

Automatic Quality Improvement of Archive Film

Xiaosong Wang



A dissertation submitted to the University of Bristol in accordance with the requirements for the degree of Doctor of Philosophy in the Faculty of Engineering, Department of Computer Science.

September 2011

Abstract

This thesis describes our work towards a unified framework for automatic restoration of dirt and blotches in archive film. The framework we present here is composed of three stages, *i.e.* defect detection, false alarm elimination, and defect removal. First, we propose a novel probabilistic approach to detect defects in digitized archive film. An HMM is characterized and trained to model the statistical changes of temporal pixel transitions over several frames before and after the current frame. The trained HMM is then applied to measure how the likelihood of an unseen observation sequence being normal varies if each single observation within it was missing, one at a time in a leave-one-out fashion. The centre pixel of the observation sequence will be marked in the defect map, if the likelihood of the observation sequence without the centre pixel is larger, by a certain degree, than the average of all likelihoods.

The resulting defect maps from our proposed defect detector encapsulates the true defects very well, but can suffer from many false alarms. Therefore, we extend the defect detection method to add a two-stage false alarm elimination process, which is developed based on investigating the characteristics and causes of false alarms. The proposed false alarm approach first applies MRF modelling on the defect map to propagate neighbouring normal pixels into the false alarm region using spatial continuity constraints. Then, the pyramidal Lucas-Kanade feature tracker is adopted to impose temporal correlation constraints on spatially isolated false alarm regions. This helps increase the accuracy of the proposed method significantly.

Finally, we present a novel restoration method for defects and missing regions in archive films. Our statistical framework is based on Random Walks to examine the spatiotemporal path of a degraded pixel, and uses texture features in addition to intensity and motion information traditionally used in previous restoration works. The degraded pixels within a frame are restored in a multiscale framework by updating their features (intensity, motion and texture) at each level with reference to the attributes of normal pixels and other defective pixels in the previous scale as long as they fall within the defective pixel's random walk-based spatiotemporal neighbourhood.

The proposed algorithms are compared against state-of-the-art and industry-standard methods to demonstrate their improved detection and restoration performance using our archive film restoration dataset.

Declaration

I declare that the work in this dissertation was carried out in accordance with the Regulations of the University of Bristol. The work is original except where indicated by special reference in the text and no part of the dissertation has been submitted for any other degree.

Any views expressed in the dissertation are those of the author and in no way represent those of the University of Bristol.

The dissertation has not been presented to any other University for examination either in the United Kingdom or overseas.

SIGNED:

DATE:

Acknowledgements

I am extremely grateful to Prof. Majid Mirmehdi for his continued support and advice throughout my PhD and this thesis owes much to him. I would also like to thank Dr. Walterio Mayol-Cuevas for introducing me to computer vision in the first place.

I also want to thank Brian Hann, John Chiverton, Siyuan Fang, Filiberto Pla, Jason Xie, Tarek Abudawood for the invaluable discussion not only on this project but also over computer science in general.

For the support of Great Western Research, ITV and University of Bristol, who provided funding and made this research possible, I am also extremely grateful.

Above all, my heartfelt thanks goes to Wenwen and our parents. I know I wouldn't finish the PhD without their endless encouragement, patience and support.

Publications

The work described in this thesis has resulted in the following publications:

1. Xiaosong Wang and Majid Mirmehdi. Archive Film Defect Detection based on a Hidden Markov Model. In *Proceedings of the 10th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2009)*, pages 292-295, May 2009. (includes material from Chapter 4).
2. Xiaosong Wang and Majid Mirmehdi. HMM based Archive Film Defect Detection with Spatial and Temporal constraints. In *Proceedings of the 20th British Machine Vision Conference (BMVC 2009)*, **Winner of the Best Industrial Paper Prize**, September 2009. (includes material from Chapters 4 and 5).
3. Xiaosong Wang and Majid Mirmehdi. Archive Film Restoration based on Spatiotemporal Random Walks. In *Proceedings of the 11th European Conference on Computer Vision (ECCV 2010)*, Lecture Notes in Computer Science, Volume 6315, pages 478-491, September 2010. (includes material from Chapter 6).
4. Xiaosong Wang and Majid Mirmehdi. Archive Film Defect Detection and Removal: an Automatic Restoration Framework. *Under review. Submitted to IEEE Transactions on Image Processing*, March 2011.

Contents

List of Acronyms	vii
List of Symbols	viii
List of Figures	ix
List of Tables	xvii
1 Introduction	1
1.1 Automated Restoration Framework for Archive Film	3
1.1.1 Defect Detection	4
1.1.2 False Alarm Elimination	6
1.1.3 Defect Removal	7
1.2 Thesis Outline	8
2 Background	11
2.1 Hardware based Automatic Restoration	12

2.2	Defect Detection	13
2.2.1	Filter-based Methods	14
2.2.2	Model-based Methods	17
2.3	False Alarm Elimination	18
2.4	Defect Removal	21
2.4.1	Image Structure Propagation based Methods	22
2.4.2	Exemplar based Methods	25
2.4.3	Combined Methods	28
2.4.4	Archive Film Restoration Algorithms	28
2.5	Summary	32
3	Defects and Data	35
3.1	Our Target: Dirt and Blotches	35
3.2	Other Defects	36
3.3	Archive Film Restoration Dataset	39
3.3.1	Training Dataset	40
3.3.2	Testing Dataset	40
3.4	Quantitative Measures	42
3.4.1	Defect Detection Stage	42
3.4.2	Defect Removal Stage	43
3.5	Summary	44
4	HAFID: HMM based Archive Film Defect Detection	49

4.1	Novelty Detection	51
4.2	HMM based Defect Detection	53
4.2.1	Fundamentals of HMM	55
4.2.2	Observations	57
4.2.3	Topological Structure of State Transitions	59
4.2.4	Training HMM for Normal Pixel Sequences	65
4.2.5	Leave-one-out Test Process	66
4.2.6	An Example Case	69
4.2.7	Notes on Pre-processing	71
4.3	Testing the Trained HMM and Discussion	71
4.4	Results and Comparative Study	73
4.4.1	Quantitative Results	73
4.4.2	Qualitative Results	76
4.5	Conclusions	82
5	HAFID-STC: False Detection Elimination	85
5.1	Category of False Alarms	86
5.1.1	Causes of False Alarms	87
5.2	Two-stage False Alarm Elimination	93
5.2.1	MRF modelling	94
5.2.2	Motion Analysis	98
5.3	Results and Comparative Study	98

5.3.1	Image Example from Real Sequences	101
5.4	Conclusions	104
6	RWDR: Spatiotemporal Random Walks based Defect Removal	105
6.1	Spatiotemporal Random Walks on Image Sequences	107
6.1.1	Neighbourhood	108
6.1.2	Edge Weight	109
6.1.3	Walk Length	110
6.2	Defect Removal in Archive Film	110
6.2.1	Random-walk based Searching	112
6.2.2	Degraded Pixel Replacement	114
6.2.3	Multiscale Refinement	117
6.3	Experimental Results and Discussion	117
6.3.1	Synthetic Defects	120
6.3.2	Real Defects	123
6.3.3	Performance and Implementation Issues	129
6.3.4	Relationship to Previous Random Walk based Methods	130
6.4	Conclusions	131
7	Conclusions and Further Work	133
7.1	Summary	133
7.2	Contributions	135
7.3	Future Work	135

Bibliography	139
A RWDR: Application to Image and Video Inpainting	149

Acronyms

BRAVA	Broadcast Restoration of Archives by Video Analysis [14]
BV	Bounded Variation
CDD	Curvature-Driven Diffusion
DCT	Discrete Cosine Transform
DTM	Double Threshold Method
FPGA	Field-Programmable Gate Array
GMM	Gaussian Mixture Modelling
HAFID	HMM based Archive FIlm Defect detection
HAFID-STC	HMM based Archive FIlm Defect detection with Spatial and Temporal Constrains
HMM	Hidden Markov Model
ICM	Iterated Conditional Mode
LUM	Lower-Upper-Middle Filter
MAP	Maximum <i>a posteriori</i>
MRF	Markov Random Field
MSE	Mean Square Error
PDE	Partial Differential Equations
PrestoSpace	Preservation towards Storage and access Standardised Practices for Audiovisual Contents in Europe [75]
PSNR	Peak Signal-to-Noise Ratio
ROC	Receiver Operating Characteristic
ROD	Rank Order Detector
RWDR	spatiotemporal Random Walk based Defect Removal
SDIp	Spike Detection Index
SIMD	Single Instruction, Multiple Data
SMF	Soft Morphological Filter
SROD	Simplified Rank Order Detector
TV	Total Variation

List of Symbols

A	Transition probability, $A = \{a_{lm}, l, m = 1, \dots, Q\}$
B	Emission probability, $B = \{b_m(o_n), m = 1, \dots, Q, n = 1, \dots, N\}$
D^t	$D^t = \{d_{\mathbf{x}}^t, d_{\mathbf{x}}^t \in \{0, 1\}\}$, map of pixels labelled as defects
G	An undirected and weighted graph, $G = (V, E)$
$I_{\mathbf{x}}^t$	Image intensity at pixel \mathbf{x} in frame t
K	Length of sequences
M	Number of random walks
$N_{\mathbf{x}}$	Spatiotemporal motion compensated neighbourhood centred at \mathbf{x}
N	Size of the discrete observation space
$Path_{0,K}$	A random walk sequence starting at pixel \mathbf{x}^0 , $Path_{0,K} = \{\mathbf{x}^0, \mathbf{x}^1, \dots, \mathbf{x}^K\}$
P	A set of M random walks on graph G , $P = \{Path_{0,K_m}^m\}_{m=1}^M$
Q	Number of states
U^t	$U^t = \{u_{\mathbf{x}}^t\}$, map of likelihood of each pixel being degraded
$\delta(\cdot)$	Dirac <i>delta</i> function
λ_{θ}	Parameters of HMM θ , $\lambda_{\theta} = (\pi, A, B)$
$\mathbf{R}_{\mathbf{x}^0}$	Set of pixels visited by the random walks in P , $\mathbf{R}_{\mathbf{x}^0} = \bigcup_{m=1}^M Path_{0,K_m}^m$
\mathbf{x}	Index of pixels in frame t
π	Start probability, $\pi = \{\pi(m), m = 1, \dots, Q\}$
\mathbf{O}	Set of observations
\mathbf{S}	Set of states
θ	A HMM trained for HAFID defect detector
$\{o(k)\}_{\mathbf{x}}^t$	Abbreviation of $\{o(k) o(k) \in \mathbf{O}, 1 \leq k \leq K\}_{\mathbf{x}}^t$, a sequence of observations centred at pixel \mathbf{x} in frame t for length K
$\{s(k)\}_{\mathbf{x}}^t$	Abbreviation of $\{s(k) s(k) \in \mathbf{S}, 1 \leq k \leq K\}_{\mathbf{x}}^t$, a sequence of states centred at pixel \mathbf{x} in frame t for length K
o_n	An observation in \mathbf{O}
$p(Path_{0,K})$	The probability of a random walk $Path_{0,K}$ starting at pixel \mathbf{x}^0
s_m	A state in \mathbf{S}
t	Index of frames in image sequence I
$w_{\mathbf{x}', \mathbf{x}''}$	A weight on edge $e_{\mathbf{x}', \mathbf{x}''}$ in graph G

List of Figures

1.1	The structure of proposed automatic restoration framework for archive film.	4
1.2	4 sample degraded frames with <i>Dirt and Blotches</i> from Squirrel (hair), Cinderella (dirt), Junior Prom (blotches) and Lamp of Memory (scratches) are shown from top to bottom, left to right.	5
1.3	Cliff - A degraded frame with degraded pixels labelled and restored by the proposed framework.	6
2.1	Sample restored frames $t - 1$, t and $t + 1$ from Archangel Ph.C-HD. Images are extracted from a sample sequence from [91].	12
2.2	Teranex VC300 with integrated touch screen GUI from [96].	13
2.3	Motion compensated temporal neighbours defined in ROD [68].	14
2.4	A comparison of image structure propagation based (c) and exemplar based (d) methods with the original image (a) and inpainting mask (b).	21
2.5	Level lines connected over an occluded region from [65].	23
2.6	An example of completing an occluded object with the gap is wider than the width of the object. The desired solution and results from TV model are also shown. Image is taken from [17].	24

2.7	The priority has been given to structure propagation in exemplar based methods [24, 34, 72, 105].	27
2.8	A sample image is decomposed into texture and non-texture subimages and restored respectively [8]. Top: the original image before and after inpainting; Middle: texture and non-texture sub-images; Bottom: inpainted texture and non-texture sub-images.	29
2.9	Sliding windows used in ML3dex from [56].	29
3.1	Junior Prom - Dirt: Top: sample degraded frames $t - 1$, t and $t + 1$; Bottom: enlargement of selected areas.	36
3.2	Personal Hygiene - Moving line scratches: Top: sample frames $t - 1$, t and $t + 1$ degraded with moving line scratches; Bottom: enlargement of selected areas.	37
3.3	Steel - Line scratches: 4 consecutive sample frames.	37
3.4	New York World Expo - Colour variation: Local colour variation possibly caused by mould is shown.	38
3.5	Junior Prom - Instability: Sample degraded frames with global vibration across 4 successive frames.	39
3.6	New York World Expo - Blurring: Examples of blurring caused by focus error (top) and long exposure time (bottom).	40
3.7	10 sample normal frames from training data (one from each sequence).	41
3.8	1-18 of 50 real degraded frames from testing data (one from each sequence).	45
3.9	19-36 of 50 real degraded frames from testing data (one from each sequence).	46
3.10	37-50 of 50 real degraded frames from testing data (one from each sequence).	47
3.11	Example frames from 10 artificially degraded testing sequences.	48

4.1	Statistics computed for both degraded temporal pixel sequences and normal ones are shown based on two types of observation, <i>i.e.</i> image intensity (top) and absolute intensity difference (bottom).	54
4.2	The diagram shows the system structure of our proposed defect detection framework.	55
4.3	Graphical illustration of a general architecture of HMM modelling.	56
4.4	Graphical illustration of an Ergodic HMM.	59
4.5	Graphical illustration of a Left-Right HMM.	60
4.6	Graphical illustration of a circular HMM.	61
4.7	Graphical illustration of state transitions of a 5-state HMM in our application.	62
4.8	Leave-one-out processing of a sample observation sequence with $K = 7$.	67
4.9	ROC graph shows a comparison of HAFID against four well known or current state-of-the-art techniques [52, 54, 67, 84], averaged across our entire handlabelled test data set.	74
4.10	ROC graph shows a comparison of HAFID with different window sizes K , averaged across our entire handlabelled test data set.	77
4.11	Seaman - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.	78
4.12	Cinderella - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.	80
4.13	Science - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.	81

5.1	The two-stage structure of our proposed false alarm elimination algorithm.	86
5.2	Man on the Land (Inter-motion) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.	88
5.3	New York World Expo (Intra-motion) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.	89
5.4	Story of Television (Complex motion) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.	90
5.5	American Girl (Shot transition) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.	91
5.6	Is This Love (Blurring) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.	92
5.7	Artificial Respiration (Spatial continuity) - From top to bottom: original images (frame n and $n + 1$); results from HAFID; results after false alarm elimination with MRF modelling only; results after the full algorithm.	95
5.8	Cinderella (Temporal correlation) - From top to bottom: original images (frame n and $n + 1$); results from HAFID; results after false alarm elimination with Motion Analysis only; results after the full algorithm.. . . .	96
5.9	ROC graph shows a comparison of HAFID-STC against four well known or current state-of-the-art techniques [52, 54, 67, 84], averaged across our entire test data set.	99
5.10	The graph shows a comparison of HAFID-STC with different window sizes K for motion tracking, averaged across our entire handlabelled test data set.	100

5.11	Henry Ford - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID and HAFID-STC. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.	102
5.12	American - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID and HAFID-STC. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.	103
6.1	Spatiotemporal motion compensated neighbourhood across 3 frames.	108
6.2	Outline of our proposed defect removal method.	111
6.3	Visual comparison between traditional searching and proposed advanced searching. Blue: Searching range; Green: Sliding Window; Red: Target pixel; dashed line: Motion trajectory.	112
6.4	Stadium - Top row: three consecutive frames $n - 1$, n and $n + 1$; Bottom three rows (from left to right): enlargement of selected areas, defect maps with the target pixel (the starting position of random walks) in blue, likelihood maps of pixels being the replacement for the target pixel after performing random walks independently for 50, 250 and 800 times. The higher grayscale level, the more likely is a pixel to be the replacement.	114
6.5	Story of Television - from top to bottom: the original frame and its defect map; enlargement of selected areas overlaid with defect maps; Initialized $\mathbf{r}(\cdot)$ maps. Green: true defects; Red: false alarms Blue: defects not detected.	116
6.6	(from left) A sample image pyramid, the defect map pyramid, and the restored results using the proposed method. The degraded regions are gradually recovered from coarse to fine and from the boundaries to their inner part.	118
6.7	Cliff - Comparing large missing area recovery. Top: original frame and the defect map in red; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.	124

6.8	Red China - Comparing large missing area recovery. Top: original frame and the defect map in red; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.	125
6.9	Policeman - Comparing restoration of false alarms. Top: original and its defect map; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.	126
6.10	Artificial Respiration - Comparing restoration of false alarms over Three consecutive frames. From top to bottom: original frames, their defect map, restoration results from Kokaram04, GD06, and the proposed method.	127
6.11	Junior Prom - Comparing motion correction. (from top left to bottom right) Original frame with overlaid defects, Original motion vectors, Corrected motion vectors from Kokaram04 and the proposed method.	128
6.12	Coffee - Comparing motion correction. (from top left to bottom right) Original frame with overlaid defects, Original motion vectors, Corrected motion vectors from Kokaram04 and the proposed method.	128
6.13	The graph shows the deviation of the optimal replacements' intensities falls as the number of random walks performed increases.	130
A.1	A painting restored by our proposed method is shown with original images on the left, masks in the middle and results on the right.	150
A.2	Two examples of image inpainting are shown with the original image on the top, inpainting mask in the middle and restored image at the bottom.	151
A.3	Bungee - Top: Original image and inpainting mask; Bottom: (left to right) inpainting results from structure propagation based method [6], exemplar based method [24] and our proposed method.	152
A.4	Container - Inpainting results from 5 consecutive frames from top to bottom. Original frames are shown on the left, mask in the middle and results on the right.	155

A.5	Stadium - Inpainting results from 5 consecutive frames from top to bottom. Original frames are shown on the left, mask in the middle and results on the right.	156
-----	---	-----

List of Tables

2.1	Definition of Pathological Motion events in terms of intensity discontinuity from [23].	20
4.1	Sample observation sequence and its associated state sequence	64
4.2	Accuracy of HMM modelling computed on 10 normal sequences	72
4.3	Seaman - Statistical results of defect detection	78
4.4	Cinderella - Statistical results of defect detection	80
4.5	Science - Statistical results of defect detection	81
4.6	Averaged computational speed for all methods	82
5.1	Henry Ford - Statistical results of false alarm elimination	102
5.2	American - Statistical results of false alarm elimination	103
5.3	Averaged computational speed for all methods	104
6.1	Comparison of MSEs on 1-5 of 10 real sequences with synthetic errors	121
6.2	Comparison of MSEs on 6-10 of 10 real sequences with synthetic errors	122

Introduction

Restoration of old, archived films is of great importance to preserve the originality of such media in terms of “a historical record”, as well as the means to quality improvement for reproduction purposes. Most filmed footage, whether recently shot or historical, will experience impairment during the storage, duplication, transmission and even at the time of filming. Repeated damage to the footage will finally degrade audience’s perception in various forms, such as increased level of noise, flicker, vibration and so on. Hence, such films require quality control and restoration before they are broadcastable again.

Currently the most reliable method is manual restoration, a tiresome course of restoration by chemical and physical means [82]. In order to recover an archive film, restoration specialists with adequate training and experience are required to perform a series of processes, *e.g.* printing, grading, restoration processing, and maintenance. The cost of staff training and film processing is high, considering it is multiplied by the numerous number of archives there exists. Besides being uneconomical, manual restoration may also cause irreversible damage to the original film copy, *e.g.* mishandling during the rewashing and drying process could result in the damage of emulsion or even broken film. Such processes could cause permanent loss of information.

Another issue of manual processing is that it is difficult to assure the quality of

restoration. There are many elements that could affect the final output of restoration in a traditional restoration process, *e.g.* restoration equipment, operator's experience, skills and even personality. Deficiency in any of these aspects will leave the quality of restoration at risk.

Based on the consideration of cost, safety and quality of the restoration process as discussed above, our motivation for implementing an automated restoration framework for archive film is threefold.

First, automated restoration in digital form is more cost-efficient. The mass availability of high capacity digital storage and broadcast technologies has had a large impact on the film industry in the last two decades. Hence, nowadays the preferred route to preservation and rebroadcasting is digitisation and automated restoration. Such techniques can speed up the whole restoration process and meanwhile greatly reduce the cost of human resources, specialized equipment and consumable materials from the traditional restoration process.

Second, digitized and automated restoration processes are able to provide a more convenient and safe solution for archive maintenance. Instead of physical transportation and repeated handling of the original film copy, remote access to the digitised copy could be achieved with a few simple mouse clicks. Furthermore, any restoration conducted on the digitised copy is reversible and will still be available for further refinement with more advanced techniques in future. And most importantly, the original film copies will remain untouched in the light of the conjecture in [82] that "Restoration is essentially duplication".

Last but not least, it is more effective to control the quality of restoration in digital form. The restoration algorithms can be adjusted (usually in forms of parameters) for different type of films in order to achieve the maximum accuracy. In the same fashion, the accuracy can also be traded for less computational cost due to possible time or budget limitations.

1.1 Automated Restoration Framework for Archive Film

Over the last two decades, there has been growing interest in developing automated quality control and improvement system for films and videos with several academic and industrial collaborations, *e.g.* BRAVA (Broadcast Restoration of Archives by Video Analysis, 1999) [14] and PrestoSpace (Preservation towards Storage and access Standardised Practices for Audiovisual Contents in Europe, 2004) [75].

In general, solutions to these projects all involve a digitised restoration system that includes a number of self-contained modules. Each module is designed to handle an individual problem which may correspond to a step in the traditional restoration framework, *e.g.* recovering flicker is related to the grading process and removing dirt and blotches corresponds to the restoration process (polishing and washing). According to different quality control and improvement requirements in each specific project, some or all of these modules are adopted to replace equivalent processes in the traditional restoration framework so as to form an automatic or semi-automatic restoration system.

Based on this system design, the work presented here focuses on the restoration problem of one group of defects, dirt and blotches (see the description below). The aim is to design a framework (as a self-contained module) to enable automatic detection and removal of the target defect. Our proposed restoration framework consists of three individual stages, *i.e.* defect detection, false alarm elimination and defect removal. An overview of our proposed restoration framework is illustrated in Figure 1.1.

There are many types of defects in archive film, such as dirt, blotches, line scratches, blurring, flickering, instability and so on. Later, we will present a brief outline of categories of defects in Chapter 3. The dirt and blotches type of defect that we deal with in this thesis can be characterised as follows.

We assume the appearance of a degraded pixel as a stochastic pixel-change event. Such changes can happen in one or more (consecutive) frames as black, white, or

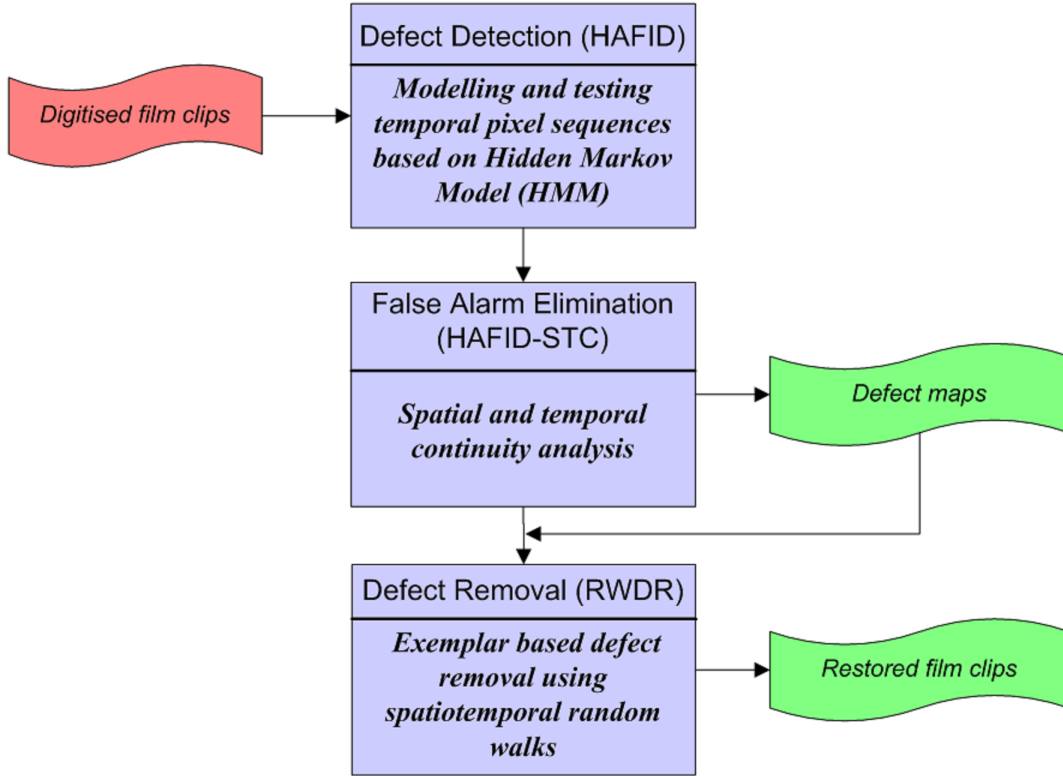


Figure 1.1: The structure of proposed automatic restoration framework for archive film.

semi-transparent regions. However, degrees of degradation and their shape and size can vary due to their random appearance.

Therefore, our target defect is not limited to any particular categories providing it has such characteristics as described above. For example, moving line scratches with different locations in each frame are considered as our target. Here, we refer to all such discontinuities as dirt and blotches, including hair, sparkle, salt and pepper like noise, digital drop out and certain line scratches. Figure 1.2 shows 4 sample degraded frames with dirt and blotches.

1.1.1 Defect Detection

Our proposed HMM based Archive Film Defect detection (HAFID) approach functions as the first step towards an automatic restoration framework. The resulting



Figure 1.2: 4 sample degraded frames with Dirt and Blotches from *Squirrel*(hair), *Cinderella*(dirt), *Junior Prom*(blotches) and *Lamp of Memory*(scratches) are shown from top to bottom, left to right.

defect maps contain the time and location information of defects. Such information could be directly employed by the follow-up removal process so as to implement a seamless automatic restoration framework without human interference. On the contrary, traditional computer aided defect removal, *e.g.* the missing data correction method from Bornard *et al.* [12], will require the input of handlabelled masks. Furthermore, defect maps also provide the possibility to obtain quantitative measures of defects, which are extremely useful for film quality control. For instance, the total number of defects in each frame indicates the degree of degradation while the distribution of a defect across a number of frames can reflect the type of defect. The central image in Figure 1.3 presents a sample degraded frame overlaid with its defect map in red from our proposed method HAFID.

The pixels labelled in our defect maps often include both true defects and false positives. Performing restoration on those falsely detected (normal) pixels might introduce new artifacts. Generally, the difficulty of defect detection lies in how



Figure 1.3: *Cliff* - A degraded frame with degraded pixels labelled and restored by the proposed framework.

to increase the correct detection rate while including fewer false detections. Our strategy is to first detect the maximum number of true defects and then process the false detections sequentially. This enables us to study both true defects and false alarms closely and design more accurate and specified approaches based on their characteristics.

Previous defect detection methods [51, 67, 84] recruit local spatial and temporal information (within 1 or 2 neighbouring frames) to help determine if a pixel is truly degraded based on the assumption that defects are single-frame events. However, our solution is to look at a larger number of frames before and after the current frame. This allows us to examine the statistical changes in pixel transitions for a longer range to help determine defects more accurately. In our proposed HAFID detector, a Hidden Markov Model (HMM) is first trained for normal pixel sequences and then applied within a framework to detect defective pixels by examining each new observation sequence and its subformations via a leave-one-out process. This approach is described in detail in Chapter 4.

1.1.2 False Alarm Elimination

Although our HAFID defect detector is capable of capturing the majority of defects, the resulting defect maps also contain quite a number of false detections. Previous works [11, 22, 23, 50, 79] have outlined some of the possible causes (see a review of

causes of false alarms in Section 5.1). These methods also made efforts to develop corresponding elimination methods for all possible causes. However, the task is difficult to complete since enumerating all possible causes itself is laborious. Therefore, in addition to looking at causes of false alarms, we also try to understand the false detections from a novel angle by investigating their characteristics. We find that false alarms caused in different circumstances share two similar characteristics and based on this, we designed a two-stage false detection elimination procedure in which each stage associates with one characteristic of false alarms. In this two-stage process, the defect map from HAFID is first modelled with a Markov Random Field (MRF) to enforce spatial continuity constraints and then the pyramidal Lucas-Kanade feature tracker [60] is applied to impose temporal correlation constraints. This entire process is referred to as HAFID-STC (Spatial and Temporal Continuity analysis) and is described in detail in Chapter 5.

1.1.3 Defect Removal

As the final step in our restoration framework, the defect removal process aims to recover the original values of pixels labelled in those defect maps. It is worth noticing that the defect maps still contain a number of false alarms even after the false alarm elimination stage. Hence, a key part of this stage is to restore truly degraded pixels while preserving the original intensities of those falsely detected pixels.

An exemplar based restoration method is our choice in this application based on the assumption that an optimal candidate for replacement of the original pixel intensity can always be located inside the target image sequence. Instead of using the traditional search scheme with sliding windows, we explore a dynamically generated region of candidate pixel-exemplars using spatiotemporal random walks. Every pixel in this region shares a significant similarity with the previous pixels in the region as defined by their features, *i.e.* intensity, motion and texture. Based on the statistics computed during the random walks, we substitute the degraded pixel with the optimal replacement selected from its region of candidate pixel-exemplars. Our proposed Random Walk based Defect Removal (RWDR) method is able to recover the degraded pixel intensities more accurately compared to traditional methods as shown later in Chapter 6. Figure 1.3 shows an example of a restored frame based

on the defect map shown in the middle.

1.2 Thesis Outline

In Chapter 2 we review the background literature for automatic archive film restoration. First, we discuss the hardware-based solutions to the automatic restoration of archive film and video, most of which are from the industry. Then, automatic restoration algorithms are grouped and introduced according to different system structures. Recent developments in defect detection, false alarm elimination and defect removal are reviewed respectively. Finally, current trends and our motivation are summarized.

Chapter 3 outlines defect and data related issues. First, we summarise the characteristics and causes of our target defect, dirt and blotches. We also introduce other categories of defects along with sample degraded frames for each category. Furthermore, the archive film restoration data set is introduced, which we use throughout the thesis for training and testing. Finally, we introduce the quantitative measures used for performance evaluations in both defect detection and restoration stages.

In Chapter 4 we introduce the proposed defect detection approach as a form of novelty detection using HMM modelling and testing. First, we train a single HMM for normal image pixel sequences, which is then applied in the testing stage to compute the likelihood of a new sequence being normal. A leave-one-out process is used to create subformations of the target observation sequence and the quality of the centre pixel is examined based on how similar the observation sequence without the centre pixel is relative to the mean of the subformations. In the experiments section, the proposed method is evaluated and compared with state-of-the-art methods using real degraded image sequences from our archive film restoration data set.

Chapter 5 first introduces the possible causes and characteristics of false detection in archive film defect detection and then provides a two-stage false alarm elimination solution by analyzing the characteristics of false positives. The defect map from our proposed defect detector is first modelled with a MRF to enforce spatial continuity

constraints and then the pyramidal Lucas-Kanade feature tracker [60] is applied to impose temporal correlation constraints. Finally, both qualitative and quantitative results are presented to demonstrate the performance of the proposed false alarm elimination method.

Chapter 6 discusses the post-detection work of archive film restoration. A novel method is proposed for restoration of missing regions in video sequences. Our statistical framework is based on random walks to examine the spatiotemporal path of a degraded pixel. The degraded pixels within a frame are restored in a multiscale framework by updating their features (intensity, motion and texture) at each level with reference to the attributes of normal pixels and other defective pixels in the previous scale as long as they fall within the defective pixel's random walk-based spatiotemporal neighbourhood. The proposed algorithm is compared against state-of-the-art methods to demonstrate improved accuracy in restoring synthetic and real degraded image sequences. In addition, we also demonstrate the application of our proposed method on image and video inpainting in Appendix A.

In Chapter 7 concluding comments are given about the work presented in this thesis and ideas for possible future works are discussed.

Background

Over the last two decades, automatic restoration algorithms have become more and more sophisticated through the mass availability of fast-computing facilities and high capacity digital storage. These methods have evolved from filter-based methods to high order mathematic models.

The structures of these restoration frameworks have nevertheless steered and adapted according to the cost-efficiency requirements from the industry. Traditionally, the filter-based methods restore the archive film by processing every pixel in the target sequence regardless of its degree and nature of degradation. On the other hand, recent restoration frameworks usually involve a quality control (defect detection) stage for quality inspection purpose. Hence, films (or scenes) with less degradation will not necessarily go through the full restoration process. In addition, the detection procedure provides not only quantitative evidence for quality control but also the time and location information of defects, which will increase the efficiency of defect removal process by only looking at labelled pixels. We will review methods from both filter-based and two-stage system structures while recent methods in defect detection, false alarm elimination and defect removal are discussed respectively.

In this chapter, techniques and algorithms are reviewed for automatic restoration in archive film while the methods are limited to those developed for detecting and restoring dirt and blotches. In the following Section 2.1, hardware based methods



Figure 2.1: *Sample restored frames $t - 1$, t and $t + 1$ from Archangel Ph.C-HD. Images are extracted from a sample sequence from [91].*

are outlined first. Defect detection methods will be reviewed in Section 2.2 and then followed by false alarm elimination methods in Section 2.3. Finally in Section 2.4, we will introduce the algorithms developed generally for filling the missing region and then those designed especially for archive film defect restoration. Approaches used to compare against our methods will be discussed later in more detail in Chapter 4 and Chapter 5 for defect detection [52, 54, 67, 84] and in Chapter 6 for defect removal [34, 51].

2.1 Hardware based Automatic Restoration

The earliest solution to the problem was hardware based and can be traced back to 1985 when BBC’s prototype equipment [93] was invented to detect and remove the film grain, dirt, and unsteadiness electronically. The idea was to examine if the pixel’s temporal forward and backward intensity differences are above a certain threshold and then replace its intensity with the median of pixels from the local neighbourhood. The equipment was reported to be capable of removing small defects automatically while large degraded regions needed to be handled manually to avoid introducing motion impairment. This detection-and-removal two-stage system structure has been inherited and popularly used in recent restoration systems compared to filter-based methods.

Recently, the Snell Group released the fully automated “Archangel Ph.C-HD” [91] restoration system for High Definition videos as the latest successor of its “Archangel” family. This state-of-the-art hardware based system is housed in a 3RU chassis. It



Figure 2.2: *Teranex VC300 with integrated touch screen GUI from [96].*

can perform motion compensated removal of noise, dirt, scratches and flicker in real-time. Figure 2.1 shows 3 sample restored frames by Archangle. Some defects (especially large degraded region) were not restored accurately.

Teranex VC300 platform [96] is another commercial solution to the archive film restoration problem. The system is also implemented in a 3RU frame based on Teranex's patented SIMD array processing technology (as shown in Figure 2.2). A family of real-time and interactive repair tools is provided to mix and match according to specific requirements for each application.

Additionally, some restoration algorithms were accelerated with hardware implementations. For example, Marshall *et al.* [41, 43] first developed a Soft Morphological Filter (SMF) based algorithm for archive film restoration, whose structuring elements were computed in advance using a genetic algorithm. Later in [40, 45], this algorithm was implemented in hardware with FPGAs. It was reported that the computational cost of attaining the optimal structuring elements was reduced from days to minutes.

2.2 Defect Detection

Archive film defect detection methods can be broadly categorised into filter-based and model-based methods.

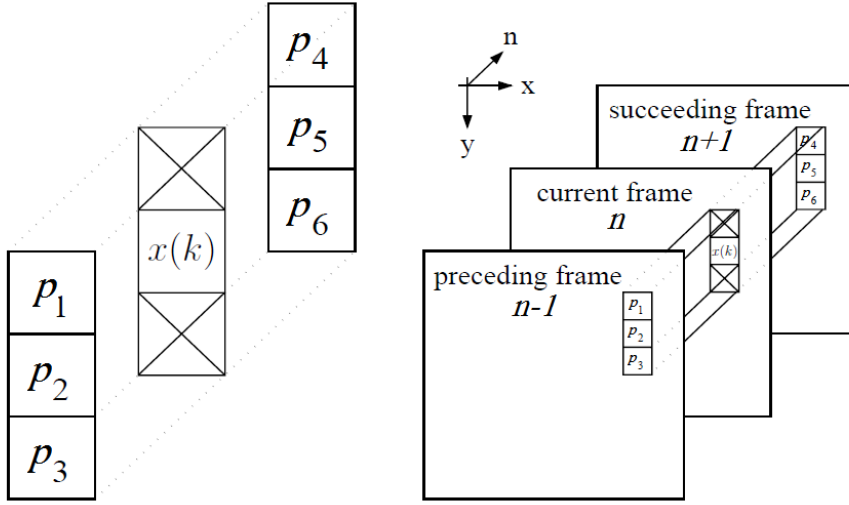


Figure 2.3: Motion compensated temporal neighbours defined in ROD [68].

2.2.1 Filter-based Methods

Probably the earliest work on filter-based archive film defect detection is the BBC's hardware-based dirt and sparkle detector [93] as mentioned above. A binary defect map was generated by examining if the temporal forward and backward intensity differences were above a certain threshold. In addition, the Double Threshold Method (DTM) from Schallauer *et al.* [90] further demanded the absolute difference between two temporal neighbours was below a second threshold.

The Spike Detection Index (SDIp) [54] is a similar example of filter-based detectors, which improved Storey's method [93] by using motion compensated pixel values while requiring the forward and backward intensity differences to have the same sign. This method is strongly sensitive to the errors of motion estimation.

Nadenau and Mitra's Rank Order Detector (ROD) [68] further extended SDIp detector [54] to use six pixel values from motion compensated temporal neighbours as $\{p_i | 1 \leq i \leq 6\}$ (illustrated in Figure 2.3). The algorithm sorted the six values as $\{s_i | 1 \leq i \leq 6, s_1 < s_2 \dots < s_6\}$ and calculated the rank-ordered mean $m_x = (s_3 + s_4)/2$. Three rank-ordered differences $diff = [diff_1, diff_2, diff_3]$ were

computed,

$$diff_1 = \begin{cases} s_i - I_{\mathbf{x}} & I_{\mathbf{x}} \leq m_{\mathbf{x}} \\ I_{\mathbf{x}} - s_{7-i} & I_{\mathbf{x}} > m_{\mathbf{x}} \end{cases} \quad (2.1)$$

A pixel $x(k)$ was then marked as a defect if $diff_1 > \tau_1$, $diff_2 > \tau_2$ and $diff_3 > \tau_3$. ($\tau_i | i = 1, 2, 3$) were thresholds, which were experimentally selected. In addition to direct temporal neighbours, the spatial information from local neighbourhood was recruited to minimize the misjudgement caused by erroneous motion information. In [35], Gangal *et al.* upgraded the ROD detector with a more accurate bi-directional motion estimator and examined the temporal neighbours in a 5-frame scheme. Furthermore, [86] has attempted to accelerate the ROD detector by only thresholding the intensity differences between the target pixel and local extremes.

