q) i_{bg,q,C}$$

where

$$\Omega_C = \int_{W \setminus S} \mu_C(w) N_w \cdot L_w \, dw,$$

$$\Lambda_C = \int_W \mu_C(w) N_w \cdot L_w \, dw,$$
are constants, and
\[ \gamma_c(q) = \frac{\Omega_C}{N_q \cdot L_q + \Lambda_C} \]
is dependent on \( q \).

It should be noted that the relationship
\[ i_{o,q,C} = \gamma_c(q) \cdot i_{b,q,C} \]
is a point-wise relationship - i.e. it is applied at \( q \) and the 'scaling' function is dependent on \( q \).

The approach of shadow detection taken in this research attempts to classify 'regions', i.e. contiguous groups of points (ultimately groups of pixels) as being in shadow, rather than classifying single pixels. The next section shows how to transform the point-wise relation into a relation that holds for a region, or neighbourhood.
3.3.2 The neighbourhood approximation

In a suitably small neighbourhood, $U_q$, of $q$, it is safe to assume:

- $N_q \cdot L_q \approx K_q$ where $K_q$ is a constant.

Hence in $U_q$ we have

$$i_{o,q^*,C} \approx \frac{\Omega_C}{K_q + \Lambda_C} i_{b,q^*,C}$$

$$= \alpha_C \ i_{b,q^*,C} \text{ for all } q^* \in U_q$$

where $\frac{\Omega_C}{K_q + \Lambda_C}$ is a constant on $U_q$; i.e. the relationship between the light incident at each point $q^*$ of the neighbourhood $U_q$ in the presence/absence of an object is approximately one of constant scale.

3.3.3 Shadows cast by two point sources

If $i^{(1)}$ and $i^{(2)}$ denote the two point sources, then

$$i^{(1)} = (i_{R,1}, i_{G,1}, i_{B,1}), \quad i^{(2)} = (i_{R,2}, i_{G,2}, i_{B,2}).$$

The light, $i_{b,q}$ and $i_{o,q}$ arriving at $q$ with no object present and with the object present respectively, is written as:

$$i_{b,q} = (i_{b,q,R}, i_{b,q,G}, i_{b,q,B}), \quad i_{o,q} = (i_{o,q,R}, i_{o,q,G}, i_{o,q,B}).$$

It follows that - see Figure 3.6:

$$i_{b,q,C} = i^{(1)}_C N_q \cdot L^{(1)}_q + i^{(2)}_C N_q \cdot L^{(2)}_q$$

$$+ i^{(1)}_C \int_W \mu_C(w) N_w \cdot L^{(1)}_w \, dw + i^{(2)}_C \int_W \mu_C(w) N_w \cdot L^{(2)}_w \, dw$$

$$= i^{(1)}_C \left[ N_q \cdot L^{(1)}_q + \int_W \mu_C(w) N_w \cdot L^{(1)}_w \, dw \right]$$

$$+ i^{(2)}_C \left[ N_q \cdot L^{(2)}_q + \int_W \mu_C(w) N_w \cdot L^{(2)}_w \, dw \right]$$
Chapter 3: Shadow Hypothesis

Figure 3.6: A simple ‘background’ environment illuminated by two point light sources.

For $q \in F$ and $q \in \text{Intersection of two shadows cast on } F$ (i.e. $q \in \text{the umbra region}$) - see Figure 3.7. In this case $q$ receives no direct light from either $p^{(1)}$ or $p^{(2)}$ but it receives reflected light from $W \setminus S_1$ (indirectly from $p^{(1)}$) and reflected light from $W \setminus S_2$ (indirectly from $p^{(2)}$). Hence, for $q$ in the umbra region the light received at $q$ is:

$$i_{o,q,C} = i^{(1)}_C \left[ \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] + i^{(2)}_C \left[ \int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right].$$

Note here that $S_1$ and $S_2$ are the shadow areas on $W$ cast by point sources $p^{(1)}$ and $p^{(2)}$ respectively. Eliminating $i^{(2)}_C$ from the expressions for $i_{o,q,C}$ and $i_{o,q,C}$ gives:

$$i^{(2)}_C = \frac{i_{o,q,C} - i^{(1)}_C \left[ N_q \cdot L_q^{(1)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right]}{N_q \cdot L_q^{(2)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(2)} \, dw}$$

from which it follows that:

$$i_{o,q,C} = i^{(1)}_C \left[ \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] + \left[ \frac{i_{o,q,C} - i^{(1)}_C \left[ N_q \cdot L_q^{(1)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right]}{N_q \cdot L_q^{(2)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(2)} \, dw} \right] \int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} \, dw.$$
Figure 3.7: Point $q$ in umbra region receives reflected light from both sources. No direct light received at this point

and rearranging

$$i_{o,q,C} = i_{bg,q,C} \left[ \frac{\int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} dw}{N_q \cdot L_q^{(2)} + \int_W \mu_C(w) N_w \cdot L_w^{(2)}} \right]$$

$$+ i_C^{(1)} \left[ \frac{N_q \cdot L_q^{(1)} + \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} dw - \left[ \frac{N_q \cdot L_q + \int_W \mu_C(w) N_w \cdot L_w^{(1)} dw}{N_q \cdot L_q + \int_W \mu_C(w) N_w \cdot L_w^{(2)} dw} \right]}{\int_W \mu_C(w) N_w \cdot L_w^{(2)} dw} \right].$$

Applying the neighbourhood condition, $N_q \cdot L_q \approx \text{constant}$ in a neighbourhood $U_q$ of $q$, gives an affine relationship

$$i_{o,q^*,C} = \alpha_2 i_{bg,q^*,C} + \beta_2$$

between $i_{o,q^*,C}$ and $i_{bg,q^*,C}$ in $U_q$.

When $q$ is in the penumbra region, it receives direct light from one source and reflected light from both sources. In the configuration shown in Figure 3.8, $q$ receives direct light from $p^{(1)}$ and reflected light from $W \setminus S_1$ (indirectly from $p^{(1)}$) and reflected
light from $W \setminus S_2$ (indirectly from $p^{(2)}$). Hence for $q$ in the penumbra region, the light received at $q$ is:

$$i_{q,C} = i^{(1)}_C N_q \cdot L^{(1)}_q + i^{(1)}_C \left( \int_{W \setminus S_1} \mu_C(w) N_w \cdot L^{(1)}_w \, dw \right) + i^{(2)}_C \left( \int_{W \setminus S_2} \mu_C(w) N_w \cdot L^{(2)}_w \, dw \right)$$

i.e. the direct light term from $p^{(1)}$ plus the 'penumbra' term. Eliminating $i^{(2)}_C$ from $i_{q,C}$ gives:

$$i_{q,C} = i_{q,C} \left[ \frac{\int_{W \setminus S_1} \mu_C(w) N_w \cdot L^{(1)}_w \, dw}{N_q \cdot L^{(1)}_q + \int_{W \setminus S_1} \mu_C(w) N_w \cdot L^{(1)}_w \, dw} \right]$$

$$+ i^{(1)}_C \left( \frac{N_q \cdot L^{(1)}_q + \int_{W \setminus S_1} \mu_C(w) N_w \cdot L^{(1)}_w \, dw}{N_q \cdot L^{(1)}_q + \int_{W \setminus S_1} \mu_C(w) N_w \cdot L^{(1)}_w \, dw} \int_{W \setminus S_2} \mu_C(w) N_w \cdot L^{(2)}_w \, dw \right)$$

Figure 3.8: Point $q$ in penumbra region receives direct light from one point source, and reflected light from both sources.

The neighbourhood condition, $N_q \cdot L_q \equiv \text{constant}$ and $N_q \cdot L^{(1)}_q \equiv \text{constant}$, in a neighbourhood $U_q$ of $q$ gives an affine relationship

$$i_{q,C} = \alpha_q \cdot i_{q,C} + \beta_q$$

between $i_{q,C}$ and $i_{q,C}$ in $U_q$ for this case too.
Two equal point sources

In this case we have \( i^{(2)} = i^{(1)} \) and hence

\[
i_{bg,q,C} = i^{(1)}_C \left[ N_q \cdot (L^{(1)}_q + L^{(2)}_q) + \int_{W} \mu C(w)N_w \cdot (L^{(1)}_w + L^{(2)}_w) \, dw \right]
\]

\[
i_{o,q,C} = i^{(1)}_C \left[ N_q \cdot L^{(1)}_q + \int_{W \backslash S_1} \mu C(w)N_w \cdot L^{(1)}_w \, dw + \int_{W \backslash S_2} \mu C(w)N_w \cdot L^{(2)}_w \, dw \right]
\]

i.e. the relationship

\[
i_{o,q,C} = i_{bg,q,C} \left[ \frac{B \cdot N_q \cdot L^{(1)}_q + \int_{W \backslash S_1} \mu C(w)N_w \cdot L^{(1)}_w \, dw + \int_{W \backslash S_2} \mu C(w)N_w \cdot L^{(2)}_w \, dw}{N_q \cdot (L^{(1)}_q + L^{(2)}_q) + \int_{W} \mu C(w)N_w \cdot (L^{(1)}_w + L^{(2)}_w) \, dw} \right]
\]

where

\[
B = \begin{cases} 0 & q \in \text{umbra}, \\ 1 & q \in \text{penumbra}. \end{cases}
\]

holds between \( i_{o,q,C} \) and \( i_{bg,q,C} \). The neighbourhood condition reduces this to a (constant) scale relationship of the form

\[
i_{o,q^*,C} = \alpha_4 i_{bg,q^*,C} \quad \text{for all} \quad q^* \in U_q.
\]

### 3.4 Reflection Modelling

Up to this point, only the light incident on \( q \) has been modelled. In this section, the light reflected from a point \( q \), whether or not it lays in shadow, will be investigated. It is shown here that, provided that a local homogeneity condition holds for the texture of the neighbourhood \( U_q \) of \( q \), the affine condition is preserved under reflection.

Given the finite resolution of all imaging devices, it follows that neither a single point nor a entire neighbourhood \( U_q \) will render to a single pixel. We assume that a subset \( V \subset U_q \) 'integrates' to determine a particular pixel value (see Figure 3.9).

If the 'mean' reflectance parameters for \( V \) are \( \mu_{V,R}, \mu_{V,G} \) and \( \mu_{V,B} \), then:
Figure 3.9: A neighbourhood \( U_q \), and two subsets \( V \) and \( V^* \) assumed to determine pixel values in the captured image.

(i) for \( U_q \) out of shadow, the reflected light from \( V \) is

\[
\tau_{bg,V,C} = i_{bg,q,C} \mu_{V,C}
\]

and

(ii) for \( U_q \) in shadow the reflected light from \( V \) is

\[
\tau_{o,V,C} = (\alpha_C i_{bg,q,C} + \beta_C)\mu_{V,C}
\]

\[
= \alpha_C i_{bg,q,C} \mu_{V,C} + \beta_C \mu_{V,C}
\]

\[
= \alpha_C \tau_{bg,V,C} + \beta_C \mu_{V,C}
\]

In a different subset \( V^* \), that also integrates to a pixel, we obtain

\[
\tau_{o,V^*,C} = \alpha_C \tau_{bg,V^*,C} + \beta_C \mu_{V^*,C}.
\]

It follows that the affine condition, that has been demonstrated for light incident on \( U_q \) in/out-of shadow, is preserved, provided \( \mu_{V,C} = \mu_{V^*,C} \) across \( U_q \). In other words local patches within neighbourhood \( U_q \) have similar mean reflectance coefficients. Such textures having this as a global property include tarred road surfaces, brick
surfaces, concrete surfaces, finely grained woods, and carpeted surfaces without strong geometric patterns. The condition applies locally to many other surfaces and is not likely to lead to significant lack of generalisation provided the neighbourhoods $U_q$ are sufficiently small.

3.5 RGB Space

With a monochrome source or with a colour source, simulated using $R, G, B$ triples, the simple model predicts that in a sufficiently small neighbourhood an affine relationship exists between reflected light from a neighbourhood $U_q$ of a point $q$ when it is in/out-of shadow, i.e. we have:

$$r_{o,q^*,C} = \alpha_C r_{b,q^*,C} + \delta_C$$

for all $q^* \in U_q$ - and provided local patches of $U_q$ having similar reflectance coefficients. Note that in different colour layers the corresponding affine parameters will
be different (e.g. $\alpha_R \neq \alpha_G$). Hence shadows may be identified in colour images by applying the affine test in each colour layer. Consequently, the shadow area will be the intersection area between the three colours, see Figure 3.10.

### 3.6 Summary

This chapter presented the proposed shadow hypothesis and computations, using the geometric and physical models of light, that lead to a local affine hypothesis for the identification of shadows in digital images, which can be summarised as follows:

When a shadow is cast over a neighbourhood, less light is received there - as compared to the fully illuminated state. Using the geometric representation of light rays and a simple reflection model, this chapter has shown that the light energy reflected/received at points $r \in n_q$ in the absence of an object casting a shadow over $n_q$ is affinely related, to a high degree of approximation, to the energy reflected/received when a shadow is cast over $n_q$ by an object. The same affine parameters being applicable to the entire neighbourhood $n_q$.

Thus, the luminance function $L : n_q \rightarrow \mathbb{R}$, when no shadow is cast over $n_q$, is affinely related to the luminance function $L^* : n_q \rightarrow \mathbb{R}$ when a shadow is cast; i.e. for $n_q$ to be in shadow we have $L^*(r) \approx \lambda L(r) + \mu$ and $L^*(r) < L(r)$, for some constants $\lambda$ and $\mu$, for all $r \in n_q$.

The scaling relationship is a special case of the affine relationship, when $\mu = 0$. Therefore, in shadow detection applications, it is safe to consider only the affine relationship as it will represent the scaling relationship when $\mu$ is estimated in a local neighbourhood to be 0. The affine relationship has been validated across wide range of environments including indoor and outdoor situations, with natural and artificial lighting.

Chapter 4 compares the proposed illumination and shadow hypothesis with the existing ones.
Chapter 4

Illumination and Shadow Models - Theoretical Comparisons

4.1 Overview

The existing illumination and shadow model presented and discussed in section 2.2, hereafter referred to as $H1$ [73], has formed the basis for many shadow detection algorithms in the literature. The shadow model discussed in section 2.3.2 [18, 20, 19, 5, 21, 23, 71], hereafter referred to as $H2$, has been used in many other applications. The proposed novel affine shadow hypothesis has been discussed in section 3.2 and derived in section 3.3. The neighbourhood affine hypothesis produced a new illumination and shadow model, hereafter referred to as $H3$. Therefore, this chapter is intended to provide comprehensive theoretical comparisons between the proposed and the existing illumination and shadow models. Section 4.2 compares the proposed and existing illumination and shadow models using single and multiple point sources. Section 4.3 presents the proposed illumination and shadow model. Section 4.4 provides a verification of the affine hypothesis across the existing model. Finally, section 4.5 summarises this chapter.
4.2 Illumination and Shadow Modelling

In hypothesis $H1$, the luminance energy (light power) received at a point $q^1$, on a surface at coordinate $(x, y)$ has been modelled as [73]:

$$\zeta(x, y) = \begin{cases} 
  c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object} \\
  \lambda_{x,y} c_P N_{x,y} \cdot L_{x,y} + c_A & \text{penumbra} \\
  c_A & \text{umbra}
\end{cases}$$

Where $0 \leq \lambda_{x,y} \leq 1$ describes the transition inside the penumbra, and depends on the light source and scene geometry. $c_P$ is the intensity of the light source $p$ incident at $q$, and $c_A$ is the ambient light incident at $q$, i.e. reflected light from the environment, here simplified with a wall 'W' and a primary reflection. In the $H1$ hypothesis the amount of ambient light $c_A$ received at $q$, is assumed to be equal in all the cases whether an object is either present or not. In addition, white illumination has been assumed, i.e. the $RGB$ colour components are all assumed to have equal intensities.

Given a simple geometrical environment, which can be safely generalised for real environments, if a 'wall' $W$ is a diffuser it reflects light equally in all directions. The strength, $i \mu(w)N_w \cdot L_w$, of the ray reflected to $q$ from a position $w = (x_w, y_w)$ on $W$ depends on $w$. Therefore, in this chapter, the ambient illumination is modelled differently, as we assume that different amounts of ambient light will be received at point $q$ when it is in or out of shadow (see Figure 3.4 and Figure 3.5).