Other recent work includes Ren and Vlachos's [83, 84] filter-based method that employed a more complicated confidence function of intensity differences to measure the likelihood of a pixel being defective. First, given three adjacent frames f_{t-1} , f_t and f_{t+1} , the difference map $diff_t$ for frame f_t was defined as

$$diff_t = \begin{cases} \frac{2 \cdot diff_{t-1} \cdot diff_{t+1}}{|diff_{t-1}| + |diff_{t+1}|} & diff_{t-1} diff_{t+1} > 0 \\ 0 & otherwise \end{cases} \quad (2.2)$$

where $diff_{t-1} = I(f_t) - I(f_{t-1})$ and $diff_{t+1} = I(f_t) - I(f_{t+1})$. Then the probability density function $P(\mathbf{x})$ is computed based on the histogram of the difference map $diff_t$. Finally, the confidence function $p_c(m)$ is defined for each value m in $diff_n$

$$p_c(m) = (M - 1) \frac{\sum_{\mathbf{x}=m_0}^m P(\mathbf{x})}{\sum_{\mathbf{x}=m_0}^M P(\mathbf{x})} \quad (2.3)$$

where M is the size of the set of all possible value in $diff_t$ and m_0 is set as a constant. This confidence function was able to highlight those pixels with large temporal differences. Pixel \mathbf{x} would be marked in defect maps if its confidence value is below the threshold T_m . In addition, local region growing algorithms were adopted to help eliminate false alarms that have strong correlations with their spatial neighbours. The correlation was measured by examining each pixel's features, *i.e.* mean and standard deviation of image intensities.

Morphological filtering [92] is another popular tool in archive film restoration. Morphological filtering based methods consider slightly larger spatial neighbourhoods (*e.g.* 5 by 5 pixels) than the previous methods. However, a common dilemma of these methods has been that while a large structuring element is usually required for detecting large degraded region, it also increases the possibility of picking up false detections.

In [30, 31, 32], Ferrandière regarded the target image sequences as a 3-D matrix and designed two-step morphological operations for defect detection. A temporal opening operation was first performed with a linear structuring element parallel to the temporal axis. Defects that were temporally isolated were erased together with a number of normal pixels which usually involved motions. Therefore, as the second step, a geodesic spatial reconstruction method [102] was adopted to recover the contours which were damaged during the first operation. Defect maps were generated by comparing the reconstructed images against the original ones.

In [15, 94, 95], more complicated structuring elements were specially designed to detect blotches by assuming target blotches are local maximum or minimum. White and black blotches were detected individually by using multiple structuring elements and thresholding D_{white} and D_{black}

$$\begin{aligned} D_{white}(I, SE_0, SE_n) &= I - (((I \ominus SE_0) \ominus SE_n) \oplus SE_0) \oplus SE_n \\ D_{black}(I, SE_0, SE_n) &= (((I \oplus SE_0) \oplus SE_n) \ominus SE_0) \ominus SE_n - I \end{aligned} \quad (2.4)$$

where \oplus and \ominus represent morphological dilation and erosion. SE_0 and SE_n were structuring elements defined as

$$SE_0 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad SE_n = \begin{pmatrix} 2n & 2n & 2n & 2n & 2n \\ 2n & n & n & n & 2n \\ 2n & n & 0 & n & 2n \\ 2n & n & n & n & 2n \\ 2n & 2n & 2n & 2n & 2n \end{pmatrix} \quad (2.5)$$

Here, structuring elements SE_0 and SE_n were designed to detect small defects and relatively larger defects with high gradients respectively. Detectors equipped with a combination of both elements (shown in 2.4) not only revealed various sizes of defects

but also reduced the false detection significantly. Furthermore, motion compensated temporal neighbours were computed and serve as assisting evidence to help prevent false detections.

In [97, 99], Tilie *et al.* applied Dempster-shafer evidence theory on the fusion of detection results from a Simplified ROD (SROD) detector [86] and spatial morphological filtering [15]. The combined detection results achieved an improved accuracy (with less false alarms) by taking advantage of both temporal and spatial information.

2.2.2 Model-based Methods

Morris's MRF based defect detector [67] is a pioneering example of model-based defect detector. The author adopted an *Ising* model to represent the prior of the defect map D ,

$$P(D) = \frac{1}{Z_D} \exp\left(-\frac{1}{T} \sum_{\mathbf{x} \in I} [-\beta_1 \sum_{\mathbf{x}' \in N_{\mathbf{x}}} d_{\mathbf{x}} d_{\mathbf{x}'} + \beta_2 \delta(1 - d_{\mathbf{x}})]\right) \quad (2.6)$$

where β_1 controls the smoothness of degraded region while β_2 assigns a penalty to pixel positions marked as defects in order to prevent a solution with every position detected as defects. Then, Gibbs sampling with annealing was applied to compute the Maximum *a posteriori* (MAP) configuration of the defect map given the image intensities from adjacent motion compensated image frames. The *a posteriori* distribution was defined as

$$P(D|I) = \frac{1}{Z} \exp\left(-\frac{1}{T} \sum_{\mathbf{x}} [\alpha(1 - d_{\mathbf{x}})(I_{\mathbf{x}} - I_{\mathbf{x}_{mc}})^2 - \beta_1 \sum_{\mathbf{x}' \in N_{\mathbf{x}}} \delta(d_{\mathbf{x}} - d_{\mathbf{x}'}) + \beta_2 \delta(1 - d_{\mathbf{x}})]\right) \quad (2.7)$$

where \mathbf{x}_{mc} was the motion compensated temporal neighbour of \mathbf{x} , either forward or backward connected and parameter α highlights the discontinuity of intensity values on the temporal axis. Pixels were marked in the final defect map if discontinuities were shown on both forward and backward adjacent frames.

In [21], Morris's method [67] was reported to be sensitive to the errors of motion estimations and cause false alarms along the moving edges. Therefore, Chong and Krishnan [21] suggested an improvement by adding a moving edge detection element into Morris's model so as to increase the accuracy. The new *a posteriori* distribution was defined as

$$P(D|I) = \frac{1}{Z} \exp\left(-\frac{1}{T} \sum_{\mathbf{x}} [\alpha(1 - d_{\mathbf{x}})(I_{\mathbf{x}} - I_{\mathbf{x}_{mc}})^2 - (\beta_1 + \phi(\mathbf{x}, \mathbf{x}_{mc})) \sum_{\mathbf{x}' \in N_{\mathbf{x}}} \delta(d_{\mathbf{x}} - d_{\mathbf{x}'} + (\beta_2 + \phi(\mathbf{x}, \mathbf{x}_{mc}))\delta(1 - d_{\mathbf{x}}))]\right) \quad (2.8)$$

where

$$\phi(\mathbf{x}, \mathbf{x}_{mc}) = \begin{cases} 0 & \mathbf{x} \text{ not on a moving edge} \\ \max_{\mathbf{x}' \in N_{\mathbf{x}}} (I_{\mathbf{x}'} - I_{\mathbf{x}'_{mc}})^2 / \tau_1^2 & \mathbf{x} \text{ on a moving edge} \end{cases} \quad (2.9)$$

Pixel \mathbf{x} was detected as a moving edge pixel if the bidirectional displacement frame differences (DFD) of every pixel \mathbf{x}' in the spatial 8-connected neighbourhood $N_{\mathbf{x}}$ of \mathbf{x} were below the threshold τ_2 .

Kokaram [52] also developed a Bayesian framework to model noise and blotches while performing motion correction. Three binary variables were used for each pixel to mark if the pixel is degraded, forward occluded or backward occluded. These variables, together with restored image values and motion vectors, were defined as unknowns. Given the pixel values of degraded frames and initial motion estimations, the method applied the ICM [9] algorithm to solve these unknowns via an iterative procedure. More details about this method, *i.e.* two-stage restoration process and motion correction, will be discussed in Section 2.4.

2.3 False Alarm Elimination

All defect detection methods reviewed above suffer from false detections to some degree. In order to improve the performance of detectors, a straightforward idea is to include an extra process to reduce false detections as much as possible while preserving the highest possible detection rate. Many algorithms [1, 10, 11, 22, 50,

58, 79, 98, 104] have adopted this route with over-detection and then false alarm elimination, rather than under-detection and then dealing with its consequences. In this preferred route, the difficulty of false alarm elimination has been transferred to how to model normal pixels, particularly false alarms, to distinguish them from true defects.

A few works [78, 79, 80] investigated the causes of false alarms and developed their corresponding solutions. One of the major causes is the irregular motion of camera and objects, especially in fast motion scenes, which was referred to as *pathological motions*. In [78, 79, 80], Rares *et al.* briefed all classes of pathological motions which may cause false detections, including motion blurring, occlusion, image overlapping, interlacing and shot change, large displacement, strong zooming and intermittent/erratic motion. Many algorithms [10, 11, 22, 23, 50, 50, 58, 59, 78, 86] have been designed to remove one or multiple classes of false alarms by focusing on their causes.

As an increment to Roosmalen’s SROD detector [86], eliminating false alarms caused by motion blurring was introduced in [78]. False alarm regions were excluded from the initial defect maps through a 3-step segment matching process using statistics computed from each segment’s intensity histogram across temporal neighbouring frames.

In [11, 22, 23, 50], the authors distinguished the pathological motions from true artifacts by defining the pathological motions on the temporal axis. A set of binary variables were used to reflect the intensity changes across a fixed number (3 or 5) of consecutive frames.

Bornard [11] first pointed out that the appearance of intensity discontinuities at same location across three consecutive frames could suggest the existence of a pathological motion event (occlusion in this case). Two binary variables were recruited to state the forward and backward occlusions on the temporal axis and MRF modelling was adopted to compute the MAP configuration of binary variables.

Additionally, Corrigan *et al.* [22, 23] further improved the work in [11] by using four binary variables instead of two, which represent the states of intensity changes across five neighbouring frames. More pathological motion events were explicitly

Table 2.1: Definition of Pathological Motion events in terms of intensity discontinuity from [23].

State	$s(\mathbf{x})$	Detection	State	$s(\mathbf{x})$	Detection
0	0,0,0,0	Normal	8	1,0,0,0	Normal
1	0,0,0,1	Normal	9	1,0,0,1	PM
2	0,0,1,0	PM	10	1,0,1,0	PM
3	0,0,1,1	PM	11	1,0,1,1	PM
4	0,1,0,0	PM	12	1,1,0,0	PM
5	0,1,0,1	PM	13	1,1,0,1	PM
6	0,1,1,0	Defect	14	1,1,1,0	PM
7	0,1,1,1	PM	15	1,1,1,1	PM

defined (as shown in Table 2.1), in which $s(\mathbf{x}) = \{s_1(\mathbf{x}), s_2(\mathbf{x}), s_3(\mathbf{x}), s_4(\mathbf{x})\}$ was the set of binary variables showing if there were large intensity changes (as $s_i(\mathbf{x}) = 1$) between temporally consecutive pixels across five frames. However, the accuracy of this method is still limited by the number of binary variables since the intensity transitions in complex motion events, *e.g.* tracking an object with fast movements or slow periodic motions, need to be examined in a longer range.

Improvement on Morris’s defect detector [67] was also reported in [50] by including a segmentation process before performing the defect detection. Images were first segmented into foreground and background, involving different types of motion. Then, Morris’ method was performed on foreground and background parts individually so that false detection caused by complex foreground motions could be reduced.

There are other false alarm elimination methods [10, 58, 59], which focused on the special kinds of false alarms generated under more specified circumstances. For instance, [10] attempted to remove isolated false detected pixels caused by Gaussian noise and Licsár *et al.* [58, 59] limited their target defect to blotches so that false detections could be eliminated with the help of using blotches’ spatial features, such as local smoothness and the difference between intensity means inside and outside blotches.

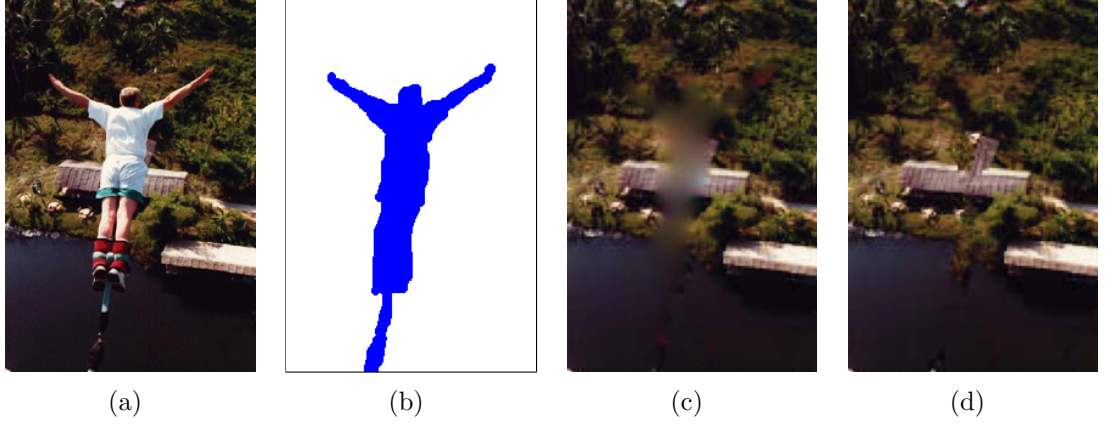


Figure 2.4: A comparison of image structure propagation based (c) and exemplar based (d) methods with the original image (a) and inpainting mask (b).

2.4 Defect Removal

The task of filling in missing regions in single or consecutive frames is often referred to as *inpainting*, which originates from restoration in the world of Art. It was first introduced into digital image restoration by Bertalmío *et al.* [6] who adapted the original idea of artistic inpainting by smoothly propagating the surrounding colour and structure into the missing area iteratively. The intensity updating function was defined as

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_t^n(i, j), \forall (i, j) \in \mathcal{D}, \quad I_t^n(i, j) = \nabla L^n(i, j) \cdot |\nabla^\perp I^n(i, j)| \quad (2.10)$$

where \mathcal{D} represents the inpainting region. $L^n(i, j) = I_{xx}^n(i, j) + I_{yy}^n(i, j)$ is the smoothness estimator and $\nabla^\perp I^n(i, j)$ provides the propagation direction.

Since then, inpainting has become a popular topic in computer vision and most of the research was concentrated on mainly two directions, *i.e.* image structure (non-texture) propagation based methods and exemplar (texture) based methods. Figure 2.4 illustrates sample inpainting results from both groups of methods, *i.e.* [6] for image structure propagation based methods and [24] for exemplar based methods.

2.4.1 Image Structure Propagation based Methods

This group of methods usually requires complex image models with high order partial differential equations (PDE) or variational energy functional. Although such methods have proven to be effective solutions to restoring small gaps in degraded images, they suffered from blurring side effects when dealing with large missing areas, *e.g.* they could fail to restore textural details within the missing regions they recover. An example is shown in Figure 2.4(c). The restored image lost the texture details in the filling region.

Atzori and De-Natale [3] and Rares *et al.* [81] believed that the priority of recovering the “skeleton” of images (edges) should be granted in an image restoration mission. Both algorithms first detected and coupled the edges that may have been disconnected with the appearance of missing regions. After being sketched, the missing region would be separated into independent ones and “flesh” would be added using patch duplication [3] or interpolation with a combination of surrounding image intensities [81].

Masnou and Morel [64, 65] were inspired by the Nitzberg *et al.* ’s variational framework [69] on image segmentation and edge completion. Instead of edges, they presented an algorithm that connected level lines that arrived at the boundary of the missing region by minimizing the sum of the angular total variations along each candidate level lines connections as follows:

$$\mathcal{C}_{i,j} = \sum \int_{L_{i,j}} (1 + |\kappa|^p) dl, \quad p \geq 1 \quad (2.11)$$

where $\kappa = |\nabla \cdot [\frac{\nabla f_I}{|\nabla f_I|}]|$ is the curvature of the connected level lines L_i and L_j . f_I is a noise-free image represented as a function of Bounded Variation (BV) and l donated the length of level lines. Figure 2.5 presents an example with possible configuration of level lines in an occluded region.

Chan and Shen [18] introduced the Total Variation (TV) inpainting model based on Rudin-Osher-Fatemi BV image model. BV images were interpolated by minimizing

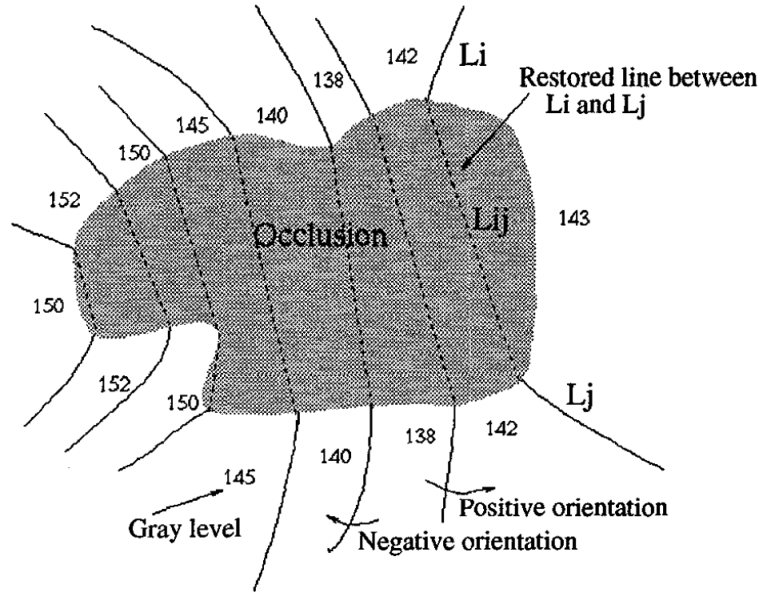


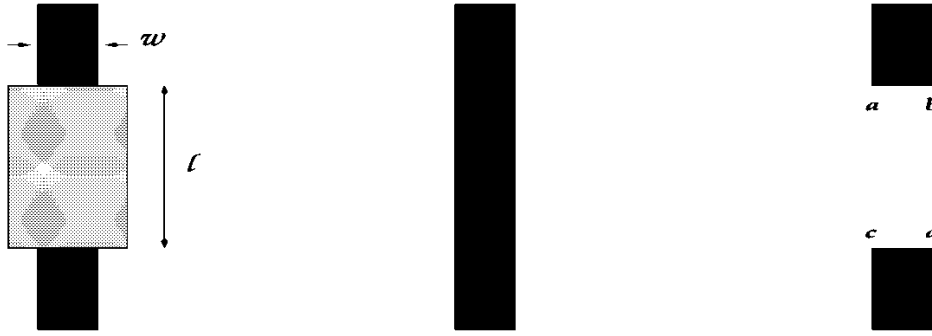
Figure 2.5: Level lines connected over an occluded region from [65].

the following posterior energy:

$$E_{TV}[f_I|f_I^0, \mathcal{D}] = \int_{\Omega} |\nabla f_I| d\mathbf{x} + \frac{\lambda}{2} \int_{\Omega \setminus \mathcal{D}} (f_I - f_I^0)^2 d\mathbf{x} \quad (2.12)$$

where λ is a predefined positive function and f_I^0 is the initial edge condition around the missing area \mathcal{D} . Inside the missing region \mathcal{D} , the TV inpainting method is equivalent to anisotropic diffusion if we leave out the second part of Equation (2.12). As reported in [19], the TV inpainting model provided an effective solution with lower complexity and easier computational implementation compared to other variational models.

However, TV inpainting model may fail to complete the occluded object when the gap is much wider than the width of the object, considering that the geometric information of isophotes is neglected in the diffusion. Therefore, Chan and Shen [17] presented an improvement based on Curvature-Driven Diffusion (CDD) to encourage the connection of edge isophotes. Figure 2.6 illustrates a sample situation with the desired recovery from a view of human beings and recovery results from previously discussed TV model. In a similar fashion to Bertalmío *et al.*'s [6] inpainting scheme,



What is behind the box? **Answer from most humans** **Answer by the TV mode**
($l \gg w$)

Figure 2.6: An example of completing an occluded object with the gap is wider than the width of the object. The desired solution and results from TV model are also shown. Image is taken from [17].

a third order PDE was employed here:

$$\frac{\partial f_I}{\partial t} = \nabla \cdot \left[\frac{G(|\kappa|)}{|\nabla f_I|} \nabla f_I \right] + \lambda_{\mathcal{D}}(f_I - f_I^0) \quad (2.13)$$

where $G(s)$ is the key factor which encourages smooth isophotes inside the missing region:

$$G(s) = \begin{cases} 1, & \mathbf{x} \in \Omega \setminus \mathcal{D} \\ s^p & \mathbf{x} \in \mathcal{D}, \end{cases} \quad \lambda_{\mathcal{D}}(\mathbf{x}) = \begin{cases} \lambda, & \mathbf{x} \in \Omega \setminus \mathcal{D} \\ 0 & \mathbf{x} \in \mathcal{D} \end{cases} \quad (2.14)$$

The CCD based method basically performed anisotropic diffusion outside the inpainting region which was similar to the TV model while the propagation inside the inpainting region was driven by a function of the curvature κ .

In addition, other improvements [4, 18, 20, 27, 101] on the TV inpainting model have been presented. The authors adopted different image models which were originally designed for image segmentation. These methods aimed to achieve smoother image interpolation based on the assumption that images are composed of a collection of smooth regions. In [18, 101], the Mumford-Shah image model was employed

$$E_{MS}[f_I, \Gamma | f_I^0, \mathcal{D}] = \frac{\gamma}{2} \int_{\Omega \setminus \Gamma} |\nabla f_I|^2 d\mathbf{x} + \alpha \text{length}(\Gamma) + \frac{\lambda}{2} \int_{\Omega \setminus \mathcal{D}} (f_I - f_I^0)^2 d\mathbf{x} \quad (2.15)$$

where Γ represents boundaries of the inpainting regions. In [4, 20], Chan *et al.* introduced the Euler's elastica image model into the inpainting task

$$E_E[f_I|f_I^0, \mathcal{D}] = \int_{\Omega} (\alpha + \beta\kappa^2) |\nabla f_I|^2 d\mathbf{x} + \frac{\lambda}{2} \int_{\Omega \setminus \mathcal{D}} (f_I - f_I^0)^2 d\mathbf{x} \quad (2.16)$$

In [27], Esedoglu and Shen combined the Euler elastica and Mumford-Shah image model and proposed the Mumford-Shah-Euler image model

$$E_{MSE}[f_I, \Gamma|f_I^0, \mathcal{D}] = \frac{\gamma}{2} \int_{\Omega \setminus \Gamma} |\nabla f_I|^2 d\mathbf{x} + \int_{\Gamma} (\alpha + \beta\kappa^2) d\mathbf{x} + \frac{\lambda}{2} \int_{\Omega \setminus \mathcal{D}} (f_I - f_I^0)^2 d\mathbf{x} \quad (2.17)$$

When we look into Equations above and compare it with (2.12), we find that a second order geometric regularity of edges $(\alpha + \beta\kappa^2)$ has been enforced on either the boundary of the occluded region or the whole domain of images, which provides more natural and geometrical modelling of the images.

Other techniques used in inpainting approaches were borrowed from other subjects, such as Navier-Stokes equation from Fluid Dynamics [7] and Ginzburg-Landau equation from Superconductors [39]. For example, Bertalmio *et al.* [7] assumed the image intensity function as the stream function for 2-D incompressible fluids and modelled it using the vorticity equation:

$$\frac{\partial w}{\partial t} + \mathbf{v} \cdot \nabla w = v \Delta w \quad (2.18)$$

where w is the Laplacian ΔI and velocity vector $\mathbf{v} = \nabla^\perp I$. Instead of solving the transport equation for I in Equation (2.18), the authors obtained the vorticity w first and then solved the following Poisson problem to recover I :

$$\Delta I = w, \quad I|_{\Gamma} = I_0. \quad (2.19)$$

2.4.2 Exemplar based Methods

Exemplar based inpainting methods, *e.g.* [11, 24, 34, 72, 105], attempted to overcome the shortcoming of image structure propagation based methods on inpainting both

texture and non-texture regions. The key point was to inpaint the missing region with the image information from the rest part of the images. The idea of texture (referred to as *Exemplar*) duplication was first suggested in a texture synthesis work [26], in which Efros and Leung assumed the texture image as a Markov Random Field and synthesized the texture in a pixel-by-pixel manner. Each pixel's intensity was computed based on its Markovian neighbours' while this Markovian neighbourhood was estimated using a patch selected from texture samples.

However, traditional texture synthesis methods [25, 26, 44, 47, 103] could not be applied for the image inpainting task directly since those methods were usually applied on pure stochastic texture images with a reference texture sample while the inpainting task on a single natural image involves working on both texture and non-texture regions.

In order to perform texture synthesis methods on natural images and image sequences, Bornard *et al.* [12] inherited Efros and Leung's idea and inpainted pixels in order by measuring the reliability of its reference Markovian neighbourhood. Furthermore, the size of spatiotemporal reference neighbourhoods was adaptive instead of the fixed one employed in [26].

In another landmark work, Criminisi *et al.* [24] also extended [26] to perform the propagation of textures using a patch-based sampling process. In [24], authors also pointed out that the order of the filling process is critical for achieving simultaneous recovery of image structure and texture while priorities were granted to those pixels with high gradients, *i.e.* image structures. Figure 2.7 shows an example of filling a missing region with the image structure (edge in this case) restored first. In order to let the structure propagation take precedence, the authors defined a Patch Priority Function $P(\mathbf{x})$ for each patch Ψ_p centred at the pixel \mathbf{x} on the boundary of inpainting area Γ :

$$P(\mathbf{x}) = Co(\mathbf{x})Da(\mathbf{x}) \quad (2.20)$$

The *confidence* term $Co(\mathbf{x})$ and the *data* term $Da(\mathbf{x})$ were defined as follows:

$$Co(\mathbf{x}) = \frac{\sum_{\mathbf{x}' \in \Psi_{\mathbf{x}} \cap (\Omega \setminus \mathcal{D})} Co(\mathbf{x}')}{|\Psi_{\mathbf{x}}|} \quad Da(\mathbf{x}) = \frac{|\nabla I_{\mathbf{x}}^{\perp} \cdot \mathbf{n}_{\mathbf{x}}|}{\alpha} \quad (2.21)$$

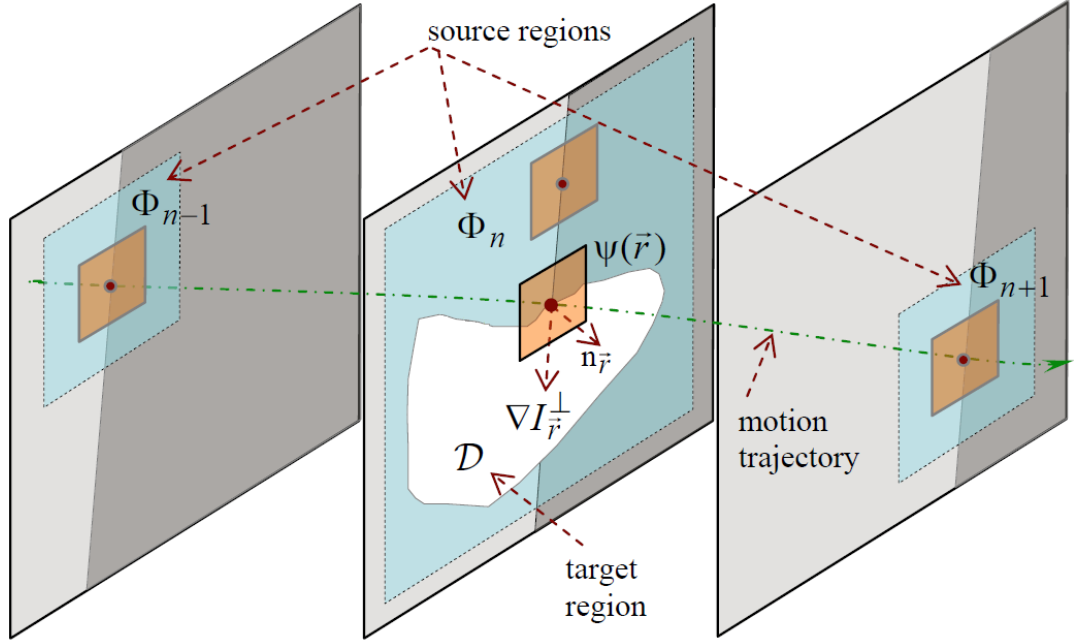


Figure 2.7: The priority has been given to structure propagation in exemplar based methods [24, 34, 72, 105].

where $|\Psi_{\mathbf{x}}|$ is the area of $\Psi_{\mathbf{x}}$, α is a normalization factor, $\nabla I_{\mathbf{x}}^{\perp}$ is the isophote at pixel \mathbf{x} , and $\mathbf{n}_{\mathbf{x}}$ is a unit vector orthogonal to Γ at pixel \mathbf{x} . The idea presented was similar to Masnou and Morel's work [64, 65] which was designed to recover edge isophotes first.

Wexler *et al.* [105, 106] and Patwardhan *et al.* [72] extended the algorithm in [24] by enlarging the sampling region to a number of temporal neighbouring frames (forwards and backwards) and both methods were designed to fill in space-time holes in video sequences with stationary background and moving foreground in periodic motions. Wexler *et al.* [106] further achieved a global optimization of patch selection in a multiscale framework. However, Patwardhan *et al.* [72, 73] considered scenes with restricted camera motions by including a motion segmentation procedure. Foreground and background parts were restored separately to preserve the spatiotemporal consistency on both intensity and motion. In [34], instead of using a global search as performed in [72, 105], Gangal *et al.* limited their search region to temporal motion compensated neighbourhoods (as shown in Figure 2.7).

These three methods ([105], [72], and [34]) inherited the shortcomings of [24]: (a) It

is difficult to choose the optimal size of an exemplar patch, considering that a larger patch will possibly bring artifacts while a smaller one may cause mismatching, and (b) A mismatching of patches in early stages will cause an incremental effect to the detriment of the final results.

In addition, methods using global image statistics were also developed. From a statistical point of view, inpainting can be regarded as a problem of estimating the missing region with the statistics from the rest parts of the image. An estimation method has been presented by Levin *et al.* [57], which intends to build an *exponential family* distribution through a training process by looking at the histograms of image features, such as gradient magnitude and pairwise gradient angle. After finding the optimized parameters for the distribution of the target image, they combined it with the image gradients computed from the pixels on the boundary to inpaint the missing region through loopy belief propagation.

2.4.3 Combined Methods

Since images usually contain both texture and non-texture regions, some works have tried to handle the challenges posed by simply treating them separately. Image decomposition algorithms and transformations, such as wavelets and DCT have been used to decompose the image into two groups of subimages, *i.e.* those exhibiting texture and those containing non-texture information (see an example in Figure 2.8). Then, corresponding methods from the two classes of structure and exemplar based approaches have been applied to the subimages separately and the final restoration was achieved by reuniting the subimages back into one, *e.g.* see [8], [71], and [33].

2.4.4 Archive Film Restoration Algorithms

We now focus on algorithms developed to restore missing scene information in archive films specifically. There is a class of methods that have used filter-based techniques applied to the entire image regardless of a defect map, *e.g.* the Lower-Upper-Median (LUM) filter [42], the ML3Dex filter [56] and the Soft Morphological



Figure 2.8: A sample image is decomposed into texture and non-texture subimages and restored respectively [8]. Top: the original image before and after inpainting; Middle: texture and non-texture sub-images; Bottom: inpainted texture and non-texture sub-images.

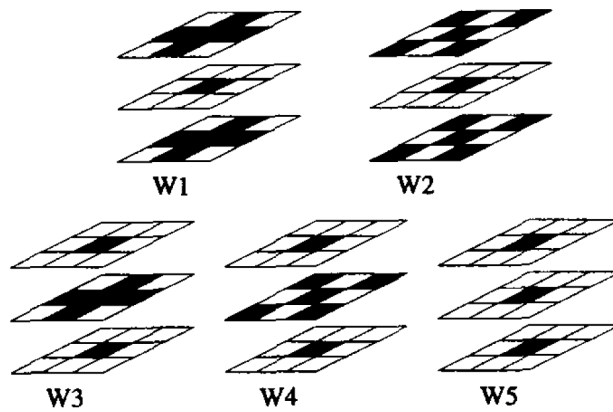


Figure 2.9: Sliding windows used in ML3dex from [56].

Filter (SMF) [41, 45]. These methods have shown good performance on removing small dirt (within 7×7 windows) in archive film but often result in artifacts elsewhere in the image by removing texture detail.

By combining LUM smoother and LUM sharpener, LUM filter [42] was essentially a rank-ordered filter using a sliding window. First, all pixels $\{\mathbf{x}_{(i)} \in W_{\mathbf{x}_c}, 1 \leq i \leq N\}$ inside the sliding window $W_{\mathbf{x}_c}$ centred at \mathbf{x}_c were ordered

$$\mathbf{x}_{(1)} \leq \mathbf{x}_{(2)} \leq \cdots \leq \mathbf{x}_{(N)} \quad (2.22)$$

Then lower and upper statistics were computed

$$\mathbf{x}^L = \text{median}\{\mathbf{x}_{(k)}, \mathbf{x}_c, \mathbf{x}_{(l)}\} \quad (2.23)$$

$$\mathbf{x}^U = \text{median}\{\mathbf{x}_{(N-k+1)}, \mathbf{x}_c, \mathbf{x}_{(N-l+1)}\} \quad (2.24)$$

where $1 \leq k \leq l \leq (N + 1)/2$ with $\mathbf{x}^L \leq \mathbf{x}^U$ and N is the number of pixels in the sliding window centred at \mathbf{x}_c . k and l are variables to control the strength of smoothing and sharpening. The final output of LUM filter was defined as

$$\bar{\mathbf{x}}_c = \begin{cases} \mathbf{x}^L, & \text{if } \mathbf{x}_c \leq (\mathbf{x}^L + \mathbf{x}^U)/2 \\ \mathbf{x}^U, & \text{otherwise} \end{cases} \quad (2.25)$$

ML3Dex filter [56] extended the LUM filter by using different shapes of sliding window across 3 adjacent frames. Totally five different windows were adopted in the filter (see Figure 2.9). The filtered pixel intensity was defined as follows:

$$\bar{x}_c = \text{median}(m_1, m_2, m_3, m_4, m_5); \quad m_i = \text{median}(W_i), \quad i = 1, \dots, 5 \quad (2.26)$$

As discussed previously, the accuracy of morphological filter based method is largely dependent on the design of structuring elements. Marshall *et al.* [41, 45] optimized their grayscale soft morphological filter using a genetic algorithm. The optimized parameters included the size and shape of structuring element's hard center and soft boundary, repetition and morphological operations. All these parameters were encoded as a "chromosome" during the evolution. The best configuration of their

SMF was given in [41] and improved results on restoring blotches were demonstrated in comparison to LUM and ML3Dex.

Additionally, a series of methods [52, 53, 55, 67, 86] have applied statistical modelling to perform the defect detection and removal in a single framework.

As a follow-up to previous MRF based defect detector[55], Morris’s missing region interpolator [67] adopted standard MRF image model using first order spatiotemporal neighbourhoods. Gibbs sampling and an improved mean field approximation method were recruited to compute the MAP configuration of restored images.

In [55], Kokaram proposed a 3-D Autoregressive (AR) model for recovering image local smoothness. Missing pixel intensities were assumed to be sums of their original intensities and errors and were then estimated using a weighted linear combination of their spatial and temporal neighbouring pixels’ intensities.

$$\hat{I}(x, y, t) = \sum_{k=1}^N a_k I(x + q_{xk}, y + q_{yk}, t + q_{tk}) \quad (2.27)$$

where $\vec{q}_k = [q_{xk}, q_{yk}, q_{tk}]$ is the location offset vector and N is the size of the spatiotemporal neighbourhood of $I(x, y, t)$. The weights $\{a_k\}$ were first predicted by minimizing the errors using normal image sequences. In another AR model based work [86, 87], instead of combining intensity values from both previous and next frames, Roosmalen interpolated the current pixel by using intensities from one direction only (either forward or backward) whose values were close to the predictions from his AR model.

As the state-of-the-art, Kokaram’s Bayesian framework [52, 53] attempted to model noise and scratches, and performed motion adjustment together. First, a set of three binary variables $s(\mathbf{x}) = [d(\mathbf{x}), O_b(\mathbf{x}), O_f(\mathbf{x})]$ were used for each pixel to mark if a pixel is degraded, forward occluded or backward occluded. In addition, the degradation model was defined as

$$G^t(\mathbf{x}) = (1 - d(\mathbf{x}))I^t(\mathbf{x}) + d(\mathbf{x})c(\mathbf{x}) + \mu(\mathbf{x}) \quad (2.28)$$

where $I^t(\mathbf{x})$ and $G^t(\mathbf{x})$ are the image intensities at time t and position \mathbf{x} before

and after the degradation. $d(\mathbf{x}) = 1$ marks pixel \mathbf{x} as a degraded position and $c(\mathbf{x})$ represents the intensity of degradation. $\mu \sim \mathcal{N}(0, \sigma_\mu^2)$ models the background noise. Binary variables $s(\mathbf{x})$, together with original image intensities $I^t(\mathbf{x})$ and motion vectors $q^{t,t-1}(\mathbf{x})$ and $q^{t,t+1}(\mathbf{x})$, were defined as unknown vector $\theta(\mathbf{x})$. Given the pixel values of degraded frames and its two neighbouring frames, the conditional distribution was defined as

$$p(\theta|I^{t-1}, G^t, I^{t+1}) \propto p(G^t|\theta, I^{t-1}, I^{t+1}) \times p(\theta|I^{t-1}, I^{t+1}). \quad (2.29)$$

where Gaussian Markov Random Field priors were used to model degraded intensity $c(\mathbf{x})$, backward occlusion $O_b(\mathbf{x})$ and forward occlusion $O_f(\mathbf{x})$. Finally, a two-stage procedure was designed to estimate $\theta(\mathbf{x})$ first and then correct the motion vectors, before repeating this process for a fixed number of iterations using the ICM [9] algorithm. The adjustment of motion vectors was block based and candidate motion vectors were selected from 8-connected spatial neighbouring blocks. The optimal motion vector was chosen provided a motion vector produced the minimum intensity difference of current block and its motion compensated one.

2.5 Summary

In this chapter, we presented a literature review on both hardware and software based archive film restoration methods. Recent developments in specified stages of automatic restoration, *i.e.* defect detection, false alarm elimination and defect removal, were discussed in detail. By studying these previous works, a few motivations are obtained for the work that we will present in this dissertation.

Previous defect detection methods assumed the defect as a stochastic event. This assumption was then followed by the practice of investigating the information from one or two previous and successive frames to determine if the current frame is suffering from degradation. Due to the high complexity of motions in archive film and possible errors introduced in motion estimation, examining the pixel intensities only in direct motion compensated neighbourhood will cause considerable number of false detections. Therefore, our approach, unlike previous methods, is to investigate

an optimized number of frames before and after the current frame so as to achieve a higher accuracy.

Previous false alarm elimination methods have paid considerable attention on the causes of false alarms. As discussed above, the major causes of false alarm in the defect detection task are the irregular and complex motions that exist in archive film. In [11, 22, 23, 50], the change of intensity values across 3 or 5 frames was investigated while patterns of intensity changes were defined for different pathological motion events. However, the accuracy of elimination was limited by the fixed number of neighbouring frames investigated and the fixed number of pathological motion events pre-defined. Instead of looking at all possible causes of false alarms, we examine the characteristics of false alarms and then design corresponding elimination methods. This is achievable since false alarms caused in different situations share similar characteristics and it is probably more efficient to focus on limited number of characteristics than to develop a solution for each cause which is laborious and sometime repetitive.

Quite a number of methods have been developed for defect removal, considering that all image/video inpainting algorithms can be directly adopted for this task. However, few methods have taken recovering falsely detected pixels into account, which can easily lead to the introduction of new artifacts especially when the number of false alarms is large. Furthermore, recovering large degraded regions remains an unsolved problem. Intensities of central pixels in a large degraded region are more difficult to restore due to the shortage of reliable neighbours. Our random walk based algorithm is designed to handle these problems with integrated multiscale scheme and reliability values.

Defects and Data

Archived films often suffer damage and quality degradation through inappropriate storage and wear and tear, and sometimes even at the time of production. These defects can be categorised according to different causes, *e.g.* dirt and blotches, line scratches, and flicker, amongst many others.

3.1 Our Target: Dirt and Blotches

As defined in Section 1.1, our target is one group of common defects in archived films, including dirt, hair, sparkle, blotches, salt and pepper like noise, digital dropout and all other temporal impulsive intensity discontinuities. We summarize the characteristics of such defective pixels as follows:

- Black, white or semi-transparent
- Random shape pattern
- Random appearance in a single frame or at the same position across several frames.
- Observed as surge or fall of image intensities

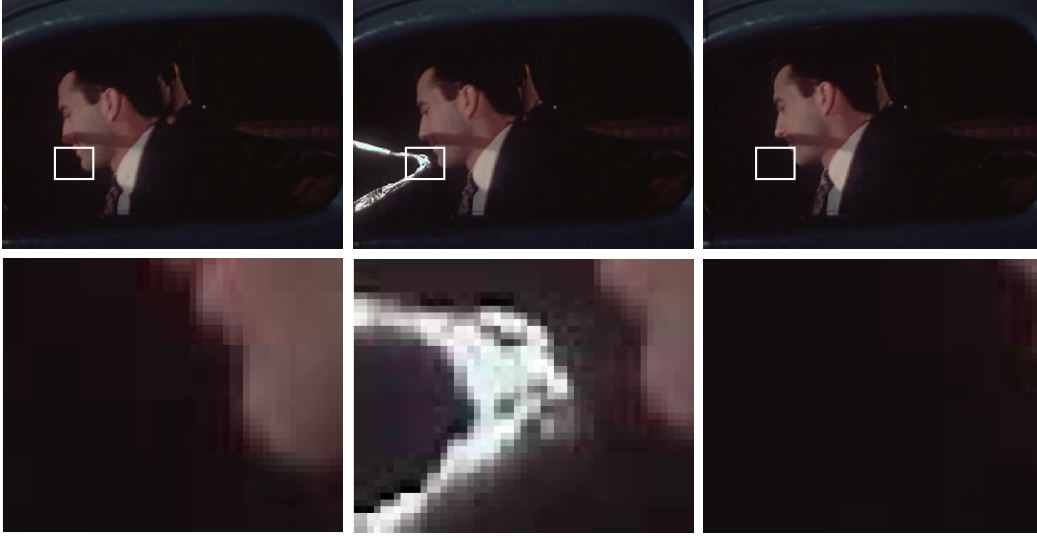


Figure 3.1: *Junior Prom - Dirt*: Top: sample degraded frames $t - 1$, t and $t + 1$; Bottom: enlargement of selected areas.

The appearance of such defects is assumed as stochastic pixel-change events due to improper storage environment or repeated damage during the acquisition, duplication, transmission and even at the time of filming. Figure 3.1 and Figure 3.2 illustrate 2 sample degraded frames along with enlargement of selected areas across 3 frames. More sample degraded frames with dirt and blotches can be found in Figure 3.8, 3.9 and 3.10.

3.2 Other Defects

In this section, we present a general introduction of other defects that commonly appear in archive film. Some of them can become our target as long as they conform to the definitions and features of our target as discussed above. An example of such kind, *i.e.* moving line scratches, is shown in Figure 3.2.

- **Line Scratches**, also known as *Tramlines*, usually appear as horizontal or vertical gaps across a number of frames. Line scratches are resulted from the film abrasion caused by ill-maintained footage transportation mechanics. A challenge in removing line scratches is how to preserve the normal line structures in the scene. An example of line scratches over 4 consecutive frames



Figure 3.2: *Personal Hygiene - Moving line scratches:* Top: sample frames $t - 1$, t and $t + 1$ degraded with moving line scratches; Bottom: enlargement of selected areas.

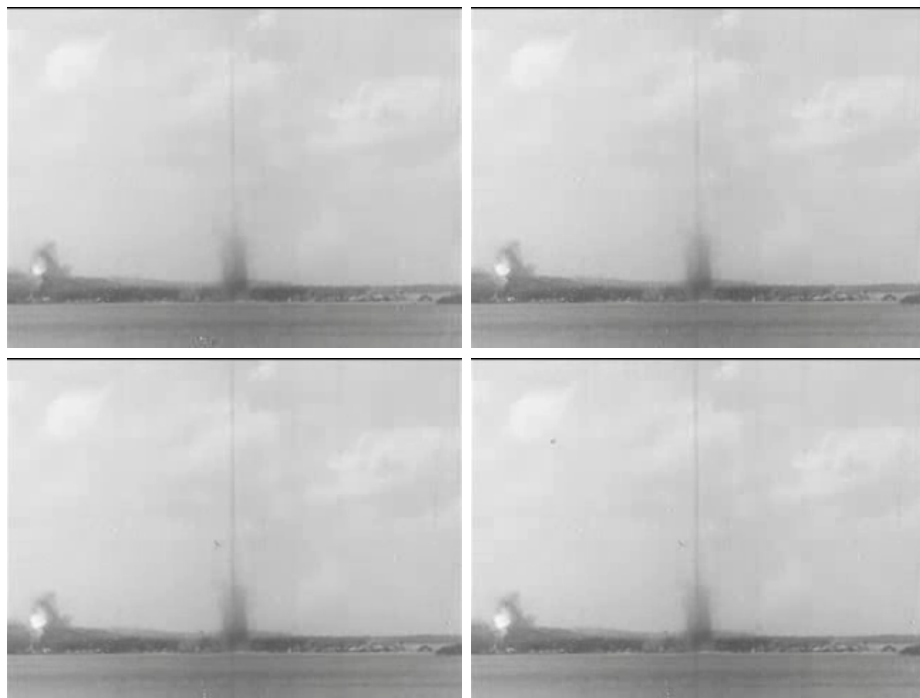


Figure 3.3: *Steel - Line scratches:* 4 consecutive sample frames.



Figure 3.4: New York World Expo - Colour variation: Local colour variation possibly caused by mould is shown.

is illustrated in Figure 3.3.

- **Brightness or colour variation** is defined here as global or local, luminance or colour variation that occurs between consecutive frames in archive film, which is usually caused by improper grading changes or inhomogeneous storage environments. An example of local colour variation is illustrated in Figure 3.4.
- **Instability** is another common distraction in old films, which are mainly caused by inaccurate transporting systems during the filming and duplication. Periodic movements can often be found across a number of frames while camera motions will further complicate the problem. Four successive frames affected by unsteadiness are shown in Figure 3.5.
- **Blurring** can be categorised into two groups according to different causes. One kind of blurring results from incorrect setting or manipulation of filming equipment, such as focus setting error and long exposure time. The resulting blur will usually influence the viewer's visual perception and the blurring we defined here is limited to this kind. On the contrary, some blurring effects are

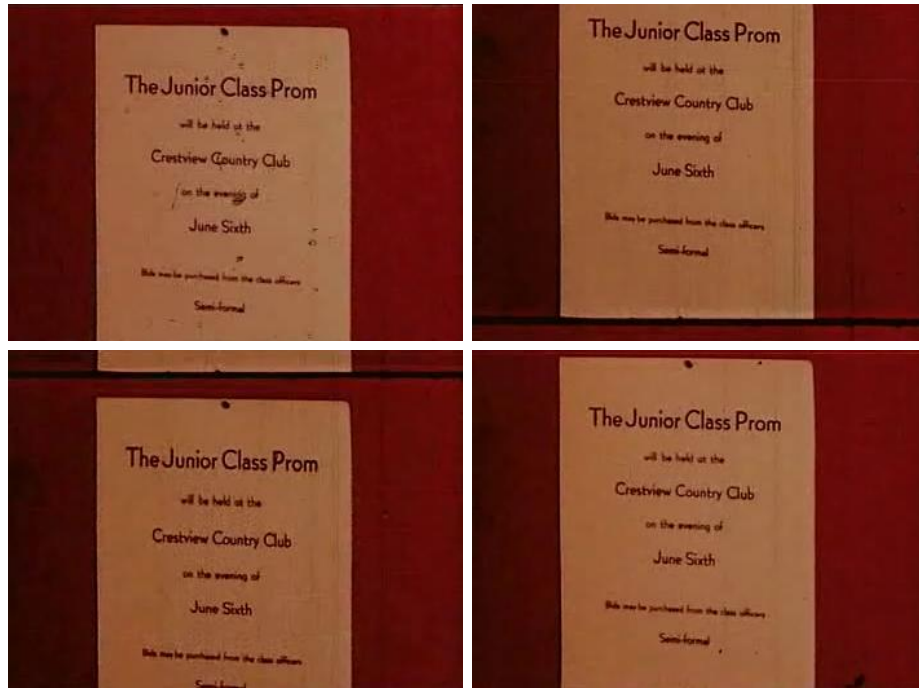


Figure 3.5: *Junior Prom - Instability:* Sample degraded frames with global vibration across 4 successive frames.

added on purpose to increase the visual coherence [107], *i.e.* the smoothness of motions. In addition, the process of frame rate converting will also introduce motion blurring into the interpolated frames. Two examples of blurring caused by focus error and long exposure time are illustrated respectively in Figure 3.6.

3.3 Archive Film Restoration Dataset

Among previous archive film restoration algorithms, few have been fully assessed using a large data set. Instead, only one or two image sequences are used to evaluate the accuracy of designed methods. The comparison with other methods can also be biased since methods may obtain better results in one particular image sequences than others. Therefore, we developed an archive film restoration dataset that comprises a variety of archive films, which will be used for comparison and evaluation of our automatic restoration methods against others. Our archive film restoration dataset is composed of two parts, *i.e.* training dataset and testing dataset.

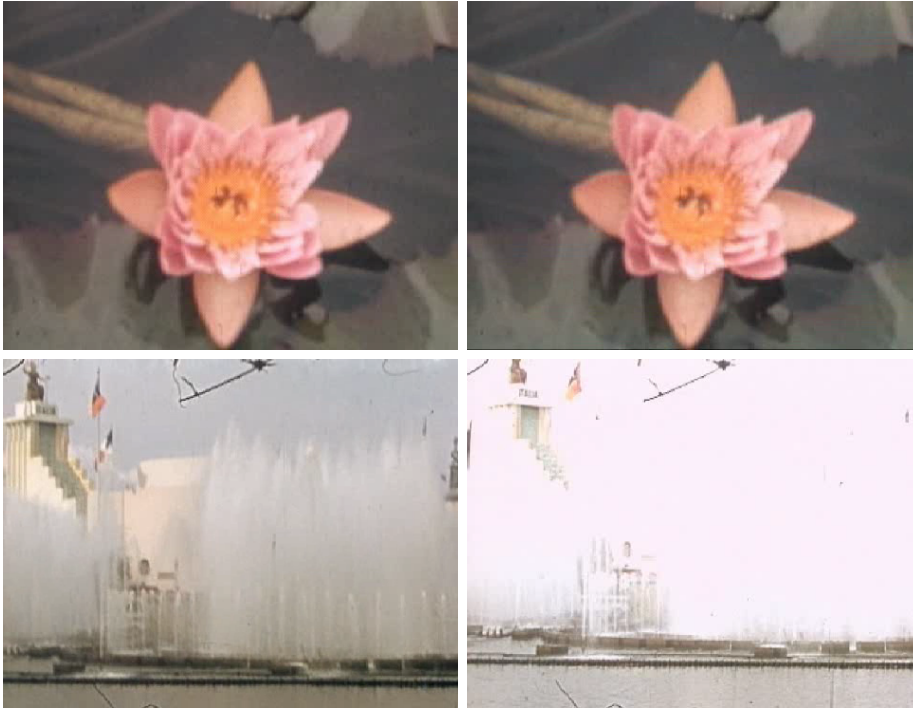


Figure 3.6: New York World Expo - Blurring: *Examples of blurring caused by focus error (top) and long exposure time (bottom).*

3.3.1 Training Dataset

The training dataset contains 10 normal archive films of different types, including grayscale, colour, indoor, outdoor, slow and fast motions, and real scenes and cartoons. These sequences only contain normal pixels and are used as the source of sample pixel sequences for HMM training. Figure 3.7 shows 10 sample frames extracted from the training dataset.

3.3.2 Testing Dataset

The testing dataset is used for both quantitative and qualitative evaluation of proposed methods. It consists of two groups of film sequences. Similar to the training data, these sequences also vary in colour, scene, degree of motions and subjects.

The first group is composed of real degraded sequences (each at a length of about one second) with handlabeled groundtruth produced from 50 film sequences totalling



Figure 3.7: 10 sample normal frames from training data (one from each sequence).

1060 frames. 50 sample degraded frames are shown in Figure 3.8, 3.9 and 3.10). This group of data is used for quantitatively measuring the performance of both the proposed defect detection algorithm and others. The archive films which we extract frames from are downloaded from [48].

The other group comprises 10 real sequences (each in a length of about 10 seconds) with artificial artifacts totalling 2952 frames (note these 10 sequences are entirely different from the ones in training data). Figure 3.11 illustrates sample frames from these sequences overlaid with artificial defects. The degraded sequences were produced by adding hand-painted black and white defects of sizes of from 1 to 6000 pixels. These data were used to evaluate and compare our proposed defect removal method against state-of-the-art methods. Quantitative assessments (*e.g.* MSE and PSNR) can be computed to measure the difference between the original defect-free frames and the restored ones. We obtained these 10 sequences from [108].

3.4 Quantitative Measures

We adopt a number of quantitative measures to objectively evaluate the performance of our proposed algorithms against previously developed techniques. For defect detection, we compare the resulting defect maps against the handlabelled groundtruth produced from real degraded sequences using binary classification test. For defect removal, quantitative results are computed based on real film sequences with artificial defects. Mean Square Error (MSE) is employed to measure the difference between the restored frames and the original defect-free ones.

3.4.1 Defect Detection Stage

As described earlier, the difficulty of defect detection lies in how to increase the correction detection ratio while keeping the false alarm ratio low. Hence, correction detection ratio and false alarm ratio are two key measurements to the accuracy of defect detection methods. They are defined in terms of binary classification test as follows:

$$\text{Correct Detection Ratio (Sensitivity)} = \frac{\text{Number of } TP}{\text{Number of } TP + \text{Number of } FN} \quad (3.1)$$

where TP stands for true positive pixels which are true defects correctly detected in our defect detection application and FN for false negatives (true defects not correctly detected).

$$\text{False Alarm Ratio (1 - Specificity)} = \frac{\text{Number of } FP}{\text{Number of } FP + \text{Number of } TN} \quad (3.2)$$

where FP stands for false positive pixels which are false alarms in our defect detection application and TN for true negatives (normal pixels not falsely detected).

Based on such definitions, we compute Receiver Operating Characteristic (ROC) curves using our testing data set with handlabelled groundtruth of defects. Then, we are able to measure the detection performance of our proposed methods and other techniques by looking at the correct detection ratios and false alarm ratios simultaneously.

3.4.2 Defect Removal Stage

The primary task of defect removal is to restore the pixels labelled in the defect maps to their original image intensities. In our experiments, artificially degraded image sequences are used as testing data. We compare the restored sequences against their original defect-free ones so as to evaluate the restoration accuracy of all defect removal methods. MSEs are computed to measure the difference between the original defect-free frame F and the restored frame \hat{F} :

$$\text{MSE}(F, \hat{F}) = \frac{1}{\text{Width} \times \text{Height} \times 3} \sum_{\mathbf{x} \in F} \sum_{i=1}^3 (F_{\mathbf{x}}^i - \hat{F}_{\mathbf{x}}^i)^2 \quad (3.3)$$

3.5 Summary

In this chapter, we discussed the defects and data related issues for our archive film restoration application. First, we briefly discussed the cause and characteristics of our target defect and other categories of defects. Then, the training and testing data sets used in our experiments were outlined along with sample frames from each film sequence. Finally, quantitative measurements were introduced for our experiments in defect detection and defect removal individually.

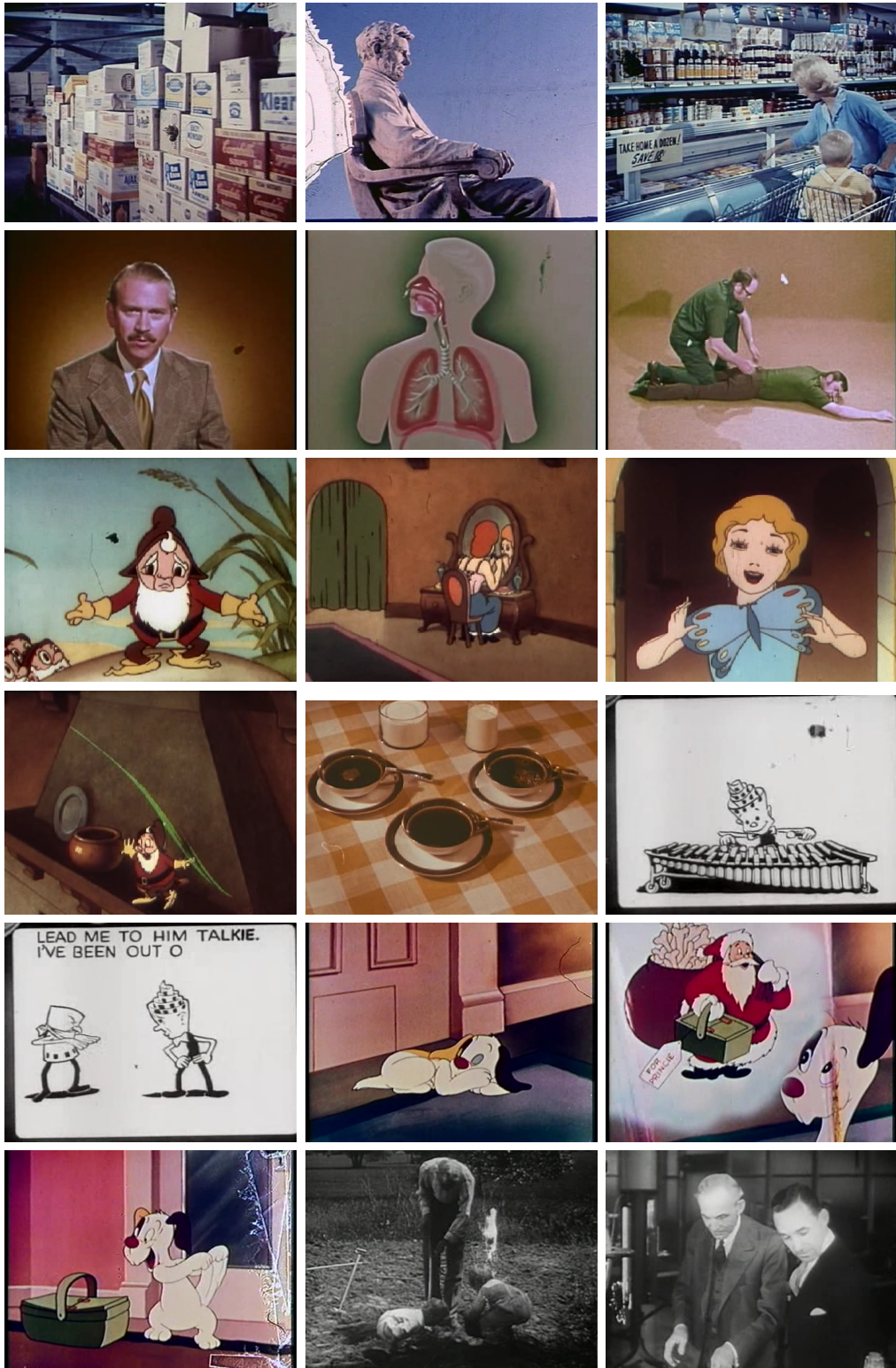


Figure 3.8: 1-18 of 50 real degraded frames from testing data (one from each sequence).



Figure 3.9: 19-36 of 50 real degraded frames from testing data (one from each sequence).

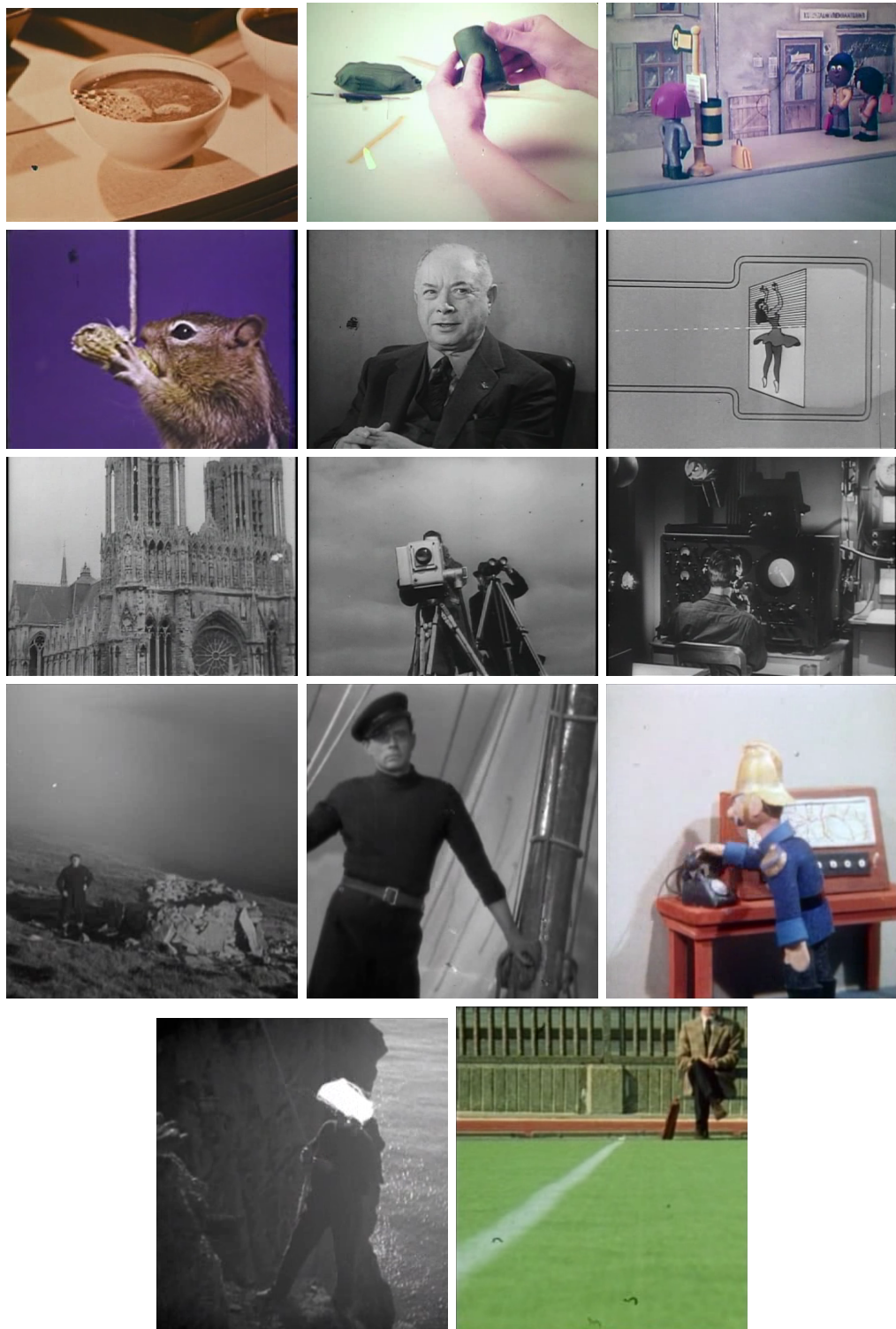


Figure 3.10: 37-50 of 50 real degraded frames from testing data (one from each sequence).

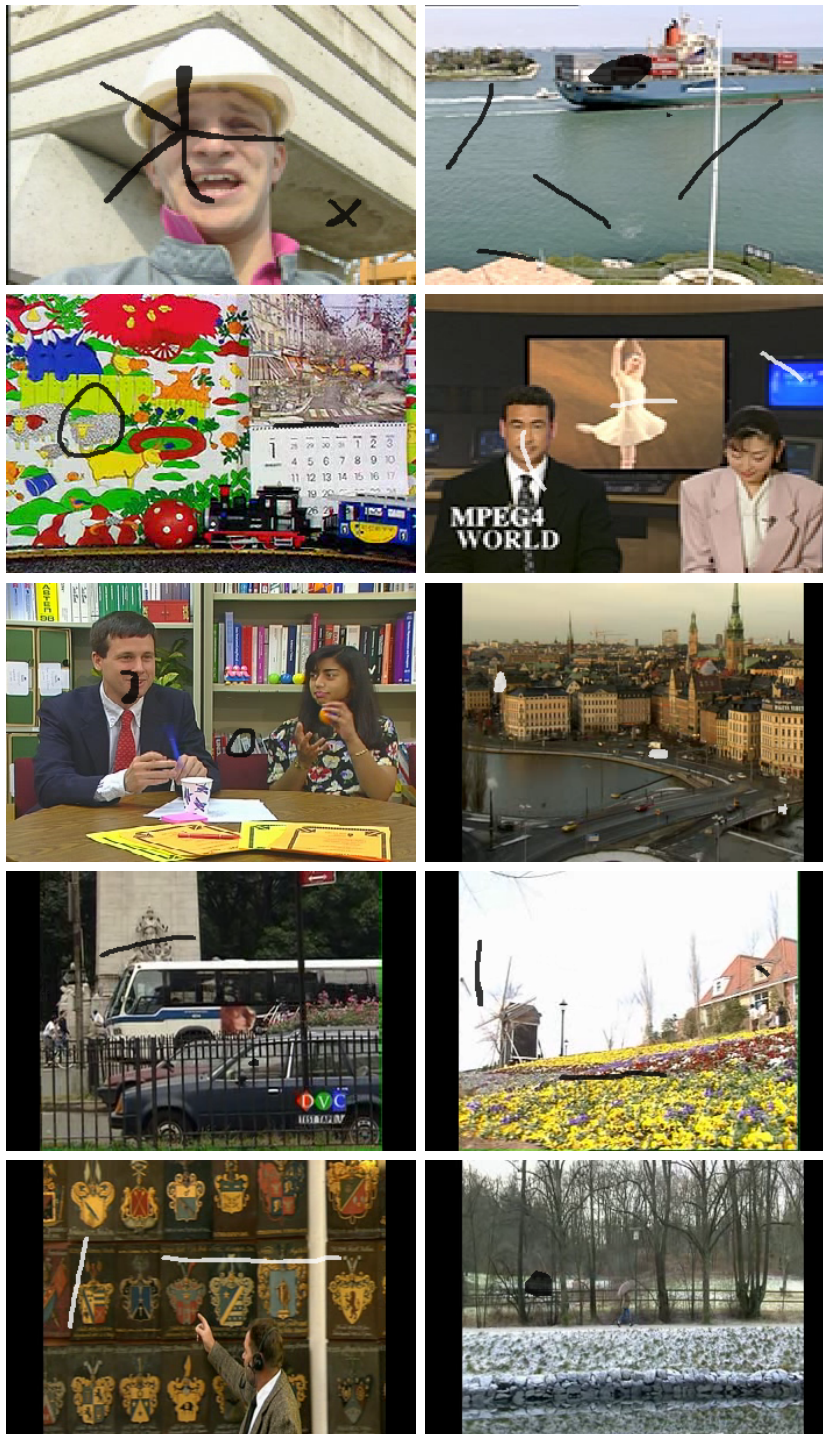


Figure 3.11: *Example frames from 10 artificially degraded testing sequences.*

HAFID: HMM based Archive Film Defect Detection

Defect detection is the first stage of our automatic archive film restoration framework. The main purpose of the detection procedure is to detect defects and recover their spatial and temporal addresses. This information can then be recorded in the form of defect maps to help restore the archive film in our proposed defect removal approach (presented in Chapter 6) or in any other restoration method by only looking at the labelled pixels.

For the proposed HAFID detector, it is of primary importance to detect the maximum number of true defects even though quite a few false alarms may also be produced. Traditional methods [15, 86, 99], nevertheless, are not able to detect all true defects and extra works were required to obtain improvements till a reasonable target is met. Although the over-detection followed by false positive elimination is preferred to under-detection and its consequences in our automatic restoration framework, we will demonstrate in Section 4.4 that the proposed method provides a higher correct detection rate and much less false alarms than current state-of-the-art techniques. These false alarms will be specially handled in our false alarm elimination approach presented in Chapter 5.

Defects have been assumed as temporal impulse events across frames. Based on this

assumption, previous archive film defect detection methods have only investigated the information from one or two previous and successive frames to determine if the current frame suffers from degradation, *e.g.* the SDIp [54] and MRF based [55] detectors. However, the approach here is to look at a larger number of frames before and after the current frame so that we are able to examine the statistical changes in pixel transitions for a longer range. This better helps model normal pixel sequences and later determine defects more accurately.

Due to the high complexity of motions in archive films and possible errors from motion estimation, a considerable number of false detections would rise by only examining the pixel intensities in direct motion compensated neighbourhoods. The majority of these false alarms, as reviewed in Section 2.3, are caused by complex motion events. Therefore, previous works [11, 22, 23, 50] have tried to distinguish these motion events from true artifacts by investigating the change of intensity values across three or five consecutive frames. Such motion events are recognised by their predefined patterns of intensity changes. However, their detection is limited by the fixed number of neighbouring frames, considering that it requires examining a longer range of frames to detect complex motion events, *e.g.* fast moving object and periodic motions. Hence, our strategy to investigate temporal pixel transitions in a longer range of frames before and after the current frame will also help recognise complex motion events and reduce the number of false detection.

Given the somewhat random nature of the occurrence of defects, rather than attempting to model pixel sequences with defects, we build a model of how normal pixels transit sequentially. This is in effect a form of novelty detection where any pixel sequences deviating from this normal model can be marked as a possible defect. A pixel \mathbf{x} would normally remain in one state and undergo a transition only when the intensity value at this location changes by a certain amount. For example, a constant state sequence may be linked to a temporal pixel sequence at a static background, whereas a single frame based defect would cause sudden state transitions at that pixel location. This temporal change of states (at a pixel location) can be considered as a Markov chain. Given some observation sequences, we are then able to model normal pixel sequences (*i.e.* state transitions) using an HMM.

In this chapter, we present our HMM based archive film defect detection method.

First, we briefly introduce the concept of novelty detection, followed by the fundamentals of HMM modelling in the context of our defect detection application. For the proposed defect detection approach, we characterize and train a single HMM for normal observation sequences and then apply the trained HMM within a framework to detect degraded pixels by examining each new observation sequence and its subformations being normal via a leave-one-out process. In order to demonstrate its improved detection performance, the proposed defect detection approach is compared against four commonly used and/or state-of-the-art techniques [52, 54, 67, 84] using real degraded archive film with handlabelled groundtruth (described in Chapter 3). For example, note although [54] is from 1992, it is still one of the most regularly used methods in industry.

4.1 Novelty Detection

As defined in Section 1.1, our target defects, dirt and blotches, are one group of defects that appear randomly as black or white sparkles or regions in one or more frames. The proposed HAFID detector aims to distinguish any such temporal intensity discontinuities against normal image pixel transitions, which is essentially a form of novelty detection. Hence, we first briefly introduce novelty detection as a data-oriented task and then discuss about the connection between our data and the proposed approach.

Novelty detection is a binary classification problem, which classifies new data into normal and abnormal ones, *e.g.* degraded pixels against normal ones in a defect detection task. Traditionally, solutions to binary classification involve learning profiles or models for both normal and abnormal data. However, due to the random nature of abnormal data and the shortage or sometime non-existence of samples, modelling abnormal data is difficult or sometime impossible. Hence, the framework of novelty detection is characterized by modelling only the normal data and then using the learned profile or models to detect abnormal data as outliers.

In [62, 63], Markou and Singh presented a comprehensive survey of novelty detection methods developed for a variety of applications, *e.g.* fault/defect detection,

mass detection in mammogram, radar target detection, intrusion detection in CCTV sequences, etc. From a view of methodology, novelty detection can be understood as a data-oriented mission. This means that the techniques used to model the data are driven by the characteristics of data. According to this, we categorize novelty detection methods into two groups, targeting at two types of data individually.

- ***Data with known distribution:*** the methods towards this class of data share the assumption that the probability distribution of data is known, *e.g.* normal distribution or Gaussian Mixture Model (GMM), etc. Such approaches mainly focus on computing the parameters of the probability density function, which usually requires knowledge and/or samples of the data. A simple example is to threshold the distance between new data and the mean of known data (following Gaussian distribution) so as to decide whether new data are normal. There are other more complex techniques, such as HMM and Hypothesis testing.
- ***Data with unknown distribution:*** the algorithms designed for data with an unknown distribution mostly rely on the sample data. The true underlying distribution of data usually remains undiscovered in real applications. Therefore, the sample data (the training data set) is of significance in estimating the probability density function of testing data. Most efforts have been made on selecting the optimal amount and formations of sample data and developing corresponding techniques to process the data. Previous solutions include histogram analysis, nearest neighbour based density estimation, neural network based approaches, etc.

We believe the temporal pixel sequence used in our archive film defect detection task should be categorized into the class of data with known distribution. We are modelling the sequential data based on the assumption that the observations (image intensities) in the sequence are controlled by the hidden states (description of scenarios in a higher level) while the state sequence is a Markov process. The probability distribution of observations for each state is determined during the training stage by learning from the training dataset provided.

We first investigate the statistics computed within each temporal pixel sequence in order to distinguish the degraded pixel sequences from normal ones. However, we found that it was difficult to decide if the central pixel of a temporal pixel sequence

is degraded by only looking at the statistics, *e.g.* mean or standard deviation of all pixel's observations against the central pixel's observation. Figure 4.1 illustrates the statistics computed for both degraded temporal pixel sequences and normal ones using two types of observations, *i.e.* image intensities and absolute temporal intensity differences. Totally 480×352 normal temporal pixel sequences and 539 degraded ones are used.

Therefore, we additionally examine the Markovian nature of our data and HMM modelling is chosen here to learn the pixel transitions across a number of neighbouring frames. HMM has been a useful tool in modelling sequential data, demonstrated in many applications, *e.g.* speech recognition [49] and handwriting recognition [46]. The traditional route for HMM based classifications is to train HMMs for each class, measure the probabilities of new data belonging to each class and categorize new data into the group with the highest probability. However, due to the random nature of defect we only train a single HMM for normal data in our defect detection approach. This trained HMM is then applied in the testing stage to compute the likelihood of a new observation sequence being normal.

Besides introducing the HAFID defect detector, the other aim of this chapter is to investigate the possibility of modelling temporal transition of pixel intensities using a single HMM. In order to testify the accuracy of our proposed HMM modelling, we test the trained HMM using 1 million normal observation sequences randomly extracted from 10 real image sequences (see Section 4.3).

4.2 HMM based Defect Detection

Figure 4.2 illustrates the outline of HAFID, which contains two stages, *i.e.* offline training of the HMM model and online testing of unseen data. In the training stage, the Baum-Welch method is adopted to estimate the parameter of our proposed HMM based on the training data extracted from real defect-free films. In the testing stage, the trained HMM is applied to compute the likelihood of a new observation sequence being normal. This involves creating subformations of the target observation sequence in each case with one observation missing, which we refer to

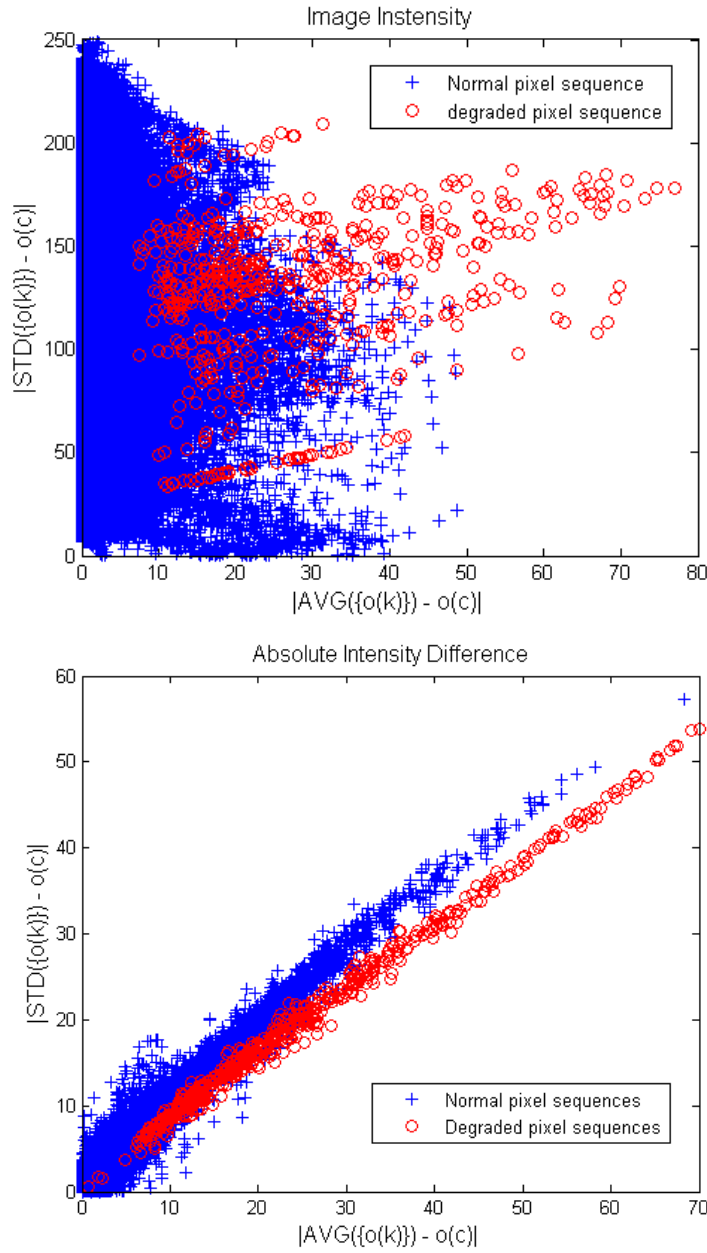


Figure 4.1: Statistics computed for both degraded temporal pixel sequences and normal ones are shown based on two types of observation, i.e. image intensity (top) and absolute intensity difference (bottom).

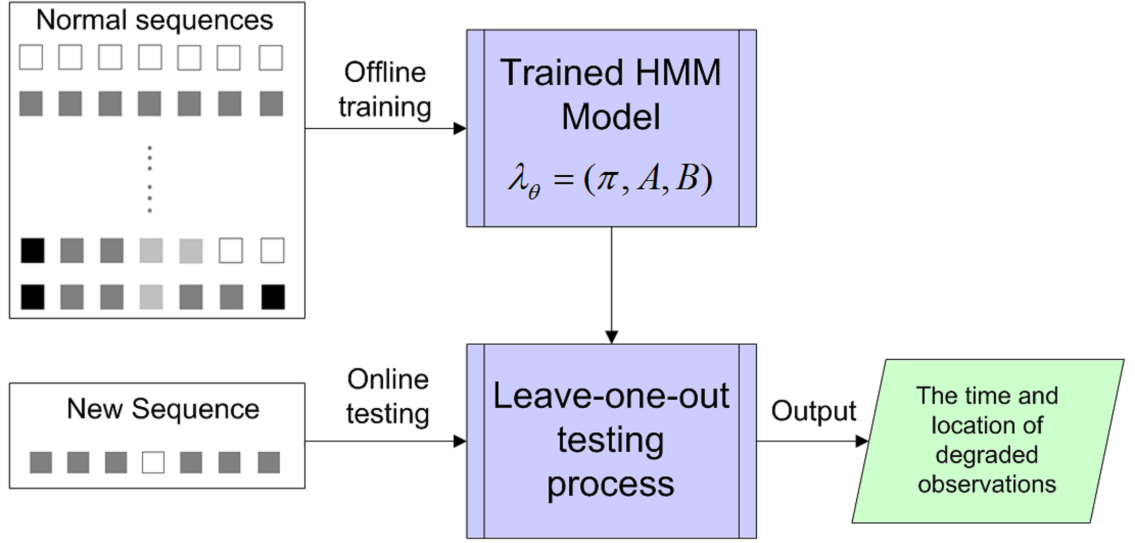


Figure 4.2: The diagram shows the system structure of our proposed defect detection framework.

here as a *leave-one-out* process. Then, the quality of the centre observation of the original observation sequence is examined based on how similar the likelihood of the observation sequence without the centre pixel being normal is relative to the mean of the likelihoods of all subformations.

The key problem in implementing this HMM based defect detection method is how to train an accurate HMM given the sample training data. In the following sections, we will discuss the principles of HMM modelling and the factors that are influential to the accuracy of our characterized HMM.