4.2.1 Using a single point source

Modelling the light source at $p$ by $(i_R, i_G, i_B)$, the light arriving at $q$ in the absence of an object may be written as:

$$(\zeta_R, \zeta_G, \zeta_B) = (i_R, i_G, i_B)N_q \cdot L_q + \int_w (i_R \mu_R(w), i_G \mu_G(w), i_B \mu_B(w))N_w \cdot L_w \, dw$$

$^1$In hypothesis $H1$, $(x, y)$ corresponds to the 2D projection of the environment, so $q$ which is $(x, y, z)$ projects to $(x, y)$ when considering the $H1$ hypothesis only
\begin{align*}
&= (i_R \left[ N_q \cdot L_q + \int_W \mu_R(w)N_w \cdot L_w \, dw \right], \\
&i_G \left[ N_q \cdot L_q + \int_W \mu_G(w)N_w \cdot L_w \, dw \right], \\
&i_B \left[ N_q \cdot L_q + \int_W \mu_B(w)N_w \cdot L_w \, dw \right); \\
\end{align*}

i.e.

\[ \zeta_C = i_C \left[ N_q \cdot L_q + \int_W \mu_C(w)N_w \cdot L_w \, dw \right] \quad \text{for } C \in \{R, G, B\} \]

Compared to hypothesis H1, where it assumes:

\[ \zeta(x, y) = c_P \, N_{x,y} \cdot L_{x,y} + c_A \]

and by ignoring the white illumination assumption, the results are equivalent. However, with object(s) added we argue that the amount of ambient light received at point q will be less compared to the case where no object is present. In Figure 3.5, the area on wall 'W' covered by a shadow will not reflect light to point q. Therefore, point q will receive less ambient light compared to the same environment with no object present, as in Figure 3.4. Note that there is no direct light incident on q, thus we have:

\[ \zeta_C = i_C \left[ \int_{W \setminus S} \mu_C(w)N_w \cdot L_w \, dw \right] \]

where S is the intersection of the shadow with W. In the case where an object is added, this will be a more precise and accurate method for modelling the ambient light compared to hypothesis H1 which assumes:

\[ \zeta(x, y) = c_A. \]

**Local appearance change due to shadows**

The local appearance change due to shadow at point q is computed using the ratio \( \xi(q) \) between the appearance of the pixel in the current frame (object present) at time \( t_2 \), and the appearance in a reference frame at time \( t_1 \) (no object present) as:
Using the reflection model $\psi_t(q) = \xi(q) \rho_t(q)$ and assuming constant reflectance through time $t_1$ and $t_2$, i.e. $\rho_{t_2}(q) = \rho_{t_1}(q) = \rho_t(q)$, it follows that:

$$\xi_t(q) = \frac{\xi_{t_2}(q)}{\xi_{t_1}(q)} \leq 1.$$ 

Therefore, by using the irradiance $\zeta(q)$, if a static background point is covered by a shadow region, the ratio $\xi(x, y)$ using the $H1$ hypothesis is written as:

$$\xi(x, y) = \frac{c_A}{c_P N_{x,y} \cdot L_{x,y} + c_A} \leq 1 \quad \text{i.e.} \quad \frac{\text{Umbra( object)}}{\text{Illuminate(no object)}}$$

However, in the proposed model, $H3$, we have:

$$\xi(q) = \frac{i_C \left[ \int_{W \setminus S} \mu_C(w) N_w \cdot L_w \, dw \right]}{i_C \left[ N_q \cdot L_q + \int_W \mu_C(w) N_w \cdot L_w \, dw \right]}.$$ 

### 4.2.2 Two point sources

In the configuration shown in Figure 3.6, let $i^{(1)}$ and $i^{(2)}$ denote the two point sources, where $i^{(1)} = (i_R^{(1)}, i_G^{(1)}, i_B^{(1)})$, $i^{(2)} = (i_R^{(2)}, i_G^{(2)}, i_B^{(2)})$, and $C \in R, G, B$, it follows that:

$$\zeta_c = i_C^{(1)} \left[ N_q \cdot L_q^{(1)} + \int_W \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ N_q \cdot L_q^{(2)} + \int_W \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right].$$

**Umbra**

In Figure 3.7, $q$ receives no direct light from either $p^{(1)}$ or $p^{(2)}$ but it receives reflected light from $W \setminus S_1$ (indirectly from $p^{(1)}$) and reflected light from $W \setminus S_2$ (indirectly from $p^{(2)}$). Hence for $q$ in the umbra region, the light received at $q$ is:
Chapter 4: Illumination and Shadow Models - Theoretical Comparisons

\[ \zeta_c = i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right]. \]

Note here, \( S_1 \) and \( S_2 \) are the shadow areas on \( W \) cast by point sources \( p^{(1)} \) and \( p^{(2)} \) respectively. It follows that the local appearance change due to shadow \( \xi(q) \) can be written as:

\[ \xi(q) = \frac{i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right]}{i_C^{(1)} \left[ N_q \cdot L_q^{(1)} + \int_W \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ N_q \cdot L_q^{(2)} + \int_W \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right]} \]

Penumbra

If a static background point is in the penumbra region, the local appearance change of a point due to shadow in model \( H1 \) can be written as:

\[ \xi(x, y) = \frac{\lambda_{x,y}C_p \cdot N_{x,y} \cdot L_{x,y} + c_A}{c_p \cdot N_{x,y} \cdot L_{x,y} + c_A} \leq 1 \quad \text{i.e.} \quad \text{Penumbra(object)} \frac{\text{Illuminate(no object)}}{} \]

When \( q \) is in the penumbra region it receives direct light from one source and reflected light from both sources. In the configuration shown in Figure 3.8, if \( q \in \text{penumbra} \), it will receive direct light from \( p^{(1)} \) and reflected light from \( W \setminus S_1 \) (indirectly from \( p^{(1)} \)) and from \( W \setminus S_2 \) (indirectly from \( p^{(2)} \)).

Hence for \( q \) in the penumbra region, according to the \( H3 \) model, the light received at \( q \) is:

\[ \zeta_c = i_C^{(1)} N_q \cdot L_q^{(1)} + i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right] \]

Accordingly, the local appearance change of a point due to shadow can be written as:

\[ \xi(q) = \frac{i_C^{(1)} N_q \cdot L_q^{(1)} + i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right]}{i_C^{(1)} \left[ N_q \cdot L_q^{(1)} + \int_W \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ N_q \cdot L_q^{(2)} + \int_W \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right]} \]
4.3 The Proposed Illumination and Shadow Model

In a single point source environment, a point $q$ on a surface will either belong to the illuminated area or a shadow umbra. Note that shadow penumbra do not exist in this particular environment. Therefore, the new illumination and shadow model $\zeta(p)$ can be written as:

$$
\zeta(q) = \begin{cases} 
    i_C \int_{W} \mu \ N_w \cdot L_w \ dw + i_C N_q \cdot L_q & \text{no object} \\
    i_C \int_{W\setminus\mu} \mu \ N_w \cdot L_w \ dw & \text{umbra}
\end{cases}
$$

To generalise this model, in a two point sources environment, a point $q$ on a surface may belong to the illuminated area, shadow umbra, or shadow penumbra. Therefore, the new illumination and shadow model ($H3$) can be written as:

$$
\zeta(q) = \begin{cases} 
    i_C^{(1)} [N_q \cdot L_q^{(1)} + \int_{W} \mu_C(w)N_w \cdot L_w^{(1)} \ dw] \\
    + i_C^{(2)} [N_q \cdot L_q^{(2)} + \int_{W} \mu_C(w)N_w \cdot L_w^{(2)} \ dw] & \text{no object} \\
    + i_C^{(1)} \int_{W\setminus\mu} \mu_C(w)N_w \cdot L_w^{(1)} \ dw & \text{penumbra} \\
    + i_C^{(2)} \int_{W\setminus\mu} \mu_C(w)N_w \cdot L_w^{(2)} \ dw & \text{umbra}
\end{cases}
$$

Compared to the $H1$ illumination and shadow model, which assumes:

$$
\zeta(x, y) = \begin{cases} 
    c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object} \\
    \lambda_{x,y} c_P N_{x,y} \cdot L_{x,y} + c_A & \text{penumbra} \\
    c_A & \text{umbra}
\end{cases}
$$

it can be noticed that the direct and ambient light received at point $q$ in case of no object present is modelled similarly in both the $H1$ and $H3$ models. However, in the case when an object is present, and the point $q$ is in the umbra or penumbra region,
the ambient light received at point $q$ in the $H1$ model does not change, compared to the $H3$ model where the ambient light received at point $q$ is modelled as:

$$\zeta_C = i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right].$$

### 4.4 The Affine Hypothesis - Verification

As introduced in section 3.2 and section 3.3, it is possible to show that the energy of light received at a point $q$ in the absence of an object casting a shadow over $n_q$ is affinely related to the energy of light received at $q$ when a shadow is cast over $n_q$ by an object, with the same affine parameters being applicable to the entire neighbourhood $n_q$.

Using the $H1$ model it is also possible to derive an affine relationship between shadow and non-shadow points, provided that $\lambda_{x,y}$ and $N_{x,y} \cdot L_{x,y}$ are constants in a sufficiently small neighbourhood. Writing $\lambda_{x,y}$ as $\lambda$ and $N_{x,y} \cdot L_{x,y}$ as $N \cdot L$ and assuming:

$$l_{ns} = c_p N \cdot L + c_A$$

$$l^*_s = \lambda c_p N \cdot L + C_A$$

$$l_s = c_A$$

The affine relationship can be derived from the $H1$ model as follows:

$$l_{ns} - c_A = c_p N \cdot L$$

$$l^*_s - c_A = \lambda c_p N \cdot L$$

$$l_{ns} - c_A = (l^*_s - c_A) / \lambda$$

$$\approx \beta (l^*_s - c_A)$$
Chapter 4: Illumination and Shadow Models - Theoretical Comparisons

\[ l_{ns} = c_A + \beta (l_s^* - c_A) \]
\[ = c_A + \beta l_s^* - \beta c_A \]
\[ = \beta l_s^* + c_A - \beta c_A \]
\[ = \beta l_s^* + c_A(1 - \beta) \]

Where the derived relation \( l_{ns} = \beta l_s^* + c_A(1 - \beta) \) is an affine relationship provided that \( \lambda \) and \( N \cdot L \) are constant over the neighbourhood.

The affine condition for individual points has been recently discussed in the work of Wang et. al [78]. In their work, the affine parameters \( \lambda \) and \( \mu \) are calculated for each point separately. However, in this thesis, the same affine parameters \( \lambda \) and \( \mu \) are applicable to the entire neighbourhood.

4.5 Summary

In this chapter a new illumination and shadow model (H3) has been proposed, which models the ambient light more precisely than the existing models, by assuming less ambient light is received at a point \( q \) on a surface when the shadow is cast. The \( H3 \) model can be summarised as follows:

\[
\zeta(q) = \begin{cases} 
|^{(1)} & i_C \left[ N_q \cdot L_q^{(1)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] \\
|^{(2)} & + i_C \left[ N_q \cdot L_q^{(2)} + \int_{W} \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right] \\
|^{(1)} & i_C \left[ \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] \\
|^{(2)} & + i_C \left[ \int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right] \\
|^{(1)} & i_C \left[ \int_{W \setminus S_1} \mu_C(w) N_w \cdot L_w^{(1)} \, dw \right] \\
|^{(2)} & + i_C \left[ \int_{W \setminus S_2} \mu_C(w) N_w \cdot L_w^{(2)} \, dw \right]
\end{cases}
\]

As an alternative to illumination and shadow model, \( H1 \);
\[ \zeta(x, y) = \begin{cases} 
  c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object} \\
  \lambda_{x,y} c_P N_{x,y} \cdot L_{x,y} + c_A & \text{penumbra} \\
  c_A & \text{umbra} 
\end{cases} \]

It is also shown that using existing illumination and shadow model \((H1)\), it is possible to derive an affine relationship between shadow and non-shadow points.

Chapter 5 introduces the multi-domain shadow detection algorithms, and illustrates the methods of applying the affine shadow test condition, and calculating the affine parameters for each domain.
Chapter 5

Spatial and Transform Domains

Shadow Detection Algorithms

5.1 Overview

Shadows are a major problem associated with segmenting and extracting moving objects. Misclassifying shadow points as foreground leads to object integration and object shape distortion. The problems related to shadows occur because shadows and objects share two important features [61]:

- shadow points are detectable as foreground points, and
- shadows have the same motion as the objects casting them.

Prati et al. [61] noted that, while the main concepts utilised for shadow analysis in still and video images are similar, the purpose behind shadow extraction is somewhat different. They continue to argue that, in the case of still images, shadows are often analysed and exploited to infer geometric properties of the objects casting the shadow, as well as to enhance object localisation and measurements. In case of video sequences, shadow detection is performed to enhance the quality of segmentation results instead of deducing some imaging or object parameters.
While most of the shadow detection algorithms proposed in the literature have looked into the shadow detection in the spatial domain, very few approaches have investigated the use of the algorithms in the frequency domain (see chapter 2). Furthermore, there is no generic model that is applicable to both domains simultaneously. Modelling a shadow detection algorithm in different domains is of significant importance for image processing and video applications, for the reason that the algorithm will be applicable in either the unprocessed, raw (uncompressed) or the processed (compressed) video sequences.

In this chapter, shadow detection algorithms are introduced in the spatial and frequency domains. Section 5.2 gives an introduction to the spatial domain, and presents the proposed pixel domain shadow detection algorithm. Section 5.3 introduces shadow detection algorithms in the Fourier and wavelet domains. Finally, section 5.4 summarises this chapter.

5.2 Pixel domain

Digital images are matrices of integer numbers whose value determines a particular shade of grey for grey level images, or a specific colour for colour images. A grey level image can be represented by a function of two variables, \( z = f(x, y) \), where \( z \) is a number corresponding to a grey level at a point \( (x, y) \) [10]. A grey level digital image can therefore be considered as a discrete function:

\[
f_{ij} \quad \text{where} \quad f_{ij} = f(x_i, y_j)
\]

where, \( f_{ij} \) is the value of the function at \( x = x_i \) and \( y = y_j \) that defines a two-dimensional array or matrix of numbers, i.e.:
Chapter 5: Spatial and Transform Domains Shadow Detection Algorithms

If a real image is taken as a map of the intensity of light at a particular point, then it must be described by the following function [10]:

$$f_{ij} = \begin{pmatrix}
    f_{11} & f_{12} & \cdots & f_{1n} \\
    f_{21} & f_{22} & \cdots & f_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    f_{n1} & f_{n2} & \cdots & f_{nn}
\end{pmatrix}$$

Blackledge [10] stated that the process of converting $f(x, y)$ into $f_{ij}$ is called digitisation (Analogue-to-Digital conversion), or spatial quantisation, where the analogue image is sampled providing a matrix of discrete values typically on a rectangular grid. Blackledge also states that there are two elements to digitisation: the spatial quantisation, and the luminance quantisation. The spatial quantisation can be taken to be an approximation to the original image, i.e. $f_{ij}$ approximates $f(x, y)$. If $n^2$ samples are taken at regular intervals, then the approximation will improve as $n$ increases. The clarity of an image depends on the total number of pixels that are used to represent the image, Figure 5.1 illustrates the effect of sampling an image with less pixels [10]. As for the luminance quantisation, each pixel is assigned a discrete value, which is the level of greyness or luminance. The number of steps in the scale available for assignment is called the pixel depth (in bits). Very small depth results in inaccurate representations and loss of information. The number of shades of grey that can be present is related to the number of bits assigned to each pixel. When images are considered using just 1 bit, then binary images become applicable. Figure 5.2 shows the effect of changing the luminance quantisation of an image using 8, 4, 3 and 1 bits.
5.2.1 Pixel domain shadow detection

The shadow detection algorithm presented in the pixel domain uses background (reference) and foreground (object) frame pairs $F_{bg}$ and $F_{obj}$ for the detection of shadows in the object frame $F_{obj}$. See Figure 5.3.

The proposed algorithm processes the frames blockwise by comparing corresponding blocks, of size $k \times k$, in $F_{obj}$ and $F_{bg}$. We denote corresponding $k \times k$ block pairs in the pixel domain by $(P, P^*)$ where $P \in F_{bg}$ and $P^* \in F_{obj}$.