4.2.1 Fundamentals of HMM

Let $I_{\mathbf{x}}^t$ represent the image intensity value at pixel location \mathbf{x} at frame t . We extract intensity values at corresponding position \mathbf{x} across frames temporally as a time series, *i.e.* $\{I^t\}_{\mathbf{x}}$. We specify θ as the HMM trained for our proposed defect detector. For HMM θ , we define $\mathbf{O} = \{o_n, n = 1, \dots, N\}$ as the set of observations where N is equal to the size of the discrete observation space. Various image features could be chosen as observations o_n , such as intensity values, local variance and so on. In this application, we use temporal absolute intensity differences (where

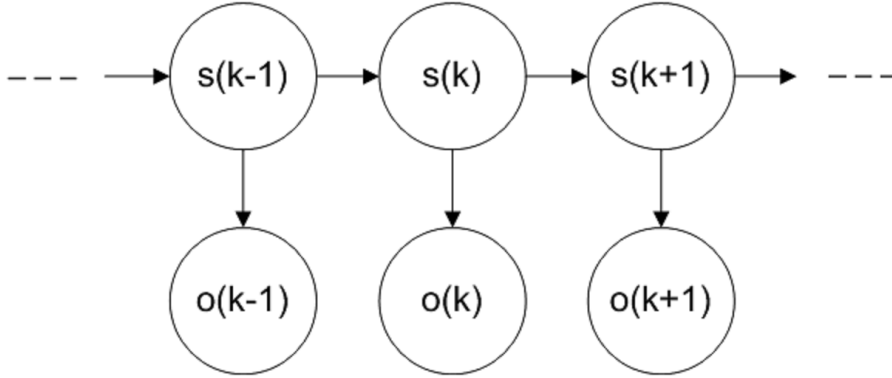


Figure 4.3: Graphical illustration of a general architecture of HMM modelling.

$N = 256$) for grayscale image sequences and temporal absolute Value differences in the HSV colour space for colour ones as observations. The reason why we adopt these observations will be stated in Subsection 4.2.2. Then, let $\{o(k)|o(k) \in \mathbf{O}, 1 \leq k \leq K\}_{\mathbf{x}}^t$ (abbreviated to $\{o(k)\}_{\mathbf{x}}^t$) be the sequence of observations centred at pixel \mathbf{x} in frame t for length K . The observation sequences are extracted from time series $\{I^t\}_{\mathbf{x}}$ within the range

$$\left[t - \left(\frac{K-1}{2}\right), t + \left(\frac{K-1}{2}\right)\right] \quad (4.1)$$

In a similar manner, let $\mathbf{S} = \{s_m, m = 1, \dots, Q\}$ be the set of states and $\{s(k)|s(k) \in \mathbf{S}, 1 \leq k \leq K\}_{\mathbf{x}}^t$ (abbreviated to $\{s(k)\}_{\mathbf{x}}^t$) be the state sequence. The topological structure of state transitions and the optimal value of Q will be discussed in Subsection 4.2.3. All observation sequences and state sequences share the same fixed length K at both training and detection stages.

A general architecture of HMM θ is illustrated in Figure 4.3. The observations are visible while they are controlled by the hidden states. The control is through HMM parameters, which are $\lambda_\theta = (\pi, A, B)$:

- **Start probability** $\pi = \{\pi(m), m = 1, \dots, Q\}$ which states the probability of $s(1)$ being s_m .

$$\pi(m) = P(s(1) = s_m), \quad 1 \leq i \leq Q; \quad (4.2)$$

For example, $\pi_i = (1, 0, 0)$ if the state sequence $\{s(k)\}_{\mathbf{x}}^t$ always starts with s_1

for a 3-state HMM.

- **Transition probability** $A = \{a_{lm}, l, m = 1, \dots, Q\}$ which states the probability of transition from current state s_l to next state s_m .

$$a_{lm} = P(s(t+1) = s_m | s(t) = s_l); \quad \sum_l a_{lm} = 1 \quad (4.3)$$

In our defect detection application, different states represent different intensity levels. Therefore, state transitions usually indicate large intensity changes. We will further discuss the state transition probability matrix of our proposed 5-state HMM in Subsection 4.2.3.

- **Emission probability** $B = \{b_m(o_n), m = 1, \dots, Q, n = 1, \dots, N\}$ which states the probability of the observation o_n given the current state s_m .

$$b_m(o_n) = P(o(t) = o_n | s(t) = s_m); \quad \sum_n b_m(o_n) = 1 \quad (4.4)$$

For example, the size of our emission probability matrix will be 5×256 if the trained HMM has 5 states and taking absolute difference of image intensities as observation.

When we model normal temporal pixel sequences using HMM, the three factors above need to be specified to characterize the HMM for our defect detection application, *i.e.* observations, topology and number of states, and HMM parameter estimation. The initialization and estimation of these parameters will be discussed in detail in Subsection 4.2.4.

4.2.2 Observations

As previously discussed, defect detection as a form of novelty detection is a data-oriented task. Therefore, choosing the observation is the first step in characterizing our HMM. Observations may vary in formation, such as discrete or continuous, single or multiple dimensions and so on. However, once the formation is decided, computed observations shall remain consistent across the entire data set. It means

that computed observations will always fall in the set of defined discrete observations or the range of continuous observation values.

As mentioned earlier, in HAFID detector we use temporal absolute intensity differences ($|I_{\mathbf{x}}^t - I_{\mathbf{x}}^{t-1}|$) as observations ($o_n \in [0, 255]$) for grayscale image sequences and temporal absolute Value (V channel) differences in the HSV colour space for colour ones. The reason why we use the temporal absolute intensity difference as observation is twofold. First, as discussed above a pixel \mathbf{x} would normally remain in one state and start a transition only when its intensity value changes by a certain amount, which will be highlighted by the temporal difference of intensities. Second, similar pixel sequences in terms of pixel transition can be mapped into a similar form of observation sequence even though they have different image intensities. For example, two pixel sequences extracted from two static background regions with different intensity levels shall share a similar observation sequence, having all its elements equal to 0 if we compute the intensity differences. Consistent observations will not only speed up the convergence of HMM parameters during training but also increase the applicability of the trained HMM on unknown data.

For colour image sequences, the Value channel represents the brightness or lightness of images in the HSV colour space, which functions in a similar manner as the grayscale intensity. Therefore, we scale values $[0, 1]$ from the V Channel to $[0, 255]$ before processing all colour films.

We also face the problem of determining the length of observation sequences. Adapting a short length of observation sequences may not be able to capture long-range pixel transitions in normal data while a longer length will introduce more complicated situations that leads to the failure of convergence in the training stage. In our defect detection application, different length of observation sequences were examined to optimize the accuracy of HMM modelling and defect detection. Values of K varying from 9 to 17 were examined for the optimal length of the observation sequences and $K = 13$ was found to give the most optimal results across our testing data set. A detailed comparison of different values of K is provided in Section 4.4.

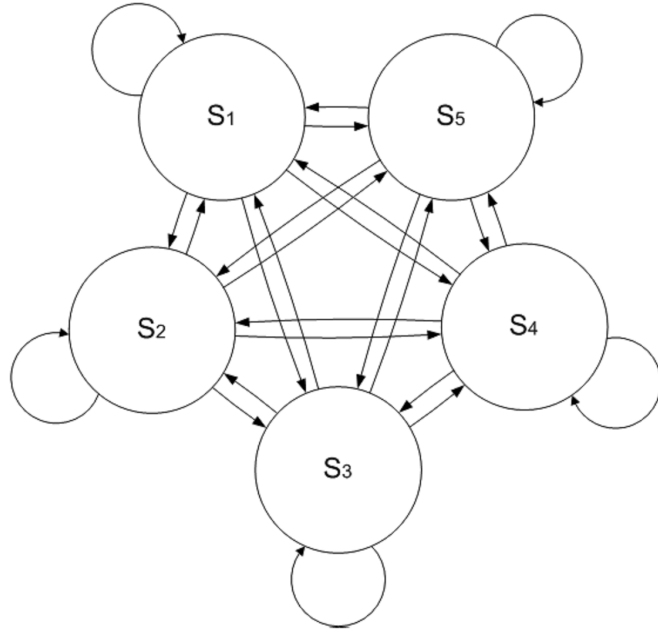


Figure 4.4: Graphical illustration of an Ergodic HMM.

4.2.3 Topological Structure of State Transitions

We begin with a brief introduction of topological structure of state transitions in HMM and three sample structures are illustrated, *i.e.* Ergodic model, Left-Right model and Circular model. Then, the topological structure of our proposed HMM is designed based on the Left-Right model and Circular model by considering both the temporal and circular characteristics of pixel sequences.

A general topological structure of HMM state transitions is shown in Figure 4.4, which is referred to as the Ergodic model [76]. In this fully connected HMM, each state can transit to all possible states including itself. Some transitions might be forbidden or unnecessary in specified applications, which leads to a number of topological variations:

- **Left-Right model** is also known as Bakis model [49]. This model was originally developed for speech recognition [76, 77] and later handwriting recognition [46]. The derivation from the Ergodic model was driven by the temporal characteristics of speech signal segments [76]. Figure 4.5 illustrates the visual topological structure of a common Left-Right model with 5 states. Each state

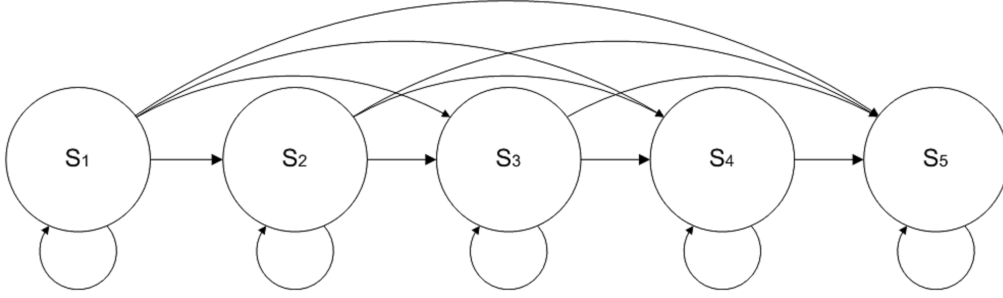


Figure 4.5: Graphical illustration of a Left-Right HMM.

sequence starts with s_1 and ends with s_K , going through all states from left to right in order. The initial state probabilities are defined as

$$\pi_m = \begin{cases} 1 & m = 1 \\ 0 & m = 2, \dots, Q \end{cases} \quad (4.5)$$

Since state transitions from right to left are forbidden in this case, all values in the bottom-left corner of the transition probability matrix are set permanently to 0:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ 0 & a_{22} & a_{23} & a_{24} & a_{25} \\ 0 & 0 & a_{33} & a_{34} & a_{35} \\ 0 & 0 & 0 & a_{44} & a_{45} \\ 0 & 0 & 0 & 0 & a_{55} \end{pmatrix}. \quad (4.6)$$

- **Circular model** is another important variation. Figure 4.6 visualizes a simple structure of Circular HMM with 5 states. Unlike the Left-Right model, the state sequence can start with any state but must end with the same state that it started with. States still transit in order, which is similar to the Left-Right model. The transition matrix for the circular structure is defined as

$$A = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} \\ a_{51} & 0 & 0 & 0 & a_{55} \end{pmatrix}. \quad (4.7)$$

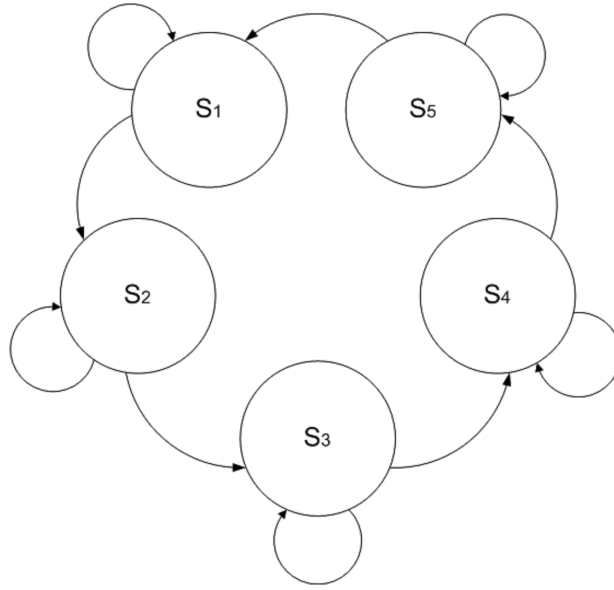


Figure 4.6: Graphical illustration of a circular HMM.

The circular topological structure was originally designed for planar shape recognition [2, 16]. In these methods, the circular structure provided the trained HMM with rotation invariance, which is important for regular shape recognition applications.

Proposed Topological Structure

There are two special features of image pixel sequences that need to be highlighted when we characterize our HMM. The temporal feature of image pixel sequences serves as the first one. Normal image pixel sequences usually contain pixels with gradual intensity transitions, which is similar to the temporal feature of speech signals. On the other hand, pixel sequences with a defect, for an example, do not have such feature. Hence, Left-Right model is ideal for modelling the temporal feature in our application.

The other feature of pixel sequences can be reviewed from a higher level understanding of temporal pixel sequences as elements of video sequences. Video sequences usually contain complex scenarios, *e.g.* multiple moving foregrounds with static or moving background, etc. Hence, pixel sequences from a single position can experi-

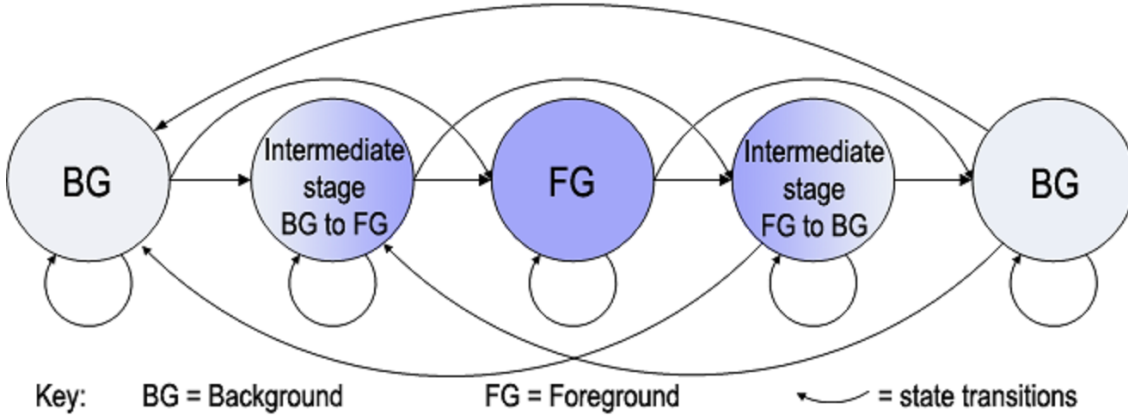


Figure 4.7: Graphical illustration of state transitions of a 5-state HMM in our application.

ence a series of state transitions, *e.g.* transition from background to foreground and then back to background. The iterative feature of state transitions is akin to the rotation invariance feature required in a regular shape recognition task, which can be modelled using a Circular HMM.

Therefore, by considering both features of image pixel sequence, the desired topological structure of our HMM can be a combination of the Left-Right and Circular models. Figure 4.7 graphically illustrates the topological structure of the proposed HMM with 5 states. The transition matrix of our proposed HMM model is defined as

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & 0 & 0 \\ 0 & a_{22} & a_{23} & a_{24} & 0 \\ 0 & 0 & a_{33} & a_{34} & a_{35} \\ a_{41} & 0 & 0 & a_{44} & a_{45} \\ a_{51} & a_{52} & 0 & 0 & a_{55} \end{pmatrix}. \quad (4.8)$$

Each state represents different level of intensities and at each time it is possible to jump by 0, 1, or 2 steps to the next state. Transitions with large intensity changes (jumping more than 2 steps) are forbidden in our proposed model, which usually imply either defects or complex motions. The initial state probabilities are defined

in the same manner as Left-Right model.

$$\pi_m = \begin{cases} 1 & m = 1 \\ 0 & m \neq 1 \end{cases} \quad (4.9)$$

A state sequence shall always start with s_1 but can end in any state. Unlike shape recognition applications, state sequences for pixel transitions in our defect detection application will not necessarily go through a complete loop. By such definition, the proposed HMM is able to model more complex pixel transition situations, *e.g.* periodic motions.

Number of States

Deciding the number of states is part of the topological design of HMM modelling. The number of states is usually determined according to designers' experience and knowledge of the training data while practical experiments could also be conducted to achieve the optimal number. Here, we present not only the logical representation of states but also experiment with different number of states in practice.

Different numbers of states of transitions ranging from 2 to 7 were experimented with and $Q = 5$ provided the best overall detection result. The value $Q = 5$ is logically more persuasive given what it represents (see Figure 4.7). The 5 states correspond to the scenes at a pixel position \mathbf{x} , *i.e.* background, intermediate stage between background and foreground, foreground, intermediate stage between foreground and background, and background again (possibly different to that initially at \mathbf{x}). Table 4.1 gives a sample observation sequence with $K = 13$ and its associated state sequence. In this example, the original intensity sequence starts with a static intensity level and then transits to another intensity level while the corresponding state sequence goes through a stable state (s_3), an intermediate state (s_4) and finally another stable state (s_5).

The representation of states can also be illustrated by looking at the transition

Table 4.1: *Sample observation sequence and its associated state sequence*

Original intensity sequence:

[46, 46, 46, 46, 47, 47, 76, 124, 125, 125, 125, 125, 125]

Observation sequence:

[0, 0, 0, 0, 1, 1, 29, 48, 1, 0, 0, 0, 0]

State sequence:

[s_1 , s_3 , s_3 , s_3 , s_3 , s_3 , s_4 , s_4 , s_5 , s_5 , s_5 , s_5 , s_5]

probability matrix of the trained 5-state HMM as follows:

$$A = \begin{pmatrix} 0.1314 & 0.4609 & 0.4077 & 0 & 0 \\ 0 & 0.8991 & 0.0001 & 0.1008 & 0 \\ 0 & 0 & 0.9629 & 0.0097 & 0.0274 \\ 0.1614 & 0 & 0 & 0.8385 & 0.0001 \\ 0.0137 & 0.0001 & 0 & 0 & 0.9862 \end{pmatrix} \quad (4.10)$$

where the transition probabilities are coherent with the state representations given above. States representing foreground and background pixels are intended to remain unchanged (with self-transition), *e.g.* states s_3 (with a probability of 0.9629) and s_5 (with a probability of 0.9862).

s_1 is a special case considering that the state sequences always start with s_1 . The initial state probabilities can also be defined as randomly or uniformly distributed. However, we prefer that the state sequences start with s_1 all the time with the assumption that pixel sequences always begin with certain intensity level and then either start transiting in an intermediate state or jump to a more stable state. As shown in Equation (4.10), a pixel observation at state s_1 has similar probabilities of transiting to either a intermediate state (s_2) or a more stable state (s_3). This preference of transitions from s_1 also well reflects that the components of training data include both static and dynamic background sequences.

If we uniformly initialized s_1 (*e.g.* $\pi_m = 0.2$, $m = 1, \dots, 5$ in the case of a 5-state HMM), the trained transition probabilities of states (shown in Equation (4.11)) would be averaged out (with all 5 states having similar probabilities for self-transition). In comparison, the transition matrix for our proposed HMM (shown in Equation

(4.10)) is more logically plausible and meaningful.

$$A = \begin{pmatrix} 0.5166 & 0.4812 & 0.0022 & 0 & 0 \\ 0 & 0.5081 & 0.2291 & 0.2628 & 0 \\ 0 & 0 & 0.6626 & 0.0545 & 0.2829 \\ 0.3063 & 0 & 0 & 0.5144 & 0.1793 \\ 0.2053 & 0.1132 & 0 & 0 & 0.6815 \end{pmatrix} \quad (4.11)$$

4.2.4 Training HMM for Normal Pixel Sequences

We train a single HMM θ , for normal image pixel sequences, which will then be applied in the testing stage to compute the likelihood of a new sequence being normal. The training procedure involves estimating HMM parameters $\lambda_\theta = (\pi, A, B)$ and selecting the training data.

HMM parameter estimation is an iterative procedure to refine the parameter $\lambda_\theta = (\pi, A, B)$ based on an initial estimation by maximizing

$$P(\{o(k)\}_{\mathbf{x}}^t | \lambda_\theta) = \sum_{m=1}^Q \alpha_K(m) \quad (4.12)$$

which represents the probability of an observation sequence $\{o(k)\}_{\mathbf{x}}^t$ being normal as defined by the HMM parameters λ_θ . $\alpha_K(m)$ is the forward variable in the Forward-Backward algorithm [76]:

$$\alpha_K(m) = P(\{o(g), g = 1, \dots, K\}_{\mathbf{x}}^t, s(K) = s_m | \lambda_\theta) \quad (4.13)$$

The value of $\alpha_K(m)$ can be computed in a recursive manner,

$$\alpha_{k+1}(m) = \left[\sum_{l=1}^Q \alpha_k(l) a_{lm} \right] b_m(o_{k+1}) \quad (4.14)$$

where $k = 1, \dots, K - 1$, $m = 1, \dots, Q$ and $\alpha_1(m) = \pi_m b_m(o_1)$.

Initial estimations of HMM parameters usually take previous experience and knowledge of the data into account considering that an accurate initial estimation would

speed up the convergence at the training stage. However, experiments [76] have proved that randomly or uniformly initialized parameters will not necessarily require longer time to achieve the convergence. Therefore, in our defect detection approach, the initial estimation of transition and emission probabilities is first set using a random distribution in the training stage.

The Baum-Welch method [5] is a common HMM parameter estimator, which is a special form of generalized EM (Expectation-Maximisation) methods. Other methods are also available, *e.g.* traditional gradient based optimization methods. However, it is the training data that plays a more important role in the HMM training procedure. The diversity and quantity of data are two basic requirements when we prepare the training data. The convergence of HMM parameters is most likely to be biased if all the data are from a single image sequence and/or only a small amount of data is used for training.

In our defect detection application, the training data set was extracted at random positions from 10 different types of archive film (5 grayscale and 5 colour image sequences), which is part of our archive film restoration dataset (as introduced in Section 3.3). In total, 207,561 normal observation sequences were used for training the HMM θ . The estimation of the parameters for our 5-state HMM is optimized by maximizing Equation (4.12) through an iterative procedure until convergence, using Baum-Welch algorithm [76].

4.2.5 Leave-one-out Test Process

The training stage results in an HMM which models the normality of transitions in temporal pixel sequences. Traditionally, this trained HMM would be only adopted to classify whether the testing sequence matches the trained model or not. However, the spatial and temporal addresses of pixels that result in the degradation of testing sequences are still unknown. Here, we design a leave-one-out process to look at both the testing observation sequence and its subformations to determine if a specified (central) image pixel in the testing observation sequence is degraded.

At the test stage, we first apply the trained HMM to compute the likelihood of an

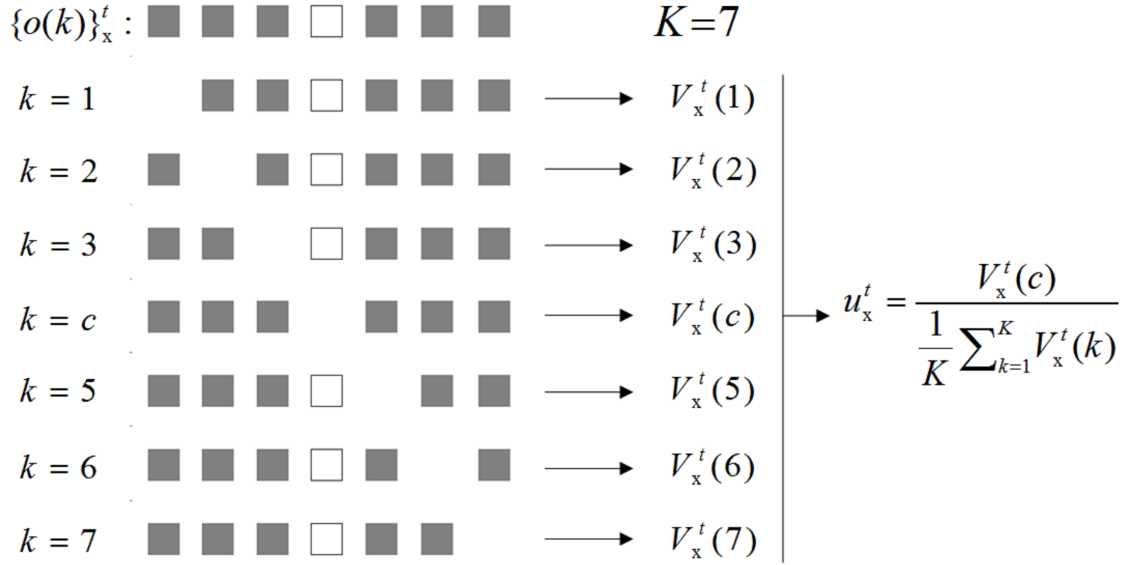


Figure 4.8: Leave-one-out processing of a sample observation sequence with $K = 7$.

observation sequence centred at position \mathbf{x} being normal, *i.e.* $P(\{o(k)\}_{\mathbf{x}}^t | \lambda_{\theta})$. We then measure how the likelihood of this observation sequence (arising from normal data) varies if each single observation within it was missing, one at a time in a leave-one-out fashion. This process enables us to observe how single observation affects the likelihood of the entire observation sequence being normal, considering that our aim is to mark the locations of degraded pixels to form a defect map. In this leave-one-out process, we will obtain a set of likelihoods for each observation sequence. The pixel at the centre of the observation sequence can then be marked as a defect, if the likelihood of the observation sequence without the centre pixel is larger, by a certain degree, than the average of all leave-one-out likelihoods computed on the observation sequence. In such manner, we are able to investigate every single pixel in the target pixel sequence and generate the corresponding defect map.

Formally, the leave-one-out process is implemented as follows and an example of how this process works will be presented in Subsection 4.2.6 for a detailed illustration. Figure 4.8 shows the leave-one-out processing of a sample observation sequence with $K = 7$.

As in the training stage, we extract a test observation sequence $\{o(k)\}_{\mathbf{x}}^t$ centred

at candidate pixel \mathbf{x} in frame t with observation $o(c)$ when centred on each image pixel location. For every element in $\{o(k)\}_{\mathbf{x}}^t$, we define a new observation sequence indexed by h that does not include the element $o(k)$ itself, *i.e.* $\{o(h), h \neq k\}_{\mathbf{x}}^t$. This will result in K observation sequences of length $K - 1$. We can then obtain, for each of these K observation sequences, the likelihood of the observation sequence $\{o(h), h \neq k\}_{\mathbf{x}}^t$ arising from normal data.

In traditional HMM based classifications, the likelihood of a testing observation sequence matching the trained HMM is computed in the same manner as the training stage using Equation (4.12). However, it is no longer feasible here to compute the likelihood using only Forward part of the Forward-Backward algorithm since observations in the original observation sequence are missing one at a time. Here, we adopt both parts of the Forward-Backward procedure to help compute the likelihood of each subformation of the original observation sequence being normal,

$$\begin{aligned} V_{\mathbf{x}}^t(k) &= P(\{o(h), h \neq k\}_{\mathbf{x}}^t | \lambda_{\theta}) \\ &= \sum_m P(\{o(h), h \neq k\}_{\mathbf{x}}^t, s(k-1) = s_m | \lambda_{\theta}) \\ &= \sum_m \alpha_{k-1}(m) \beta_{k-1}(m) \end{aligned} \quad (4.15)$$

where $k = 1, \dots, K$. $\alpha_{k-1}(m)$ and $\beta_{k-1}(m)$ are the forward and backward variables defined in the Forward-Backward algorithm [77], which individually represent the likelihoods of two parts of the subformation $\{o(h), h \neq k\}_{\mathbf{x}}^t$ (*i.e.* $\{o(g), g = 1, \dots, k-1\}_{\mathbf{x}}^t$ and $\{o(g), g = k+1, \dots, K\}_{\mathbf{x}}^t$) being normal given $s(k-1) = s_m$.

$$\begin{aligned} \alpha_{k-1}(m) &= P(\{o(g), g = 1, \dots, k-1\}_{\mathbf{x}}^t, s(y) = s_m | \lambda_{\theta}) \\ \beta_{k-1}(m) &= P(\{o(g), g = k+1, \dots, K\}_{\mathbf{x}}^t | s(y) = s_m, \lambda_{\theta}) \end{aligned} \quad (4.16)$$

where $\alpha_{k-1}(m)$ and $\beta_{k-1}(m)$ are specified when $k = 1$ and $k = K$, *i.e.* when the first or last observation is missing from the original observation sequence.

$$\begin{aligned} \alpha_0(m) &= P(\{o(g), g = 2\}_{\mathbf{x}}^t, s(2) = s_m | \lambda_{\theta}) \\ \beta_0(m) &= P(\{o(g), g = 3, \dots, K\}_{\mathbf{x}}^t | s(2) = s_m, \lambda_{\theta}) \\ \alpha_{K-1}(m) &= P(\{o(g), g = 1, \dots, K-2\}_{\mathbf{x}}^t, s(K-2) = s_m | \lambda_{\theta}) \\ \beta_{K-1}(m) &= P(\{o(g), g = K-1\}_{\mathbf{x}}^t | s(K-2) = s_m, \lambda_{\theta}) \end{aligned} \quad (4.17)$$

In a similar manner to $\alpha_{k-1}(m)$, $\beta_{k-1}(m)$ is computed through a backward recursion,

$$\beta_{k-1}(m) = \sum_{l=1}^Q a_{ml} b_l(o_k) \beta_k(l) \quad (4.18)$$

where $k = K, \dots, 2$, $m = 1, \dots, Q$ and $\beta_K(m) = 1$.

Although we have considered the diversity and quantity of data when we trained the HMM, exceptional situations like intensity level changes caused by motion can result in a high value of $V_{\mathbf{x}}^t(k)$. Enormous false alarms will be produced if we simply threshold the $V_{\mathbf{x}}^t(k)$ of pixel \mathbf{x} . Therefore, a mean of all $V_{\mathbf{x}}^t(k)$ values can be computed to average out the effect of such situations.

$$u_{\mathbf{x}}^t = \frac{V_{\mathbf{x}}^t(c)}{\frac{1}{K} \sum_{k=1}^K V_{\mathbf{x}}^t(k)} \quad (4.19)$$

$u_{\mathbf{x}}^t$ will reach a high value when \mathbf{x} is a truly degraded pixel position. For normal pixel positions, $u_{\mathbf{x}}^t$ values will be relatively smaller (see an example in the following Subsection 4.2.6).

After computing $u_{\mathbf{x}}^t$ for every pixel \mathbf{x} in frame t , we obtain the likelihood map $U^t = \{u_{\mathbf{x}}^t\}$ for all \mathbf{x} in frame t . Finally, any pixel \mathbf{x} in frame t is marked as a defect in the binary defect map $D^t = \{d_{\mathbf{x}}^t, d_{\mathbf{x}}^t \in \{0, 1\}\}$ if $u_{\mathbf{x}}^t > \tau_{\theta}$ where τ_{θ} is a threshold (see Section 4.4 for discussion of how this threshold is determined).

4.2.6 An Example Case

Here, we present a detailed example of how the likelihood $V_{\mathbf{x}}^t(k)$ is computed in our proposed leave-one-out process for a normal observation sequence that involves motions. This example also illustrates how false detections caused by complex motion events are prevented by comparing the likelihood of the subformation without the centre pixel with the mean of all likelihoods (*i.e.* Equation (4.19)).

First, we give a sample normal pixel sequence with a length of 13 and its intensities

as follows:

$$25, 25, 24, 48, 79, 106, 127, 152, 179, 179, 178, 178, 178 \quad (4.20)$$

This pixel sequence is extracted from a pixel position which experiences a transition from background to foreground. Its associated observation (absolute difference of intensities) sequence is:

$$0, 0, 1, 24, 31, 27, 21, 25, 27, 0, 1, 0, 0 \quad (4.21)$$

We first compute the likelihood $V_{\mathbf{x}}^t(1)$ of the subformation $\{o(g), g \neq 1\}_{\mathbf{x}}^t$ being normal (with the first observation 0 missing), which is a special case using $\alpha_0(m)$ and $\beta_0(m)$ defined in Equation (4.17),

$$\begin{aligned} V_{\mathbf{x}}^t(1) &= P(\{o(g), g \neq 1\}_{\mathbf{x}}^t | \lambda_{\theta}) \\ &= \sum_m P(o(2), s(2) = s_m | \lambda_{\theta}) P(\{o(g), g = 3, \dots, 13\} | s(2) = s_m, \lambda_{\theta}) \\ &= \sum_m \alpha_0(m) \beta_0(m) \\ &= \sum_m \pi_m b_m(o_2) \sum_{l=1}^Q a_{m,l} b_l(o_3) \beta_3(l) \end{aligned} \quad (4.22)$$

where $\beta_3(m)$ can be computed by a recursive process using Equation (4.18).

Next, let us look at the subformation $\{o(g), g \neq 7\}_{\mathbf{x}}^t$ with the centre pixel missing. The likelihood $V_{\mathbf{x}}^t(7)$ computed using Equation (4.16),

$$\begin{aligned} V_{\mathbf{x}}^t(7) &= P(\{o(g), g \neq 7\}_{\mathbf{x}}^t | \lambda_{\theta}) \\ &= \sum_m P(\{o(g), g = 1, \dots, 6\}, s(6) = s_m | \lambda_{\theta}) P(\{o(g), g = 8, \dots, 13\} | s(6) = s_m, \lambda_{\theta}) \\ &= \sum_m \alpha_6(m) \beta_6(m) \\ &= \sum_m \left[\sum_{l=1}^Q \alpha_5(l) a_{lm} \right] b_m(o_6) \sum_{l=1}^Q a_{m,l} b_l(o_8) \beta_8(l) \end{aligned} \quad (4.23)$$

where $\alpha_5(l)$ and $\beta_8(l)$ are computed in a recursive manner using Equation (4.14) and (4.18) respectively. In a similar manner, we can obtain the likelihood for every observation in the sequence,

$$\{4, 4, 6, 110, 130, 115, 97, 110, 115, 4, 15, 6, 6\} \times 10^{-5} \quad (4.24)$$

The mean of all likelihoods is equal to 58×10^{-5} . Although the likelihood $V_{\mathbf{x}}^t(c) = 97 \times 10^{-5}$ is quite large (in comparison to $V_{\mathbf{x}}^t(1) = 4 \times 10^{-5}$), this high value can be averaged out by comparing it with the mean of all likelihoods so that false detections caused by motions can be prevented.

4.2.7 Notes on Pre-processing

During the detailed implementation, we design a couple of processes before the main algorithm to increase the efficiency and accuracy of defect detection. The overall effect of such pre-processing is similar to BBC's hardware-based dirt and sparkle detector [93].

Before applying the leave-one-out process, we exclude those pixels with the absolute temporal forward and backward intensity differences below a certain threshold since only a large difference value implies a significant pixel transition or a defect. This will help dramatically increase the processing speed.

In addition, we also require the sign of forward and backward intensity differences to be the same. This allows us to leave pixels with opposing signs that imply mainly false alarms caused by continuous motions out of the main algorithm.

4.3 Testing the Trained HMM and Discussion

As stated at the very beginning of this chapter, the primary task of the proposed method is to detect the maximum number of true defects and false alarms are affordable expenses at this stage. Hence, we modelled the normal sequences using

Table 4.2: Accuracy of HMM modelling computed on 10 normal sequences

Sequence Name	# of sequence	Tested as normal	Accuracy (%)
Mobile Calender	100000	97067	97.1
Container		99971	99.9
Foreman		99202	99.2
News		99703	99.7
Paris		99677	99.7
Stockholm		97801	97.8
Flower		97717	97.7
Shields		97154	97.2
Park		96677	96.7
Bus		95595	95.6
Average	100000	98066	98.1

a single HMM, which is rather coarse but probably accurate enough for our defect detection task.

Here, we demonstrate the accuracy and limitation of HMM modelling on general normal sequences by testing the trained HMM with normal observation sequences randomly extracted from 10 unseen defect-free film sequences, namely *Mobile Calendar*, *Container*, *Foreman*, *News*, *Paris*, *Stockholm*, *Flower*, *Shields*, *Park* and *Bus*. These sequences are part of the archive film testing dataset (see Section 3.3), which will also be adopted in Chapter 6 to generate artificially degraded sequences.

We randomly select 100,000 normal observation sequences from each film sequence. Hence, 1 million sequences in total go through the leave-one-out testing process. The number of observation sequences tested without degraded pixels are counted as correct detections for each film sequence. Table 4.2 presents the computed statistics, giving the number of observation sequences tested, number of sequences correctly tested as normal, and the accuracy of HMM modelling.

As shown in Table 4.2, the trained HMM is capable of accurately modelling pixel sequences from all 10 films with correct detection rates of 95.6% to 99.9%. On average, only 1.9% of normal observation sequences were detected as false alarms. We believe it is reasonable to model temporal pixel transitions using a single HMM though the number of false detections increases provided the testing sequences contain complex

motion events, *e.g.* the *Bus* sequence (with the lowest accuracy) is composed of multiple objects and camera motions, *i.e.* camera tracking and zooming out on a fast moving bus with many occlusions and disocclusions of other vehicles.

These false alarms reflect the limitation of the proposed HMM modelling. Although the optimal length of observation sequences is computed through experiments, the decision is not made in the best consideration of modelling all possible motion events but the convergence of HMM parameters and the overall performance of defect detection. Some complex motion events will still not be recognized by the proposed HMM due to the length limitation while longer lengths of observation sequences will cause the reduction of correct detection ratio as we experimentally verified in Section 4.4. Although the number of such events is small as shown in Table 4.2. Therefore, we prefer to handle this small number of exceptions with specific elimination solutions according to their characteristics, which will be discussed in Chapter 5.

4.4 Results and Comparative Study

To demonstrate the accuracy of the proposed defect detection algorithm, we compare our method against state-of-the-art methods using both quantitative and qualitative results. The specificity and sensitivity (as defined in Section 3.4) of the proposed method HAFID is measured with reference to handlabelled groundtruth produced from the real degraded image sequences in our archive film restoration database. Image samples from real degraded sequences are also shown to present a subjective comparison amongst all the methods.

4.4.1 Quantitative Results

The ROC graph in Figure 4.9 shows a comparison of HAFID against those of four commonly used and/or state-of-the-art techniques: SDIp [54], Morris's MRF based defect detector [67] (referred to as Morris95), a Bayesian defect detection framework [52] (referred to as Kokaram04) and Ren and Vlachos's recent work [84] (referred to as RV07). These methods and the manner by which they are applied for

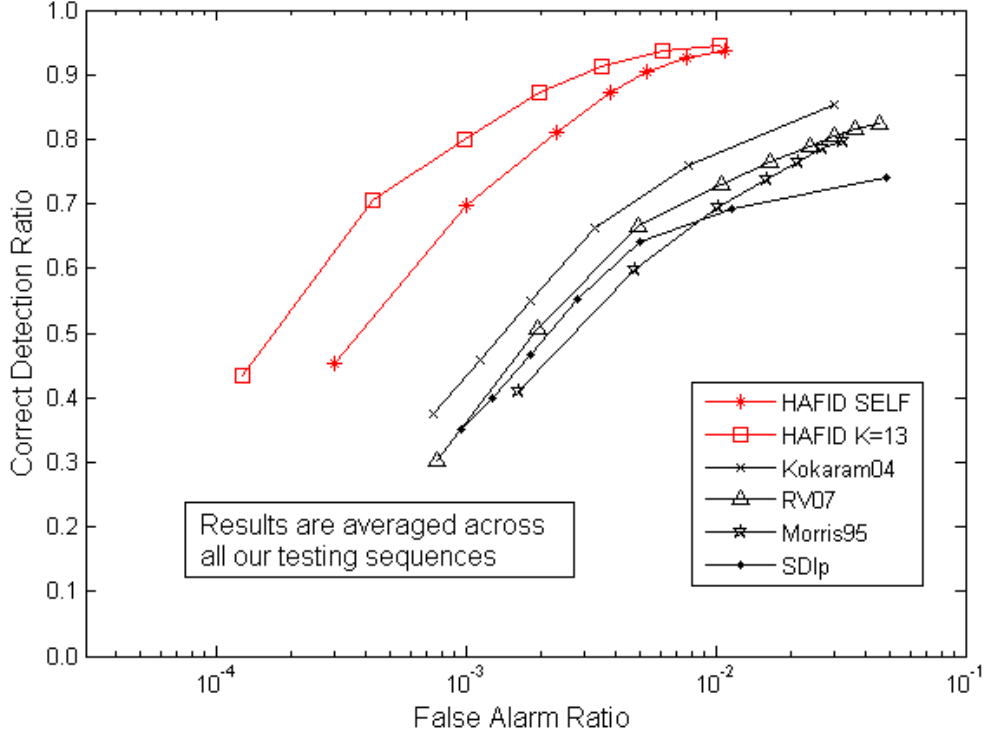


Figure 4.9: ROC graph shows a comparison of HAFID against four well known or current state-of-the-art techniques [52, 54, 67, 84], averaged across our entire handlabelled test data set.

fair comparison here are now outlined as follows:

SDIp: The Spike Detection Index (SDIp) [54] was a typical filter based defect detector, which improved Storey’s method [93]. A binary defect map was generated by examining if the motion compensated temporal forward and backward intensity differences were above a certain threshold. In addition, it also required the forward and backward intensity differences to have the same sign to further reduce false detections. This method was strongly sensitive to motion estimation errors. The threshold τ is the only variable that needs to be adjusted. In our experiments, τ is set to values in $\{2, 5, 8, 11, 14, 17, 20\}$ (abbreviated to $[2 : 3 : 20]$).

Morris95: The work of Morris [67] is a good example of model-based defect detector in which the author adopted an Ising model to represent the prior of the defect map. Gibbs sampling with annealing was then applied to compute the MAP configuration of the defect map given the image intensities from two adjacent motion

compensated frames. Sites in the final defect map were marked if discontinuities were shown on both adjacent frames. As described in Section 2.2, there are three parameters in this method, *i.e.* α , β_1 and β_2 . α controls the strength of temporal intensity discontinuity while β_1 and β_2 enforce the smoothness of degraded regions. These parameters are set to the values recommended in [67]. Only α varied in our experiments in the range $[0.125 : 0.125 : 0.75]$.

Kakaram04: Kokaram [52] developed a Bayesian framework to model noise and scratches while performing motion correction. Three binary variables were used for each pixel to mark if the pixel is degraded, forward occluded or backward occluded. These variables, together with restored image values and motion vectors, were defined as unknowns. Given the pixel values of degraded frames and initial motion estimations, the method applied the ICM [9] algorithm to solve these unknowns via an iterative two-step procedure. In our experiments, parameter \wedge^b and \wedge^c varied so as to control the local smoothness of binary defect maps and restore image intensities. \wedge^b was set to values in $\{2^n, n = [0 : 1 : 5]\}$ while \wedge^c was set to a constant, which is equal to 0.1.

RV07: Ren and Vlachos [83, 84] created a confidence function of intensity differences to measure the possibility of a pixel being degraded. The confidence function was computed by comparing candidate pixel's intensity against the intensity histogram for the entire frame. The candidate pixel would be categorized as degraded if its confidence value is below the threshold T_m . Image segmentation and local correlation information was also included to help eliminate false alarms. In our experiments, T_m varied in a range from $[180 : 10 : 250]$.

HAFID: For our HMM based detector HAFID, the likelihood $u_{\mathbf{x}}^t$ of pixel \mathbf{x} being degraded was thresholded by τ_θ , which varied in $[1 : 1 : 7]$.

These methods were all tuned for optimal performance. In the ROC graph in Figure 4.9, the correct detection ratio (sensitivity) is plotted against the false alarm ratio (1-specificity) computed from the average across 50 test sequences. These quantitative measures were described in detail in Section 3.4. The false alarm ratio was computed as a logarithm which is visually more sensible for the representation of small numbers (*i.e.* in a range of 10^{-5} to 10^{-1}). In general, the closer the curve is to the top-left

corner, the better the method performs in terms of both higher correct detection rates and lower false alarm rates.

As shown in Figure 4.9, HAFID comfortably outperforms all previous methods with both higher correct detection rates and much less false detections. Kokaram04 performed relatively better than the other three methods, considering that a motion correction process was integrated to prevent false detections caused by motions.

In Figure 4.9, we also examine the accuracy of our proposed method by applying the training and the testing on the same image sequence, *i.e.* *American Girl*. The image sequence is composed of 50 normal frames (for training) and another 250 degraded frames (for detection). As shown in Figure 4.9, HAFID SELF's accuracy is no better than HAFID, considering that training data extracted from the first 50 frames may not well represent the pixel transitions in the succeeding frames. It also suggests that training data extracted from a variety of image sequences provide a better modelling of normal pixel transitions on the temporal axis and superior detection accuracies on unseen data.

The results for HAFID with different sequence length K are shown in Figure 4.10. We experiment with difference K in a range from 9 to 17. As the length of the observation sequences increases, HAFID first achieves better accuracy and then falls back. The optimal results across our data set was reached when $K = 13$. As discussed in Subsection 4.2.1, it is because investigating a short length of observation sequences could cause false alarms while a longer length would introduce more complicated situations that lead to a under-detection (low correct detection rate).

4.4.2 Qualitative Results

Figure 4.11 and Table 4.3 show a comparative visual example of defect detection for a sample degraded frame from sequence *Seaman*. *Seaman* is composed of scenes with a slow wave-like global motion and small movements of the canvas in the background. SDIp detected 92.1% of the defects but also produced 717 false alarm pixels (out of a total frame size of 136704 pixels). Morris95 and RV07 were able to achieve correct detection rates of 93.8% and 93.3% but still resulted in 674 and 725

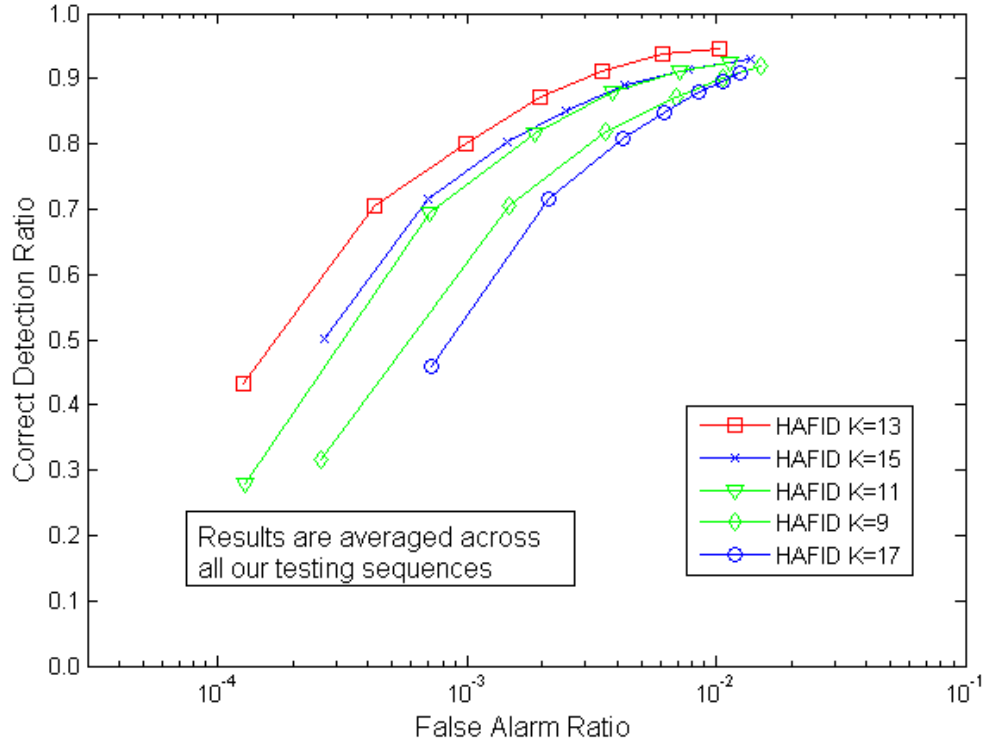


Figure 4.10: ROC graph shows a comparison of HAFID with different window sizes K , averaged across our entire handlabelled test data set.

false detections respectively. Those falsely detected pixels are mainly distributed around the seaman and the canvas, which indicates that they can be attributed to the motion of those objects. Kokaram04 produced far fewer false alarms (253) but only detected 86.7% of the true defects. The HAFID method was better at both detecting more true positives, *i.e.* 94.6%, and fewer false alarm pixels at 97 only. It shows our proposed HAFID detector is immune to some complex motion events (*i.e.* periodic movements in this case), which is a direct benefit derived from investigating a longer range of data at both training and testing stage.

In Figure 4.12 and Table 4.4, another visual example with a sample degraded frame from sequence *Cinderella* is shown. *Cinderella* had a static background and slow motions of the girl. SDIp, Morris95 and RV07 all detected 88.4% of the defects but also produced 2543, 889 and 5297 false alarm pixels respectively (out of a total frame size of 135424 pixels). These false alarms concentrated on the head and hands of the girl, which are the parts that exhibit motions. Kokaram04 and HAFID were

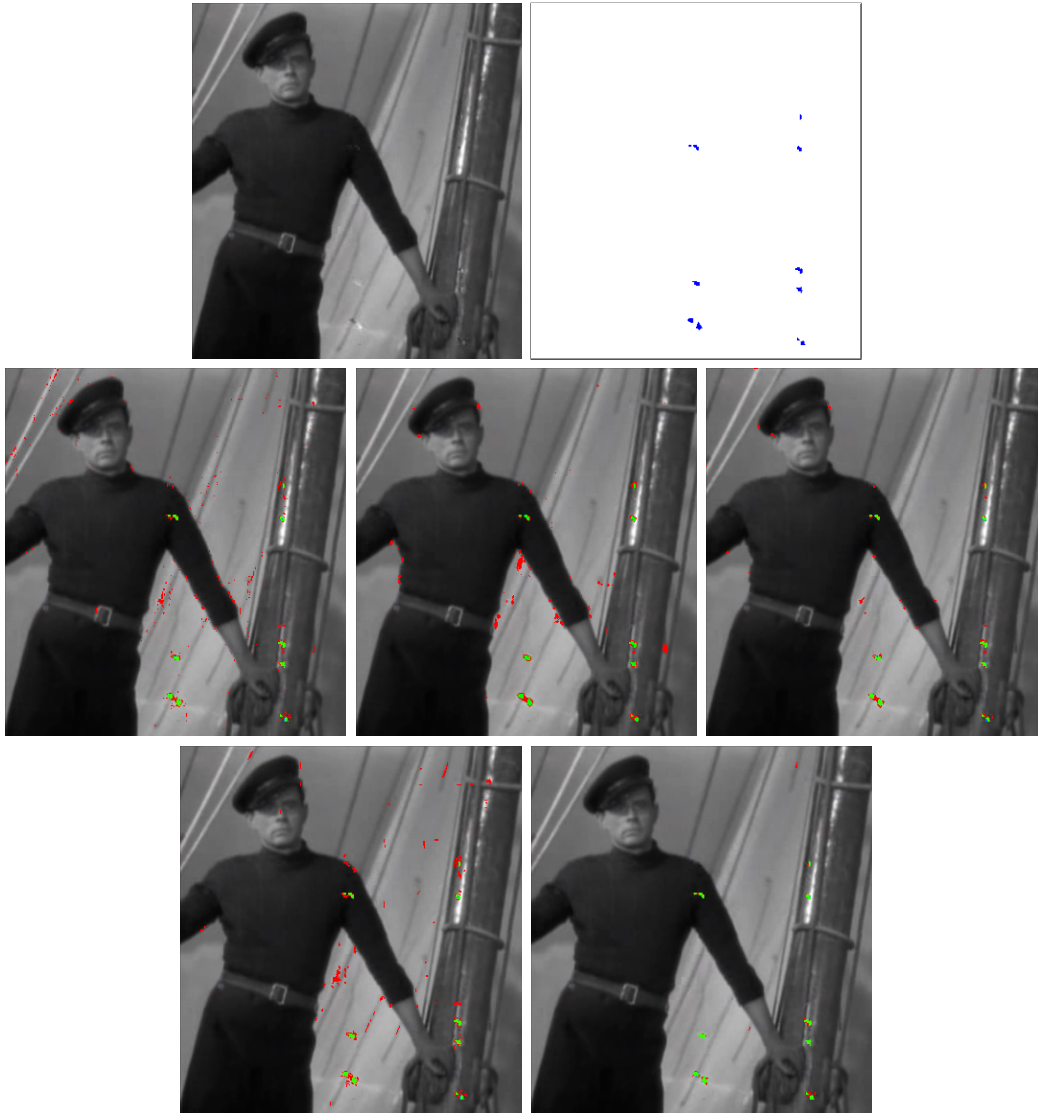


Figure 4.11: Seaman - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.

Table 4.3: Seaman - Statistical results of defect detection

Method	SDIp	Morris95	Kokaram04	RV07	HAFID
Detected true defects	201	204	189	203	206
Defects not detected	17	14	29	15	12
False detection	717	674	253	725	97
Correct detection(%)	92.1	93.8	86.7	93.3	94.6
False alarm(%)	0.5	0.5	0.2	0.5	0.01

able to decrease the number of false detections down to 401 and 381 respectively. However, compared to Kokaram04, the HAFID method achieved a much higher correct detection rate, *i.e.* a rise from 80.8% to 97.1%.

Figure 4.13 and Table 4.5 show a comparative example of preventing false detections from sequence *Science*. *Science* contains ego motion, *i.e.* the camera tilts down at an approximate speed of 5 pixels per frame. In this case, all methods achieved a high detection rate (above 90%). However, Morris95 and RV07 resulted in 3514 and 3290 false alarm pixels respectively (out of a total frame size of 54970 pixels). As SDIp includes motion compensation, it was able to produce fewer false detections at 1400 in this case. Kokaram04 produced far less false detections at 411 from the help of its integrated motion restoration algorithm. Finally, our HAFID method resulted in the least number of false alarm pixels at 281 only.

These results show that even though HAFID's emphasis is on modelling normal pixels at the expense of false alarms, our normal observation modelling is rather very accurate and the rate of false alarms is significantly lower than the current state-of-the-art techniques.

Computational Issues

Table 4.6 shows a comparison of the computational speed of all the methods in terms of average time per frame¹ in seconds. All implementations were coded and computed in MATLAB on a laptop with Intel Core Duo 2.4 GHz processors and 2GB RAM. The proposed algorithm is somewhat slower than SDIp, considering a longer temporal range is investigated to help determine defects, but it is a lot more accurate and also outperforms all the other methods.

It is worth to note that the HAFID defect detector only uses the temporal information so that each pixel site could be processed independently. Therefore, a promising acceleration to the current HAFID detector is to implement it using parallel computing techniques with multiple processors. It will result in a dramatically reduction

¹The average frame size in our data is 480×356 pixels

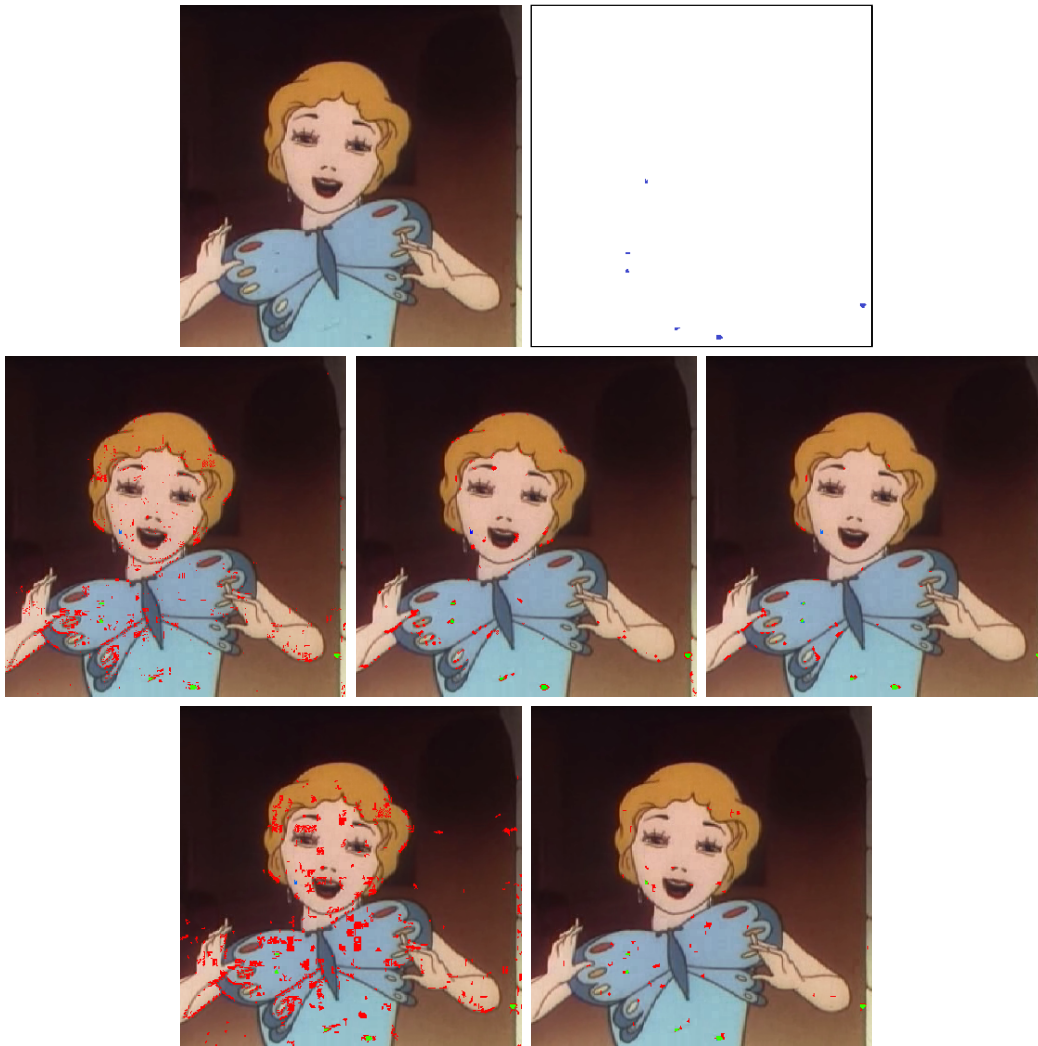


Figure 4.12: *Cinderella* - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.

Table 4.4: *Cinderella* - Statistical results of defect detection

Method	SDIp	Morris95	Kokaram04	RV07	HAFID
Detected true defects	92	92	84	92	101
Defects not detected	12	12	20	12	3
False detection	2543	889	401	5297	381
Correct detection(%)	88.4	88.4	80.8	88.4	97.1
False alarm(%)	1.9	0.6	0.3	3.9	0.3

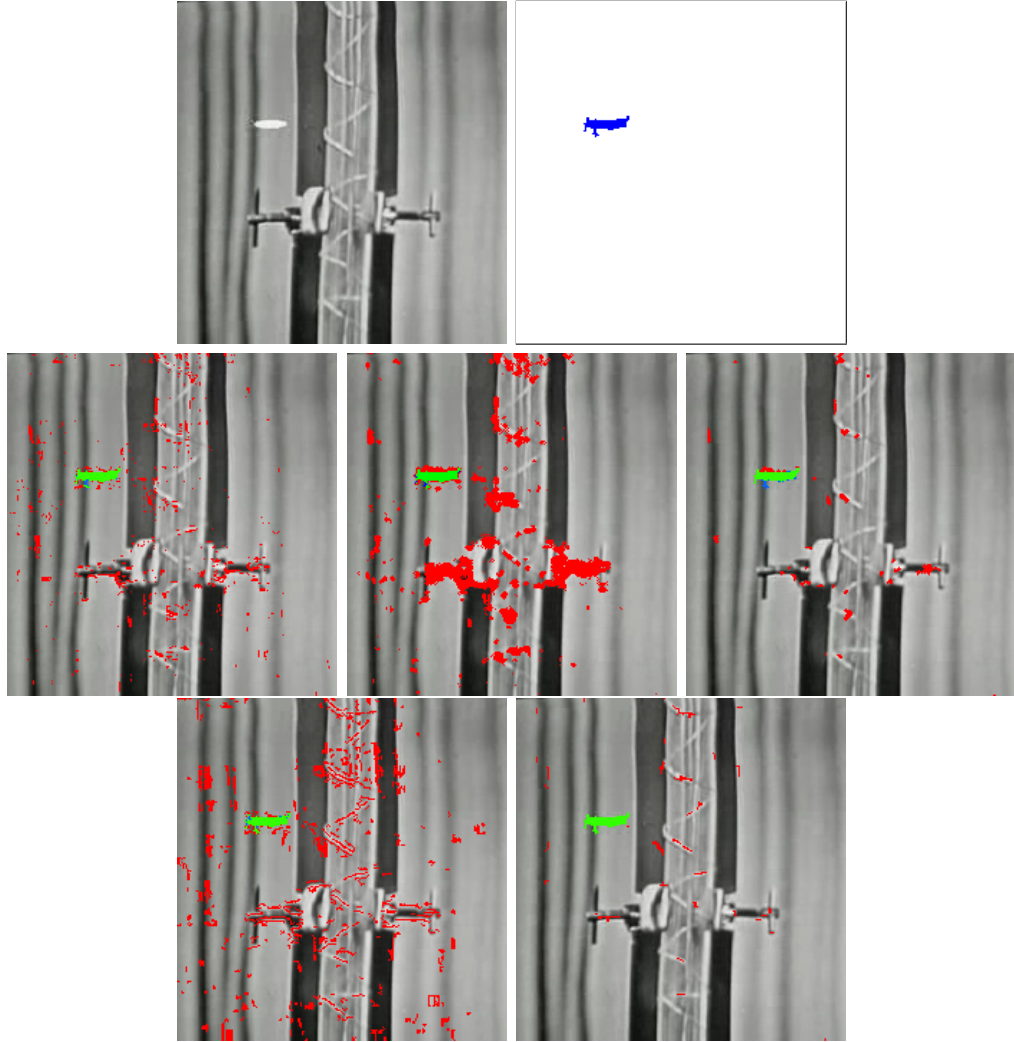


Figure 4.13: Science - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.

Table 4.5: Science - Statistical results of defect detection

Method	SDIp	Morris95	Kokaram04	RV07	HAFID
Detected true defects	190	194	185	191	205
Defects not detected	15	11	20	14	0
False detection	1400	3514	411	3290	281
Correct detection(%)	92.6	94.6	90.2	93.2	100
False alarm(%)	2.5	6.3	0.7	6.0	0.5

Table 4.6: Averaged computational speed for all methods

Method	SDIp	Morris95	RV07	Kokaram04	HAFID K=13
seconds	3	288	62	174	11

of computational cost.

4.5 Conclusions

In this chapter, we have presented a novel defect detection approach based on HMM modelling. We first discussed the fundamental factors which characterize HMMs, *i.e.* states, observations and HMM parameters. Then, an HMM was designed and trained for normal pixel sequences and used to compute the likelihood of a testing observation sequence and its subformations being normal in a leave-one-out process. The resulting defect maps presented the time and location information of defects. In addition, the accuracy of HMM modelling was tested using 1 million randomly selected normal observation sequences. The proposed HAFID was able to detect true defects with a high accuracy while keeping the false alarm rates low. It was achieved by investigating the pixel intensity transitions on the temporal axis in a relatively longer range. A few false detections were also produced due to complex motions that exist in archive film, which will be handled with false alarm elimination methods in the following Chapter.

The proposed HAFID detector fails to respond to a defective pixel if all pixels across a single observation sequence were defective pixels, although this rarely happens except for continuous vertical line scratches. However, overall, the proposed HAFID performs much more accurately than the current state-of-the-art and is considerably faster too (except for SDIp which is not so accurate).

The defect detection approach we proposed here is indeed a form of novelty detection. We believe it is straightforward to adapt it for other novelty detection applications in sequential data analysis. For instance, it can be applied to detect vertical line scratches by using spatial pixel sequences parallel to the horizon instead of temporal ones. For a more generic application, our method can be applied to detect tool

breakage in an end milling operation by observing the cutting force.

HAFID-STC: False Detection Elimination

In Chapter 4, we presented HAFID, a novel defect detection method based on HMM modelling. The resulting defect map from HAFID encapsulates the true defects very well, but suffers from a few false alarms. Large number of false positives will increase not only the computational cost of the follow-up restoration process but also the possibility of introducing new artifacts at those normal pixel locations.

In this chapter we try to refine our defect detection results based on the defect maps produced by HAFID. This is achieved by introducing a two-stage false alarm elimination process imposing spatial and temporal constraints respectively. Figure 5.1 illustrates the two-stage structure of our proposed elimination approach. First, the defect map from HAFID is modelled with a MRF to enforce spatial continuity constraints, and then the pyramidal Lucas-Kanade feature tracker [13] is applied to impose temporal correlation constraints. This extended method is referred to as HAFID-STC (HMM based Archive Film Defect detection with Spatial and Temporal Constraints). In a similar manner to HAFID, a comparison of HAFID-STC against four commonly used and/or state-of-the-art techniques [52, 54, 67, 84] is outlined to demonstrate the accuracy of our false alarm elimination method.

Previous archive film defect detection methods share a common dilemma - that is how to increase correct detection rates while reducing the number of false alarms. One popular strategy is to integrate the false alarm elimination scheme into the

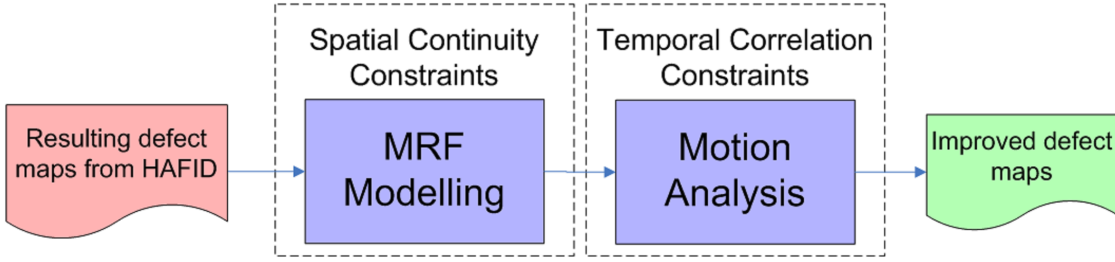


Figure 5.1: The two-stage structure of our proposed false alarm elimination algorithm.

defect detection algorithm by handling false alarms due to one particular cause, *e.g.* blurring in [79, 80], occlusions in [51], intermittent motions in [22], etc. Nevertheless, our strategy is to start with an over-detection to mark the maximum number of true defects and then handle the false detections specifically. These false alarms may belong to multiple categories with different causes but they can be handled together by looking at their shared characteristics. Here, we focus on two groups of false alarms with their own unique characteristics, which we observed in the detection results from HAFID. Accordingly, a two-stage false alarm elimination approach is developed to handle them respectively.

In Section 5.1, we first give a detailed introduction of false alarms by looking at both their causes and characteristics. Based on two distinctive characteristics, we develop the two-stage false alarm elimination process presented in Section 5.2. Finally, the elimination results and experiments are demonstrated in Section 5.3.

5.1 Category of False Alarms

In order to distinguish false alarms from true defects, previous works have focused on investigating the causes of false alarms and then developing corresponding solutions for elimination. In [79, 80], Rares *et al.* reported that the majority of false alarms are caused by irregular and complex motions in image sequences, which were referred to as *pathological motion*. In addition to pathological motion events, there are other causes, *e.g.* shot transitions, blurring, etc. It is rather difficult and sometime repetitive to develop an individual solution for each cause of false alarms. Therefore,

our strategy here is to investigate the characteristics of false alarms, considering false alarms caused in different situations can share a similar characteristics. In the following section, we will briefly discuss a few common causes of false alarms and demonstrate that all these false alarms share two distinctive characteristics.

5.1.1 Causes of False Alarms

As mentioned before, irregular and complex motion events are the major cause of false alarms. We further categorize them into inter-motion, intra-motion and other complex motions based on the assumption that motion scenes in image sequences can always be separated into foreground object(s) and the background. In addition, two other causes are also discussed, *i.e.* blurring and shot transition. Though they will not complete the list of all possible causes of false alarms, we demonstrate here that false alarms share some common characteristics regardless of the causes.

We illustrate the causes of false alarms along with image examples for each category, including two consecutive sample frames, the resulting defect maps from HAFID overlaid on the original frames and the elimination results from our proposed method. The detected true defects are marked in green, defects not detected in blue, and the false alarms in red.

Inter-motion: The inter-motion of foreground object(s) and background is a common cause of false alarms, *e.g.* the background is often occluded by the moving foreground objects. It can be seen as an interaction event amongst two or multiple objects. The reappearance of occluded objects is also categorized into this group. An occlusion or reappearance only causes the discontinuity of image intensities in one temporal direction, either forward or backward. Therefore, we can explore temporally in one of these directions for the relationship between false alarms and their temporal neighbours. In Figure 5.2, we show two consecutive sample frames from *Man on the Land* with a truck shifting into the scene from the left.

Intra-motion: In terms of image intensity changes, false alarms caused by the transformation of a nonrigid object generate similar discontinuities as true defects. Transformations of objects, such as zooming and shape transformation can cause



Figure 5.2: *Man on the Land (Inter-motion)* - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.

sudden appearance and disappearance of novel pixel intensities. However, these novel pixel intensities are usually connected with their spatial or temporal neighbours, such as sharing similar texture and colour. In Figure 5.3, we illustrate false detections caused by the transformation of a nonrigid object (flag in this case) on sample frames from *New York World Expo*.

Other complex motions: It is still a difficult task to estimate complex and irregular motions in film sequences, *e.g.* fast motions, periodic motions and rotations of

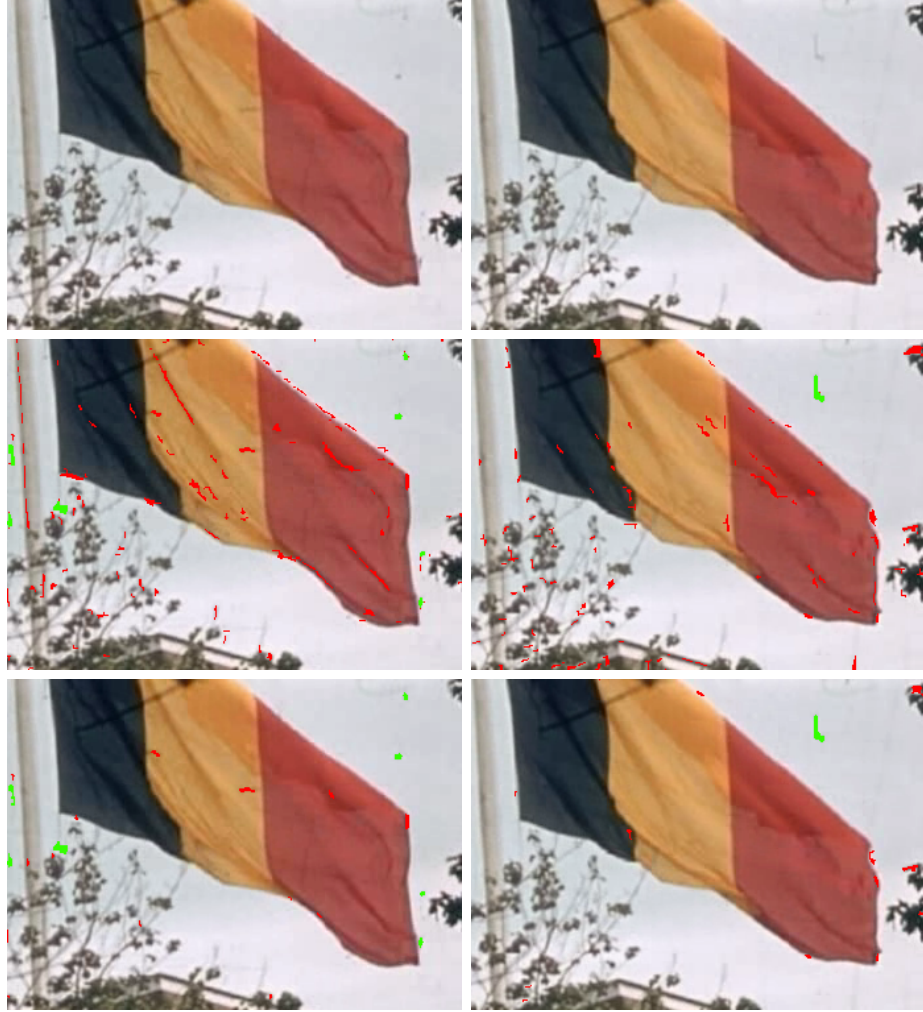


Figure 5.3: New York World Expo (Intra-motion) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.

objects. Erroneous motion estimation will further cause false detection of defects for those motion-dependent defect detectors, *e.g.* [54, 67, 68]. However, the continuity of intensities can still be tracked by looking at either spatial or temporal neighbouring pixel's intensities. Figure 5.4 gives an example of false detections caused by object rotation (as the camera inside the image was panning towards left) from *Story of Television*. It is worth noticing that false alarms in this case are mainly located around the camera with strong correlation with their spatial neighbours. This characteristic helps us eliminate these false alarms.

Shot transition: There are varieties of shot cut and transition means used in film



Figure 5.4: *Story of Television (Complex motion)* - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.

post production, such as fade-in, fade-out, dissolve, wipe, push and so on. These transition methods will produce similar false alarms as those caused by inter-motion and intra-motion of objects or a combination of both. For example, Figure 5.5 shows two sample frames before and after a shot cut. The resulting defect maps from our HAFID detector show a large number of false detections. Our false alarm elimination method is more likely to lose tracking of these falsely detected regions since their corresponding pixels on the temporal axis can only be found in one direction, either forward or backward.



Figure 5.5: *American Girl* (Shot transition) - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.

However, this group of false alarms could be easily avoided by introducing a pre-processing step into our automated restoration framework. A shot transition detection procedure, *e.g.* [74], could be performed before the defect detection so as to divide a long video sequence into a number of separate shots. Defect detection could then be conducted on each shot to avoid false alarms caused by shot transitions.

Blurring: As discussed in Section 3.2, two types of blurring exist in archive film. What we discuss here is the second kind which is added on purpose to increase the visual coherence, *i.e.* the smoothness of motions, to provide the human visual

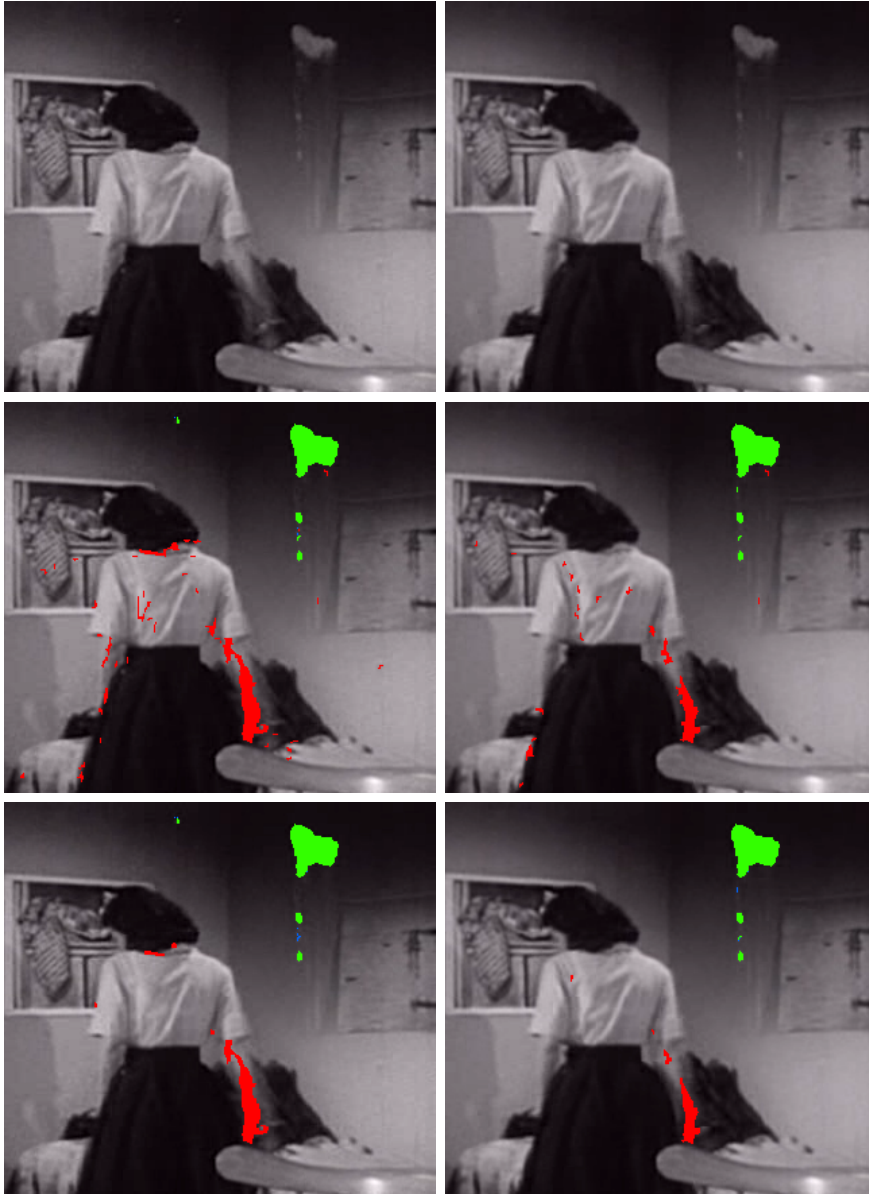


Figure 5.6: *Is This Love (Blurring)* - Top: original images (frame n and $n + 1$); Middle: HAFID results before false alarm elimination; Bottom: after false alarm elimination by our proposed method.

system with a more natural perception of object motions [107] in high frame rates. The blurring effect could also be introduced in the interpolated frames during the processing of frame rate conversion for the same consideration. Figure 5.6 presents an example with sample frames extracted from *Is This Love*. The girl's arm is blurred and falsely detected as defects. Since the pixel intensities caused by blurring are unique compared to their spatial and temporal neighbours, it is difficult to

separate them from the true defects. Therefore, we will handle these false alarms in the defect removal stage to maximally preserve the original intensities of pixels at these locations.

5.2 Two-stage False Alarm Elimination

The HMM model performs extremely well in locating true defects. However, it is rather sensitive to scene motion leading to false positives. As shown in Subsection 4.2.5, we have already integrated some anti-false alarm mechanisms into our proposed HAFID detector, *e.g.* computing the mean of all $V_{\mathbf{x}}^t(k)$ values (*i.e.* the likelihood of all observation sequences) to average out the effect of intensity level transitions caused by motion. However, if the length of such transitions compared to the entire state sequence is short, as true defects are, then false alarms arise in or around moving regions caused by object and/or ego motion. Therefore, it is necessary to introduce an extra false alarm elimination procedure after we perform an over-detection to maximize the detection of true defects. In order to identify and remove the false alarms, we extend the defect detector to apply a two-stage process enforcing (a) spatial continuity and (b) temporal correlation constraints based on the two characteristics of false alarms that we learned by investigating the causes of false alarms.

When investigating pixel transitions over time on a single location, we find that false detection usually appears as a discontinuity of image intensity on temporal axis, which is similar to true defects. However, by taking their spatial or temporally motion compensated neighbours into consideration, we can summarize two common characteristics of false alarms against true defects as follows.

For those false alarms that locate around the edges, *e.g.* the red pixels around the TV presenter's head shown in the second row of Figure 5.7. They are mainly caused by irregular motion or transformation of objects so that only partial pixels around the objects are falsely detected as defects. Those pixels are novel to their direct temporal neighbours while strong correlation could be found between those pixels and their spatial neighbours as defined by their features, *e.g.* intensities and gradients. In

such cases local smoothness can be exploited by modelling the defect map D^t and the likelihood map U^t with MRFs to encourage grouping defects into connected regions while removing false positives by propagating neighbouring non-degraded pixel locations (see Subsection 5.2.1).

For those false alarms that make up an entire moving region, *e.g.* the shadows in the curtain folds (shown in Figure 5.8) which leave the scene as the camera pans and zooms in towards the girl. These isolated false alarm regions are mostly caused by fast motion or interactions of objects. No relationship with other pixels may be found unless we trace forwards and backwards on the temporal axis. The pyramidal Lucas-Kanade feature tracker [13] is used here in order to impose temporal constraints (see Subsection 5.2.2).