The proposed algorithm in the pixel domain initially sets $k$ to 16\(^1\). If a block of this size fails to classify as a shadow block, it is subdivided into four, $8 \times 8$ blocks for re-testing. This process is repeated on blocks that initially fail to classify as shadows, down to a block size $3 \times 3$. Blocks that fail to classify as shadow at this lowest level

\(^1\)The choice of initial block size is relative to the original image size, the object size, and the shadow size. Therefore, the initial block size of $16 \times 16$ is reasonable according to the given video sequences-see chapter 6. The lowest level of sub-division is $3 \times 3$, and is found to give better results than $4 \times 4$, and $2 \times 2$, where it is expected to perform better in shadow edges regions (compared to $4 \times 4$ and to be less sensitive to image noise compared to $2 \times 2$.
The mean and standard deviation of each block will be used throughout the algorithm, and are defined as follows: let $\bar{P}$ denotes the average of the pixel values in the block $P$, and $\sigma(P)$ denote the standard deviation, i.e.

$$\bar{P} = \frac{1}{k^2} \sum_{0 \leq i,j \leq k-1} p_{ij} \quad \text{and} \quad \sigma(P) = \left( \sum_{0 \leq i,j \leq k-1} (\bar{P} - p_{ij})^2 \right)^{\frac{1}{2}}$$

Similarly,

$$\bar{P}^* = \frac{1}{k^2} \sum_{0 \leq i,j \leq k-1} p_{ij}^* \quad \text{and} \quad \sigma(P^*) = \left( \sum_{0 \leq i,j \leq k-1} (\bar{P}^* - p_{ij}^*)^2 \right)^{\frac{1}{2}}$$

The $L^2$-metric is used throughout and denoted $\| \cdot \|_2$, as for instance:

$$\|P\|_2 = \left( \sum_{0 \leq i,j \leq k-1} p_{ij}^2 \right)^{\frac{1}{2}}$$

The proposed shadow detection algorithm in the pixel domain, is based on the shadow affine hypothesis described in section 3.2, where the shadow/non-shadow luminance values are related by the affine transformation:
Figure 5.3: Frames $F_{bg}$ (background frame) and $F_{obj}$ (object frame), are used for the detection of shadows in the object frame $F_{obj}$.

\[
L_o = \lambda L_e + \mu
\]

The parameter $\lambda$ can be estimated from the variance of the pixel values in block pairs $(P^*, P)$ as:

\[
\lambda = \frac{\sigma(P^*)}{\sigma(P)}
\]

Applying the luminance relation $L_o = \lambda L_e + \mu$ in the pixel domain, where $\bar{P}$ represents the average luminance value of a block, gives:

\[
\bar{P}^* = \lambda \bar{P} + \mu
\]

Therefore, $\mu$ can be computed as:

\[
\mu = \bar{P}^* - \lambda \bar{P}
\]

Based on the affine hypothesis proposed in section 3.2 and derived in section 3.3, the expression for $\delta$ can be written as:

\[
\delta = \frac{\|P^* - (\lambda P + \mu J_P)\|_2}{\|P^*\|_2} \approx 0,
\]
where $J_P$ is the $k \times k$ matrix defined by

$$(J_P)_{i,j} = 1 \quad \text{for all} \quad 0 \leq i, j < k$$

**Shadow condition:** As discussed in section 3.2, the light energy received at points $r \in n_q$ in the absence of an object casting a shadow over $n_q$ is affinely related, with a high degree of approximation, to the energy received when a shadow is cast over $n_q$ by an object. The same affine parameters being applicable to the entire neighbourhood $n_q$. Here, the algorithm is applied over the neighbourhood of a block of size $k \times k$.

The shadow condition for the block pair $(P, P^*)$ comprises two parts, namely that the luminance value of $P^*$ is lower than $P$: i.e.

$$\frac{P^*}{P} < 1$$

and that the affine condition

$$\frac{\|P^* - (\lambda P + \mu J_P)\|_2}{\|P^*\|_2} \approx 0.$$ 

holds for $(P, P^*)$.

Note that for a $4 \times 4$ block we have:

$$J_P = \begin{bmatrix}
1,1,1,1 \\
1,1,1,1 \\
1,1,1,1 \\
1,1,1,1 \\
\end{bmatrix}.$$ 

The pixel domain shadow detection algorithm can be summarised as shown in Algorithm 1.
Algorithm 1 Pixel Domain Shadow Detection Algorithm

1: \( k \leftarrow 16 \)

2: Partition \( F \) \& \( F^* \) into disjoint blocks \( P \) \& \( P^* \) of size \( k \times k \)

3: \( \forall P, P^* \) DO

4: \( \lambda = \frac{\sigma(P^*)}{\sigma(P)} \)

5: \( \mu = \tilde{P}^* - \lambda \tilde{P} \)

6: if \( \frac{P^*}{\tilde{P}} < 1 \land \frac{\|P^*-(\lambda P + \mu P)\|_2}{\|P^*\|_2} \approx 0 \) then

7: Block \( P^* \) is a SHADOW BLOCK

8: else

9: \( k \leftarrow 8 \), partition \( P \) \& \( P^* \) into disjoint blocks \( P_2 \) \& \( P_2^* \) of size \( k \times k \)

10: \( \forall P_2, P_2^* \) DO

11: \( \lambda = \frac{\sigma(P_2^*)}{\sigma(P_2)} \)

12: \( \mu = P_2^* - \lambda P_2 \)

13: if \( \frac{P_2^*}{P_2} < 1 \land \frac{\|P_2^*-(\lambda P_2 + \mu P_2)\|_2}{\|P_2^*\|_2} \approx 0 \) then

14: Block \( P_2^* \) is a SHADOW BLOCK

15: else

16: \( k \leftarrow 3 \), partition \( P_2 \) \& \( P_2^* \) into disjoint blocks \( P_3 \) \& \( P_3^* \) of size \( k \times k \)

17: \( \forall P_3, P_3^* \) DO

18: \( \lambda = \frac{\sigma(P_3^*)}{\sigma(P_3)} \)

19: \( \mu = P_3^* - \lambda P_3 \)

20: if \( \frac{P_3^*}{P_3} < 1 \land \frac{\|P_3^*-(\lambda P_3 + \mu P_3)\|_2}{\|P_3^*\|_2} \approx 0 \) then

21: Block \( P_3^* \) is a SHADOW BLOCK

22: else

23: BLOCK \( P^* \) DOES NOT CONTAIN SHADOW

24: end if

25: end if

26: end if
5.3 The Transform Domains

5.3.1 The Discrete Cosine Transform (DCT)

The DCT is an orthogonal transform in 2D signal processing. It is known to be almost optimal in terms of its energy compaction capabilities and can be computed via a fast algorithm [63]. The DCT is used in most of the international image/video compression standards of Joint Photographic Experts Group (JPEG), and Motion Picture Experts Group (MPEG) [34]. Technically, DCT is a Fourier-related transform similar to the Discrete Fourier transform (DFT), but applies for real numbers only [62].

The DCT operates on a block of $N \times N$ samples (pixels), the output is $\hat{B}$, a block of $N \times N$ coefficients, [64].

$$\hat{B}_{xy} = a_x a_y \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} P_{ij} \cos \left(\frac{(2j + 1)y\pi}{2N}\right) \cos \left(\frac{(2i + 1)x\pi}{2N}\right)$$

The inverse DCT transform (IDCT) can be written as:

$$P_{ij} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} a_x a_y \hat{B}_{xy} \cos \left(\frac{(2j + 1)y\pi}{2N}\right) \cos \left(\frac{(2i + 1)x\pi}{2N}\right)$$

The DCT (and its inverse, the IDCT) can be described in terms of a transformation matrix $A$. The Forward DCT (FDCT) of a $N \times N$ pixel block $P$ can be expressed as [64]:

$$\hat{B} = APA^T$$

and the Inverse DCT can be expressed as follows:

$$P = A^T \hat{B} A$$

where, $P$ is the matrix of samples, $\hat{B}$ is the matrix of the transform coefficients and $A$ is the $N \times N$ DCT transformation matrix. The elements of $A$ are defined as [64]:
\[ A_{ij} = a_i \cos \frac{(2j + 1)i\pi}{2N} \]

where \((i, j), 0 \leq i \leq N - 1 \) and \(0 \leq j \leq N - 1\) represent an element location in the \(N \times N\) matrix, \(a_i = \sqrt{\frac{1}{n}}\) when \(i = 0\) and \(a_i = \sqrt{\frac{2}{n}}\) when \(i > 0\).

According to Richardson [64], when \(N = 4\) the elements of the transform matrix \(A\) can be represented as:

\[
A = \begin{pmatrix}
\frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) & \frac{1}{2} \cos(0) \\
\sqrt{\frac{1}{2}} \cos(\frac{\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{5\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{7\pi}{8}) \\
\sqrt{\frac{1}{2}} \cos(\frac{3\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{5\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{11\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{13\pi}{8}) \\
\sqrt{\frac{1}{2}} \cos(\frac{5\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{11\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{15\pi}{8}) & \sqrt{\frac{1}{2}} \cos(\frac{21\pi}{8})
\end{pmatrix}
\]

i.e.

\[
A = \begin{pmatrix}
0.5000 & 0.5000 & 0.5000 & 0.5000 \\
0.6533 & 0.2706 & -0.2700 & -0.6530 \\
0.5000 & -0.5000 & -0.5000 & 0.5000 \\
0.2706 & -0.6530 & 0.6530 & -0.2700
\end{pmatrix}
\]

For instance, given a block of pixels \(P_1\) of size \(4 \times 4\), where:

\[
P_1 = \begin{pmatrix}
172 & 123 & 193 & 124 \\
181 & 140 & 139 & 135 \\
170 & 165 & 146 & 184 \\
147 & 197 & 189 & 128
\end{pmatrix}
\]

Applying the relation \(\hat{B} = AP_1A^T\) results in:

\[
\hat{B} = \begin{pmatrix}
121.25 & 26.654 & -12.750 & 27.1135 \\
-25.47 & 7.8765 & 28.9125 & 45.9905 \\
3.25 & 0.3379 & -52.750 & 31.5200 \\
\end{pmatrix}
\]
The block matrix now consists of 16 DCT coefficients, $a_{ij}$ where $i$ and $j$ range from 0 to 3. The top-left coefficient, $a_{00}$ (in image processing, this value is referred to as the dc value of the block) correlates to the lowest frequency of the original image block, which represents the average luminance value of the block. As we move away from $a_{00}$ in a zigzag order, the DCT coefficients (in image processing, these values are referred to as the ac values of the block) correlate to increasingly higher frequencies of the image block, where $a_{nn}$ corresponds to the highest frequency. The DCT coefficients represent the pattern of the texture inside the image block [64].

5.3.2 Fourier domain shadow detection

The shadow detection algorithm presented in the DCT domain uses background and object frame pairs $F_{bg}$ and $F_{obj}$ for the detection of shadows in the object frame $F_{obj}$. See Figure 5.3.

The proposed algorithm processes the frames blockwise by comparing corresponding blocks of size $k \times k$, in $F_{obj}$ and $F_{bg}$. We denote corresponding $k \times k$ block pairs in the DCT domain by $(B, B^*)$, where $B \in DCT(F_{bg})$ and $B^* \in DCT(F_{obj})$.

Similar to the method used in 5.2.1, the proposed algorithm in the DCT domain initially sets $k$ to 16. If a block of this size fails to classify as a shadow block, it is subdivided into four $8 \times 8$ blocks for re-testing. This process is repeated on blocks that initially fail to classify as shadows, down to a block size $3 \times 3$. Blocks that fail to classify as a shadow block at this lowest level of sub-division are classified as non-shadow.

The proposed shadow detection algorithm in the DCT domain, is based on the shadow affine hypothesis described in section 3.2, where the shadow/non-shadow luminance values are related as:

$$L_o = \lambda L_e + \mu.$$

Writing $f$ instead of $L$ the relationship may be represented as the transformation
Chapter 5: Spatial and Transform Domains Shadow Detection Algorithms

\( f \rightarrow \lambda f + \mu \). Writing [8, 9]:

\[
f(x, y) = a_{00} + \sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m} e^{inx} e^{imy}
\]

Note that, it is possible to define the transformation of the dc and ac coefficients from the Fourier series. Where \( a_{00} \) (dc) is essentially the average value of the transformation, \( a_{n,m} \) (ac) are the rest of the Fourier coefficients. The transformation \( f \rightarrow \lambda f + \mu \), in the Fourier representation in 2D is:

\[
a_{00} + \sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m} e^{inx} e^{imy} \rightarrow \lambda a_{00} + \lambda \left( \sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m} e^{inx} e^{imy} \right) + \mu
\]

Thus, the Fourier coefficients transform as:

\[
\{a_{00}, \{a_{n,m}\}\} \rightarrow \{\lambda a_{00} + \mu, \{\lambda a_{n,m}\}\}
\]

In particular, the dc coefficient is affinely transformed by \( (\lambda, \mu) \) and the ac coefficients are all scaled by \( \lambda \).

The parameter \( \lambda \) may be estimated from the averages (or sums) of the ac coefficients of block pairs. We have:

\[
\lambda = \frac{\sum ac^*}{\sum ac}
\]

Alternatively, to compute \( \lambda \), we have

\[
f(x, y) = a_{00} + \sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m} e^{inx} e^{imy}
\]

hence

\[
f(00) = a_{00} + \sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m}
\]

and

\[
\sum_{n=-\infty, n \neq 0}^{\infty} \sum_{m=-\infty, m \neq 0}^{\infty} a_{n,m} = f(0, 0) - a_{00}.
\]
The left-hand side is the sum of the ac values, \( f(0,0) \) is the function value at \( x = 0, y = 0 \) and is to be interpreted as the top left pixel value of a block; \( a_{00} \) is the dc value. This gives

\[
\lambda = \frac{P'(0,0) - a_{00}}{P(0,0) - a_{00}}
\]

Using the shadow/non-shadow luminance relation \( L_o = \lambda L_e + \mu \) in the DCT domain, where \( a_{00} \) represents the average luminance value of a block, this gives:

\[
a_{00}^* = \lambda a_{00} + \mu
\]

Thus, \( \mu \) can be computed as:

\[
\mu = a_{00}^* - \lambda a_{00} \quad \text{i.e.} \quad \mu = dc^* - \lambda dc
\]

Based on the affine hypothesis proposed in section 3.2 and derived in section 3.3, the expression for \( \delta \) can be written as:

\[
\delta = \frac{\|B^* - (\lambda B + \mu J_{DCT})\|_2}{\|B^*\|_2} \approx 0,
\]

where \( J_{DCT} \) is the \( k \times k \) matrix defined by

\[
(J_{DCT})_{i,j} = \begin{cases} 1 & \text{for } i = j = 0 \\ 0 & \text{otherwise.} \end{cases}
\]

**Shadow condition:** As discussed in section 3.2, the light energy received at points \( r \in n_q \) in the absence of an object casting a shadow over \( n_q \) is affinely related, with a high degree of approximation, to the energy received when a shadow is cast over \( n_q \) by an object. The same affine parameters being applicable to the entire neighbourhood \( n_q \). Here, the algorithm is applied over the neighbourhood of a block of size \( k \times k \).
Algorithm 2: DCT Domain Shadow Detection Algorithm

1: \( k \leftarrow 16 \)

2: Partition \( F \) and \( F^* \) into disjoint blocks \( P \) and \( P^* \) of size \( k \times k \)

3: \( \forall_{P, P^*} \) DO

4: \( B = DCT(P) \) and \( B^* = DCT(P^*) \)

5: \( \lambda = \frac{p_{00}^{*} - p_{00}^{*}}{p_{00}^{*} - p_{00}^{*}} \)

6: \( \mu = a_{00}^{*} - \lambda a_{00}^{*} \)

7: if \( \frac{p_{00}^{*}}{p_{00}^{*}} < 1 \wedge \frac{\|B^* - (\lambda B + \mu J_{DCT})\|_2}{\|B^*\|_2} \approx 0 \) then

8: BLOCK \( P^* \) is a SHADOW BLOCK

9: else

10: \( k \leftarrow 8 \), partition \( P \) and \( P^* \) into disjoint blocks \( P_2 \) and \( P_2^* \) of size \( k \times k \)

11: \( \forall_{P_2, P_2^*} \) DO

12: \( B_2 = DCT(P_2) \) and \( B_2^* = DCT(P_2^*) \)

13: \( \lambda = \frac{p_{20}^{*} - a_{20}^{*}}{p_{20}^{*} - a_{20}^{*}} \)

14: \( \mu = a_{20}^{*} - \lambda a_{20}^{*} \)

15: if \( \frac{p_{20}^{*}}{p_{20}^{*}} < 1 \wedge \frac{\|B_2^* - (\lambda B_2 + \mu J_{DCT})\|_2}{\|B_2^*\|_2} \approx 0 \) then

16: BLOCK \( P_2^* \) is a SHADOW BLOCK

17: else

18: \( k \leftarrow 3 \), partition \( P_2 \) and \( P_2^* \) into disjoint blocks \( P_3 \) and \( P_3^* \) of size \( k \times k \)

19: \( \forall_{P_3, P_3^*} \) DO

20: \( B_3 = DCT(P_3) \) and \( B_3^* = DCT(P_3^*) \)

21: \( \lambda = \frac{p_{30}^{*} - a_{30}^{*}}{p_{30}^{*} - a_{30}^{*}} \)

22: \( \mu = a_{30}^{*} - \lambda a_{30}^{*} \)

23: if \( \frac{p_{30}^{*}}{p_{30}^{*}} < 1 \wedge \frac{\|B_3^* - (\lambda B_3 + \mu J_{DCT})\|_2}{\|B_3^*\|_2} \approx 0 \) then

24: BLOCK \( P_3^* \) is a SHADOW BLOCK

25: else

26: BLOCK \( P^* \) DOES NOT CONTAIN SHADOW

27: end if

28: end if

29: end if
The shadow condition for the block pair \((B, B^*)\) comprises two parts, namely that the luminance value of \(B^*\) is lower: i.e.

\[
\frac{a_{00}^*}{a_{00}} < 1 \quad \text{i.e.} \quad \frac{dc^*}{dc} < 1
\]

and that the affine condition

\[
\frac{\|B^* - (\lambda B + \mu J_{DCT})\|_2}{\|B^*\|_2} \approx 0.
\]

holds for \((B, B^*)\).