5.2.1 MRF modelling

In [67], Morris applied an Ising model to represent the prior of what he referred to as his *detection frame* (effectively an initial defect map). Gibbs sampling with annealing was then applied to achieve the MAP configuration of the defect map given the image intensities from two adjacent motion compensated image frames. Positions in his final defect map were marked if discontinuities were shown on both adjacent frames. In this work, in contrast to Morris's initial zero-valued detection frame, D^t becomes our initial defect map, effectively providing advanced-stage prior information. False alarms are then eliminated iteratively by computing the MAP configuration of D^t given the likelihood map U^t . Thus, again unlike Morris [67], who investigated intensities at this point, we use the original likelihood map values in U^t to support the MAP estimation of D^t .

This process is implemented as follows. According to Bayes' theorem:

$$P(D^t|U^t) \propto P(U^t|D^t)P(D^t) \quad (5.1)$$

The joint probability distribution of U^t modelled with an MRF is equivalent to a Gibbs distribution. The function $\phi(\cdot) = (\cdot)^2$ is used to denote the potential for all possible connected 2-element cliques in an 8-connected neighbourhood. For each



Figure 5.7: Artificial Respiration (Spatial continuity) - From top to bottom: original images (frame n and $n + 1$); results from HAFID; results after false alarm elimination with MRF modelling only; results after the full algorithm.



Figure 5.8: Cinderella (Temporal correlation) - From top to bottom: original images (frame n and $n + 1$); results from HAFID; results after false alarm elimination with Motion Analysis only; results after the full algorithm..

defect position in map D^t , we compute the probability of a pixel having value $u_{\mathbf{x}}^t$ as a function of all its spatial neighbourhood ($N_{\mathbf{x}}^t$) pixel likelihoods and (previous frame) temporal neighbour $u_{\mathbf{x}}^{t-1}$. So the joint probability distribution of U^t given D^t is:

$$P(U^t|D^t) = \frac{1}{Z_{U^t}} \exp\left(-\sum_{\mathbf{x}} [\alpha' \sum_{\mathbf{x}' \in N_{\mathbf{x}}^t} \phi(u_{\mathbf{x}}^t - u_{\mathbf{x}'}^t) + \alpha(1 - d_{\mathbf{x}}^t) \phi(u_{\mathbf{x}}^t - u_{\mathbf{x}}^{t-1})]\right) \quad (5.2)$$

where Z_{U^t} is a normalizing constant and α', α are weights. $\phi(u_{\mathbf{x}}^t - u_{\mathbf{x}}^{t-1})$ measures the difference of likelihoods between corresponding pixels on the temporal axis. It will be switched on as a penalty for any pixel position that is not labelled as defect and the strength of this penalty is controlled by α . The prior of defect map D^t is:

$$P(D^t) = \frac{1}{Z_{D^t}} \exp\left(-\sum_{\mathbf{x}} [-\beta_1 \sum_{\mathbf{x}' \in N_{\mathbf{x}}^t} \delta(d_{\mathbf{x}}^t - d_{\mathbf{x}'}^t) + \beta_2 \delta(1 - d_{\mathbf{x}}^t)]\right) \quad (5.3)$$

where $\delta(\cdot)$ is the delta function, β_1, β_2 are weights and Z_{D^t} is also a normalizing constant. β_1 enforces the continuity of defect maps and β_2 is a penalty to any pixel position that is labelled as defect. The bigger β_2 is, the more likely labelled pixel positions are eliminated as false alarms. Finally, combining (5.2) and (5.3), the *a posteriori* likelihood function is:

$$P(D^t|U^t) \propto \frac{1}{Z^t} \exp\left(-\sum_{\mathbf{x}} [\alpha(1 - d_{\mathbf{x}}^t)(u_{\mathbf{x}}^t - u_{\mathbf{x}}^{t-1})^2 - \beta_1 \sum_{\mathbf{x}' \in N_{\mathbf{x}}^t} \delta(d_{\mathbf{x}}^t - d_{\mathbf{x}'}^t) + \beta_2 \delta(1 - d_{\mathbf{x}}^t)]\right) \quad (5.4)$$

where $Z^t = Z_{U^t} Z_{D^t}$. Gibbs sampling with annealing [36] is applied to compute the MAP configuration for eliminating false alarms. Weights (α and β_2) work together to balance the number of labelled pixels to be eliminated as false alarms. We set $\beta_1 = 1$, as suggested in [85], and apply a similar parameter estimation procedure as [67] for (α, β_2) . The images at the second row (from bottom) of Figure 5.7 show the TV presenter scene after false alarm elimination using this approach and the images at the bottom row show the final results after applying the two-stage method.

5.2.2 Motion Analysis

Some false alarms make up an entire (but small) moving region. The MRF modelling introduced above requires strong spatial correlation upon local neighbourhood to propagate surrounding non-degraded pixels into the false alarms regions. Nonetheless, for those spatially isolated false alarm regions, we look for temporal correlation constraints instead.

We apply pyramidal Lucas-Kanade motion tracking [13] on pixels corresponding to map D^t positions across K frames centred at the current frame to determine the stability of the pixels being tracked. If the tracking fails (in more than $\frac{K-1}{2}$ frames), then the position in D^t is left unchanged, otherwise it is eliminated as a false positive. The images at the second row (from bottom) of Figure 5.8 show examples of a scene after false alarm elimination using this approach. The proposed method is able to remove most false alarm regions on the curtain.

The target of this approach is comprised of those false alarms caused by fast motions, occlusions and many other complex motion events. In many of these situations, falsely detected pixels can only be tracked in one temporal direction, either forwards or backwards. Therefore, we will eliminate a false alarm pixel only if it is trackable in at least half ($\frac{K-1}{2}$) of the $K-1$ neighbouring frames. We also experimented with other values of this threshold in a range from $K-1$ to 1. A high elimination rate also comes with a fall in the overall correct detection rate of true defects. $\frac{K-1}{2}$ is able to provide the highest elimination rate while preserving the highest correct detection rate. We also examined different values of K and $K = 13$ provided the highest elimination accuracy across our testing data set (described in detail in Section 5.3). This value also matches the length of observation sequences on which we examined the pixel intensity transitions in HAFID.

5.3 Results and Comparative Study

In order to illustrate the improvement of HAFID-STC over HAFID, we now show the HAFID-STC results on the same graph as in Figure 4.9. The ROC graph in

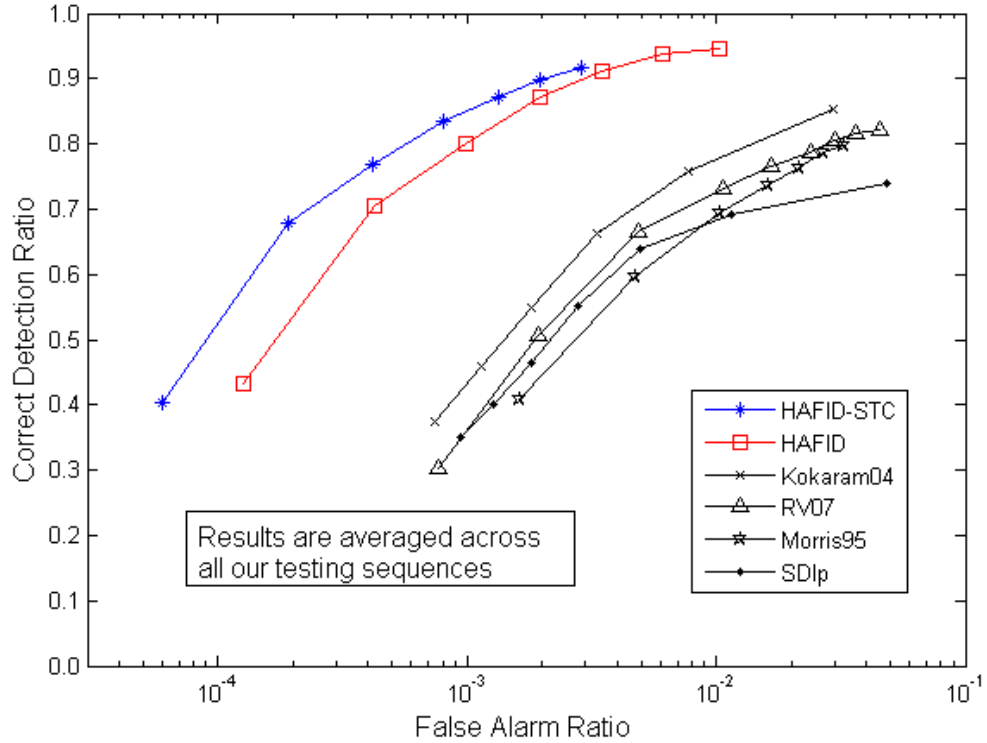


Figure 5.9: ROC graph shows a comparison of HAFID-STC against four well known or current state-of-the-art techniques [52, 54, 67, 84], averaged across our entire test data set.

Figure 5.9 illustrates a comparison of HAFID-STC against HAFID, Kokaram04, RV07, Morris95, SDIp as defined previously in Section 4.4.

It is worth noting that both Kokaram04 and RV07 have integrated false alarm elimination techniques into their defect detectors. In Kokaram04, two binary variables are used to indicate if the current pixel is forward and backward occluded. These variables, together with defect map, restored image intensities and motion vectors achieve the optimal values through the ICM algorithm in an iterative two-stage procedure. Hence, Kokaram04 is able to handle false alarms caused by occlusion. In RV07, false alarm elimination is implemented as an incremental to the main detection algorithm, whose system structure is similar to the one of our HAFID-STC detector. Region growing based segmentation and local correlation information (*i.e.* mean and standard deviation of intensities) are adopted to eliminate those pixels that impair local smoothness. Both Kokaram04 and RV07 target only one

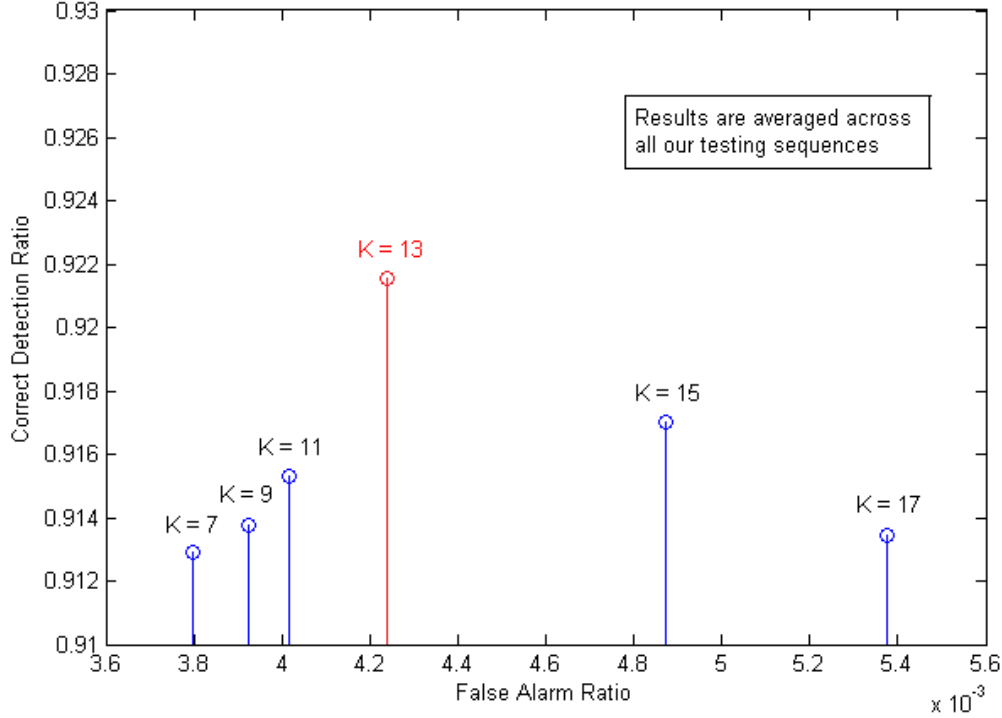


Figure 5.10: The graph shows a comparison of HAFID-STC with different window sizes K for motion tracking, averaged across our entire handlabelled test data set.

category of false alarms while our proposed false alarm elimination algorithm, nevertheless, handles a larger collection of false alarms and also achieves a higher accuracy.

For HAFID-STC, the threshold τ_θ was varied (as with HAFID) while we found the optimum α and β_2 for each τ_θ . α controls the strength of temporal continuity constraint while β_2 enforces the elimination of (possibly both correct and false) detections. Two parameters work together to eliminate false detections with strong local correlations. As shown in Figure 5.9, HAFID comfortably outperforms all previous methods while HAFID-STC achieves the best results overall (with lower false alarm ratios and higher correct detection ratios).

The results for HAFID-STC with different window sizes K for motion tracking are shown in Figure 5.10. The false alarm ratios increase as K varies from 7 to 17. However, $K = 13$ was found to give the highest correct detection ratio across our testing data set while eliminating over 70% of the false alarms.

5.3.1 Image Example from Real Sequences

Figure 5.11 shows a comparative visual example for a sample degraded frame from *Henry Ford*. In Table 5.1, we also reproduce the quantitative results for HAFID and other methods from previous chapter and additionally show results from HAFID-STC for a better comparison of the defect map before and after false alarm elimination. The *Henry Ford* sequence is composed of a stable indoor background and some object motions in the foreground. The corresponding statistics of detection results for all methods are listed in Table 5.1. As shown in the table, SDIp and RV07 achieved correct detection rates of 87.3% and 82.1% but still resulted in 3236 and 5107 false detections respectively (out of a total frame size of 118800 pixels). Morris95 was able to detect 96.8% of the defects but also produced 3868 false alarm pixels. Kokaram04 produced far fewer false alarms (1094) but only detected 72.6% of the true defects. The HAFID method was better at both detecting more true positives, *i.e.* 100%, and fewer false alarm pixels at 536 only, while the full proposed method HAFID-STC improved the result further by reducing the false alarms down to 13 pixels. HAFID-STC also falsely eliminates 7 true defects, which causes a drop of correct detection ratio from 100% to 92.6%.

In Figure 5.12 and Table 5.2, another comparative example on a sample degraded frame from *American* is shown. The camera is slowly tracking in circle with the statue as the centre. As illustrated in Table 5.2, both SDIp and Morris95 detect over 82% of the defects while also producing large numbers of false detections, *i.e.* 9397 and 4356 respectively. RV07 was capable of reducing the false alarms down to 3008 but its correct detection ratio also decreased to 73.9%. Kokaram04 was able to detect 85.8% of the defects while keeping the number of false alarm below 4000. The HAFID method was again better at both detecting more true defects, *i.e.* 96.2%, and fewer false alarm pixels at 1917. Our proposed HAFID-STC improved the result further by eliminating over 90% of all false alarms but also cause a decrease of the correct detection ratio down to 87.7% in this case.

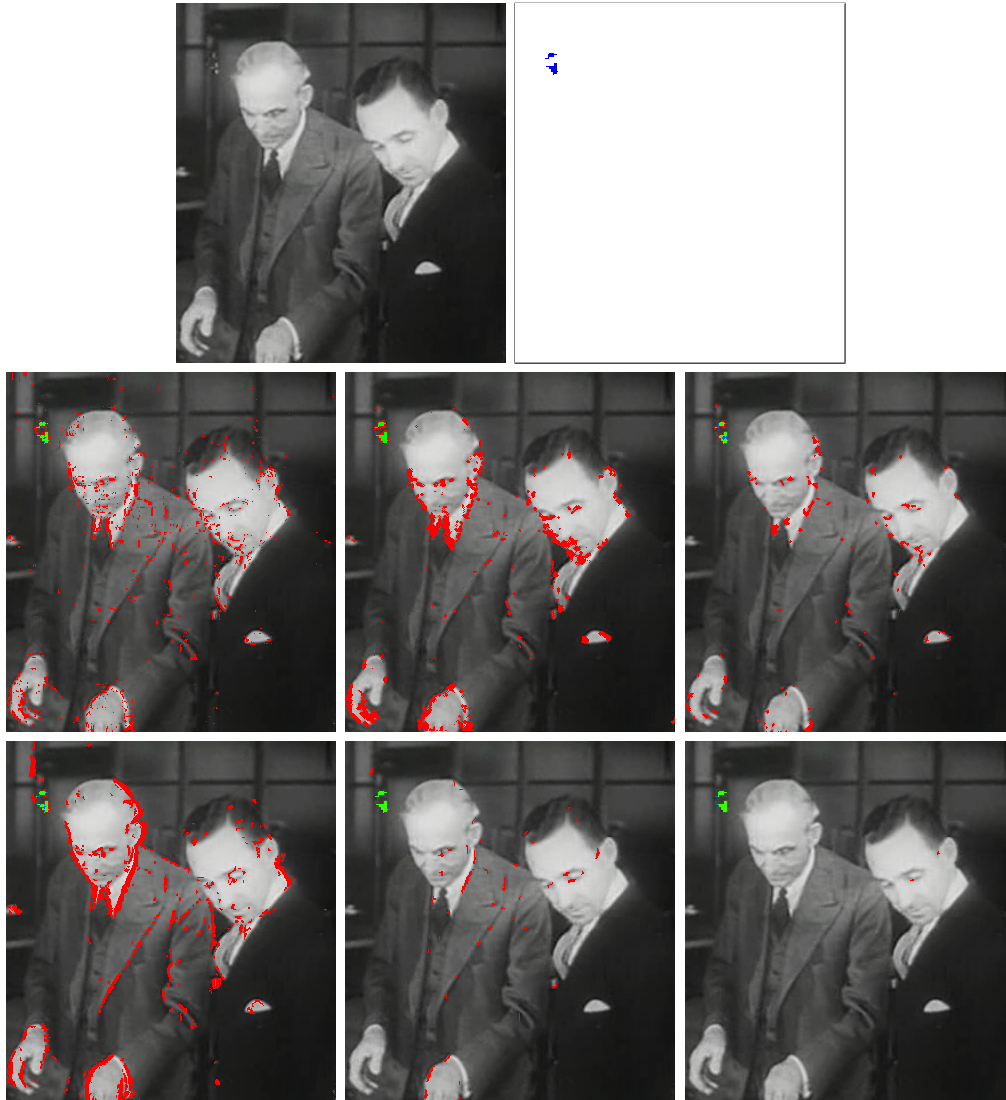


Figure 5.11: Henry Ford - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID and HAFID-STC. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.

Table 5.1: Henry Ford - Statistical results of false alarm elimination

Method	SDIp	Morris95	Kokaram04	RV07	HAFID	HAFID-STC
Detected true defects	83	92	69	78	95	88
Defects not detected	12	3	26	17	0	7
False detection	3236	3868	1094	5107	536	13
Correct detection(%)	87.3	96.8	72.6	82.1	100	92.6
False alarm(%)	2.7	3.2	0.9	4.3	0.5	0.01

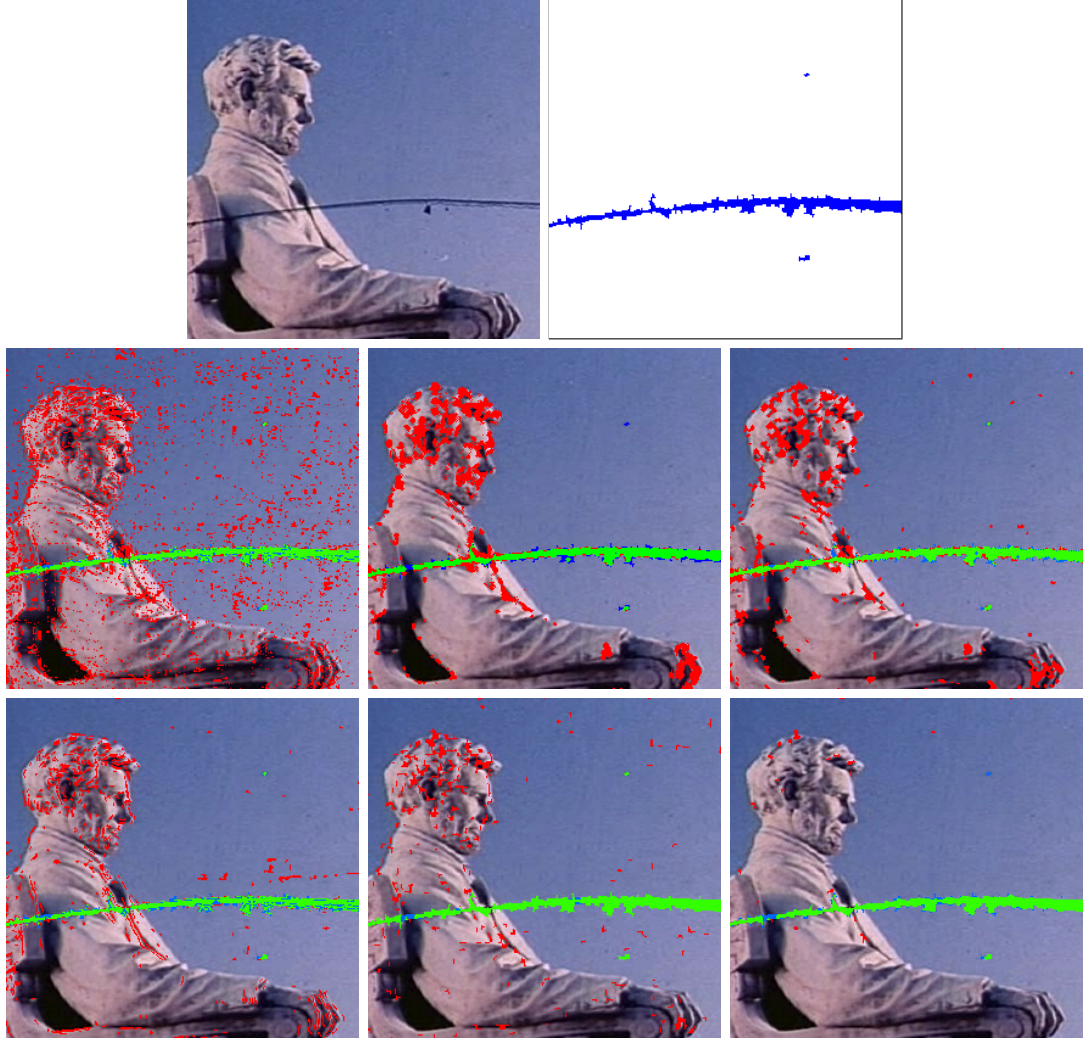


Figure 5.12: American - Top: Degraded frame and its groundtruth mask; Middle: detection results from SDIp, Morris95, Kokaram04; Bottom: results from RV07, HAFID and HAFID-STC. Green: correctly detected defect; Red: false alarms; Blue: defects not detected.

Table 5.2: American - Statistical results of false alarm elimination

Method	SDIp	Morris95	Kokaram04	RV07	HAFID	HAFID-STC
Detected true defects	1707	1712	1780	1534	1997	1819
Defects not detected	368	363	295	541	78	256
False detection	9397	4356	3938	3008	1917	191
Correct detection(%)	82.3	82.5	85.8	73.9	96.2	87.7
False alarm(%)	11.6	5.4	4.8	3.7	2.3	0.2

Table 5.3: *Averaged computational speed for all methods*

Method	SDIp	Morris95	RV07	Kokaram04	HAFID	HAFID-STC
seconds	3	288	62	174	11	17

Computational Issues

In Table 5.3, we reproduce the comparison results of the computational speed of all the methods from the previous chapter and additionally show the computational cost of HAFID-STC. As other methods, HAFID-STC is implemented and computed in MATLAB on a laptop with Intel Core Duo 2.4 GHz and 2GB RAM. On average, HAFID-STC adds a further 6 seconds on the entire algorithm, but this additional computational cost is affordable in archive film restoration, considering that HAFID-STC is able to remove over 70% of the false alarms and still detect more true defects than all the other methods.

5.4 Conclusions

In this chapter, we first briefly outlined the categories of false alarms by looking at both causes and characteristics. Two distinctive characteristics of false alarms were summarized and analyzed, which is shared by the majority of false alarms caused in defect detection. Then, our two-stage false alarm elimination algorithm was proposed to eliminate false alarms according to their characteristics based on (a) MRF modelling for false alarms that have strong correlation with their neighbours and (b) localised feature tracking for those that can be traced temporally. The resulting defect maps after false alarm elimination achieved improved accuracy in comparison to other commonly used or state-of-the-art techniques.

Our proposed false alarm elimination algorithm was not able to eliminate false alarms caused by blurring due to its unique features discussed earlier. Since some of the blurring effects are added on purpose, we will pass them to the follow-up defect removal stage in which algorithms have been designed to preserve the original intensities of those falsely detected pixels.

RWDR: Spatiotemporal Random Walks based Defect Removal

Given the defect maps from defect detection, the aim of the defect removal stage is to restore the missing data on the degraded pixel positions labelled in the defect maps. Although HAFID-STC is capable of eliminating the majority of false alarms, the defect maps still comprise a number of falsely detected pixel positions. Therefore, we also need to prevent introducing new artifacts at those false detection pixels during the defect removal process.

As described in Section 2.4, a number of inpainting methods have been developed for filling missing regions. Most of these algorithms can be applied directly to our defect removal application in a frame-by-frame manner, however the accuracy of these methods is limited without the use of temporal correlation information among frames. Our proposed defect removal method aims to achieve improved accuracy by looking into both the spatial and temporal information contained in a dynamically generated neighbourhood for each missing pixel. As categorized in Section 2.4, our proposed spatiotemporal Random Walk based Defect Removal (RWDR) method is an exemplar based restoration approach which involves a search-and-replace procedure.

One approach to the restoration of a degraded pixel is to replace it with an original

corresponding pixel along its projected motion trajectory from (temporal) neighbouring frames. This clearly involves an accurate estimation of the degraded pixel's motion through space and time and helps enforce a local consistency by imposing features besides just image intensities, *i.e.* motion vectors (leading to consistent optical flows). The chances of more accurate recovery can be increased by recruiting more significant features, *e.g.* texture features such as the Local Binary Pattern (LBP) [70] (leading to consistent region representation). Unlike previous methods such as [34, 53, 67, 86], we consider multiple features *in an integrated fashion* and show that this provides better restoration than treating the features separately. The computational expense incurred due to the use of more features is an affordable tariff in our archive film restoration application where accuracy is of paramount importance.

In order to locate the optimal replacement for a degraded pixel, we establish a region of candidate pixels formed by a number of 3D random walks on the *spatiotemporal* domain, starting from the defective pixel. In [29], spatial-only random walks were applied for noise reduction by taking a weighted average over all spatial pixels visited by the random walks, whereas we select the optimal pixel-exemplar as the pixel which has the maximum likelihood of being the original pixel - as defined by its intensity, motion and texture characteristics - from this dynamically generated spatiotemporal region. The relationship between the proposed method and previous random-walk based methods will be discussed in detail in Subsection 6.3.4. We perform this search-and-replace procedure for each degraded pixel in the defect map in a multiscale framework to refine the restored pixels from coarse to fine. This multiscale refinement particularly helps with large degraded regions which are forced to implode gradually through the propagation of reliable outer pixels into the region.

The contributions of our approach are therefore as follows. We present a novel pixel-exemplar based restoration algorithm using spatiotemporal random walks. In comparison to current state-of-the-art archive film restoration techniques, our method is more accurate by using more reliable statistics produced during the random walks. Also, in addition to intensity and motion features, we employ a higher order texture feature, *i.e.* one that is more complex than raw intensities. Finally, degraded pixels within a frame are collectively restored in a multiscale framework by updating all their features (intensity, motion and texture), which leads to more effective searching for optimal replacements (and significantly helps in the restoration of degraded

regions that are considerably larger than typical defects). This means that at each scale the attributes of a defective pixel are updated with reference to the attributes of normal pixels and other defective pixels updated in the previous higher scale as long as they fall within the defective pixel’s random walk-based spatiotemporal neighbourhood. Thus, there are more constraints to contribute to the restoration of intensities.

The fundamentals of 3D random walks are introduced first in Section 6.1. Then, our restoration algorithm is described in Section 6.2, in which an advanced algorithm for exemplar searching and a multiscale restoration framework are discussed in Subsections 6.2.1 and 6.2.3 respectively. In Section 6.3, we evaluate and compare our proposed method against two state-of-the-art methods, *i.e.* [52] and [34], on a variety of artificially degraded and real films. Finally, we demonstrate the applications of our proposed algorithm in image and video inpainting using handlabelled masks.

6.1 Spatiotemporal Random Walks on Image Sequences

A random walk is a simple stochastic process, which has been applied to model random processes in many areas, such as share prices [61], random movement of molecules [88], image segmentation in computer vision [38] and so on. Here, we establish the fundamentals of 3D random walks on an image sequence and then express the probability of a random walk sequence in the context of our application.

First, we define the input image sequence as an undirected and weighted graph $G = (V, E)$ with vertices (nodes) $v_{\mathbf{x}} \in V$ and edges $e_{\mathbf{x}', \mathbf{x}''} \in E \subseteq V \times V$. Each edge $e_{\mathbf{x}', \mathbf{x}''}$ is assigned a weight $w_{\mathbf{x}', \mathbf{x}''}$ where $w_{\mathbf{x}', \mathbf{x}''} > 0$ and $w_{\mathbf{x}', \mathbf{x}''} = w_{\mathbf{x}'', \mathbf{x}'}$. An image pixel \mathbf{x} at location (i, j, t) ($1 \leq i \leq Width, 1 \leq j \leq Height, 1 \leq t \leq Length$) is represented as a node $v_{\mathbf{x}} (v_{\mathbf{x}} \in V)$ in graph G where $Width \times Height \times Length$ defines the image sequence volume.

A random walk sequence $Path_{0,K} = \{\mathbf{x}^0, \mathbf{x}^1, \dots, \mathbf{x}^K\}$ with length $K+1$ on graph G is

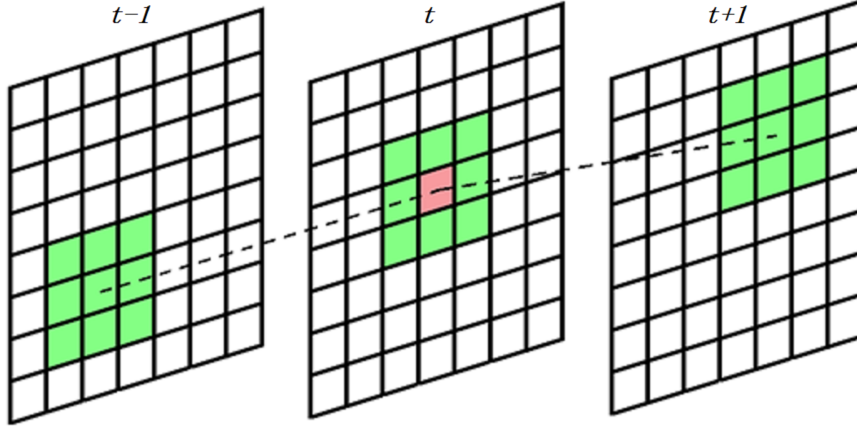


Figure 6.1: Spatiotemporal motion compensated neighbourhood across 3 frames.

specified as a sequence of nodes (pixels) which is a Markov process. The probability of the transition $p(\mathbf{x}^k | \mathbf{x}^{k-1})$ between consecutive pixels \mathbf{x}^{k-1} and \mathbf{x}^k is given as the weight $w_{\mathbf{x}^{k-1}, \mathbf{x}^k}$ on the edge $e_{\mathbf{x}^{k-1}, \mathbf{x}^k}$. According to the Markov property of $Path_{0,K}$, the probability of a $Path_{0,K}$ starting at pixel \mathbf{x}^0 is defined as

$$p(Path_{0,K}) = \prod_{k=1}^K p(\mathbf{x}^k | \mathbf{x}^{k-1}) = \prod_{k=1}^K w_{\mathbf{x}^k, \mathbf{x}^{k-1}}, \quad (6.1)$$

where $P = \{Path_{0,K_m}^m\}_{m=1}^M$ is a set of M random walks on graph G , with each walk starting from \mathbf{x}^0 . Furthermore, we define the region of candidates or pixel-exemplars $\mathbf{R}_{\mathbf{x}^0} = \bigcup_{m=1}^M Path_{0,K_m}^m$ as the set of all pixels visited by the random walks in P . An example of $\mathbf{R}_{\mathbf{x}^0}$ is illustrated later in Figure 6.4. The neighbourhood for a pixel, the associated edge weights, and the walk length are discussed in detail in the following sections.

6.1.1 Neighbourhood

For each pixel \mathbf{x} on a walk, we define a $3 \times 3 \times 3$ spatiotemporal motion compensated neighbourhood $N_{\mathbf{x}}$ centred at \mathbf{x} . Figure 6.1 shows a sample 3-D neighbourhood across 3 frames centred at a pixel marked in red. At $N_{\mathbf{x}}$, we denote the connection between pixel \mathbf{x} and $\mathbf{x}' (\mathbf{x}' \in N_{\mathbf{x}}, \mathbf{x}' \neq \mathbf{x})$ as edge $e_{\mathbf{x}, \mathbf{x}'}$ with a weight $w_{\mathbf{x}, \mathbf{x}'}$. For each

step in a random walk, a transition from the current pixel \mathbf{x} to one of its 26 direct neighbours $\mathbf{x}' (\mathbf{x}' \in N_{\mathbf{x}})$ is permitted.

6.1.2 Edge Weight

In the same fashion as previous graph-based methods, *e.g.* [37], the edge weights are defined by a function that evaluates the similarity of two consecutive pixels during a random walk so as to bias it to stop the walk when a significant decrease in similarity is observed. Here, we define edge weights as the probability of pixels \mathbf{x} and \mathbf{x}' being identical, measured by using a number of different pixel features,

$$w_{\mathbf{x}, \mathbf{x}'} = \frac{1}{T} \prod_{q=1}^Q \exp\left\{-\frac{\varphi_q^2(\mathbf{x}, \mathbf{x}')}{2\sigma_q^2}\right\}, \quad (6.2)$$

where T is a normalization constant, σ_q is the standard deviation for pixel feature q , and $\varphi_q(\cdot)$ measures the Euclidean distance between pixel \mathbf{x} and \mathbf{x}' in feature space \mathfrak{F}_q . A variety of pixel features can be used to measure the similarity between two pixels and here we apply four (*i.e.* $Q = 4$); these are intensity, forward and backward motion, and the local LBP texture pattern:

$$\varphi_q^2(\mathbf{x}, \mathbf{x}') = \frac{1}{J_q} \sum_{j=1}^{J_q} (Z_q^j(\mathbf{x}) - Z_q^j(\mathbf{x}'))^2, \quad (6.3)$$

where $Z_q = \{\mathbf{I}, \mathbf{V}^f, \mathbf{V}^b, \mathbf{L}\}$ for $q = \{1..4\}$, \mathbf{I} represents RGB intensity maps with $J_1 = 3$, \mathbf{V}_x^f and \mathbf{V}_x^b represent forward and backward motion vector maps with $J_2 = J_3 = 2$ respectively, and \mathbf{L} represents maps of 2D image LBP patterns in a spatial 3×3 neighbourhood with $J_4 = 8$. The addition of a texture feature, along with a more integrated contribution of all the features used through (6.3), and subsequently (6.5), is an essential improvement on other works in archive film restoration, such as [52] and [34]. The extra texture feature is specifically appropriate to enforce a constraint in textured regions to help select the pixels that can be included in the region of candidates during the random walks.

6.1.3 Walk Length

We control the length of a random walk by monitoring $p(Path_{0,K})$ in the same manner as proposed in [29]. Since we are performing a biased random walk by encouraging transitions between similar neighbours, the random walk will be terminated if $p(Path_{0,K})$ is smaller than a threshold. This will prevent random walks from stepping across strong boundaries in terms of significant changes of all pixel features. A walk will also be terminated if it hits a hard boundary, *i.e.* the image boundaries on the spatial domain and the first and last frames on the temporal axis.

6.2 Defect Removal in Archive Film

Traditional pixel-exemplar or patch-exemplar based restoration methods such as [26, 34, 105], search for the optimal exemplar amongst a square or rectangular region of pixels using sliding windows. A novel feature of our proposed method is that for each defective pixel examined, we explore a dynamically generated, random-walk based region of candidate pixel-exemplars to select the optimal replacement from. Every pixel in this region shares a significant similarity with the previous pixels in the region as defined by their features, *i.e.* intensity, motion and texture. A random walk starts from a degraded pixel and stops when it reaches a strong boundary in terms of a significant change in all the pixel features. The size of the region is thus determined on-the-fly and is based on the length of all the random walks (for the current defective pixel). We perform an experimentally-determined fixed number (see Section 6.2.1 for details) of random walks for a degraded pixel to form a region.

After building the region of candidate pixel-exemplars for a degraded pixel, we assign to each of them a likelihood of being the optimal replacement for the degraded pixel. This is obtained for each pixel-exemplar by first computing the average (geometric mean) of transition probabilities during each random walk which starts from the degraded pixel and visits the pixel-exemplar. Then the averaged probabilities from these random walks are summed up to get a measure of the similarity between the pixel-exemplar and the rest of the pixel-exemplars in the region (recalling that the transition probabilities are an indication of pixel similarities in a path). The higher

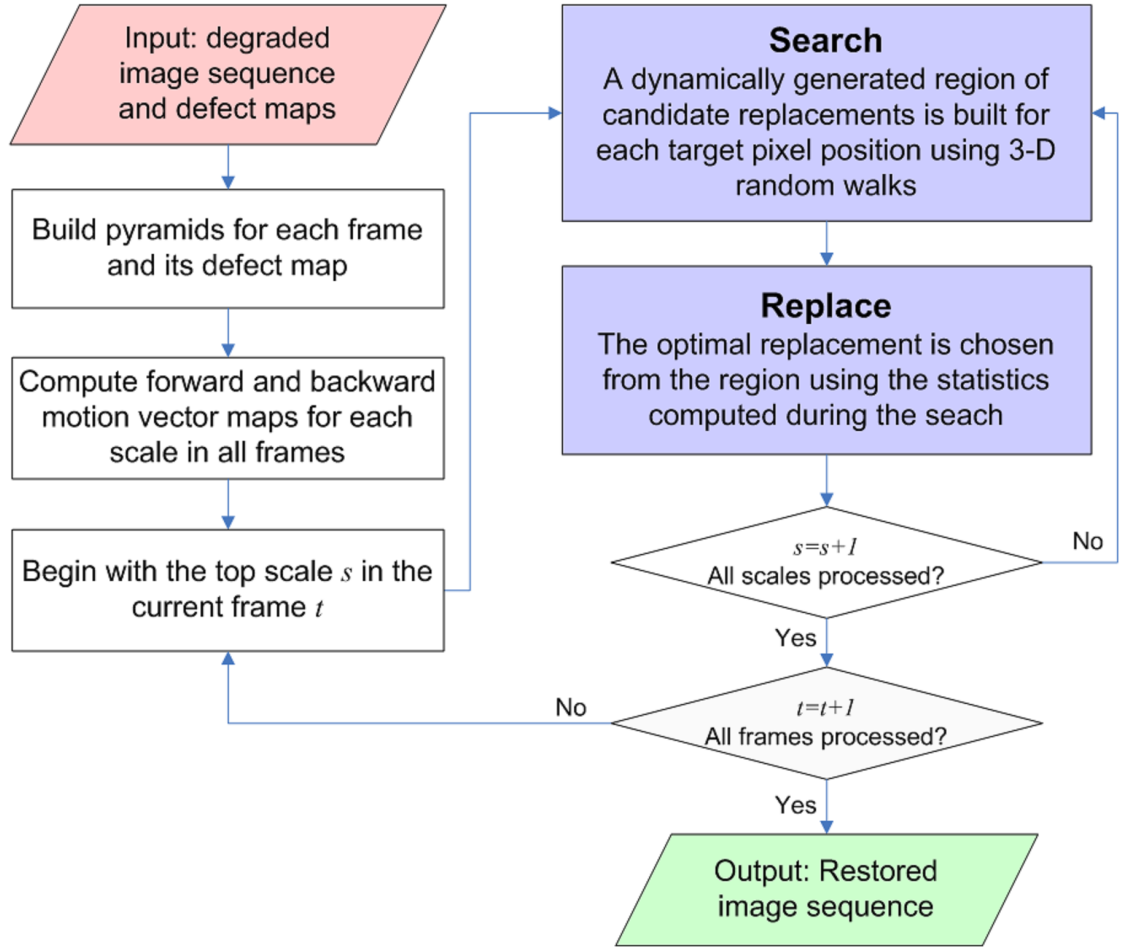


Figure 6.2: Outline of our proposed defect removal method.

this value, the higher is the similarity. This is then weighted by a reliability value, which measures the degree of degradation for each pixel-exemplar, to obtain its likelihood value. The pixel-exemplar with the maximal likelihood will be selected to replace the target degraded pixel. This means that the selected pixel is the optimal representation of the spatiotemporal random walk-based region of candidates - with relatively low (to possibly no) degree of degradation - to restore the current degraded pixel. The above processing is performed in multiscale for all degraded pixels within a frame along with their reliability values, refining the updated pixels' features from coarse to fine. Figure 6.2 outlines the structure of our defect removal approach.