Note that for a \(4 \times 4\) block we have:

\[
J_{DCT} = \begin{bmatrix}
1, 0, 0, 0 \\
0, 0, 0, 0 \\
0, 0, 0, 0 \\
0, 0, 0, 0
\end{bmatrix}
\]

The DCT domain shadow detection algorithm can be summarised as shown in Algorithm 2:

### 5.3.3 The Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) is named after Alfred Haar, a Hungarian mathematician. According to Richardson [64], the DWT is applied to a discrete signal containing \(N\) samples, the signal is decomposed into a low frequency band \((L)\) and a high frequency band \((H)\), using a low-pass filter and a high-pass filter, respectively. Each band is sub-sampled by a factor of two, i.e. each contains \(N/2\) samples. For a \(2D\) signal such as an intensity image. Firstly, each row of the image is filtered with a low-pass and a high-pass filter \((L_x\) and \(H_x\) ) and the output of each filter is sub-sampled by a factor of two to produce the intermediate images \(L\) and \(H\). \(L\) is the low-pass filtered image and sub-sampled in the \(x\)-direction and \(H\) is the high-pass filtered and also sub-sampled in the \(x\) direction [64].
Secondly, each column of these new row-wise transformed images is filtered with low-pass and high-pass filters \((L_y \text{ and } H_y)\) and sub-sampled by a factor of two to produce four sub-images \((LL, LH, HL \text{ and } HH)\). These four sub-band images together comprise the same number of samples as the original. \(LL\) is the original image, low-pass filtered in horizontal and vertical directions and sub-sampled by a factor of 2. \(HL\) is high-pass filtered in the vertical direction and contains residual vertical frequencies. \(LH\) is high-pass filtered in the horizontal direction and contains residual horizontal frequencies and \(HH\) is high-pass filtered in both horizontal and vertical directions. The 2D wavelet decomposition can be applied again to the \(LL\) image, forming four new sub-band images. The resulting low-pass image (always the top-left sub-band image) is iteratively filtered to create a tree of sub-band images [64] (see Figure 5.4).
5.3.4 Wavelet domain shadow detection

The proposed DWT shadow detection algorithm uses background and object frame pairs \( F_{bg} \) and \( F_{obj} \) for the detection of shadows in the object frame \( F_{obj} \) (see Figure 5.3). The proposed algorithm processes the frames blockwise by comparing corresponding blocks, of size \( k \times k \), in \( F_{obj} \) and \( F_{bg} \). We denote corresponding \( k \times k \) block pairs in the DWT domain by \( (D, D^*) \), where \( D \in DWT(F_{bg}) \) and \( D^* \in DWT(F_{obj}) \).

The 'Haar' wavelet is used in the proposed DWT shadow detection algorithm. The decomposition process repeats until \( k = 2 \). Therefore, the total number of decomposition levels required depends on the block-size. If a \( 16 \times 16 \) block is used, then 3 level block decomposition is required. If \( 8 \times 8 \) is used then 2 level block decomposition is required. If \( 4 \times 4 \) block size is used then 1 level decomposition is required (see figure 5.5).

The proposed shadow detection algorithm in the DWT domain, is based on the affine shadow hypothesis described in section 3.2, where the shadow/non-shadow
luminance values are related as:

\[ L_o = \lambda L_e + \mu. \]

Writing \( f \) instead of \( L \), the relationship may be represented as the transformation \( f \rightarrow \lambda f + \mu \). Writing [8, 9]:

\[ f(x, y) = b_{00} + \sum_{n \neq 0} \sum_{m \neq 0} b_{n,m} h_n(x) h_m(y) \]

where \( b_{00} \) is essentially the average value of the block, \( b_{n,m} \) are the rest of the wavelet coefficients. \( h \) is the mother wavelet (here the Haar wavelet is used).

The transformation \( f \rightarrow \lambda f + \mu \), in the wavelet representation is (note that the transformation is presented here in 2D):

\[
\begin{align*}
b_{00} + \sum_{n \neq 0} \sum_{m \neq 0} b_{n,m} h_n(x) h_m(y) & \rightarrow \lambda b_{00} + \lambda \left( \sum_{n \neq 0} \sum_{m \neq 0} b_{n,m} h_n(x) h_m(y) \right) + \mu \\
& = (\lambda b_{00} + \mu) + \sum_{n \neq 0} \sum_{m \neq 0} \lambda (b_{n,m}) h_n(x) h_m(y).
\end{align*}
\]

and the wavelet coefficients transform as:

\[
\{b_{00}, \{b_{n,m}\}\} \rightarrow \{\lambda b_{00} + \mu, \{\lambda b_{n,m}\}\}
\]

In particular the \( b_{00} \) corresponds to the average of the \( LL \) coefficients \( (LL) \), which is affinely transformed by \( (\lambda, \mu) \) and \( b_{n,m} \) corresponds to the rest of the coefficients in \( LH, HL, HH \) which all are scaled by \( \lambda \).

The parameter \( \lambda \) can be estimated from the averages of the \( LH, HL, HH \) coefficients of block pairs. We have:

\[
\lambda = \frac{LH^* + HL^* + HH^*}{LH + HL + HH}
\]
Alternatively, to compute $\lambda$, we have:

$$f(x, y) = b_{00} + \sum_{n \neq 0} \sum_{m \neq 0} b_{n,m} h_n(x) h_m(y)$$

hence

$$f(0, 0) = b_{00} + \sum_{n \neq 0} \sum_{m \neq 0} b_{n,m}$$

and

$$\sum_{n \neq 0} \sum_{m \neq 0} b_{n,m} = f(0, 0) - b_{00}.$$

The left-hand side is the sum of the $b_{n,m}$ coefficients ($LH$, $HL$, $HH$). $f(0, 0)$ is the function value at $x = 0$, $y = 0$ and is to be interpreted as the top left pixel value of a block; $b_{00}$ is the average luminance value of the block. This gives:

$$\lambda = \frac{P^*(0, 0) - LL^*}{P(0, 0) - LL}$$

Using the shadow/non-shadow luminance relation $L_o = \lambda L + \mu$ in the DWT domain, where $LL$ represents the average luminance value of a block, this gives:

$$LL^* = \lambda LL + \mu$$

Thus, $\mu$ can be computed as:

$$\mu = LL^* - \lambda LL$$

Based on the affine hypothesis proposed in section 3.2 and the derived in section 3.3, the expression for $\delta$ can be written as:

$$\delta = \frac{\|D^* - (\lambda D + \mu J_{DWT})\|_2}{\|D^*\|_2} \approx 0,$$

where $J_{DWT}$ is the $k \times k$ matrix defined by

$$(J_{DWT})_{i,j} = \begin{cases} 1 & \text{for } i = j = 0 \\ 0 & \text{otherwise.} \end{cases}$$
Shadow condition: As discussed in section 3.2, the light energy received at points \( r \in n_q \) in the absence of an object casting a shadow over \( n_q \) is affinely related, with a high degree of approximation, to the energy received when a shadow is cast over \( n_q \) by an object. The same affine parameters being applicable to the entire neighbourhood \( n_q \). Here, the algorithm is applied over the neighbourhood of a block of size \( k \times k \).

The shadow condition for the block pair \((D, D^*)\) again comprises two components, namely that the luminance value of \( D^* \) is lower: i.e.

\[
\frac{b^*_{00}}{b_{00}} < 1 \quad \text{i.e.} \quad \frac{L^*}{L} < 1
\]

and that the affine condition

\[
\frac{\|D^* - (\lambda D + \mu J_{DWT})\|_2}{\|D^*\|_2} \approx 0.
\]

holds for \((D, D^*)\).

Note that for a \(4 \times 4\) block we have:

\[
J_{DWT} = \begin{bmatrix}
1,0,0,0 \\
0,0,0,0 \\
0,0,0,0 \\
0,0,0,0
\end{bmatrix}.
\]

The DWT domain shadow detection algorithm can be summarised as shown in Algorithm 3.

5.4 Summary

This chapter discussed the proposed shadow detection algorithms in the spatial (pixel) and transform domains (Fourier and wavelet). They use background and object frame pairs \( F \) and \( F^* \), for the detection of moving shadows in the object frame.
Algorithm 3 DWT Domain Shadow Detection Algorithm

1: \( k \leftarrow 16 \)

2: Partition \( F \land F^* \) into disjoint blocks \( P \land P^* \) of size \( k \times k \)

3: \( \forall_{P,P^*} \) DO

4: \( D = DWT(P) \land D^* = DWT(P^*) \) \{1st decomposition level\}

5: \( D2 = DWT(D) \land D2^* = DWT(D^*) \) \{2nd decomposition level\}

6: \( D3 = DWT(D2) \land D3^* = DWT(D2^*) \) \{3rd decomposition level\}

7: \( \lambda = \frac{LH^* + HL^* + HH^*}{LH + HL + HH} \)

8: \( \mu = LL^* - \lambda LL \)

9: if \( \frac{\|D3^* - (\lambda D3 + \mu DWT)\|_2}{\|D3^*\|_2} \approx 0 \) then

10: BLOCK \( P^* \) is a SHADOW BLOCK

11: else

12: \( k \leftarrow 8 \), partition \( P \land P^* \) into disjoint blocks \( P2 \land P2^* \) of size \( k \times k \)

13: \( \forall_{P2,P2^*} \) DO

14: \( D = DWT(P2) \land D^* = DWT(P2^*) \) \{1st decomposition level\}

15: \( D2 = DWT(D) \land D2^* = DWT(D^*) \) \{2nd decomposition level\}

16: \( \lambda = \frac{LH^* + HL^* + HH^*}{LH + HL + HH} \)

17: \( \mu = LL^* - \lambda LL \)

18: if \( \frac{\|D2^* - (\lambda D2 + \mu DWT)\|_2}{\|D2^*\|_2} \approx 0 \) then

19: BLOCK \( P2^* \) is a SHADOW BLOCK

20: else

21: \( k \leftarrow 4 \), partition \( P2 \land P2^* \) into disjoint blocks \( P3 \land P3^* \) of size \( k \times k \)

22: \( \forall_{P3,P3^*} \) DO

23: \( D = DWT(P3) \land D^* = DWT(P3^*) \) \{1st decomposition level\}

24: \( \lambda = \frac{LH^* + HL^* + HH^*}{LH + HL + HH} \)

25: \( \mu = LL^* - \lambda LL \)

26: if \( \frac{\|D3^* - (\lambda D3 + \mu DWT)\|_2}{\|D3^*\|_2} \approx 0 \) then

27: BLOCK \( P3^* \) is a SHADOW BLOCK

28: else

29: BLOCK \( P^* \) DOES NOT CONTAIN SHADOW

30: end if

31: end if

32: end if
The algorithms work either in the pixel, DCT, or DWT domains and process the frames blockwise by comparing corresponding blocks, of size $k \times k$, in $F$ and $F^*$. 

As discussed in section 3.2 the light energy received at points $r \in n_q$ in the absence of an object casting a shadow over $n_q$ is affinely related, to a high degree of approximation, to the energy received when a shadow is cast over $n_q$ by an object. The same affine parameters applicable to the entire neighbourhood $n_q$. Here, the algorithm is applied over the neighbourhood of a block of size $k \times k$. In all of the proposed algorithms, the shadow condition for the corresponding block pair comprises two components, namely, that the luminance value of the shadow block in the object frame is lower than the luminance value of the non-shadow block in the background frame, and that the proposed affine shadow condition holds for the selected block pair.

The success of the approach depends crucially on the accurate determination of candidate affine parameters $\lambda$ and $\mu$, for each block pair in the pixel or the transform domains with which to test the neighbourhood affine condition. As video data is inherently noisy, this suggests estimating candidate values using statistical measures that have error and noise reducing properties. To this effect in the DCT domain the dc value and the sum of the ac values suggest effective measurements, while in the DWT domain, the LL value and the sum of $LH, HL, HH$ values also suggest effective measurements.

Chapter 6, provides the performance evaluation metrics, compares the results with the state of the art algorithms, and discusses other methods for choosing the most appropriate reference frame.
Chapter 6

Shadow Detection in Video Sequences

6.1 Overview

Richardson [64] describes a digital video as a representation of a natural visual scene, sampled spatially and temporally. A scene is sampled at a point in time using progressive sampling to produce a frame or using interlaced sampling to produce a field. Sampling is repeated at intervals (e.g. 1/25 or 1/30 second intervals) to produce a moving video signal. Each spatio-temporal sample (pixel) is represented as a number or set of numbers that describes the brightness (luminance) and colour of the sample. To obtain a 2D sampled image, a camera focuses a 2D projection of the video scene onto a sensor, such as an array of Charge Coupled Devices (CCD). In the case of colour image capture, each colour component is separately filtered and projected onto a CCD array [64].

A video is captured by taking rectangular pictures of the scene at periodic time intervals. Playing back the series of frames produces the appearance of motion. A higher temporal sampling rate (frame rate, measured by number of frames per second $f/s$) gives smoother motion in the video scene, but requires more samples to be
Chapter 6: Shadow Detection in Video Sequences

captured and stored. Frame rates below 10 f/s are used for very low bit-rate video communications (because the amount of data is relatively small). Sampling at 25 or 30 f/s is standard for television pictures; 50 or 60 f/s produces smooth motion at the expense of a very high data rate [34, 64].

Amongst the wide variety of video applications, video object tracking and recognition are of high importance for security systems and video surveillance applications, where shadow detection and removal stands at the core of successful tracking and recognition systems. In this chapter, section 6.2 gives an introduction to video surveillance, and object tracking applications. Section 6.3 introduces the experimental results and the performance evaluation of the proposed algorithms, and compares the results against the state-of-art algorithms. Section 6.4 discusses other methods of choosing the most appropriate reference frame. Finally, section 6.5 summarises this chapter.

6.2 Introduction

Automated surveillance systems address real time observation of people and vehicles within busy environments. However, object tracking and recognition is considered as the most important, yet error prone component of a surveillance system [1]. A large number of surveillance systems have been designed in recent years. The approach used by Stauffer and Grimson [37] uses an adaptive multi-modal background subtraction method that can deal with slow changes in illumination, and repeated motion estimation from background clutter and long term scene changes. After background subtraction, the detected objects are tracked using a multiple hypothesis tracker. The PFinder tracking system [79] uses a background model to locate the objects, and tracks the full body of a person. The system assumes only a single person is present in the scene. The W4 system [38] uses dynamic appearance models to track people. Individual human figures and groups are distinguished using projection histograms,
and are tracked by their heads. Ricquebourg and Bouthemy [65] in their system, track people by exploiting spatio-temporal slices. The detection scheme involves the use of intensity and temporal differences between successive images. Al-Najdawi et al. [1] proposed a real-time object tracking system based on a limited, discontinuous feature set. The proposed system uses a simplified version of the Kanade-Lucas-Tomasi (KLT) [74] algorithm to detect features of both continuous and discontinuous nature.

Beside the problems caused by cast shadows, a number of problems including object occlusion and background changes arise in real environments. These problems need to be resolved. Occlusion occurs when an object of interest is not visible in the scene, since another object is blocking its view. Tracking objects under occlusion is complicated because it is difficult to determine the accurate position and velocity of an occluded object [44].

According to the Comaniciu et al. [17], motion predictors and estimators have been used in the literature to solve the problem of occlusion. The choice of these algorithms and the data association methodologies has direct impact on tracking systems. The right choice is highly related to the tracking scenario. As for instance, the Particle Filtering introduced by Isard and Blake [41]. Al-Najdawi et al. [1], employed a Kalman filter [47] to seek optimal estimates in object tracking. Boykov and Huttenlocher [11] employed the Kalman filter to track vehicles in an adaptive framework. Rosales and Sclaroff [66] used the Extended Kalman Filter to estimate a 3D object trajectory from 2D image motion. Blake and MacCormick [53] introduced the probabilistic exclusion for tracking multiple objects. Wu and Huang [80] developed an algorithm to integrate multiple target clues. Li and Chellappa [51] proposed simultaneous tracking and verification based on particle filters applied to vehicles and faces. Chen et al. [15] used the Hidden Markov Model formulation for tracking, combined with JPDAF data association. Rui and Chen [68] proposed to track the contour of a human face based on a particle filter.

Most of the automated surveillance systems employ static background subtrac-
tion methods to extract objects from a scene in order to track them. However, background subtraction methods cannot deal with changes in lighting or large illumination variations, which have the impact of altering the colour characteristics of the background with time [44]. Using a static background is applicable for short duration video sequences, where scene illumination is relatively stable. As for long duration video sequences, a dynamic background model is required to continuously update the background frame. Most of the existing tracking applications apply the dynamic background generation technique proposed by Stauffer and Grimson [37] as a pre-processing step to the background subtraction stage.