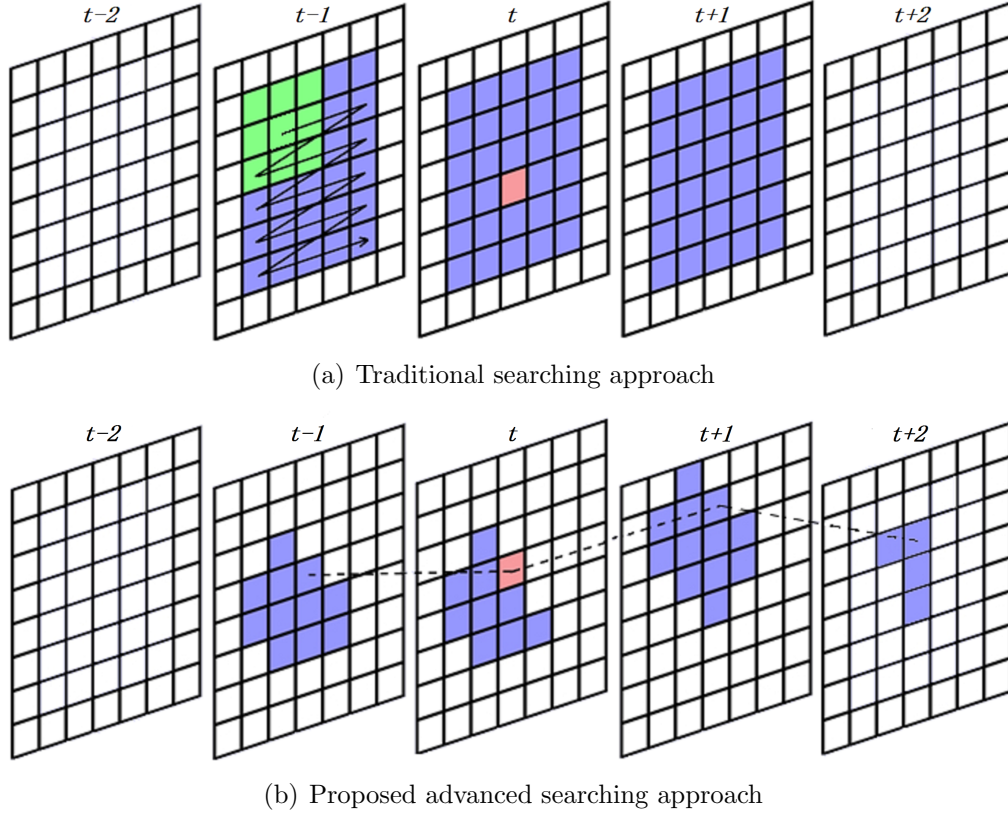


Figure 6.3: Visual comparison between traditional searching and proposed advanced searching. Blue: Searching range; Green: Sliding Window; Red: Target pixel; dashed line: Motion trajectory.

6.2.1 Random-walk based Searching

Figure 6.3 illustrates two different searching approaches: traditional sliding-window based method and our proposed random-walk based method. Traditionally, sliding-window based searching is mostly used in exemplar based inpainting methods [26, 34, 105]. The difficulty of adopting this searching approach lies in how to decide the size of the search region and the size of the sliding window. A larger search region will increase the computational cost while a smaller one may miss the optimal replacement. In a similar sense, a larger sliding window will possibly result in under-matching while a smaller one may cause over-matching. It will require experiments to compute the optimal sizes for each application.

This dilemma can be resolved by investigating a region of candidates with an adap-

tive size. Our motivation is also based on the assumption that the optimal replacement for a degraded pixel always exists somewhere near the degraded pixel position spatially and temporally. Our task here is to outline this adaptive-size neighbourhood to include all the pixels that have higher probability of being the original pixel than those located outside this region. Then, we are able to select the optimal pixel-exemplar to replace the degraded pixel from this region.

For each degraded pixel position, we start a random walk from the degraded pixel and randomly select one of its 26 spatiotemporal neighbour as the next step. The probability of this random walk path $p(path_{0,K})$ is updated with the edge weight between two pixels using Equation (6.1). The value of edge weight $w_{\mathbf{x}',\mathbf{x}''}$ decreases when the similarity between two pixels connected by the edge $e_{\mathbf{x}',\mathbf{x}''}$ reduces along the random walk path, as the similarity is defined by pixel's intensity, motion and texture characteristics. The random walk continues until the probability $p(path_{0,K})$ is below a threshold. By performing such random walks for M number of times (discussed in detail in Section 6.3), we are able to dynamically generate the region of candidate pixel-exemplars for degraded pixel position \mathbf{x}^0 . The size and shape of this region vary according to the neighbouring pixels' characteristics, *e.g.* this region will be larger if the degraded pixel is in a more homogeneous area while it will be smaller in a high frequency area.

Figure 6.4 illustrates the generation of a region $\mathbf{R}_{\mathbf{x}^0}$ of candidate pixel-exemplars in three consecutive frames. It is worth noticing that the entire $\mathbf{R}_{\mathbf{x}^0}$ is larger than the region shown in these frames, which is across 9 frames. The likelihoods of being the replacement of the target pixel are also shown for pixels in $\mathbf{R}_{\mathbf{x}^0}$ after we perform the random walks independently for $M = 50, 250$ and 800 times. This likelihood $A_{\mathbf{x}} \cdot \mathbf{r}(\mathbf{x})$ will be defined in Equations 6.4 and 6.6 in the following Subsection 6.2.2. As shown in Figure 6.4, $\mathbf{R}_{\mathbf{x}^0}$ grows bigger when we perform more random walks and a few pixels surrounding the target pixel are highlighted with relatively higher likelihoods.

In addition to a dynamically generated region of candidates, the statistics computed during the random walks provide us with the evidence for choosing the optimal replacement, which also increases the efficiency by reducing the cost of including other optimization methods. The procedures of computing the statistics and selecting the

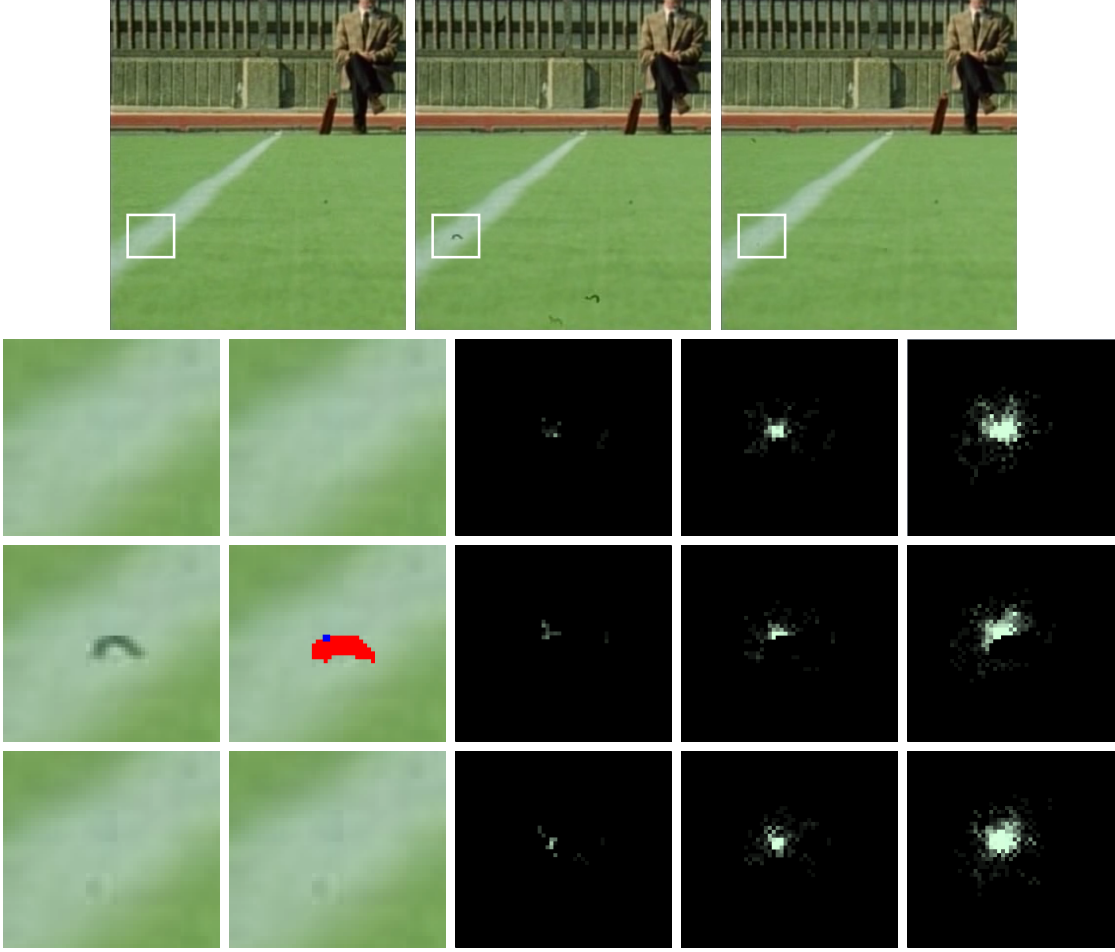


Figure 6.4: Stadium - Top row: three consecutive frames $n - 1$, n and $n + 1$; Bottom three rows (from left to right): enlargement of selected areas, defect maps with the target pixel (the starting position of random walks) in blue, likelihood maps of pixels being the replacement for the target pixel after performing random walks independently for 50, 250 and 800 times. The higher grayscale level, the more likely is a pixel to be the replacement.

optimal pixel-exemplar are introduced in detail in the following section.

6.2.2 Degraded Pixel Replacement

For each pixel-exemplar \mathbf{x} in a degraded pixel's region of candidates, *i.e.* $\mathbf{R}_{\mathbf{x}^0}$, the similarity between \mathbf{x} and the rest of the pixel-exemplars in the region is measured based on the probabilities of random walk paths which start from the degraded pixel

and visit the pixel-exemplar, represented as

$$A_{\mathbf{x}} = \sum_{m=1}^M \sum_{k=1}^{K_m} (p(Path_{0,k}^m)^{1/k} \cdot \delta(\mathbf{x}^k = \mathbf{x})), \quad (6.4)$$

where $\delta(\cdot)$ is the Dirac *delta* function. In order to measure the similarity among all pixel-exemplars in a random walk path regardless of the length of the path, we compute the geometric mean of their transition probabilities. Provided we perform a sufficient number of spatiotemporal random walks, the sum of their averaged probabilities suggests the similarity between the pixel-exemplar and the rest of the pixel-exemplars in the region. This is influenced by the way spatial random walks are used in [29] to examine the transition probabilities (*i.e.* similarity) of pixels along a path in their image denoising application. The reason why we use this value instead of using other measurements, *e.g.* a count of random walks that visit the pixel-exemplar, is because this value indicates if the pixel-exemplar provides random walks with a smooth transition from their previous locations to this pixel-exemplar, *e.g.* if the probabilities of random walk paths decrease significantly after they visit this pixel-exemplar, this value will be probably small even though this pixel-exemplar has been visited by a large number of random walks. The optimal pixel-exemplar is then selected as

$$\hat{\mathbf{x}}^0 = \arg \max_{\mathbf{x} \in \mathbf{R}_{\mathbf{x}^0}} (A_{\mathbf{x}} \cdot \mathbf{r}(\mathbf{x})), \quad (6.5)$$

where $\mathbf{r}(\cdot)$ indicates the reliability of a pixel-exemplar based on its degree of degradation. For normal pixels, $\mathbf{r}(\cdot)$ is 1 while a degraded pixel is initialised to the likelihood of being identical to all its defect-free neighbours in $N_{\mathbf{x}}$:

$$\mathbf{r}(\mathbf{x}) = \begin{cases} \frac{1}{\sum_{\mathbf{x}' \in N_{\mathbf{x}}} \delta(d_{\mathbf{x}'})} \sum_{\mathbf{x}' \in N_{\mathbf{x}}} w_{\mathbf{x}\mathbf{x}'} \delta(d_{\mathbf{x}'}) & d_{\mathbf{x}} = 1 \\ 1 & d_{\mathbf{x}} = 0 \end{cases} \quad (6.6)$$

Note also that by this definition, a false alarm pixel is more likely to be initialised with a high $\mathbf{r}(\cdot)$ value, given it is likely to be more similar to its defect-free spatiotemporal neighbours than to real degraded pixels. Figure 6.5 shows different initializations for a true degraded region and false alarm regions. The higher grayscale level, the higher is the value of reliability. As shown in Figure 6.5, normal pixels are set to the highest value (white) and true defect positions are mostly black. False alarm

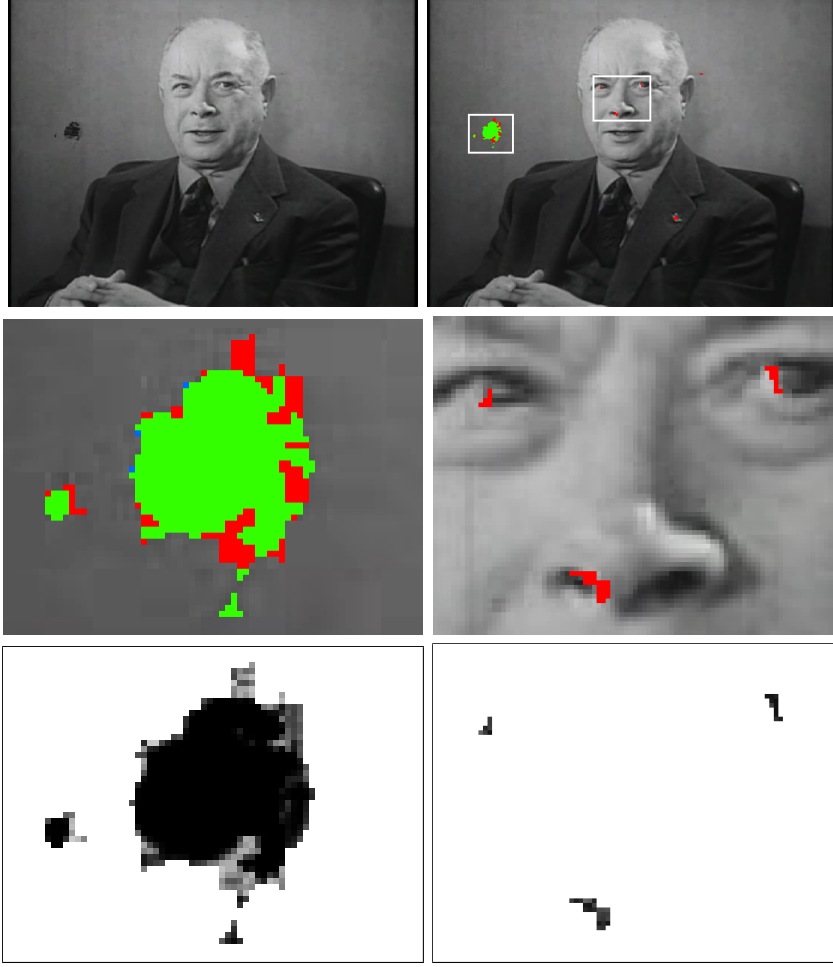


Figure 6.5: *Story of Television* - from top to bottom: the original frame and its defect map; enlargement of selected areas overlaid with defect maps; Initialized $\mathbf{r}(\cdot)$ maps. Green: true defects; Red: false alarms Blue: defects not detected.

pixels are granted with higher reliability values than real degraded pixels (*e.g.* the false alarm region on man's nose) so that the false alarm pixel is more likely to obtain an optimal replacement considering more reliable candidates (nearby false alarm pixels) are present in its random walk-based neighbourhood.

After a degraded pixel is replaced with a specific pixel-exemplar, its reliability value is updated with:

$$\hat{\mathbf{r}}(\mathbf{x}) = \frac{1}{|N_{\mathbf{x}}|} \sum_{\mathbf{x}' \in N_{\mathbf{x}}} \mathbf{r}(\mathbf{x}'). \quad (6.7)$$

During the multiscale updating algorithm (reviewed next), the $\mathbf{r}(\cdot)$ value for a degraded pixel will approach 1 after a number of updates. For a degraded pixel near the boundary of a degraded region, the $\mathbf{r}(\cdot)$ value will reach 1 faster than an inner pixel considering it is surrounded by more reliable spatiotemporal neighbours (normal pixels). Thus, during the multiscale refinement, a degraded region will gradually implode through the propagation of more reliable outer pixels in the region. For an example see Figure 6.6.

6.2.3 Multiscale Refinement

Given an image sequence and its defect map, we build pyramids for each frame and its corresponding defect map by downsampling the original by a factor of 2 after smoothing with a 5×5 Gaussian kernel. A sample image pyramid and its associated defect pyramid are shown in Figure 6.6. After restoring the degraded pixels' features on a current level of the pyramid, we upsample these pixels to the next level and then update their corresponding pixels' features in that level. This level-by-level refinement and restoration process continues until it reaches the lowest level of the pyramid (see Algorithm 6.1).

6.3 Experimental Results and Discussion

We present the restoration performance of the proposed algorithm on both artificially degraded and real sequences, and compare our results against two state-of-the-art techniques: Kokaram's Bayesian framework [52] and Gangal and Dizdaroglu's exemplar-based method [34], hereafter referred to as Kokaram04 and GD06 respectively.

Kokaram04: As the state-of-the-art, Kokaram's statistical approach performed defect detection and defect removal in a single framework. As discussed in Section 2.4, the method attempted to remove noise and dirt, and perform motion correction together in a two-stage restoration process. Three binary variables were used for each pixel to mark if a pixel is degraded, forward occluded or backward occluded.

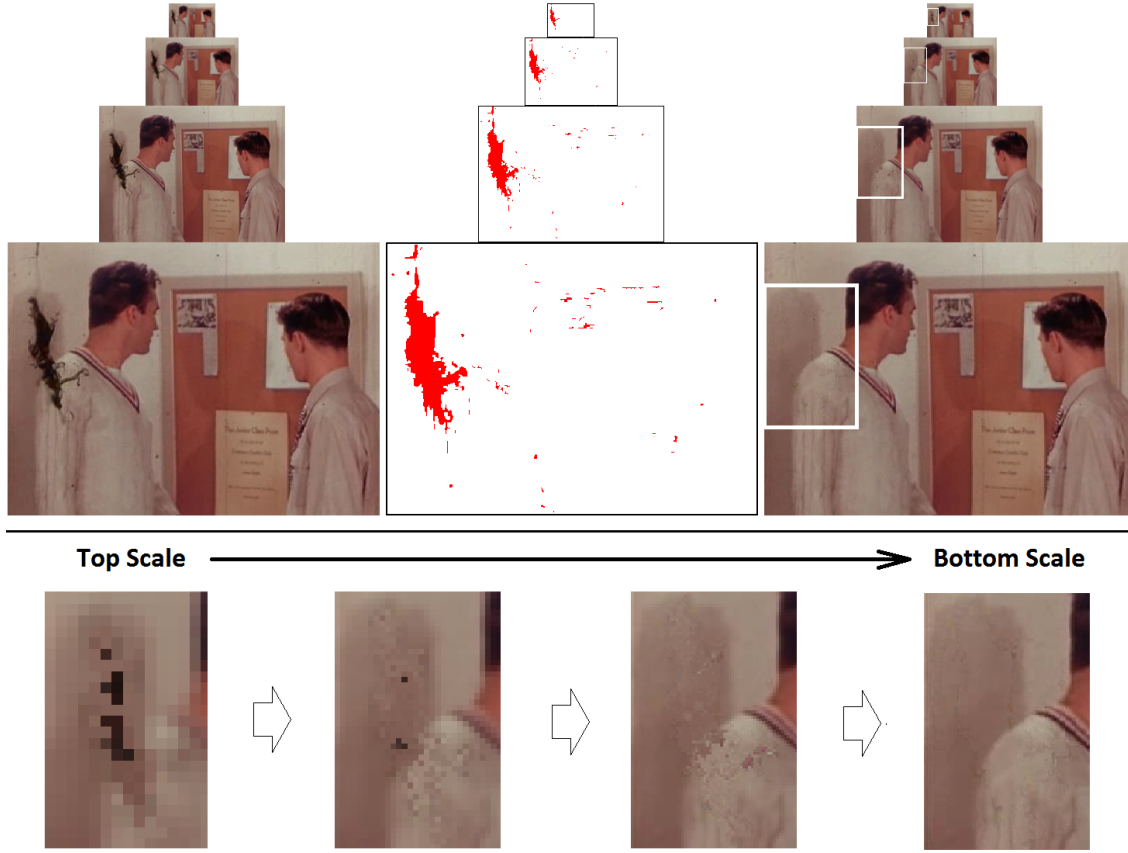


Figure 6.6: (from left) A sample image pyramid, the defect map pyramid, and the restored results using the proposed method. The degraded regions are gradually recovered from coarse to fine and from the boundaries to their inner part.

These variables, together with restored image intensities and motion vectors, were defined as unknowns. Given the pixel values of degraded frames and initial motion estimations, a two-stage procedure was designed to estimate the variables and image intensities first and then adjust the motion vectors according to neighbouring motion vectors, before repeating this process for a fixed number of iterations.

It is worth noting that to perform motion adjustment for a degraded pixel, Kokaram04 relied on the accuracy of the pixel's surrounding motion vectors. During the iterative processing, motion information was improved separately and with no reference to the improved intensities. However as stated earlier, we update the motion vectors of a defective pixel in a multiscale process with reference to all attributes (*i.e.* intensity, motion and texture) of normal pixels and other defective pixels (updated in


```

1: Build pyramids  $\{\mathbf{I}_s\}_{s=1}^S$  and  $\{\mathbf{D}_s\}_{s=1}^S$ ;
2: Initialize scale  $s = 1$ : Compute motion vector maps  $\mathbf{V}_1^f$ ,  $\mathbf{V}_1^b$ ,  $\mathbf{L}_1$ , and  $\mathbf{r}_1$ ;
3: while  $s \leq S$  do
4:   if  $s > 1$  then
5:     Update  $\mathbf{I}_s$ ,  $\mathbf{V}_s^f$ ,  $\mathbf{V}_s^b$ ,  $\mathbf{L}_s$  by  $\mathbf{I}_{s-1}$ ,  $\mathbf{V}_{s-1}^f$ ,  $\mathbf{V}_{s-1}^b$ ;
6:     /*Only on degraded sites*/
7:   end if
8:    $(\mathbf{I}_{s+1}, \mathbf{V}_{s+1}^f, \mathbf{V}_{s+1}^b, \mathbf{L}_{s+1}) = \text{Restoration}(\mathbf{I}_s, \mathbf{V}_s^f, \mathbf{V}_s^b, \mathbf{L}_s, \mathbf{D}_s)$ ;
9:   Update  $\mathbf{r}_{s+1}$  using Equation (6.7);
10:  if  $s < S$  then
11:    Upsample  $\mathbf{I}_{s+1}, \mathbf{V}_{s+1}^f, \mathbf{V}_{s+1}^b, \mathbf{r}_{s+1}$  by factor 2;
12:    /*For every scale but the last*/
13:  end if
14:   $s = s + 1$ ;
15: end while

```

Algorithm 6.1: *The multiscale restoration algorithm*

the higher scale) as long as they fall within its spatiotemporal random walk-based neighbourhood.

GD06: Gangal and Dizdaroglu extended Criminisi’s exemplar based image inpainting method [24] into the spatiotemporal domain. A patch matching based process was conducted to propagate the textures into the degraded region by giving the priorities to those pixels that have higher gradients, *i.e.* image structures, so that the recovery of image structure and texture can be achieved simultaneously. In GD06, the searching and matching were restricted in motion compensated spatiotemporal neighbourhoods for better accuracy and efficiency.

The defect maps for both GD06 and the proposed method were produced in advance using the HAFID-STC defect detector (described in Chapter 4 and 5) while Kokaram04 has an integrated defect detector. All methods were tuned for optimal performance using constant parameter values across all experiments.

6.3.1 Synthetic Defects

The proposed method was compared against Kokaram04 and GD06 on restoring 10 artificially degraded real sequences totalling 2534 frames, namely *Mobile Calendar*, *Container*, *Foreman*, *News*, *Paris*, *Stockholm*, *Flower*, *Shields*, *Park* and *Bus*. The degraded sequences were produced by adding synthetic black and white defects of sizes of between 1 and 6000 pixels on a random basis (see examples in Section 3.3). For each method, the Mean Square Error (MSE) to measure the difference between the original defect-free frame F and the restored frame \hat{F} was computed. The MSE measurement was previously described in Section 3.4.

Columns 2 through to 5 in Table 6.1 and Table 6.2 show the MSEs for four randomly selected sample frames from the *Mobile Calendar*, *Container*, *Foreman*, *News*, *Paris*, *Stockholm*, *Flower*, *Shields*, *Park* and *Bus* sequences respectively. The percentage of degraded pixels in each frame is listed along with the frame number along the top row. The raw, unrestored frame error rate is shown along the ‘Degraded’ row in each case. The last column in Table 6.1 and Table 6.2 shows the average MSEs across all degraded frames in each of the synthetic-error sequences for each method; for example for the *Foreman* sequence, given the average true MSE rate of 153.4, the proposed method resulted in the lowest error at 44.7 compared to Kokaram04 and GD06 at 130.3 and 103.1 respectively.

The proposed method performed much better (with lower MSEs) than Kokaram04 and GD06 in all the sequences, *e.g.* for sequences *Container* and *Stockholm*, the averaged MSEs resulted from the proposed method was less than 20% of the MSEs computed on the original frame with synthetic defects and about 50% of Kokaram04 and GD06’s. Additionally, the proposed method was also more capable of restoring large defects. For instance, frame 233 of the *News* sequence (shown in Table 6.1) was artificially degraded with a large black blotch. The proposed method reduced the MSE from 751.5 to 119.8 while Kokaram04 and GD06 resulted in higher errors at 289.1 and 205.3 respectively.

The proposed method was also able to avoid the creation of too many artifacts on these false detection pixel positions in comparison to Kokaram04 and GD06. For example, the *Bus* sequence consisted of complex motions as mentioned in Section

Table 6.1: Comparison of MSEs on 1-5 of 10 real sequences with synthetic errors

<i>Mobile Calendar</i>					
Frame #	32 (0.07%)	58 (0.11%)	181 (3.47%)	233 (0.03%)	Avg (0.62%)
Degraded	16.3	31.8	651.5	56.6	183.5
Kokaram04	210.6	89.9	293.6	92.5	157.3
GD06	128.7	80.7	196.8	123.5	135.9
Proposed	23.4	19.5	105.9	46.7	49.2
<i>Container</i>					
Frame #	9 (0.06%)	23 (0.21%)	138 (2.44%)	210 (0.06%)	Avg (0.60%)
Degraded	15.7	80.7	451.1	15.7	119.1
Kokaram04	7.6	3.5	93.4	2.8	33.9
GD06	0.9	0.8	96.3	3.2	23.8
Proposed	0.6	0.6	45.1	3.3	10.5
<i>Foreman</i>					
Frame #	33 (0.12%)	65 (0.24%)	98 (0.23%)	199 (0.02%)	Avg (0.55%)
Degraded	29.7	45.8	40.8	5.3	153.4
Kokaram04	70.5	155.32	89.84	74.0	130.3
GD06	95.3	149.12	95.87	68.5	103.1
Proposed	21.4	51.38	40.37	10.7	44.7
<i>News</i>					
Frame #	113 (0.08%)	165 (0.05%)	233 (2.47%)	291 (0.21%)	Avg (0.61%)
Degraded	31.5	15.2	751.5	66.9	154.5
Kokaram04	159.9	218.3	289.1	89.5	140.7
GD06	140.7	125.7	205.3	113.5	125.3
Proposed	49.5	27.7	119.8	54.4	54.2
<i>Paris</i>					
Frame #	81 (0.16%)	123 (0.01%)	158 (1.34%)	280 (0.10%)	Avg (0.60%)
Degraded	80.7	15.7	537.1	85.3	122.1
Kokaram04	30.5	25.6	107.1	11.3	63.9
GD06	21.8	27.9	122.3	15.4	83.8
Proposed	13.4	14.3	69.3	6.3	33.5

Table 6.2: Comparison of MSEs on 6-10 of 10 real sequences with synthetic errors

<i>Stockholm</i>					
Frame #	12(0.06%)	40(0.22%)	81(0.16%)	158(1.34%)	Avg(0.60%)
Degraded	88.1	64.7	274.2	676.2	132.2
Kokaram04	34.7	43.2	44.9	106.5	51.7
GD06	18.5	69.5	76.8	97.1	55.4
Proposed	5.5	47.0	16.5	80.5	26.2
<i>Flower</i>					
Frame #	7(0.15%)	65(0.24%)	119(0.65%)	152(0.07%)	Avg(0.46%)
Degraded	37.4	195.8	173.5	31.1	142.8
Kokaram04	120.6	112.3	73.5	51.1	90.3
GD06	67.2	242.1	154.5	69.8	145.2
Proposed	11.4	83.3	35.7	17.6	33.7
<i>Shields</i>					
Frame #	32(0.07%)	33(0.12%)	58(0.11%)	133(1.21%)	Avg(0.56%)
Degraded	64.1	112.6	53.2	40.2	78.4
Kokaram04	48.2	52.7	44.2	52.6	49.1
GD06	52.9	60.5	31.1	18.8	36.8
Proposed	10.6	12.9	7.2	12.7	12.7
<i>Park</i>					
Frame #	23(0.21%)	81(0.16%)	85(0.19%)	158(1.34%)	Avg(0.63%)
Degraded	84.8	208.2	29.4	440.9	108.4
Kokaram04	105.5	99.2	56.1	167.5	63.6
GD06	193.9	186.9	21.7	179.5	80.1
Proposed	18.2	28.9	12.8	88.9	27.9
<i>Bus</i>					
Frame #	81(3.47%)	105(0.09%)	109(0.05%)	138(2.44%)	Avg(0.62%)
Degraded	197.1	190.9	17.5	488.4	113.7
Kokaram04	236.8	230.2	152.4	190.4	153.1
GD06	210.3	337.5	298.8	272.8	279.4
Proposed	97.9	87.4	36.3	119.5	66.9

4.3 and its defect map from HAFID-STC contained a considerable number of false detections. As shown in Table 6.2, MSEs errors resulted from both Kokaram04 and GD06 were constantly larger than the raw original whose MSE is only based on synthetic defects. However, the proposed method resulted in an averaged MSE at 66.9, about half of the original MSE.

6.3.2 Real Defects

We compared the three methods in restoring a variety of real degraded image sequences, including grayscale and colour, indoor and outdoor scenes, and slow and fast motions, and in all cases the proposed method produced the best results. In the following, three sets of sample results are illustrated to inspect three aspects of the proposed method, *i.e.* recovering a large degraded region and substantially avoiding artifacts in Figures 6.7 and 6.8, handling defect-free (false alarm) pixels in Figures 6.9 and 6.10, and correcting motions in Figures 6.11 and 6.12.

Figure 6.7 shows the results on a sample degraded frame with a large missing area. The original frame and the defect map (in red) are shown in the top row. The results by Kokaram04, GD06, and the proposed method are in the middle row and a close-up of the degraded area is shown in the bottom row for each method. Kokaram04 resulted in a considerable number of artifacts in the restored frame because its performance strongly depends on the accuracy of motion information. Its motion correction procedure is not designed for such large missing areas, but rather for small degraded areas with accurate motion information provided in their spatial neighbouring regions. While GD06 was able to restore the outline of the man's head, it introduced some artifacts in the inner region due to the mismatching of patches in an early stage. Although the proposed method still caused some small artifacts, both the image structure and texture were recovered well.

Figure 6.8 shows another example with a large degraded area in a sample frame from *Red China*. In a similar manner as the previous example, the restoration results are shown in the middle and a close-up of the degraded area in the bottom row for each method. In this case, Kokaram04 was only able to decrease the degradation, considering its estimation of the original intensity is computed based on an average over

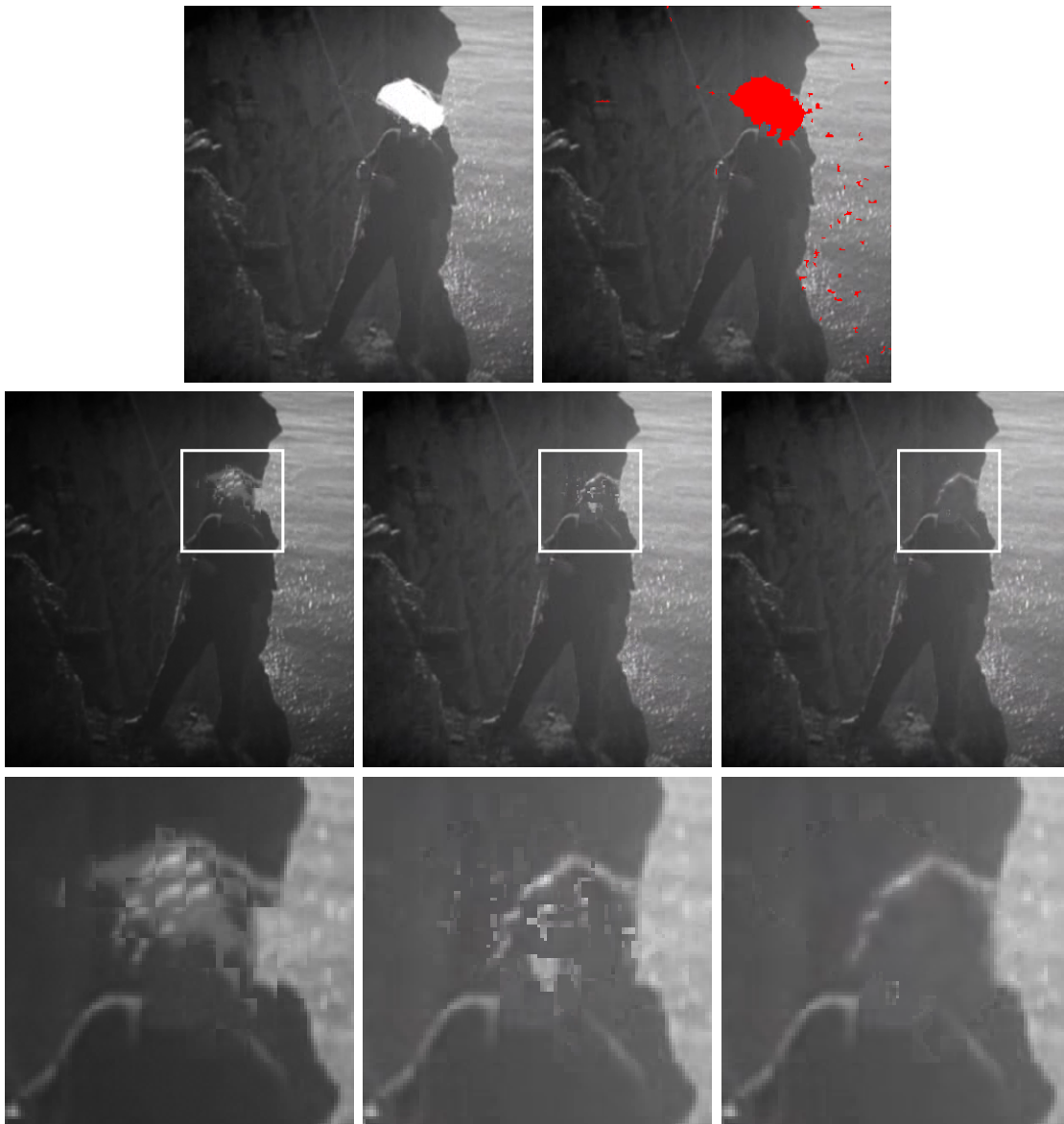


Figure 6.7: *Cliff* - Comparing large missing area recovery. Top: original frame and the defect map in red; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.

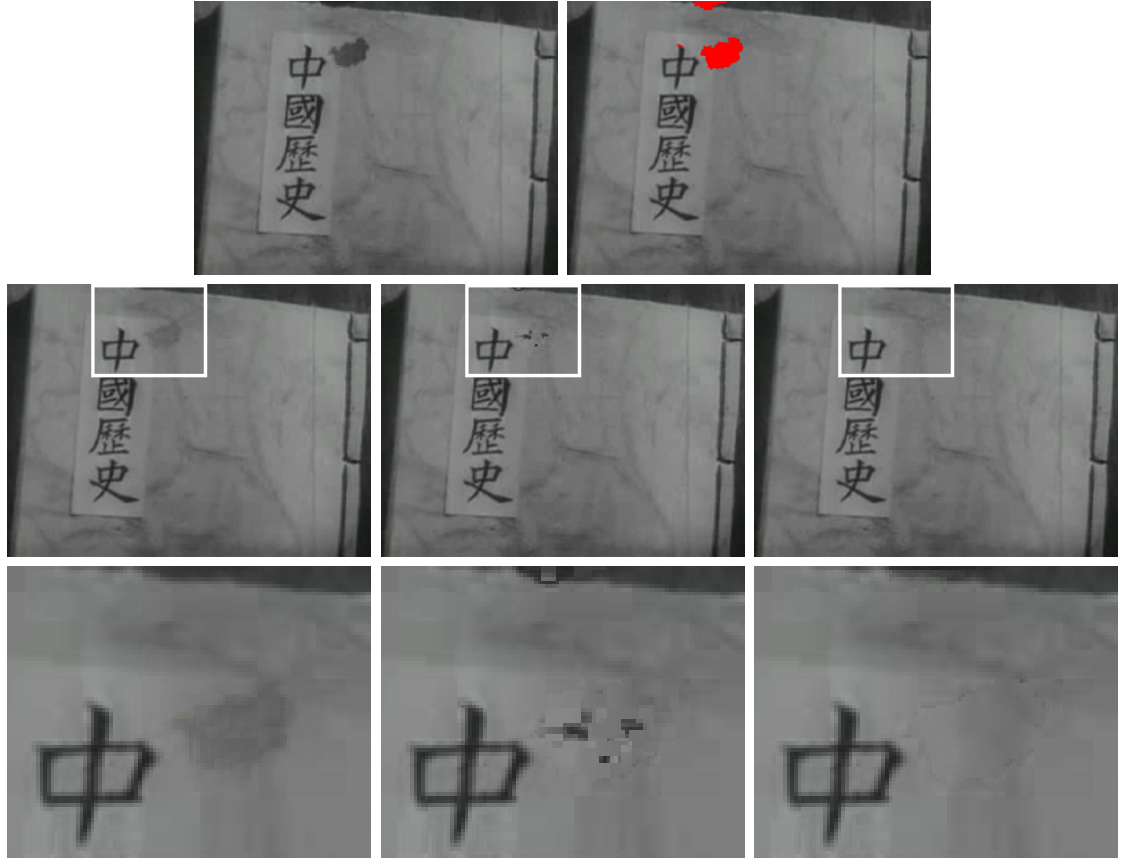


Figure 6.8: Red China - Comparing large missing area recovery. Top: original frame and the defect map in red; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.

the local spatiotemporal neighbourhood. Due to mismatching, GD06 introduced some artifacts in the centre of the missing region. The proposed method was able to restore the image structure (edge of the label) well while inconsistency can still be found over the boundary of the missing area when we examine closely.