6.3 Shadow Detection in Video Sequences with Dedicated Background Image

As stated earlier, the shadow detection algorithms presented use background and object frame pairs $F$ and $F^*$ for the detection of moving shadows in the object frame $F^*$, as in Figure 6.1. The algorithms work either in the pixel, DCT, or DWT domains and process the frames blockwise by comparing corresponding blocks, of size $k \times k$, in $F$ and $F^*$. In any given domain, if both of the corresponding shadow conditions presented in sections 5.2.1, 5.3.2, and 5.3.4 hold, then the block is considered as a shadow block. Therefore, the process of shadow removal can be done by replacing the shadow detected block in $F^*$ with the corresponding background block in $F$. Figure 6.1 and Figure 6.5 represent a sample of the results obtained by using the pixel domain shadow detection and removal algorithm. Figure 6.2 and Figure 6.4 respectively, represent samples of the results obtained by using the DCT and DWT domain shadow detection and removal algorithms.

Additional results of the proposed algorithms in the pixel and transform domains are available in Appendix B, Figures: B.1-B.11.

As discussed in chapter 5, the proposed algorithms in the pixel and Fourier do-
Figure 6.1: The benchmark “Hall” video sequence, represents an indoor video sequence, with multiple combinations of light sources, spectrally equal and of equal intensities. This figure illustrates the use of the background and object frame to detect shadows. In the processed frame, shadows have been removed using the pixel domain shadow detection and removal algorithm.

The algorithm proceeds as follows. Each pixel is classified either as shadow or non-shadow. If the pixel is classified as a shadow, the pixel is classified as shadow. Otherwise, if the pixel is classified as non-shadow, it is subdivided into four, 8x8 blocks for re-testing. This process is repeated on blocks that fail to classify as shadow, down to block size 3x3. Blocks that fail to classify as shadow at this level of sub-division are classified as non-shadow blocks (see Figure 6.3). As for the DWT domain, the algorithm initially sets the block size k to 16 and a 3 level decomposition is done. If a block of this size fails to classify as shadow, the original pixel block is then subdivided into four, 8x8 blocks for re-testing, and a 2 level decomposition is done. This process repeats on blocks that fail to classify as shadow, down to a block size 4x4 where 1 level decomposition is done. Blocks that fail to classify as shadow at this level of sub-division are classified...
Figure 6.2: "Side" video sequence, represents an indoor video sequence, with arbitrary combination of light sources, with different intensities. This figure illustrates the use of the DCT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence, frames (A', B', C') are the corresponding processed frames.

Experiments have been performed in both high and low-resolution video sequences. The low-resolution camera captures less texture details and has higher noise levels, creating a significant challenge for any shadow detection algorithm. The algorithms have been tested with more than 1000 randomly selected frames, chosen from a set of 14 test video sequences. These sequences comprise more than 30,000 frames representing simple/complex, indoor/outdoor scenes with different numbers of objects, under different lighting and environmental conditions. Four of these video sequences are popular benchmark videos available in the public domain (see Figure 6.1, and Figure 6.6). Table 6.1 illustrates those videos with a classified description of their complexity. The benchmark sequences are used by existing shadow detection approaches to compare and evaluate their results against other approaches. In addition
Figure 6.3: Example of using different block sizes in the shadow detection and removal algorithm. The algorithm starts with a $16 \times 16$ block size down to size $3 \times 3$ as required.

to the benchmark videos, new video sequences were created, where scenes and cast shadows complexity are increased, in order to test and evaluate the performance of the proposed algorithms under different conditions. Table 6.2 lists the new video sequences and gives a classified description for each of them. In the new video sequences, the camera settings (aperture, shutter, speed, sampling rate) are all set in the automatic mode. The physical conditions in the benchmark and the new video sequences include:

- Background surface: carpets, wooden floors, concrete textured walls, concrete neutral walls, asphalt roads.

- Distance to objects: 2-250 feet.

- Lighting conditions in indoor environments: single point source and neutral walls, single point source and textured walls, multiple point sources spectrally equal but of different intensities, multiple point sources spectrally equal and of equal intensities, multiple point sources with arbitrary combination of lights.
Figure 6.4: “Sara”, represents an indoor video sequence, with multiple combination of light sources, spectrally equal and of equal intensities. This figure illustrates the use the DWT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence, frames (A', B', C') are the corresponding processed frames.

- Lighting conditions in outdoor environment: sunlight, overcast, dusk and dawn with artificial lights, dusk and dawn without artificial lights, night with single light source, and night with multiple light sources.

- Surface orientation: vertical, horizontal, and sloping.

The proposed algorithms have been implemented in Matlab, using the image processing tools box. As shown in Appendix A, Table A.1-Table A.2 provide a qualitative evaluation and classification for the proposed algorithms and a comparison against the existing approaches in terms of object and environment dependency, and the applicability in the other domains. Choosing the best colour space for shadow detection algorithms is a critical task in designing a good shadow detection algorithm. Colour spaces are different bases for representing intensity and colour information in colour
Figure 6.5: "Holywell", represents an outdoor video sequence with strong sun light, and strong shadows cast on the road. The left hand side image represents the original image and the right hand side image is the corresponding processed one. This figure illustrates the use the pixel domain shadow detection and removal algorithm.

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Sequence Type</th>
<th>Image size</th>
<th>Shadow Strength</th>
<th>Shadow Size</th>
<th>Object Size</th>
<th>Noise Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hall</td>
<td>indoor</td>
<td>320x240</td>
<td>low</td>
<td>medium</td>
<td>large</td>
<td>very low</td>
</tr>
<tr>
<td>Campus</td>
<td>outdoor</td>
<td>352x288</td>
<td>low</td>
<td>large</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Laboratory</td>
<td>indoor</td>
<td>320x240</td>
<td>low</td>
<td>medium</td>
<td>medium</td>
<td>medium</td>
</tr>
<tr>
<td>Intelligent room</td>
<td>indoor</td>
<td>320x240</td>
<td>very low</td>
<td>small</td>
<td>small</td>
<td>medium</td>
</tr>
</tbody>
</table>

Table 6.1: Benchmark video sequences, conclusion drawn from sources.

images. Usually colour spaces have three components or channels for representing all possible colour and intensity information. As seen in the literature (see Chapter 2), many different approaches are modelled using different colour spaces. Kumar et al. [49] presented a comprehensive comparative study of shadow detection algorithms that use different colour spaces. In their study, the colour spaces considered were: RGB, XYZ, YUV, HSV, and the normalised rgb. Quantitative and qualitative results are provided, evaluation of the correctness of the detection is based on true detection, false detection, and the total number of misclassifications. To summarise their results, it is shown that the YUV colour space is the best for optimal shadow de-
According to Kumar et al's. accuracy measurement results, the YUV colour space stands on the top of the list with the highest true detection rate, lowest number of misclassifications, and the lowest false detection rate. YUV is followed in descending order of accuracy by RGB, rgb, XYZ, and HSV.

### 6.3.1 Performance evaluation metrics

In order to systematically evaluate shadow detectors, it is useful to identify the following two important quality measures: good detection (low probability of miscla-
Table 6.2: The new video sequences description.

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Sequence Type</th>
<th>Image size</th>
<th>Shadow Strength</th>
<th>Shadow Size</th>
<th>Object Size</th>
<th>Noise Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corridor</td>
<td>indoor</td>
<td>320x240</td>
<td>medium</td>
<td>large</td>
<td>large</td>
<td>high</td>
</tr>
<tr>
<td>Sara</td>
<td>indoor</td>
<td>320x240</td>
<td>medium</td>
<td>large</td>
<td>large</td>
<td>medium</td>
</tr>
<tr>
<td>Asphalt</td>
<td>outdoor</td>
<td>320x240</td>
<td>very low</td>
<td>large</td>
<td>large</td>
<td>medium</td>
</tr>
<tr>
<td>Side</td>
<td>indoor</td>
<td>320x240</td>
<td>high</td>
<td>large</td>
<td>large</td>
<td>medium</td>
</tr>
<tr>
<td>Ghost</td>
<td>indoor</td>
<td>320x240</td>
<td>high</td>
<td>very large</td>
<td>very large</td>
<td>low</td>
</tr>
<tr>
<td>Legs</td>
<td>indoor</td>
<td>320x240</td>
<td>very high</td>
<td>large</td>
<td>large</td>
<td>medium</td>
</tr>
<tr>
<td>Holywell</td>
<td>outdoor</td>
<td>320x240</td>
<td>high</td>
<td>large</td>
<td>large</td>
<td>medium</td>
</tr>
<tr>
<td>Farnham</td>
<td>outdoor</td>
<td>320x240</td>
<td>medium</td>
<td>medium</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Door</td>
<td>outdoor</td>
<td>320x240</td>
<td>low</td>
<td>small</td>
<td>medium</td>
<td>low</td>
</tr>
<tr>
<td>Falkner</td>
<td>outdoor</td>
<td>320x240</td>
<td>very high</td>
<td>large</td>
<td>large</td>
<td>very high</td>
</tr>
<tr>
<td>Eggington</td>
<td>outdoor</td>
<td>320x240</td>
<td>very high</td>
<td>large</td>
<td>large</td>
<td>very high</td>
</tr>
</tbody>
</table>

Classifying a shadow point and good discrimination (the probability of classifying non-shadow points as shadow should be low, i.e. low false alarm rate). Good detection corresponds to minimising the number of false negatives (FN), i.e. the shadow points classified as background/foreground. Good discrimination corresponds to minimising the number of false positives (FP), i.e. the foreground/background points detected as shadows [61].

Onoguchi in his work [58] proposed two metrics for moving object detection evaluation: the Detection Rate (DR) and the False Alarm Rate (FAR). Assuming TP as the number of true positives (i.e. the shadow points correctly identified), these two metrics are defined as follows:

\[
DR = \frac{TP}{TP + FN} \quad \text{and} \quad FAR = \frac{FP}{TP + FP}
\]

Prati et al. [61] in their work showed that Onguchi metrics are not selective enough for the evaluation of shadow detection methods, since the metrics do not take into account whether a point detected as shadow belongs to a foreground object or to the background. Therefore, if shadow detection is used to improve moving object...
Chapter 6: Shadow Detection in Video Sequences

Detection, only the first case is problematic, since false positives belonging to the background affect neither the object detection nor the object shape. To account for this, they have modified the above metrics, defining the shadow detection rate $\eta$ and the shadow discrimination rate $\nu$ as follows:

$$\eta = \frac{TP_s}{TP_s + FN_s} \quad \text{and} \quad \nu = \frac{T\bar{P}_F}{T\bar{P}_F + F\bar{N}_F}$$

where $S$ denotes shadow and $F$ denotes foreground. $T\bar{P}_F$ is the number of ground-truth points of the foreground objects minus the number of points detected as shadows, but belonging to foreground objects.

Despite the above efforts, a reliable and objective way to evaluate the robustness of a shadow detection algorithm is still lacking in the literature. To compute the evaluation metrics described above, the ground truth for each frame is necessary. The ground truth is obtained by segmenting the images with an accurate manual classification of points in the foreground, background, and shadow regions. Based on the metrics of Prati et al., Table 6.3 provides a quantitative comparison between the proposed algorithms and the state of the art algorithms, based on the benchmark video sequences. Similarly, Table 6.4 provides a quantitative evaluation of the proposed algorithms based on the new video sequences.

The performance measurements show that the proposed algorithms, when applied to the benchmark videos, outperform most of the state of the art approaches, with significant enhancements of good detection and discrimination rates. The proposed algorithms also perform well when applied to most of the new video sequences. However, as shown in Table 6.4, the algorithms did not perform as expected when applied to the new video sequences "Falkner" and "Eggington". The $\nu$ metric results show high numbers of misclassified foreground points. These two sequences are created using low resolution cameras at night time with single and multiple street light sources. The noise levels in these two sequences are relatively high, and the cast shadows are relatively strong. It is clear that the proposed algorithms failed to classify shadows
Table 6.3: Quantitative evaluation of the proposed and the state-of-art algorithms, based on the benchmark video sequences. The results of the first four algorithms are obtained from Prati et al. work [61]. Siala et al [72] provided their own results as shown in the fifth algorithm.

Correctly in these two video sequences, with poor shadow detection and discrimination results. However, in such environments, there is no known algorithm that can give better results. Shadow detection results can be enhanced in such environments provided that a higher resolution camera with wider aperture settings can be used to capture more texture details and less noise.

Selected threshold values throughout the algorithms are \( \approx 0 \), and range within the following scale [0.1 ... 1.0]. Selecting threshold values is done experimentally, with an approximate assumption of the value based on the scene environment. Strong shadows in outdoor environments with strong sunlight, or scenes at night time tend to destroy texture information. Therefore the threshold value will approach the upper limit of the scale. Weak shadows in indoor environments and overcast situations do not affect the texture information as much as the strong shadows do. Therefore threshold values will approach the lower bound of the scale. An automatic approach to select threshold values is feasible, as a pre-processing stage of the shadow detection algorithms. The automatic approach should consider the scene environment, and lighting conditions.
Chapter 6: Shadow Detection in Video Sequences

<table>
<thead>
<tr>
<th>Video Sequence</th>
<th>Pixel Domain ( \eta % )</th>
<th>( \nu % )</th>
<th>DCT Domain ( \eta % )</th>
<th>( \nu % )</th>
<th>DWT Domain ( \eta % )</th>
<th>( \nu % )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corridor</td>
<td>84.36%</td>
<td>82.65%</td>
<td>86.84%</td>
<td>90.43%</td>
<td>84.32%</td>
<td>82.78%</td>
</tr>
<tr>
<td>Sara</td>
<td>92.12%</td>
<td>91.34%</td>
<td>93.99%</td>
<td>90.84%</td>
<td>92.76%</td>
<td>91.46%</td>
</tr>
<tr>
<td>Asphalt</td>
<td>92.51%</td>
<td>90.08%</td>
<td>95.61%</td>
<td>94.58%</td>
<td>96.19%</td>
<td>91.27%</td>
</tr>
<tr>
<td>Side</td>
<td>89.82%</td>
<td>87.35%</td>
<td>90.27%</td>
<td>89.48%</td>
<td>91.24%</td>
<td>86.40%</td>
</tr>
<tr>
<td>Ghost</td>
<td>93.02%</td>
<td>84.25%</td>
<td>92.63%</td>
<td>92.70%</td>
<td>92.04%</td>
<td>90.47%</td>
</tr>
<tr>
<td>Legs</td>
<td>85.22%</td>
<td>87.46%</td>
<td>80.57%</td>
<td>78.61%</td>
<td>82.88%</td>
<td>86.29%</td>
</tr>
<tr>
<td>Holywell</td>
<td>85.52%</td>
<td>82.39%</td>
<td>84.41%</td>
<td>88.75%</td>
<td>87.54%</td>
<td>90.30%</td>
</tr>
<tr>
<td>Farnham</td>
<td>95.49%</td>
<td>94.24%</td>
<td>92.47%</td>
<td>96.82%</td>
<td>94.35%</td>
<td>92.33%</td>
</tr>
<tr>
<td>Door</td>
<td>88.15%</td>
<td>92.77%</td>
<td>90.01%</td>
<td>89.34%</td>
<td>92.66%</td>
<td>94.21%</td>
</tr>
<tr>
<td>Falkner</td>
<td>76.56%</td>
<td>58.61%</td>
<td>73.25%</td>
<td>67.18%</td>
<td>78.82%</td>
<td>52.53%</td>
</tr>
<tr>
<td>Eggington</td>
<td>71.24%</td>
<td>39.35%</td>
<td>68.98%</td>
<td>42.51%</td>
<td>66.12%</td>
<td>48.74%</td>
</tr>
</tbody>
</table>

Table 6.4: Quantitative evaluation of the proposed algorithms, based on the new video sequences.

in order to decide on the best possible threshold values.

6.4 Shadow Detection in Video Sequences with a Non-Dedicated Background Image

Although the use of dynamic background generation techniques enhance the background subtraction process for long video sequences, they are still incapable of modelling fast changes in illumination. Chapter 5 introduced the proposed shadow detection algorithms in the spatial and frequency domains using a static background model. The algorithms use static background and object frame pairs, i.e. \( F \) and \( F^* \), for the detection of moving shadows in the object frame \( F^* \). These algorithms compare the shadow blocks in frame \( F^* \) with the corresponding non-shadow blocks in the background frame, which is used as a reference. However, in the proposed algorithms, the use of a fixed background frame as a reference is not required if any of the
Figure 6.7: "Farnham" video sequence, (a) the background frame, (b) the current frame, (c) the binary image resulting from the background subtraction process, (d) the shadow detection frame, detected shadow blocks are presented as white blocks, (e) the shadow removed frame, detected shadow blocks are replaced with corresponding non-shadow blocks from frame a, (f) the binary image of the processed frame.

previous frames in the video sequence contains the corresponding non-shadow area, and therefore can be used as a reference frame. For instance, the area covered by the shadow in frame (c) in Figure 6.4 has corresponding non-shadow areas in frame (a), and frame (b). Therefore, frame (a) or frame (b) can be used as a reference frame $F$, in order to detect shadows in frame (c). This technique would overcome the problems related to the use of static and dynamic background techniques.

An automated approach\textsuperscript{1} is more appropriate in determining which previous frame

\textsuperscript{1}This automatic method should account for: the frame rate, the total number of the objects in the scene, objects displacements, and velocities of the moving objects.
is best used as a reference, i.e. given a sequence of frames \( \{I_1 \ldots I_n\} \), the algorithm decides to use \( I_m \) as a reference frame, where \( 1 < m < n \).

The reminder of this section presents an exemplar for the proposed approach. This exemplar is designed for particular environments such as: pedestrians surveillance applications, with restrictions on the total number of humans, human sizes, velocities, and their distances from the camera. The end of this section discusses a method for generalising the exemplar in order to make it applicable for general objects and environments. Note that the core of the algorithm is based on the proposed shadow detection hypothesis discussed in section 3.2, and can be applied to any of the domains discussed in chapter 5.

This exemplar algorithm consists of three stages as illustrated in Figure 6.8, and is discussed thoroughly during the rest of the chapter.

![Figure 6.8: An exemplar shadow detection algorithm diagram for non-dedicated background algorithms.](image)
6.4.1 Initialisation

In this stage a decision on the number of frames $k$ to be used to bootstrap the algorithm should be made. Then for each frame $I_i$ where $1 < i \leq k$ the appropriate shadow detection and removal algorithm discussed in sections 5.2.1, 5.3.2, and 5.3.4 should be applied. Consequently, for each frame $I_i$ the second and third stages of the method should be applied.

6.4.2 Locating and extracting objects

Information provided by binary images is used in this step to locate and extract the objects from the scene. Note that, the scene up to this stage contains the objects only, with no background or shadow details, see Figure 6.10. Given the binary frame $B_i$, which contains pixels with values 1 (Black) representing the object and 0 (White) representing the rest of the frame, the algorithm starts searching for the first pixel with the value of 1 in the scene.

Due to noise and background subtraction, it is possible that the binary image contains some isolated pixels of value 1. Therefore, the algorithm should look into the surrounding pixels, within a relatively small window size, and if the total number of black pixels in that window is less than a certain threshold then the pixel should be disregarded and considered as noise. Otherwise, the pixel is assumed to represent the top of the first object head in the scene.

After the top of the head is located, and by assuming that the head is relatively of round shape, two vertical tangent lines of the circular shape are constructed (on the left and right of the head), and the distance between the two intersection points of the vertical tangents to the line through the point of top of head can be used to measure the diameter (measurements are represented in terms of pixels, where each pixel represents 1 unit). Note that the diameter in this particular application represents the head height or width.
Based on anthropometric \(^2\) information, the head width approximately equals to half of the body width and the head length is approximately 1/7 of the total body height [69, 59]. Using the calculated measurements, each object is extracted from the scene into a separate layer, and the algorithm repeats the search for the next object in the scene until all objects are extracted from the scene into different layers.

### 6.4.3 Calculating average object velocities

Motion estimation examines the movement of objects in an image sequence, in order to obtain vectors representing the estimated motion [64]. In motion estimation techniques, each frame is divided into blocks, and then each block in the present frame is matched against candidate blocks within a search area, in the reference frame [34].

The full search motion estimation algorithm gives optimal performance at the cost of very expensive computation, where it compares each block in the current frame with every single block in the reference frame within a window. To reduce the computational load, fast search algorithms have been developed such as: the 2D Logarithmic Search (TDL) [43], the Four-Step-Search (FSS) [60], the Cross Search Algorithm (CSA) [33], and the Spiral Search (SS) [84]. Amongst the fast search algorithms, the Three-Steps-Search (TSS) algorithm is the most popular algorithm. It is a block-based search technique, and uses a maximum of three steps. It has a fixed number of three search steps, and a maximum number of searching points of 25. The TSS technique starts with a step size slightly larger or equal to half of the maximum range. At the end of the search the step size becomes 1 pixel [48].

In the proposed algorithm the TSS algorithm is used, the blocks to be matched in the reference frame are only the heads of the objects. The matching criteria is the Mean Square Error (MSE), where the block that gives the least MSE is considered as the matching block. Once the matching block is found, a motion vector \(\vec{V}_t\) is

\(^2\)Anthropometry means measurement of humans, it refers to the measurement of living human individuals for the purposes of understanding human physical variation [36]
calculated for that particular object, where \( l \) represents the object number.

![Image of video sequence](image.png)

Figure 6.9: "Corridor", represents an indoor video sequence, with multiple combination of light sources, spectrally equal and of equal intensities. This figure illustrates the use the background non-dedicated approach based on the DCT domain shadow detection and removal algorithm. Frames (A, B, C) represent the original frames in the sequence; frames (A', B', C') are the corresponding processed frames.

The resulting motion vector \( \vec{V}_l \) is considered as the object displacement. The process repeats calculating the motion vectors for the rest of the objects in the scene and for all the frames in the initialisation stage. Once all the motion vectors are calculated, the average displacement \( \Delta S \) of the objects is calculated and stored for this particular frame. Based on this information simple motion calculations are used to predict how many frames the algorithm has to look back in order to choose the reference frame. Figure 6.9 illustrates the results of the algorithm.

### 6.4.4 Evaluation and discussion

As stated earlier, although the use of this exemplar minimises the dependency on the background frame, it is designed for particular environments, with the following
restrictions:

- Objects in the scene are humans and their sizes are relatively similar.
- Full human figures should be available in the scene.
- Objects are moving relative to one another in similar directions and at similar speeds.
- The number of objects in the scene is relatively low compared to the frame size, i.e. scenes should not be crowded.

The above constraints make the algorithm applicable for particular environments such as paths and corridors. However, the core of the algorithm is based on the proposed shadow detection algorithms presented in sections 5.2.1, 5.3.2, and 5.3.4.

To generalise this exemplar and to overcome the restrictions associated with it, motion estimation can be done for the entire frame (over the entire block set). Thus,
the average motion vectors can be used to calculate the average displacements in the scene. The technique overcomes the drawback of having the algorithm designed for a particular object type.

6.5 Summary

Video object tracking and recognition is of prime importance for security systems and video surveillance applications, where shadow detection stands at the core of successful tracking and recognition systems. This chapter illustrated the process of shadow detection in the pixel, DCT and DWT domains and discussed the experimental results associated with it. The performance evaluation shows that the proposed methods outperform most of the existing methods with significant contributions to detection accuracy.

Benchmark test video sequences are used in the proposed approaches to compare and evaluate the results with the state of the art approaches. In addition to the benchmark videos, new test video sequences have been created, with scenes and cast shadows of increased complexity, in order to test and evaluate the performance of the proposed algorithms under different conditions. The different physical conditions incorporated in the benchmark and the new video sequences include: a variety of background surfaces, wide range of object distances from the camera, and indoor/outdoor environments.

The proposed algorithms are proven theoretically and experimentally to be independent of object types and scene environments, and account for neither scene geometry nor lighting conditions. Unlike most of the other approaches, the proposed methods are capable of detecting shadow umbra and penumbra in indoor and outdoor scene environments. The strengths of the proposed approaches are a result of the high accuracy of the proposed theories and models fundamental to the algorithms. The accuracy, simplicity, and adaptability to different domains, makes the proposed
algorithms competitive when compared with the state-of-art approaches.

The use of the background frame as a reference is not required, if any of the previous frames in the video sequence contains the corresponding non-shadow area, and therefore can be used as a reference frame. An automated approach is more appropriate to determine which previous frame is the best to be used as a reference. For this purpose, an exemplar algorithm for shadow detection of pedestrians is presented in this chapter, which can be adapted or generalised for other environments and object types.

Chapter 7 discusses applying the proposed algorithms to still images where no other frames are available as a reference for the current image.
Chapter 7

Shadow Boundary Detection in Still Images

7.1 Overview

When considering shadow detection in still images, there is no additional image to be used as a reference frame (i.e. neither the background nor the reference frames are available). Therefore, many of the techniques proposed for shadow detection in video sequences cannot be directly applied to still images. However, some approaches that are merely designed for shadow detection and removal in still images have been proposed, based on the properties of colour spaces, such as the approaches proposed by Baba et al. [3, 4]. These approaches investigated the identification of shadow areas based on the RGB colour space using colour clustering. The regions of the cluster that need to be considered must be manually identified through human intervention. Therefore, this method fails to provide a truly autonomous approach to the problem of shadow detection.

The approach presented by Charit and Loew [14] looked at the detection of shadows based on the light source vector illuminating the image. It is assumed that the illumination vector information is available, and the image is illuminated by white
light. The same assumption was made by Gevers [32] when looking at adaptive image segmentation through the combination of both photometric invariant regions and edge information.

The approach proposed by Levine and Battacharyya [50] looked at the detection and removal of shadows using a learning system that identifies shadows based on their boundary properties. The work is based on the initial findings of Barnard and Finlayson [6]. This approach uses AI-based properties to identify probable boundary regions.

There have been a variety of different attempts to solve the problem of shadow detection in still images. However, the majority of these approaches are either based upon assumptions that limit their possible scope or have been designed for specific applications. There appears to be no single technique that has managed to create a generic algorithm, that is capable of detecting shadows from still images, and is usable in a broad range of applications.

In this chapter, section 7.2 introduces the proposed shadow edge detection algorithms. Section 7.2.1 illustrates the edge detection algorithms. Sections 7.2.2, and 7.2.3 introduces the spatial and transform domains shadow edge detection algorithms. Section 7.2 provides a subjective evaluation for the proposed algorithms. Finally, section 7.3 summarises this chapter.

## 7.2 Shadow Edge Detection Algorithms

The cast shadow detection algorithms presented in sections 5.2.1, 5.3.2, and 5.3.4, for the spatial and transform domains, use block-based approaches to compare shadow regions in the background frame $F^*$ with non-shadow regions in the foreground frame $F$. In still images, the unavailability of background and reference frames limits the use of the above comparison-based approach. However, regions at shadow boundaries can be used for the same comparison. Therefore, the novel approach proposed in this
Chapter 7: Shadow Boundary Detection in Still Images

The chapter suggests using 1-D blocks of size $1 \times k$ pixels instead of $k \times k$ square blocks, hereafter called intervals, for the detection of shadow boundaries in still images. It is important to note here, that this method is capable of simultaneously detecting self and cast shadow edges.

Instead of processing the entire frame in search of shadow regions, edge detectors can be used as a pre-processing step (see Figure 7.1). Based on the resulting edge detected image, the proposed shadow boundary detection algorithm compares regions at each edge point, in order to decide whether or not that edge point forms a shadow boundary.

The proposed approach suggests the use of so-called vertical and horizontal intervals, to verify the shadow edge point, where each edge point is considered as a shadow edge if both the vertical and horizontal intervals satisfy the shadow test conditions, presented in sections 5.2.1, 5.3.2, and 5.3.4 (see Figure 7.2). However, in using vertical and horizontal intervals the algorithm fails to classify the shadow edge point, if the shadow edge is oriented either in a vertical or horizontal direction with respect to the image axis. Therefore, the use of additional diagonal intervals will solve the problem, and is thus more appropriate in general for shadow edge detection, see Figure 7.3.

![Figure 7.1: Shadow boundary detection algorithm diagram](image-url)
7.2.1 Edge detection

Edge detection is simply a method of segmenting an image into regions of discontinuity. In other words, it allows the user to observe features of an image where there is a more or less abrupt change in grey level or texture, indicating the end of one region in the image and the beginning of another. Similar to other methods of image analysis, edge detection is sensitive to noise. For this reason, undetected edges can occur in places where the transition between regions is not abrupt enough or else edges can be detected in regions of an image where the texture is uniform [10]. Edge detection makes use of differential operators to detect changes in the gradients of the grey levels. A wide variety of edge detectors have been proposed in literature, such as: Roberts Cross Edge Detector [12], Sobel Edge Detector [35], Prewitt Edge Detector [22], Compass Edge Detector [77], and Canny Edge Detector [13]. The Canny edge detector is widely accepted to perform optimally under various image conditions, and thus is used in this proposed approach as the pre-processing stage.
Figure 7.3: A combination of vertical, horizontal, and Diagonal intervals used in the 1-D shadow boundary detection algorithms.

Figure 7.4 visually compares selected edge detectors, which clearly illustrates the superiority of the Canny edge detection. Readers who are interested to read more about the Canny edge detector are referred to [13].

### 7.2.2 Spatial domain shadow edge detection algorithm

Initially, the proposed pixel domain shadow boundary detection algorithm uses the Canny edge detector to extract all possible edges from a given image. Subsequently, the algorithm processes the image by comparing corresponding intervals of size $1 \times k$, at each edge point. We denote corresponding $1 \times k$ interval pairs in the pixel domain by $(P_1, P_1^*)$, $(P_2, P_2^*)$, $(P_3, P_3^*)$, $(P_4, P_4^*)$ where $P_1, P_2, P_3, P_4$ are on one side of the edge and $P_1^*, P_2^*, P_3^*, P_4^*$ are on the other side.

The proposed algorithm initially sets the size $k$ of diagonals, vertical and horizontal (V/H) intervals to 32 pixels, with 16 pixels on either side of the edge. If both diagonals and V/H intervals at this size, fail to classify the shadow edge point, then the size of the intervals, $k$, is set to half of its original size, i.e. 16 pixels, with 8 pixels on either side of the edge. This process repeats on intervals that fail to classify as shadow, down to an interval size of 4 pixels. Intervals that fail to classify as shadow edges at
Figure 7.4: Edge detection using different detectors

this level of sub-division are classified as non-shadow edges.

The pixel domain shadow detection algorithm presented here is a modified version of the block-based pixel domain algorithm presented in section 5.2.1, where the affine parameters \( \lambda \) and \( \mu \) can be computed as:

\[
\lambda_1 = \frac{\sigma(P_1^*)}{\sigma(P_1)} \quad \lambda_2 = \frac{\sigma(P_2^*)}{\sigma(P_2)} \quad \lambda_3 = \frac{\sigma(P_3^*)}{\sigma(P_3)} \quad \lambda_4 = \frac{\sigma(P_4^*)}{\sigma(P_4)}
\]

\[
\mu_1 = \tilde{P}_1^* - \lambda_1 \tilde{P}_1 \quad \mu_2 = \tilde{P}_2^* - \lambda_2 \tilde{P}_2 \\
\mu_3 = \tilde{P}_3^* - \lambda_3 \tilde{P}_3 \quad \mu_4 = \tilde{P}_4^* - \lambda_4 \tilde{P}_4
\]

Based on the affine hypothesis proposed in section 3.2 and derived in section 3.3, the expression for \( \delta_1, \delta_2, \delta_3, \) and \( \delta_4 \) can be written as:

\[
\delta_1 = \frac{\|P_1^* - (\lambda_1 P_1 + \mu_1 L_P)\|_2}{\|P_1^*\|_2} \quad \delta_2 = \frac{\|P_2^* - (\lambda_2 P_2 + \mu_2 L_P)\|_2}{\|P_2^*\|_2}
\]
Chapter 7: Shadow Boundary Detection in Still Images

\[ \delta_3 = \frac{\|P_3^* - (\lambda_3 P_3 + \mu_3 L_P)\|_2}{\|P_3^*\|_2}, \quad \delta_4 = \frac{\|P_4^* - (\lambda_4 P_4 + \mu_4 L_P)\|_2}{\|P_4^*\|_2}, \]

where \( L_P \) is the \( 1 \times k \) matrix defined by

\[ (L_P)_j = 1 \quad \text{for all } 0 \leq j < k \]

Shadow condition: As discussed in section 3.2, the light energy received at points \( r \in n_q \) in the absence of an object casting a shadow over \( n_q \) is affinely related, with a high degree of approximation, to the energy received when a shadow is cast over \( n_q \) by an object. The same affine parameters are applicable to the entire neighbourhood \( n_q \). Here, the algorithm is applied over the neighbourhood of an interval of size \( 1 \times k \).

As compared to the shadow condition discussed in section 5.2.1, similarly the shadow condition here for the interval pairs \((P_1, P_1^*), (P_2, P_2^*), (P_3, P_3^*), (P_4, P_4^*)\) is:

\[ \left( \frac{P_1^*}{P_1} < 1 \land \frac{P_2^*}{P_2} < 1 \right) \land \left( \frac{P_1^* - (\lambda_1 P_1 + \mu_1 L_P)}{\|P_1\|_2} \approx 0 \right) \land \left( \frac{P_2^* - (\lambda_2 P_2 + \mu_2 L_P)}{\|P_2\|_2} \approx 0 \right) \]

holds for \(((P_1, P_1^*) \text{ and } (P_2, P_2^*))\)

\[ \lor \]

\[ \left( \frac{P_3^*}{P_3} < 1 \land \frac{P_4^*}{P_4} < 1 \right) \land \left( \frac{P_3^* - (\lambda_3 P_3 + \mu_3 L_P)}{\|P_3\|_2} \approx 0 \right) \land \left( \frac{P_4^* - (\lambda_4 P_4 + \mu_4 L_P)}{\|P_4\|_2} \approx 0 \right) \]

holds for \(((P_3, P_3^*) \text{ and } (P_4, P_4^*))\).

Note that for a \( 1 \times 4 \) interval we have:

\[ L_P = \left[ 1, 1, 1, 1 \right]. \]

The pixel domain shadow edge detection algorithm is summarised and presented in Algorithm 4 for ease of revision. Samples of the algorithm results are provided in Figure 7.5. It clearly illustrates the efficiency of the algorithm.
Algorithm 4 Linear Pixel Domain Shadow Detection Algorithm

1: $k = 16$

2: $P^* = \text{EdgeDetection}(F^*, \text{Canny})$

3: $\forall \text{EdgePoints} \text{ DO}$

4: Find the intervals $P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^*$ each of size $k$

5: $\lambda_1 = \frac{\sigma(P_1^*)}{\sigma(P_1)}, \mu_1 = P_1^* - \lambda_1 P_1, \lambda_2 = \frac{\sigma(P_2^*)}{\sigma(P_2)}, \mu_2 = P_2^* - \lambda_2 P_2, \lambda_3 = \frac{\sigma(P_3^*)}{\sigma(P_3)}, \mu_3 = P_3^* - \lambda_3 P_3, \lambda_4 = \frac{\sigma(P_4^*)}{\sigma(P_4)}, \mu_4 = P_4^* - \lambda_4 P_4$

6: if $(\frac{P_1}{P_1^*} < 1 \land \frac{||P_1^* - (\lambda_1 P_1 + \mu_1 L_P)||_2}{||P_1||_2} < 1 \land \frac{||P_2^* - (\lambda_2 P_2 + \mu_2 L_P)||_2}{||P_2||_2} < 1 \land \frac{||P_3^* - (\lambda_3 P_3 + \mu_3 L_P)||_2}{||P_3||_2} < 1 \land \frac{||P_4^* - (\lambda_4 P_4 + \mu_4 L_P)||_2}{||P_4||_2} < 1)$ then

7: \text{THIS IS A SHADOW EDGE POINT}

8: else

9: $k = 8$

10: Find the intervals $P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^*$ each of size $k$

11: $\lambda_1 = \frac{\sigma(P_1^*)}{\sigma(P_1)}, \mu_1 = P_1^* - \lambda_1 P_1, \lambda_2 = \frac{\sigma(P_2^*)}{\sigma(P_2)}, \mu_2 = P_2^* - \lambda_2 P_2, \lambda_3 = \frac{\sigma(P_3^*)}{\sigma(P_3)}, \mu_3 = P_3^* - \lambda_3 P_3, \lambda_4 = \frac{\sigma(P_4^*)}{\sigma(P_4)}, \mu_4 = P_4^* - \lambda_4 P_4$

12: if $(\frac{P_1}{P_1^*} < 1 \land \frac{||P_1^* - (\lambda_1 P_1 + \mu_1 L_P)||_2}{||P_1||_2} < 1 \land \frac{||P_2^* - (\lambda_2 P_2 + \mu_2 L_P)||_2}{||P_2||_2} < 1 \land \frac{||P_3^* - (\lambda_3 P_3 + \mu_3 L_P)||_2}{||P_3||_2} < 1 \land \frac{||P_4^* - (\lambda_4 P_4 + \mu_4 L_P)||_2}{||P_4||_2} < 1)$ then

13: \text{THIS IS A SHADOW EDGE POINT}

14: else

15: $k = 4$

16: Find the intervals $P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^*$ each of size $k$

17: $\lambda_1 = \frac{\sigma(P_1^*)}{\sigma(P_1)}, \mu_1 = P_1^* - \lambda_1 P_1, \lambda_2 = \frac{\sigma(P_2^*)}{\sigma(P_2)}, \mu_2 = P_2^* - \lambda_2 P_2, \lambda_3 = \frac{\sigma(P_3^*)}{\sigma(P_3)}, \mu_3 = P_3^* - \lambda_3 P_3, \lambda_4 = \frac{\sigma(P_4^*)}{\sigma(P_4)}, \mu_4 = P_4^* - \lambda_4 P_4$

18: if $(\frac{P_1}{P_1^*} < 1 \land \frac{||P_1^* - (\lambda_1 P_1 + \mu_1 L_P)||_2}{||P_1||_2} < 1 \land \frac{||P_2^* - (\lambda_2 P_2 + \mu_2 L_P)||_2}{||P_2||_2} < 1 \land \frac{||P_3^* - (\lambda_3 P_3 + \mu_3 L_P)||_2}{||P_3||_2} < 1 \land \frac{||P_4^* - (\lambda_4 P_4 + \mu_4 L_P)||_2}{||P_4||_2} < 1)$ then

19: \text{THIS IS A SHADOW EDGE POINT}

20: else

21: \text{THIS IS NOT A SHADOW EDGE POINT}

22: end if

23: end if

24: end if
7.2.3 Transform domain shadow edge detection algorithms

The proposed transform domain approaches to the shadow boundary detection use the Canny edge detector to extract all edges from a given image in a manner identical to that discussed in section 7.2.2. Further the analysis is based on diagonals and V/H intervals of size $1 \times k$ as discussed in section 7.2.2. However, the processing of these intervals is done in either the DCT or DWT domains, i.e. the pixels are transformed to the above domain before further processing. We denote corresponding $1 \times k$ interval pairs in the DCT domain by $(B_1, B_1^*), (B_2, B_2^*), (B_3, B_3^*), (B_4, B_4^*)$ where $B_1 \in DCT(P_1), B_1^* \in DCT(P_1^*), B_2 \in DCT(P_2), B_2^* \in DCT(P_2^*), B_3 \in DCT(P_3), B_3^* \in DCT(P_3^*),$ and $B_4 \in DCT(P_4), B_4^* \in DCT(P_4^*)$. In the DWT domain, the author denote corresponding $1 \times k$ interval pairs by $(D_1, D_1^*), (D_2, D_2^*), (D_3, D_3^*),$ and $(D_4, D_4^*)$ where $D_1 \in DWT(P_1), D_1^* \in DWT(P_1^*), D_2 \in DWT(P_2)$ and $D_2^* \in DWT(P_2^*), D_3 \in DWT(P_3), D_3^* \in DWT(P_3^*),$ and $D_4 \in DWT(P_4), D_4^* \in DWT(P_4^*)$.

Initially, an interval size $k = 32$ is considered. If the edge point fails to satisfy the shadow conditions, the interval size is reduced to 16. This process is continued as described in section 7.2.2, up to a minimum interval size $k = 4$.

In the proposed DWT domain approach a modified version of the general block-based algorithm presented in section 5.3.4 can be used, where 1-D DWT decomposition for each interval replaces the 2-D block-based approach. Similarly, in the DCT domain, the proposed shadow edge detection algorithm is a modified version of the block-based DCT domain algorithm presented in section 5.3.2, where the affine parameters $\lambda$ and $\mu$ can be computed as:

$$
\lambda_1 = \frac{\sum ac_1^*}{\sum ac_1}, \quad \lambda_2 = \frac{\sum ac_2^*}{\sum ac_2}, \quad \lambda_3 = \frac{\sum ac_3^*}{\sum ac_3}, \quad \lambda_4 = \frac{\sum ac_4^*}{\sum ac_4}.
$$

Alternatively:

As the modifications required for the DCT and DWT block-based algorithms are relatively similar in both domains, only the DCT approach is discussed here.
\[
\lambda_1 = \frac{P_1^*(0,0) - a_{100}^*}{P_1(0,0) - a_{100}} \quad \lambda_2 = \frac{P_2^*(0,0) - a_{200}^*}{P_2(0,0) - a_{200}} \quad \lambda_3 = \frac{P_3^*(0,0) - a_{300}^*}{P_3(0,0) - a_{300}} \quad \lambda_4 = \frac{P_4^*(0,0) - a_{400}^*}{P_4(0,0) - a_{400}}
\]

\[
\mu_1 = dc_1^* - \lambda_1 dc_1 \quad , \quad \mu_2 = dc_2^* - \lambda_2 dc_2
\]

\[
\mu_3 = dc_3^* - \lambda_3 dc_3 \quad , \quad \mu_4 = dc_4^* - \lambda_4 dc_4
\]

Therefore, based on the affine hypothesis proposed in section 3.2 and derived in section 3.3, the expressions for \(\delta_1, \delta_2, \delta_3, \delta_4\) can be written as:

\[
\delta_1 = \frac{\|B_1^* - (\lambda_1 B_1 + \mu_1 L_D)\|_2}{\|B_1^*\|_2} \quad \delta_2 = \frac{\|B_2^* - (\lambda_2 B_2 + \mu_2 L_D)\|_2}{\|B_2^*\|_2} \\
\delta_3 = \frac{\|B_3^* - (\lambda_3 B_3 + \mu_3 L_D)\|_2}{\|B_3^*\|_2} \quad \delta_4 = \frac{\|B_4^* - (\lambda_4 B_4 + \mu_4 L_D)\|_2}{\|B_4^*\|_2}
\]

where \(L_D\) is the \(1 \times k\) matrix defined by

\[
(L_D)_j = \begin{cases} 
1 & \text{for } j = 0 \\
0 & \text{for } 0 < j < k
\end{cases}
\]

**Shadow condition:** As discussed in section 3.2, the light energy received at points \(r \in n_q\) in the absence of an object casting a shadow over \(n_q\) is affinely related, with a high degree of approximation, to the energy received when a shadow is cast over \(n_q\) by an object. The same affine parameters are applicable to the entire neighbourhood \(n_q\). Here, the algorithm is applied over the neighbourhood of an interval of size \(1 \times k\).

As compared to the shadow condition discussed in section 5.3.2, similarly the shadow condition here for the interval pairs \((B_1, B_1^*)\), \((B_2, B_2^*)\), and \((B_3, B_3^*)\), \((B_4, B_4^*)\) is:
Algorithm 5 Linear DCT Domain Shadow Detection Algorithm

1:  \( k \leftarrow 16 \)
2:  \( F^* = \text{EdgeDetection}(F^*, \text{Canny}) \)
3:  \( \forall \text{EdgePoints} \ do \)
4:  Find the intervals \( P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^* \) each of size \( k \)
5:  \( B_1 = \text{DCT}(P_1), B_1^* = \text{DCT}(P_1^*), B_2 = \text{DCT}(P_2), B_2^* = \text{DCT}(P_2^*), B_3 = \text{DCT}(P_3), B_3^* = \text{DCT}(P_3^*), \)
\( B_4 = \text{DCT}(P_4), B_4^* = \text{DCT}(P_4^*) \)
6:  \( \lambda_1 = \sum_{ac_1}^{ac_1^*} \mu_1 = a_{100}^* - \lambda_1a_{100}, \lambda_2 = \sum_{ac_2}^{ac_2^*} \mu_2 = a_{200}^* - \lambda_2a_{200}, \lambda_3 = \sum_{ac_3}^{ac_3^*} \mu_3 = a_{300}^* - \lambda_3a_{300}, \)
\( \lambda_4 = \sum_{ac_4}^{ac_4^*} \mu_4 = a_{400}^* - \lambda_4a_{400} \)
7:  if \( \left( \begin{array}{c} b_{100}^* \\ b_{200}^* \\ b_{300}^* \\ b_{400}^* \end{array} \right) < \left( \begin{array}{c} \| B_1^* - (\lambda_1 B_1 + \mu_1 L) \|_2 \\ \| B_2^* - (\lambda_2 B_2 + \mu_2 L) \|_2 \\ \| B_3^* - (\lambda_3 B_3 + \mu_3 L) \|_2 \\ \| B_4^* - (\lambda_4 B_4 + \mu_4 L) \|_2 \end{array} \right) \approx 0 \) then
8:  THIS IS A SHADOW EDGE POINT
9:  else
10:  \( k \leftarrow 8 \)
11:  Find the intervals \( P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^* \) each of size \( k \)
12:  \( B_1 = \text{DCT}(P_1), B_1^* = \text{DCT}(P_1^*), B_2 = \text{DCT}(P_2), B_2^* = \text{DCT}(P_2^*), B_3 = \text{DCT}(P_3), B_3^* = \text{DCT}(P_3^*), \)
\( B_4 = \text{DCT}(P_4), B_4^* = \text{DCT}(P_4^*) \)
13:  \( \lambda_1 = \sum_{ac_1}^{ac_1^*} \mu_1 = a_{100}^* - \lambda_1a_{100}, \lambda_2 = \sum_{ac_2}^{ac_2^*} \mu_2 = a_{200}^* - \lambda_2a_{200}, \lambda_3 = \sum_{ac_3}^{ac_3^*} \mu_3 = a_{300}^* - \lambda_3a_{300}, \)
\( \lambda_4 = \sum_{ac_4}^{ac_4^*} \mu_4 = a_{400}^* - \lambda_4a_{400} \)
14:  if \( \left( \begin{array}{c} b_{100}^* \\ b_{200}^* \\ b_{300}^* \\ b_{400}^* \end{array} \right) < \left( \begin{array}{c} \| B_1^* - (\lambda_1 B_1 + \mu_1 L) \|_2 \\ \| B_2^* - (\lambda_2 B_2 + \mu_2 L) \|_2 \\ \| B_3^* - (\lambda_3 B_3 + \mu_3 L) \|_2 \\ \| B_4^* - (\lambda_4 B_4 + \mu_4 L) \|_2 \end{array} \right) \approx 0 \) then
15:  THIS IS A SHADOW EDGE POINT
16:  else
17:  \( k \leftarrow 4 \)
18:  Find the intervals \( P_1, P_1^*, P_2, P_2^*, P_3, P_3^*, P_4, P_4^* \) each of size \( k \)
19:  \( B_1 = \text{DCT}(P_1), B_1^* = \text{DCT}(P_1^*), B_2 = \text{DCT}(P_2), B_2^* = \text{DCT}(P_2^*), B_3 = \text{DCT}(P_3), B_3^* = \text{DCT}(P_3^*), \)
\( B_4 = \text{DCT}(P_4), B_4^* = \text{DCT}(P_4^*) \)
20:  \( \lambda_1 = \sum_{ac_1}^{ac_1^*} \mu_1 = a_{100}^* - \lambda_1a_{100}, \lambda_2 = \sum_{ac_2}^{ac_2^*} \mu_2 = a_{200}^* - \lambda_2a_{200}, \lambda_3 = \sum_{ac_3}^{ac_3^*} \mu_3 = a_{300}^* - \lambda_3a_{300}, \)
\( \lambda_4 = \sum_{ac_4}^{ac_4^*} \mu_4 = a_{400}^* - \lambda_4a_{400} \)
21:  if \( \left( \begin{array}{c} b_{100}^* \\ b_{200}^* \\ b_{300}^* \\ b_{400}^* \end{array} \right) < \left( \begin{array}{c} \| B_1^* - (\lambda_1 B_1 + \mu_1 L) \|_2 \\ \| B_2^* - (\lambda_2 B_2 + \mu_2 L) \|_2 \\ \| B_3^* - (\lambda_3 B_3 + \mu_3 L) \|_2 \\ \| B_4^* - (\lambda_4 B_4 + \mu_4 L) \|_2 \end{array} \right) \approx 0 \) then
22:  THIS IS A SHADOW EDGE POINT
23:  else
24:  THIS IS NOT A SHADOW EDGE POINT
25:  end if
26:  end if
27:  end if
\[ \left( \frac{a_{100}^*}{a_{100}} < 1 \right) \land \left( \frac{a_{200}^*}{a_{200}} < 1 \right) \land \left( \frac{1}{\|B_1^* - (\lambda_1 B_1 + \mu_1 L_D)\|_2} \approx 0 \right) \land \left( \frac{1}{\|B_2^* - (\lambda_2 B_2 + \mu_2 L_D)\|_2} \approx 0 \right) \]

holds for \((B_1, B_1^*)\) and \((B_2, B_2^*)\).

\[ \lor \]

\[ \left( \frac{a_{300}^*}{a_{300}} < 1 \right) \land \left( \frac{a_{400}^*}{a_{400}} < 1 \right) \land \left( \frac{1}{\|B_3^* - (\lambda_3 B_3 + \mu_3 L_D)\|_2} \approx 0 \right) \land \left( \frac{1}{\|B_4^* - (\lambda_4 B_4 + \mu_4 L_D)\|_2} \approx 0 \right) \]

holds for \((B_3, B_3^*)\) and \((B_4, B_4^*)\).

Note that for a 1 x 4 block we have:

\[ L_D = \begin{bmatrix} 1, 0, 0, 0 \end{bmatrix}. \]

The DCT domain shadow edge detection algorithm is summarised and presented in Algorithm 5 for ease of revision, samples of the algorithm results are provided in Figure 7.6. It clearly illustrates the performance of the algorithm.

### 7.2.4 Subjective evaluation

Experiments are performed using the video frames (i.e. images) of the test video sequences used in chapter 6. The performance evaluation metrics used in section 6.3.1 for shadow detection in video applications cannot be applied in this work for the reason that self and cast shadows together complicate the evaluation metrics. Further in this chapter, we only aim for shadow boundary detection instead of cast shadow detection. Therefore, subjective results are used to evaluate the effectiveness of the proposed algorithm.

Figure 7.5 shows the results of the spatial domain shadow edge detection algorithm, the figure contains three test images (original frames) each with the Canny edge detection, the output of the algorithm is represented in the detected shadow edges section where potential shadow edges have been spotted in white. The last
Figure 7.5: Shadow edge detection in the spatial domain. Still images are extracted from the following video sequences: “Holywell”, “Black”, and “Sara”, respectively. The presented shadow edge detection figures show that the algorithm succeeds to detect self and cast shadows in the images.

The section of the figure illustrates more clearly the shadow edge detected points. Note that self and cast shadow edges have been detected as shown in this figure. Figure 7.6 shows the results of the transform (DCT) domain shadow edge detection algorithm, the representation of the figure is identical to Figure 7.5 with same images. Output results from both domain algorithms are almost identical with major shadow edges being detected in both domains. However, very minor differences can be reported without any significant conclusion regarding the differences.

The minor differences are related the affine parameter estimation in the spatial and transform domain, where the parameters are computed in a comparable but not fully identical methods.
Figure 7.6: Shadow edge detection in the transform domain. Still images are extracted from the following video sequences: “Holywell”, “Black”, and “Sara” respectively. The presented shadow edge detection figures show that the algorithm succeeds to detect self and cast shadows in the images. Results are relatively similar to the pixel domain shadow edge detection algorithm, with minor differences.

7.3 Summary

The absence of background and reference frames in still images limits the use of the block based shadow detection approach presented in chapter 5. However, regions at shadow boundaries can be used for the same comparison. Therefore, in this chapter interval-based shadow edge detection algorithms for still images have been proposed in the spatial and frequency domains. The algorithms in both domains use the Canny edge detector as a pre-processing stage to locate possible edges in the scene. Based on the resulting detected edges, a shadow boundary detection algorithm compares
regions at each edge points in order to decide whether or not that edge point forms a shadow boundary. The proposed algorithms use diagonal, vertical and horizontal intervals to verify the shadow edge point, where each edge point is considered as a shadow edge if either the diagonal, or the vertical and horizontal intervals satisfy the shadow test conditions.

Experiments are performed on video frames (i.e. images) similar to the one used in chapter 6. The performance evaluation metrics used in section 6.3.1 for shadow detection in video applications cannot be applied in this part of the work for the reason that self and cast shadows together complicate the evaluation metrics. At the same time, this work aims at shadow boundary detection instead of shadow detection. However, a subjective evaluation is carried out to test and compare the results of the spatial and transform domains shadow boundary detection algorithms. As shown in the results, the algorithms succeeded to detect self and cast shadows in the scenes with minor differences in the results obtained from multiple domains.
Chapter 8

Conclusion and Future Work

This chapter summarises the key results presented in the previous chapters, draws conclusions, and emphasises the significant contributions of this thesis. It also presents possible future directions of research in order to expand the functionalities and efficiency of the proposed algorithms.

The research work reported in this thesis commenced with potential improvements to the current illumination and shadow model $H1$, which models the ambient light more precisely than the existing models, by assuming that a lesser amount of ambient light is received at a point $q$ on a surface when the shadow is cast. In the $H3$ model, this thesis proposes:

$$ C(q) = \begin{cases} 
  i_C^{(1)} \left[ N_q \cdot L_q^{(1)} + \int_{W} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] \\
  + i_C^{(2)} \left[ N_q \cdot L_q^{(2)} + \int_{W} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right] \\
  \text{no object} \\
  \\
  i_C^{(1)} N_q \cdot L_q^{(1)} \\
  + i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] \\
  + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right] \\
  \text{penumbra} \\
  \\
  i_C^{(1)} \left[ \int_{W \setminus S_1} \mu_C(w)N_w \cdot L_w^{(1)} \, dw \right] \\
  + i_C^{(2)} \left[ \int_{W \setminus S_2} \mu_C(w)N_w \cdot L_w^{(2)} \, dw \right] \\
  \text{umbra} 
\end{cases} $$

113
As an alternative to illumination and shadow model, \( H1; \)

\[
\zeta(q) = \begin{cases} 
  c_P N_{x,y} \cdot L_{x,y} + c_A & \text{no object} \\
  \lambda x,y c_P N_{x,y} \cdot L_{x,y} + c_A & \text{penumbra} \\
  c_A & \text{umbra}
\end{cases}
\]

The main contribution of the thesis is presented by the proposed shadow hypothesis and computations, using the geometric and physical models of light, that showed the way to a local affine hypothesis for the identification of shadows in digital images. The hypothesis can be summarised as follows: When a shadow is cast over a neighbourhood, a smaller amount of light is received there - as compared to the entirely illuminated state. Using the geometric depiction of light rays and a simple reflection model, it is possible to show that the light energy received at points \( r \in n_q \) in the non-appearance of an object casting a shadow over \( n_q \) is affinely related, to a high degree of approximation, to the energy received when a shadow is cast over \( n_q \) by an object. The same affine parameters being applicable to the entire neighbourhood \( n_q \).

Consequently, the luminance function \( L : n_q \to \mathbb{R} \) when no shadow is cast over \( n_q \) is affinely related to the luminance function \( L^* : n_q \to \mathbb{R} \) when a shadow is cast; i.e. for \( n_q \) to be in shadow we have \( L^*(r) \approx \lambda L(r) + \mu \) and \( L^*(r) < L(r) \), for some constants \( \lambda \) and \( \mu \), for all \( r \in n_q \).

There are analogies here with several other physical phenomena that may be highly non-linear when considered globally (or over a long time period). But nevertheless, are linear or affine, to a high degree of approximation, when considered in a sufficiently small spatial neighbourhood (or over a sufficiently small time interval). Therefore, in shadow detection applications it is safe to consider only the affine relationship, as it will represent the scaling relationship when \( \mu \) is estimated in a local neighbourhood to be 0.

It follows that with a monochrome source or with a colour source, simulated using \( R, G, B \) triples, the simple model predicts that in a sufficiently small neighbourhood an affine relationship exists between reflected light from a neighbourhood \( U_q \) of a
point $q$ when it is in/out-of shadow, i.e. we have:

$$r_{q,q^*,c} = \alpha_c r_{bg,q^*,c} + \delta_c$$

for all $q^* \in U_q$ - and provided local patches of $U_q$ have similar reflectance coefficients.

In the different colour layers the corresponding affine parameters will be different. Hence shadows may be identified in colour images by applying the affine test in each colour layer. Consequently, the shadow area will be the intersection area between the three colours.

The accomplishment of the approach depends vitally on the accurate determination of candidate affine parameters $\lambda$ and $\mu$, with which to test the neighbourhood affine condition. As video data is inherently noisy, this suggests estimating candidate values using statistical measures that have error and noise reducing properties. To this effect in the DCT domain the $dc$ value and the sum of the $ac$ values suggest effective measurements, in the DWT the $LL$ value and the sum of $LH, HL, HH$ also suggest effective measurements. The affine parameters' computations in the spatial and transform domains are comparable but not fully identical, which leads to minor differences between results obtained from different domains. Further research can be done to compute identical affine parameters from different domains.

As a result of the proposed shadow-affine hypothesis, shadow detection algorithms are developed in the spatial and frequency domains, (pixel, DCT, and DWT) which are specifically proposed for video applications where the background frame is available. The shadow detection algorithms presented in this work use background and object frame pairs $F$ and $F^*$ for the detection of moving shadows in the object frame $F^*$. The algorithms work either in the pixel, DCT, or DWT domains and process the frames blockwise by comparing corresponding blocks, of size $k \times k$, in $F$ and $F^*$.

The proposed algorithm in any domain initially sets $k$ to 16. If a block of this size fails to classify as a shadow block, it is subdivided into four, $8 \times 8$ blocks for re-testing. This process is repeated, on blocks that initially fail to classify as shadows,
down to a block size $3 \times 3$. Blocks that fail to classify as shadow at this lowest level of sub-division are classified as non-shadow.

The following metrics are used to systematically evaluate the proposed shadow detectors:

$$\eta = \frac{TP_s}{TP_s + FN_s} \quad \text{and} \quad \nu = \frac{TP_F}{TP_F + FN_F}$$

The performance measurements show that the proposed algorithms when applied to the benchmark videos outperform most of state of the art approaches with significant enhancements of the good detection and discrimination rate.

As for videos with no available background image, an automated approach is more appropriate to be used to determine which previous frame is the best to be used as a reference. For this purpose, an exemplar is designed for particular environments such as: pedestrians surveillance applications, and a method is discussed for generalising the algorithm to be applied for general objects. However, the core of the algorithm is based on the proposed shadow detection hypothesis, and can be used in any of the discussed spatial or transform domains.

Finally, interval-based shadow edge detection algorithms for still images are proposed in the spatial and frequency domains. The proposed algorithms in the pixel and transform domains use the Canny edge detector to locate edges in the scene as a pre-processing step. Based on the resulting edge detected image, a shadow boundary detection algorithm compares regions at each edge point in order to decide whether or not that edge point forms a shadow boundary point.

In the proposed approach 1-D diagonal intervals are used to verify the shadow edge point, where each edge point is considered as a shadow edge if either the diagonal intervals, or the vertical and horizontal intervals satisfy the shadow test conditions. The algorithms initially set the interval size $k$ of each interval to 32, which passes the edge point with equal length on both sides, i.e. 16 pixels before the edge point and 16 pixels after the edge point. If both of the diagonal and V/H intervals of this size fail
to classify the edge point as a shadow edge point, the sizes of the diagonal intervals $k$
is set to 16, with 8 pixels before and 8 pixels after the edge point for re-testing. This
process is repeated on intervals that fail to classify as shadows, down to an interval
size of 4. Intervals that fail to classify as shadow edges at this level of sub-division
are classified as non-shadow edges.

The proposed algorithms succeeded to detect self and cast shadow edges in still
images. However, this approach aims at shadow boundary detection instead of shadow
detection. Shadow removal in still images is also a possible future direction of research,
where region growing can be used to eliminate detected shadow edges.
Bibliography


Appendix A

Taxonomy of Moving Cast Shadow Detection Algorithms

For the purpose of this research, moving cast shadow detection algorithms are categorised into a four-layer taxonomy, as shown in Table A.1. The first layer of classification considers whether the approach is independent/dependent of object types. The second layer considers whether the approach is environment independent/dependent. The third layer considers whether the decision process introduces and exploits uncertainty. The third layer is subdivided into deterministic and statistical approaches, the former uses an on/off decision process, and the later uses statistical measurements, and introduces uncertainty to reduce noise sensitivity. The deterministic class can be further subdivided based on whether the on/off decision can be supported by model-based knowledge or not. Another classification has been created based on the domain of the shadow detection algorithms, shown in Table A.2. This classification considers whether the approach is applied in the spatial (pixel), or in the transform (frequency) domain, and whether it is based on the Hue-Saturation-Value (HSV), the Red-Green-Blue (RGB), or YUV colour spaces.
### Table A.1: A Four-Layer taxonomy, Independent/Dependent of object type, Independent/Dependent of the environment, Deterministic/Statistical.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Object Type</th>
<th>Environment</th>
<th>Deterministic</th>
<th>Statistical</th>
</tr>
</thead>
<tbody>
<tr>
<td>54</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>76</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>39</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>30</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>73</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>18</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>29</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>16</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>56</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>67</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>5</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>23</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>55</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>71</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>42</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>31</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>83</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>82</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>72</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>75</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>57</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>This approach</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
### Table A.2: Taxonomy based on the Domain -pixel (spatial) or transform (frequency) and colour spaces $RGB$, $HSV$, and $YUV$.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Pixel Domain</th>
<th>Transform Domain</th>
<th>Colour space</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DCT</td>
<td>DWT</td>
</tr>
<tr>
<td>54</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>76</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>24</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>39</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>30</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>45</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>73</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>18</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>20</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>40</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>70</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>81</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>58</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>2</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>29</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>16</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>56</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>44</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>19</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>7</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>46</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>67</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>5</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>21</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>23</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>55</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>71</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>42</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>31</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>83</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>82</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>72</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>75</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>57</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td><strong>This approach</strong></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

This table categorizes moving cast shadow detection algorithms based on their approach in the pixel domain or transform domain and their use of colour spaces. The ✔ symbol indicates the presence of the domain or colour space in the approach.
Appendix B

Additional Experimental Results

Further experimental results related to the algorithms proposed in chapter 5 are presented in this Appendix. For clarity of presentation, each figure is divided into 4 frames. The first and second frames are processed using the pixel domain shadow detection algorithm presented in section 5.2.1. The third frame is processed using the Fourier domain shadow detection algorithm presented in section 5.3.2. The fourth frame is processed using the wavelet domain shadow detection algorithm presented in section 5.3.4. To add more clarity to the presentation, the input video frame and the output video frames are both accompanied with their corresponding binary images.

Experiments are performed in both high and low-resolution video sequences. The video sequences shown here, are the new video sequences created for the purpose of this research, they represent simple/complex, indoor/outdoor scenes with different numbers of objects under different lighting and environmental conditions. New video sequences are created with scene and cast shadow complexity is varied, in order to test and evaluate the performance of the proposed algorithms under different conditions. The camera settings (aperture, shutter, speed, sampling rate) are all set in automatic mode.
Figure B.1: Indoor environment with single point source and neutral walls, ‘Ghost’ video sequence, frames: 50, 52, 56, 85. The frames shown in this figure are captured using a high resolution camera. The distance between the camera and the object is relatively small. In this sequence, the shadow is cast vertically on the wall and horizontally on a wooden floor. The cast shadow covers a wide area of the frame. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
## Figure B.2: Indoor environment with single point source and textured walls, ‘Legs’ video sequence, frames: 9, 11, 13, 15. The frames shown in this figure are captured using a high resolution camera. The distance between the camera and the object is relatively small. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.3: Indoor environment with multiple arbitrary combination of light sources, 'Black' video sequence, Frames: 21-24. The frames shown in this figure are captured using a low resolution camera. Multiple shadows in this sequence are cast vertically onto the wall. In this particular scene, due to the use of static a background subtraction technique and a change in illumination, it is clear in the output video that the brightness of the replaced blocks is slightly different from the brightness in the current frame. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.4: Indoor environment with multiple combination of light sources spectrally equal with different intensities, 'Side' video sequence, frames: 63-66. The frames shown in this figure were captured using a low resolution camera. Multiple shadows in this sequence are cast vertically onto the wall. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.5: Indoor environment with multiple combination of light sources spectrally equal with equal intensities, 'Sara' video sequence, frames: 35-38. The frames shown in this figure were captured using a high resolution camera. In this sequence the distance between the objects and the camera varies from relatively high to relatively low. The shadows are cast vertically onto the walls and horizontally onto the carpet. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.6: Indoor environment with multiple combination of light sources spectrally equal with equal intensities. The well-known ‘Hall’ video, frames: 31, 35, 42, 132. In this sequence the distance between the objects and the camera varies from relatively high to relatively low. The shadows are cast vertically onto the walls and horizontally onto the carpet. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.7: Outdoor in overcast condition, ‘Asphalt’ video sequence, frames: 25-28. The frames shown in this figure were captured using a low resolution camera. In this sequence, shadows are cast horizontally onto the asphalt street. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.8: Outdoor environment in sunlight, ‘Holywell’ video sequence, frames: 122, 125, 128, 131. The frames shown in this figure were captured using the low resolution camera. This sequence was captured in bright sunlight, casting strong shadows with well-defined edges onto the asphalt street. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.9: Outdoor at Dusk time without any artificial lights, 'Farnham' video sequence, frames: 8-11. The frames shown in this figure were captured using the high resolution camera. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.10: Outdoor at Dusk time with artificial lights, 'Door' video sequence. frames: 54-57. The frames shown in this figure were captured using a high resolution camera. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.
Figure B.11: Outdoor video sequence at night with multiple street lights, ‘Falkner’ video sequence, frames: 21, 23, 25, 27. The frames shown in this figure were captured using a high resolution camera. In this sequence, shadows create a great challenge for the algorithm because of their darkness and lack of texture information. Table 6.2 provides a classified description of the video scene complexity. Table 6.4 provides quantitative results and a performance evaluation of the proposed algorithms for this particular video sequence.