In Figure 6.9 we investigate the restoration performance of the three methods on handling false alarm pixels. A sample degraded frame and its corresponding defect map are in the top row. Restoration results from Kokaram04, GD06 and the proposed method are in the middle row and close-up of the false alarm areas follow in the bottom row. All methods did well in restoring the real degraded pixels. However, both Kokaram04 and GD06 introduced artifacts on those false detection pixel positions, *e.g.* artifacts were introduced across the telephone and the policeman's

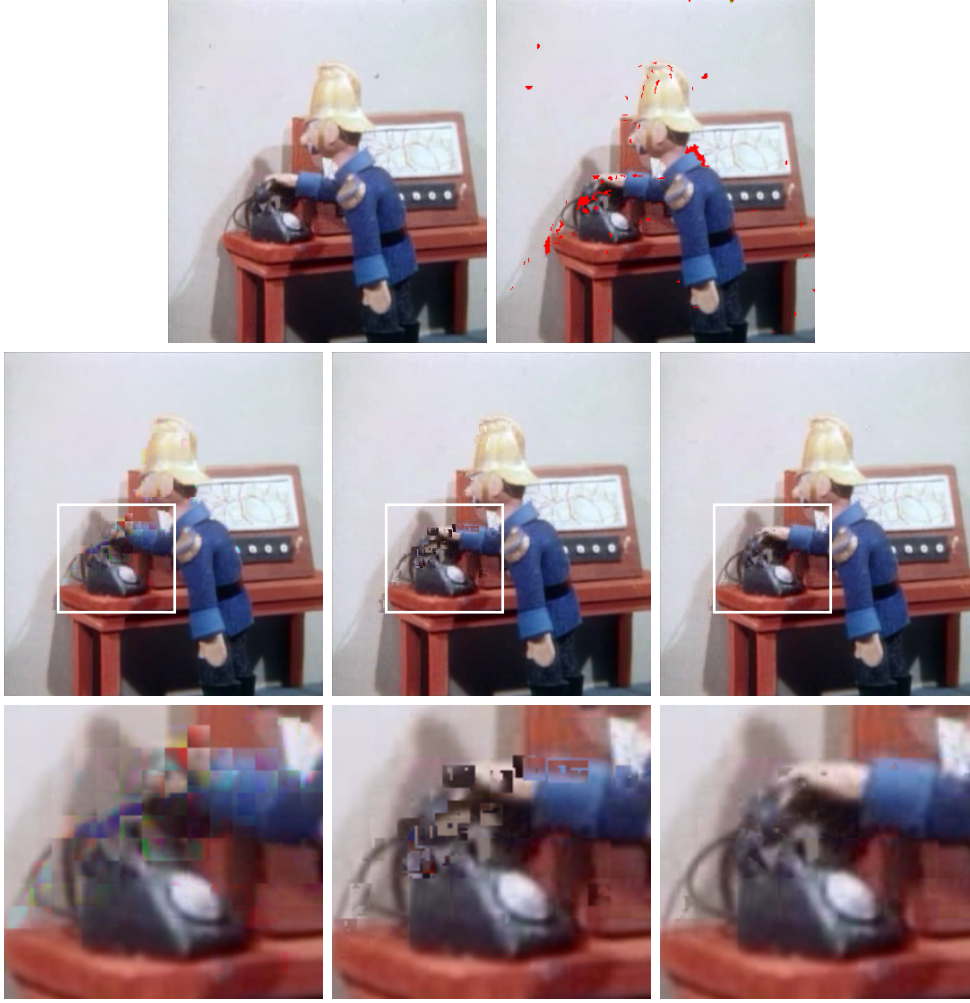


Figure 6.9: *Policeman* - Comparing restoration of false alarms. Top: original and its defect map; Middle: restoration results from Kokaram04, GD06, and the proposed method; Bottom: enlargement of selected areas.

hand in the frame restored by Kokaram04 and GD06.

Figure 6.10 illustrates another example of handling false alarm pixels. Three consecutive frames are shown with their corresponding defect maps and restoration results from Kokaram04, GD06 and the proposed method (from top to bottom). All methods also restored the real degraded pixels well in this example, but both Kokaram04 and GD06 lose considerable detail at positions of false alarm defects, *e.g.* the man's nose is missing in the frame restored by Kokaram04, and the crouching man's arms and face are only restored partially in the results from GD06.

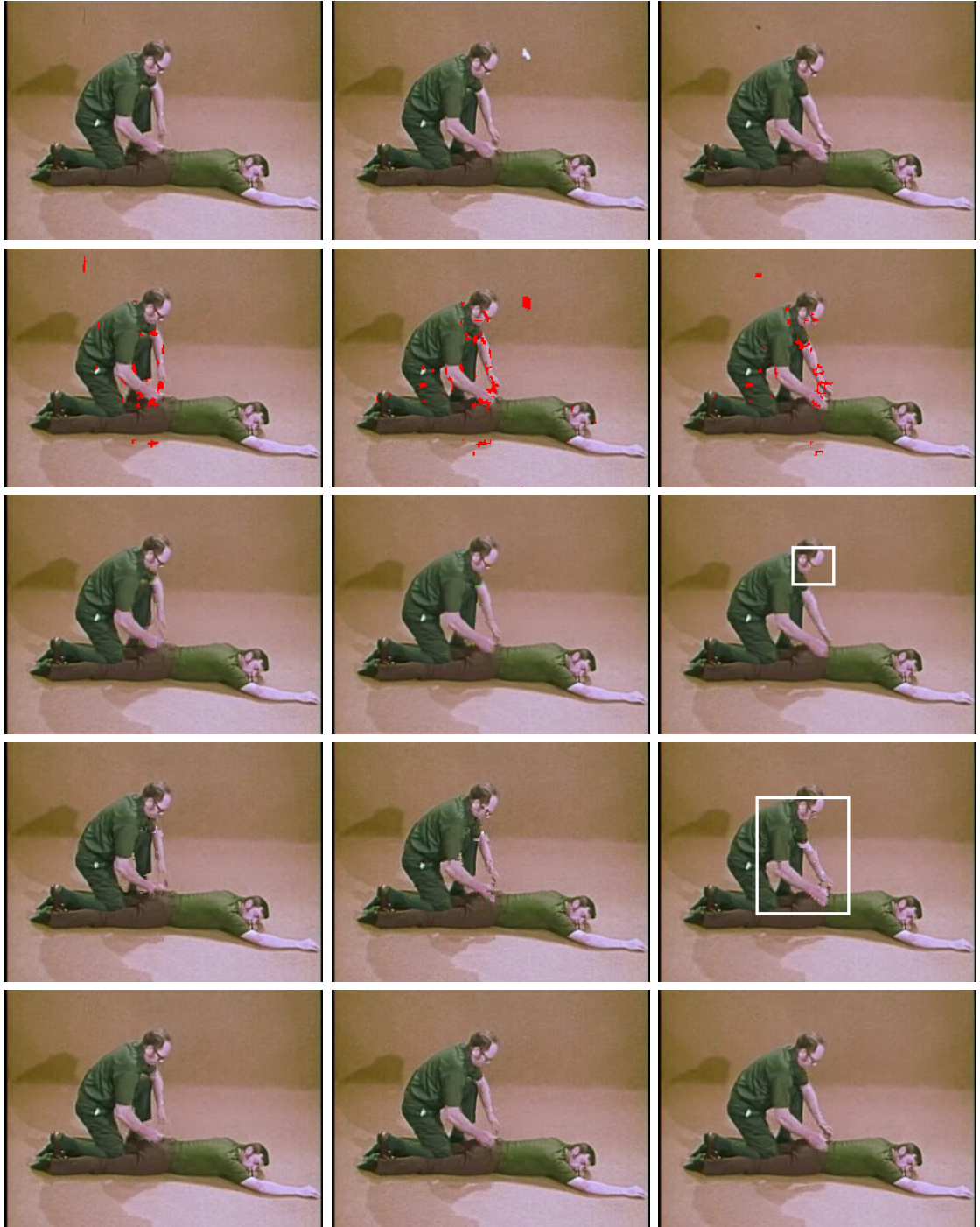


Figure 6.10: Artificial Respiration - Comparing restoration of false alarms over Three consecutive frames. From top to bottom: original frames, their defect map, restoration results from Kokaram04, GD06, and the proposed method.



Figure 6.11: *Junior Prom* - Comparing motion correction. (from top left to bottom right) Original frame with overlaid defects, Original motion vectors, Corrected motion vectors from Kokaram04 and the proposed method.

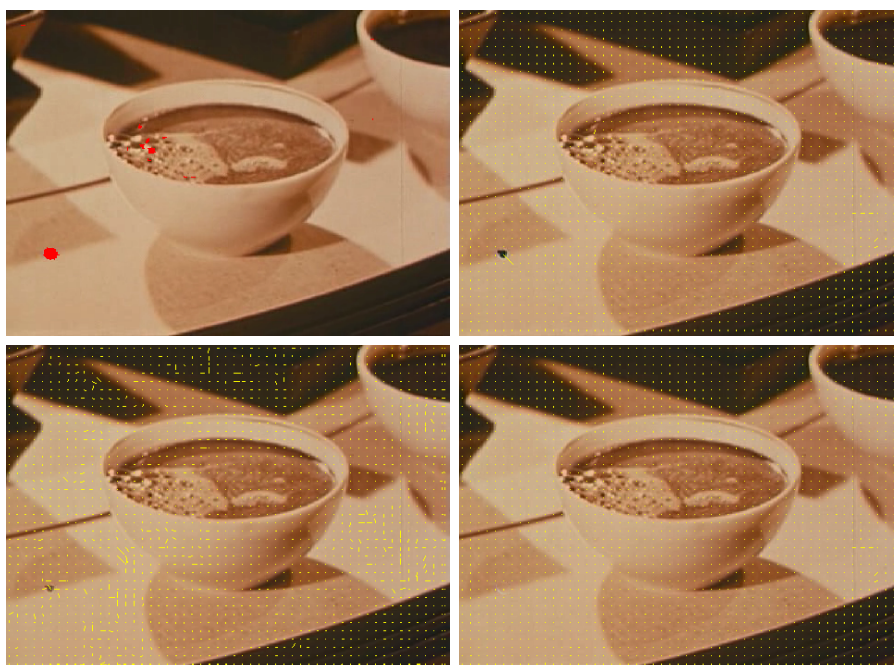


Figure 6.12: *Coffee* - Comparing motion correction. (from top left to bottom right) Original frame with overlaid defects, Original motion vectors, Corrected motion vectors from Kokaram04 and the proposed method.

The final two examples present a comparison between Kokaram04 and the proposed method on correcting motions for degraded pixels. The motion vectors overlaid on original frames are shown in the second image from left in Figures 6.11 and 6.12. The correction based on this by the result from Kokaram04's integrated motion correction algorithm and the proposed method follow this respectively. During Kokaram04's iterative process, motion information was improved separately and with no reference to the improved intensities; this means its correction was limited by the accuracy of initial motion estimations which were often inaccurate by the presence of defects. The proposed method outperformed Kokaram04 by achieving more accurate motion correction for each defective pixel by reference to their spatiotemporal random-walk neighbours through the multiscale process.

6.3.3 Performance and Implementation Issues

All methods were implemented in MATLAB on a laptop with Intel Core Duo 2.4 GHz and 2GB RAM. The average speed for a degraded frame of average size of 480×360 was 406 seconds for our proposed method, while Kokaram04 and GD06 needed 174 and 265 seconds respectively. Our proposed algorithm is slower but more accurate than Kokaram04 and GD06, since it considers an extra feature and requires considerable sampling by the random walks. The number of steps in each random walk often varied from 2 to 48. Since accuracy is critical for the restoration of archive films, the extra computational burden is a tolerable cost.

Due to the random nature of random walks, the optimal replacements located may not be the same, as we perform the search-and-replace procedure on the same pixel twice. Therefore, we experiment with different numbers of random walks (M) performed for each degraded pixel position so that we can find the optimal number of random walks required to achieve a stable and accurate result. Figure 6.13 illustrates the standard deviations of the optimal replacements' intensities with different values of M , resulting from 1000 separate runs of our proposed restoration method on 10689 randomly selected degraded pixel positions using different random seeds for each random walk. $M = 800$ was found to provide stable results (with the standard deviation below 1) and reasonable computing costs.

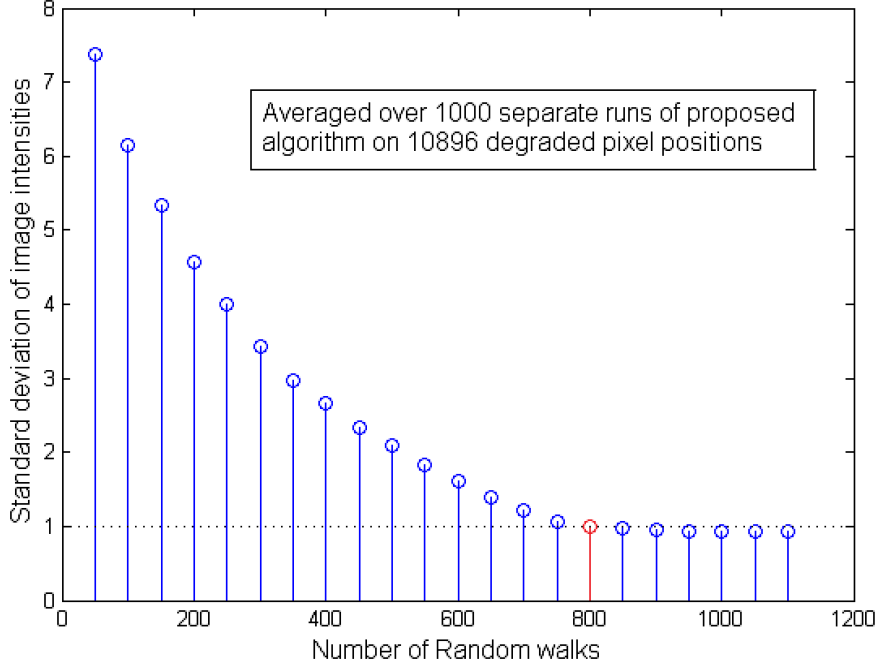


Figure 6.13: The graph shows the deviation of the optimal replacements' intensities falls as the number of random walks performed increases.

6.3.4 Relationship to Previous Random Walk based Methods

Previous methods [28, 66] applied image segmentation by investigating the transition matrix of the random walks and studying the eigenvalues and eigenvectors of this matrix without actually performing random walks. This transition matrix was computed using the pairwise similarities of all pixels in the target image. Then, normalised cut [66] or mini cut [28] were performed on this matrix to group pixels that share similar image features. The segmented local regions are what we want in our defect removal application as sources of candidate exemplars for the degraded pixels located inside these regions. However, this segmentation was optimized globally and results may change significantly with the present of defects, especially large degraded regions. Furthermore, a full segmentation of the image may not be necessary for our defect removal application considering the number of defects is relatively small compared to the entire image. This motivated us to generate a localized and random walk based segmentation around each degraded pixel position.

Estrada *et al.* [29] proposed an image denoising algorithm by performing random walks in the image domain. For each pixel \mathbf{x} in the target image, a weighted average was computed over all pixels that are passed by a number of random walks started from pixel \mathbf{x} . However, instead of only looking at the pairwise similarities for adjacent pixels, the algorithm also measured the similarity between current pixel \mathbf{x}^k and pixel \mathbf{x} in random walk step k . This would particularly benefit the denoising process since image intensities are usually not highly degraded by image noise (Gaussian noise in this case). In such manner, all the pixels used to compute the average share similarities not only with each other but with the starting pixel \mathbf{x} as well. However, the information on the degraded pixel position is often completely lost in our defect removal application. Here, we perform random walks to generate a local segmented region for each degraded pixel by only measuring the pairwise similarity of multiple image features between two adjacent pixels in each random walk step.

We believe our proposed method is not only a spatiotemporal extension to [29] in the archive film restoration application but also an advancement in terms of methodology due to several respects, *e.g.* : (a) In our method, we select the optimal pixel-exemplar from a dynamically generated spatio-temporal neighbourhood, which has the maximal likelihood of being the original pixel, whereas [29] simply takes a weighted average over all spatial pixels, (b) The multiscale algorithm and reliability values are integral in our method to deal with large degraded regions while [29] has no such mechanism and is only designed to recover individual Gaussian noise infected pixels, and (c) We employ additional features, *i.e.* motion and texture, which play an important role in the algorithm and hugely influence the outcome of our archive film restoration application. Both motion and texture features help measure the similarity of pixels during random walks and motion information is essential to guide the random walks on the temporal axis.

6.4 Conclusions

In this chapter, we presented a novel pixel-exemplar based restoration algorithm using spatiotemporal random walks. The random walks were performed by consid-

ering pixel similarities using multiple features. The method was applicable given a defect map generated by any archive film defect detection algorithm. While the use of multiple features added to our computational costs, we obtained much more accurate and artifact-free results than current state-of-the-art techniques.

Conclusions and Further Work

The basic motivation of the work presented here is to find an effective, safe and convenient solution for digitised and automatic archive film restoration. This dissertation not only demonstrates that automatic restoration of archive films is feasible and accurate but also attains insights on the restoration problem itself, *e.g.* characteristics of defects and false alarms, which also leads to a few directions for future investigation.

7.1 Summary

In this thesis, we presented a framework towards automatic restoration of a group of common defects in archive film, *i.e.* dirt and blotches. It was achieved by adapting HMM modelling, MRF modelling and motion analysis, and random walks in three individual stages of this framework, *i.e.* defect detection, false alarm elimination and defect removal.

In Chapter 4 we proposed an archive film defect detection approach involving HMM modelling of temporal pixel sequences and then a leave-one-out process to pinpoint the degraded pixels. An HMM was first characterized by combining the Left-Right and Circular models according to the features of pixel transitions on the temporal

axis. The trained 5-state HMM was accurate for our defect detection task, which was backed up by a testing experiment using real normal pixel sequences. Then, we applied the trained HMM in a leave-one-out process to investigate how the likelihood of an observation sequence being normal varied if each pixel in the sequence was missing one at a time. Most importantly, this process enabled the method to detect if the centre pixel was degraded, instead of only computing if an observation sequence was degraded as in traditional HMM based classifications. Furthermore, we found that it was better to have an overzealous defect detector and then remove false positives, rather than apply a conservative method where not all true defects are found and extra steps have to be implemented till a reasonable target is met. Although we applied a rather coarse modelling of normal pixel observations at the expense of false alarms, the proposed HAFID defect detector resulted in not only a higher correct detection rate but also a lower rate of false detections than the current state-of-the-art techniques, which was testified using a large number of real degraded films with handlabelled groundtruth from a generalized catalogue.

In Chapter 5, we used the resulting defect map from HAFID as a prior in an MRF scheme, to iteratively remove false positive defect candidates by examining spatial constraints followed by a motion tracking stage to remove more false positives based on temporal constraints. This two-stage elimination procedure was developed by analyzing the characteristics of false alarms in addition to causes that were often investigated in previous methods. This strategy was chosen based on our finding that false alarms share similar characteristics though they are caused in different circumstances. The proposed approach proved to be an effective solution to remove false alarms from our HAFID defect detector through experiments reproduced from Chapter 4.

In Chapter 6, we presented a novel pixel-exemplar based restoration algorithm by performing the spatiotemporal random walks on image sequences. An advanced random walk based searching algorithm was first proposed to dynamically generate a region of candidate pixel-exemplars for each degraded pixel position. The random walks were performed by computing pixel similarities using multiple features (intensity, motion and texture). Then, the optimal exemplar was chosen from this region to replace the degraded pixel using the statistics produced during the random walks. Large degraded regions were specially handled using multiscale updating and

reliability values. The proposed algorithm was compared against two state-of-the-art methods to demonstrate the improved accuracy in restoring synthetic and real degraded image sequences. We also demonstrated the performance of our proposed defect removal approach on inpainting.

7.2 Contributions

The contributions of this thesis are:

- A defect detection approach was proposed based on HMM modelling of pixel transitions over a long range and a leave-one-out test process.
- A two-stage false alarm elimination method was developed using spatial and temporal continuity analysis based on two characteristics of false alarms.
- A pixel-exemplar based restoration algorithm was proposed based on spatiotemporal random walks.

7.3 Future Work

- **Defect detection:** Modelling the normal pixel sequences using multiple HMMs can be applied in HAFID detector in the benefit of not only a higher correct detection rate but less false alarms as well although our strategy is always to detect the maximum number of true defects first and followed by false alarms elimination. Modelling of normal pixel sequences using a single HMM was rather coarse as shown in Section 4.3. Complex motion events that last for a longer range can be modelled using additional HMMs.

As mentioned before, our defect detection approach can be generalised as a novelty detection framework with applications in other areas, such as detecting vertical line scratches by using spatial pixel sequences parallel to the horizon instead of temporal ones and detecting tool breakage in an end milling operation by observing the cutting force signal.

- ***False Alarm Elimination:*** Our two-stage false alarm elimination method was designed based on two characteristics that were discovered during the investigation of the false detections from HAFID. A further understanding of the characteristics of false alarms is necessary to give a more generalised categorization of false alarms, where the investigation should not be limited to results from our HAFID detector. Additionally, the accuracy of our proposed elimination method shall be evaluated in a comparison with other false alarms elimination methods.

As previously mentioned, false alarms caused by shot transitions can be prevented by including a video segmentation process into our archive film restoration system. Hence, a film will be separated into a number of clips, each of which only contains one scene.

- ***Defect Removal:*** our proposed defect removal method fails to restore some pixels on image structures, which was easier to observe when it was applied in image inpainting. Higher accuracy can be achieved through a better optimized selection in the search-and-replace procedure. The current optimal pixel-exemplar was chosen based on the statistics generated during the random walk by computing the pairwise similarities of pixels inside the region of candidates. However, propagating image structures requires emphasises on the correlation between the optimal replacement pixel and the degraded pixel's neighbours. Hence, we can improve the restoration accuracy if we additionally measured the similarities of image features between the pixel-exemplar candidates and the degraded pixel's neighbours.

The proposed 3-D random walk based search can be useful in some computer vision applications, such as finding corresponding pixels in tracking and image registration, creating a support region to compute key point descriptions and so on. However, it was limited by its high computational cost. A possible solution will be to perform fast random walks [89, 100] to reduce the number of random walks being performed. Another choice will be to study the eigenvalues of the random walk transition matrix computed in a local neighbourhood instead of in the entire image.

- ***Other types of defects:*** as mentioned in Chapter 4, the proposed HAFID detector fails to respond to defective pixels caused by other types of defects,

such as continuous vertical line scratches. Therefore, one of the future works involves developing specified solutions to handle other defects (discussed in Chapter 3) so as to further replace corresponding manual processes in the automatic archive film restoration system that we outlined in Chapter 1.

Bibliography

- [1] H. Ammar-Badri and A. Benazza-Benyahia. Wavelet-based blotch detection in old movies. *Image Processing Theory, Tool and Applications*, pages 1–7, 2008.
- [2] N. Arica and F. Vural. A shape descriptor based on circular hidden markov model. In *Proceedings of International Conference on Pattern Recognition*, volume 1, pages 924–927, 2000.
- [3] L. Atzori and F. De-Natale. Error concealment in video transmission over packet networks by a sketch-based approach. *Signal Processing: Image Communication*, 15(1-2):57–76, 1999.
- [4] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transaction on Image Processing*, 10:1200–1211, 2001.
- [5] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.
- [6] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Proceedings of SIGGRAPH*, pages 417–424, 2000.
- [7] M. Bertalmio, A. Bertozzi, and G. Sapiro. Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 355–362, 2001.
- [8] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. Simultaneous structure and texture image inpainting. *IEEE Transaction on Image Processing*, 12(8):882–889, 2003.
- [9] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society Series B*, 48:259–302, 1986.

- [10] J. Biemond, P. M. B. van Roosmalen, and R. L. Lagendijk. Improved blotch detection by postprocessing. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 3101–3104, 1999.
- [11] R. Bornard. *Probabilistic approaches for the digital restoration of television archives*. PhD thesis, Ecole Centrale Paris, 2002.
- [12] R. Bornard, E. Lecan, L. Laborelli, and J.-H. Chenot. Missing data correction in still images and image sequences. In *Proceedings of the tenth ACM international conference on Multimedia*, MULTIMEDIA, pages 355–361, 2002.
- [13] J. Y. Bouguet. Pyramidal implementation of the lucas kanade feature tracker. Technical report, OpenCV Documentation, Intel Corporation, Miroprocessor Research Labs, 2000.
- [14] BRAVA. <http://brava.ina.fr>.
- [15] O. Buisson, B. Besserer, S. Boukir, and F. Helt. Deterioration detection for digital film restoration. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 78–84, 1997.
- [16] J. Cai. Constrained circular hidden markov models for recognizing deformed shapes. In *International Conference on Computational Intelligence for Modelling, Control and Automation*, page 40, 2006.
- [17] T. F. Chan and J. Shen. Non-texture inpainting by curvature driven diffusions (cdd). *Journal of Visual Communication and Image Representation*, 12(4):436–449, 2001.
- [18] T. F. Chan and J. Shen. Mathematical models for local nontexture inpaintings. *SIAM Journal on Applied Mathematics*, 62(3):1019–1043, 2001.
- [19] T. F. Chan and J. Shen. *Image Processing and Analysis*. SIAM, Philadelphia, 2005.
- [20] T. F. Chan, S. H. Kang, and J. Shen. Euler’s elastica and curvature driven diffusions. *SIAM Journal of Applied Mathematics*, 2:564–592, 2002.
- [21] M. Chong and D. Krishnan. An edge-preserving MRF model for the detection of missing data in image sequences. *IEEE Transaction on Signal Processing Letters*, 5(4):81–83, 1998.
- [22] D. Corrigan. *Motion Estimation Reliability and the Digital Restoration of Archived Media*. PhD thesis, Trinity College Dublin, 2008.
- [23] D. Corrigan, N. Harte, and A. Kokaram. Pathological motion detection for robust missing data treatment in degraded archived media. In *Proceedings of IEEE International Conference on Image Processing*, pages 621–624, 2006.

- [24] A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transaction on Image Processing*, 13(9):1–13, 2004.
- [25] A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of SIGGRAPH*, pages 341–346, 2001.
- [26] A. Efros and T. Leung. Texture synthesis by non-parametric sampling. In *Proceedings of IEEE International Conference on Computer Vision*, volume 2, pages 1033–1038, 1999.
- [27] S. Esedoglu and J. Shen. Digital inpainting based on the mumford-shah-euler image model. *European Journal of Applied Mathematics*, 13(4):353–370, 2002.
- [28] F. Estrada, A. Jepson, and C. Chennubhotla. Spectral embedding and min-cut for image segmentation. In *Proceedings of British Machine Vision Conference*, pages 317–326, 2004.
- [29] F. Estrada, D. Fleet, and A. Jepson. Stochastic image denoising. In *Proceedings of British Machine Vision Conference*, 2009.
- [30] E. Ferrandière. *Motion Picture Restoration Using Morphological Tools*. Norwell, MA: Kluwer, 1999.
- [31] E. Ferrandière. *Mathematical Morphology and Motion Picture Restoration*. New York: Wiley, 2001.
- [32] E. Ferrandière and J. Serra. Detection of local defects in old motion pictures. *Proceedings of VII National Symposium on Pattern Recognition and Image Analysis*, pages 145–150, 1997.
- [33] M. Fornasier. Nonlinear projection recovery in digital inpainting for color image restoration. *Journal of Mathematical Imaging and Vision*, 24(3):359–373, 2006.
- [34] A. Gangal and B. Dizdaroglu. Automatic restoration of old motion picture films using spatiotemporal exemplar-based inpainting. In *Proceedings of conference on Advanced Concepts for Intelligent Visual System*, pages 55–66, 2006.
- [35] A. Gangal, T. Kayikcioglu, and B. Dizdaroglu. An improved motion-compensated restoration method for damaged color motion films. *Signal Processing: Image Communication*, 19(4):353–368, 2004.
- [36] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

- [37] L. Grady. Random walk for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.
- [38] L. Grady, Y. Sun, and J. Williams. Interactive graph-based segmentation methods in cardiovascular imaging. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Handbook of Mathematical Models in Computer Vision*, pages 453–469. Springer, 2006.
- [39] H. Grossauer and O. Scherzer. Using the complex ginzburg-landau equation for digital inpainting in 2d and 3d. In *Scale Space Methods in Computer Vision*, volume 2695 of *Lecture Notes in Computer Science*, pages 225–236. 2003.
- [40] M. Hamid and S. Marshall. Fpga realisation of the genetic algorithm for the design of grey-scale soft morphological filters. In *Proceedings of International Conference on Visual Information Engineering*, pages 141–144, 2003.
- [41] M. Hamid, N. Harvey, and S. Marshall. Genetic algorithm optimization of multidimensional grayscale soft morphological filters with applications in film archive restoration. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(5):406–416, 2003.
- [42] R. Hardie and C. Boncelet. Lum filters: a class of rank-order-based filters for smoothing and sharpening. *IEEE Transactions on Signal Processing*, 41(3):1061–1076, 1993.
- [43] N. Harvey and S. Marshall. Film restoration using soft morphological filters. In *Proceedings of International Conference on Image Processing and Its Application*, volume 1, pages 279–282, 1997.
- [44] D. J. Heeger and J. R. Bergen. Pyramid-based texture analysis/synthesis. In *Proceedings of SIGGRAPH*, pages 229–238, 1995.
- [45] K. Hough and S. Marshall. Soft morphological filters applied to the removal of noise from cctv footage. In *Proceedings of IEE International Symposium on Imaging for Crime Detection and Prevention*, pages 61–66, 2005.
- [46] J. Hu, M. Brown, and W. Turin. Hmm based online handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):1039–1045, 1996.
- [47] H. Igehy and L. Pereira. Image replacement through texture synthesis. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 186–189, 1997.
- [48] Internet Archive. <http://www.archive.org>.

- [49] F. Jelinek. Continuous speech recognition by statistical methods. In *IEEE Proceedings*, volume 64, pages 532–536, 1976.
- [50] B. Kent, A. Kokaram, B. Collis, and S. Robinson. Two layer segmentation for handling pathological motion in degraded post production media. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 299–302, 2004.
- [51] A. Kokaram. Practical, unified, motion and missing data treatment in degraded video. *Journal of Mathematical Imaging and Vision*, 20(1):163–177, 2004.
- [52] A. Kokaram. On missing data treatment for degraded video and film archives: a survey and a new bayesian approach. *IEEE Transactions on Image Processing*, 13(3):397–415, 2004.
- [53] A. Kokaram and S. Godsill. MCMC for joint noise reduction and missing data treatment in degraded video. *IEEE Transactions on Signal Processing*, 50(2):189–205, 2002.
- [54] A. Kokaram and P. Rayner. System for the removal of impulsive noise in image sequences. *Journal of Visual Communication and Image Representation*, 1818(1):322–331, 1992.
- [55] A. Kokaram, R. Morris, W. Fitzgerald, and P. Rayner. Detection of missing data in image sequences. *IEEE Transactions on Image Processing*, 4(11):1496–1508, 1995.
- [56] A. Kokaram, R. Morris, W. Fitzgerald, and P. Rayner. Interpolation of missing data in image sequences. *IEEE Transaction on Image Processing*, 4(11):1509–1519, 1995.
- [57] A. Levin, A. Zomet, and Y. Weiss. Learning how to inpaint from global image statistics. In *Proceedings of IEEE International Conference on Computer Vision*, volume 1, pages 305–312, 2003.
- [58] A. Licsar, L. Czuni, and T. Sziranyi. Trainable postprocessing method to reduce false alarms in the detection of small blotches of archive films. In *Proceedings of IEEE International Conference on Image Processing*, pages 562–565, 2005.
- [59] A. Licsar, L. Czuni, and T. Sziranyi. Trainable blotch detection on high resolution archive films minimizing the human interaction. *Machine Vision and Application*, 21:767–777, 2010.

- [60] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of International Conference on Artificial Intelligence*, pages 674–679, 1981.
- [61] R. S. Mamon and R. J. Elliott. *Hidden Markov Models in Finance*. Springer US, 2007.
- [62] M. Markou and S. Singh. Noveltydetection: a review – part 1: statistical approaches. *IEEE Transactions on Signal Processing*, 83:2481–2497, 2003.
- [63] M. Markou and S. Singh. Noveltydetection: a review – part 2: neural network based approaches. *IEEE Transactions on Signal Processing*, 83:2499–2521, 2003.
- [64] S. Masnou. Disocclusion: a variational approach using level lines. *IEEE Transaction on Image Processing*, 11(2):68–76, 2002.
- [65] S. Masnou and J.-M. Morel. Level lines based disocclusion. In *Proceedings of IEEE International Conference on Image Processing*, volume 3, pages 259–263, 1998.
- [66] M. Meila and J. Shi. Learning segmentation by random walks. In *Proceedings of Neural Information Processing Systems*, pages 873–879, 2000.
- [67] R. Morris. *Image Sequence Restoration using Gibbs Distributions*. PhD thesis, Cambridge University, 1995.
- [68] M. Nadenau and S. Mitra. Blotch and scratch detection in image sequences based on rank ordered differences. In *Proceedings of International Workshop on Time-Varying Image Processing and Moving Object Recognition*, pages 1–7, 1997.
- [69] M. Nitzberg, D. Mumford, and T. Shiota. *Filtering, Segmentation and Depth*. Lecture Notes in Computer Science, volume 662, Springer-Verlag, Berlin, 1993.
- [70] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
- [71] K. Patwardhan and G. Sapiro. Projection based image and video inpainting using wavelets. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 857–860, 2003.
- [72] K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting of occluding and occluded objects. In *Proceedings of IEEE International Conference on Image Processing*, volume 2, pages 69–72, 2005.

- [73] K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting under constrained camera motion. In *Proceedings of IEEE International Conference on Image Processing*, volume 16, pages 545–553, 2007.
- [74] S. Porter. *Video Segmentation and Indexing using Motion Estimation*. PhD thesis, University of Bristol, 2004.
- [75] PrestoSpace. <http://prestospace.org>.
- [76] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *IEEE Proceedings*, 77(2):257–286, 1989.
- [77] L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE Acoustics, Speech, and Signal Processing Magazine*, 3(1):4–16, 1986.
- [78] A. Rares. *Archive Film Analysis and Restoration*. PhD thesis, Technische Universiteit Delft, 2004.
- [79] A. Rares, M. Reinders, and J. Biemond. Statistical analysis of pathological motion areas. In *IEE seminar on Digital Restoration of Film and Video Archives*, volume 8, pages 1–28, 2001.
- [80] A. Rares, M. Reinders, and J. Biemond. Complex event classification in degraded image sequences. In *Proceedings of IEEE International Conference on Image Processing*, volume 1, pages 253–256, 2001.
- [81] A. Rares, M. Reinders, and J. Biemond. Edge-based image restoration. *IEEE Transaction on Image Processing*, 14(10):1458–1464, 2005.
- [82] P. Read and M. Meyer. *Restoration of Motion Picture Film*. Butterworth Heinemann, 2000.
- [83] J. Ren and T. Vlachos. Segmentation-assisted dirt detection for the restoration of archived films. In *Proceedings of British Machine Vision Conference*, volume I, pages 359–368, 2005.
- [84] J. Ren and T. Vlachos. Efficient detection of temporally impulsive dirt impairments in archived films. *IEEE Transactions on Signal Processing*, 87(3): 541–551, 2007.
- [85] B. D. Ripley. *Statistical inference for spatial processes*. Cambridge University Press, 1988.
- [86] P. V. Roosmalen. *Restoration of Archived Film and Video*. PhD thesis, Delft University, 1999.

- [87] P. V. M. Roosmalen, A. Kokaram, and J. Biemond. Fast high quality interpolation of missing data in image sequences using a controlled pasting scheme. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 3105–3108, 1999.
- [88] M. Rubinstein and R. H. Colby. *Polymer Physics*. Oxford University Press, 2003.
- [89] A. Sarma, D. Nanongkai, and G. Pandurangan. Fast distributed random walks. In *Proceedings of ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, 2009.
- [90] P. Schallauer, A. Pinz, and W. Haas. Automatic restoration algorithms for 35 mm film. *Journal of Computer Visual Research*, 1(2):60–85, 1999.
- [91] Snell Group. Archangel ph.c-hd, <http://www.snellgroup.com/products/conversion-and-restoration/restoration>.
- [92] S. R. Sternberg. Grayscale morphology. *Computer Vision, Graphic and Image Processing*, 35(3):333–355, 1986. doi: [http://dx.doi.org/10.1016/0734-189X\(86\)90004-6](http://dx.doi.org/10.1016/0734-189X(86)90004-6).
- [93] R. Storey. Electronic detection and concealment of film dirt. *Journal of Society of Motion Picture and Television Engineers*, pages 642–647, 1985.
- [94] L. Tenze, G. Ramponi, and S. Carrato. Blotches correction and contrast enhancement for old film pictures. *Proceedings of IEEE International Conference on Image Processing*, 2000.
- [95] L. Tenze, G. Ramponi, and S. Carrato. Robust detection and correction of blotches in old films using spatiotemporal information. *SPIE Image Processing: Algorithms and Systems*, pages 348–357, 2000.
- [96] Teranex VC300 platform. <http://www.teranex.com/products/vc300>.
- [97] S. Tilie, L. Laborelli, and I. Bloch. Blotch detection for digital archives restoration based on the fusion of spatial and temporal detectors. In *Proceedings of International Conference on Information Fusion*, pages 1–8, 2006.
- [98] S. Tilie, L. Laborelli, and I. Bloch. A contrario false alarms removal for improving blotch detection in digitized films restoration. In *Proceedings of European Conference on Signals and Image Processing*, pages 410–413, 2007.
- [99] S. Tilie, L. Laborelli, and I. Bloch. Fusion of complementary detectors for improving blotch detection in digitized films. *Pattern Recognition letters*, 27: 1735–1746, 2007.

-
- [100] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast randomwalk with restart and its applications. In *Proceedings of International Conference on Data Mining*, 2006.
 - [101] A. Tsai, A. Yezzi, and A. Willsky. Curve evolution implementation of the mumford-shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Transaction on Image Processing*, 10:1169–1186, 2001.
 - [102] L. Vincent. Morphological gray scale reconstruction in image analysis: Applications and efficient algorithms. *IEEE Transaction on Image Processing*, (4): 176–201, 1993.
 - [103] L. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *Proceedings of SIGGRAPH*, pages 479–488, 2000.
 - [104] S. Wei, R. Zhang, P. Hao, and Y. Ding. Blotch detection based on texture matching and adaptive multi-threshold. *Proceedings of International Conference on Image and Graphics*, 0:256–261, 2009.
 - [105] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 120–127, 2004.
 - [106] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):463–476, 2007.
 - [107] M. Wloka and R. Zeleznik. Interactive real-time motion blur. *Visual Computer*, 12:283–294, 1996.
 - [108] Xiph.org Test Media. <http://media.xiph.org/video/derf/>.

RWDR: Application to Image and Video Inpainting

In this appendix, we demonstrate the application of our proposed defect removal approach to image and video inpainting. As mentioned in Section 2.4, defect removal is essentially a task of filling in missing regions. It is straightforward to apply our method to inpainting problems using handlabelled mask instead of defect maps. In the following sections, we will discuss the applications in detail and give some sample results along with comparative results from both image structure propagation based [6] and exemplar based [24] inpainting methods.

The term “inpainting” originates from painting restoration in the world of Art (see an example in Figure A.1). The original idea of artistic inpainting is to smoothly propagate the surrounding colour and structure into the missing area iteratively. Like [6], the proposed approach also adapted this idea by introducing the multiscale algorithm and reliability values to enforce a (large) degraded region to implode gradually through the propagation of reliable outer pixels into the region.



Figure A.1: A painting restored by our proposed method is shown with original images on the left, masks in the middle and results on the right.

Image Inpainting

In the case of image inpainting, the random walking is limited to the spatial domain. Instead of 26 spatiotemporal neighbours, a pixel can only transit to one of its 8 spatial neighbours in each random walk step. The spatial-only regions of candidate pixel-exemplars are generated much faster than using 3-D random walks. Figure A.2 shows two sample image inpainting results from our proposed algorithm. In both examples, the inpainting regions are small and thin so that our proposed method is able to achieve subjectively good restoration of these regions.

Figure A.3 gives an example of filling in a large missing region. The inpainting result from our proposed method are compared with two landmark works, *i.e.* image structure propagation based method [6] and exemplar based method [24]. It is truly difficult to measure the quality of restoration when we restore large missing regions. Subjectively speaking, results from the exemplar based method [24] is more reasonable in this case since the house is better restored, but quite a few artifacts are also introduced in some parts of the image, *e.g.* trees on the roof and lake. For our proposed method, the inpainting result is rather plain and lacks of textures. This is because patch-exemplar based methods perform more effectively in propagating non-stochastic (regular or structured) textures than pixel-exemplar based methods and



Figure A.2: Two examples of image inpainting are shown with the original image on the top, inpainting mask in the middle and restored image at the bottom.

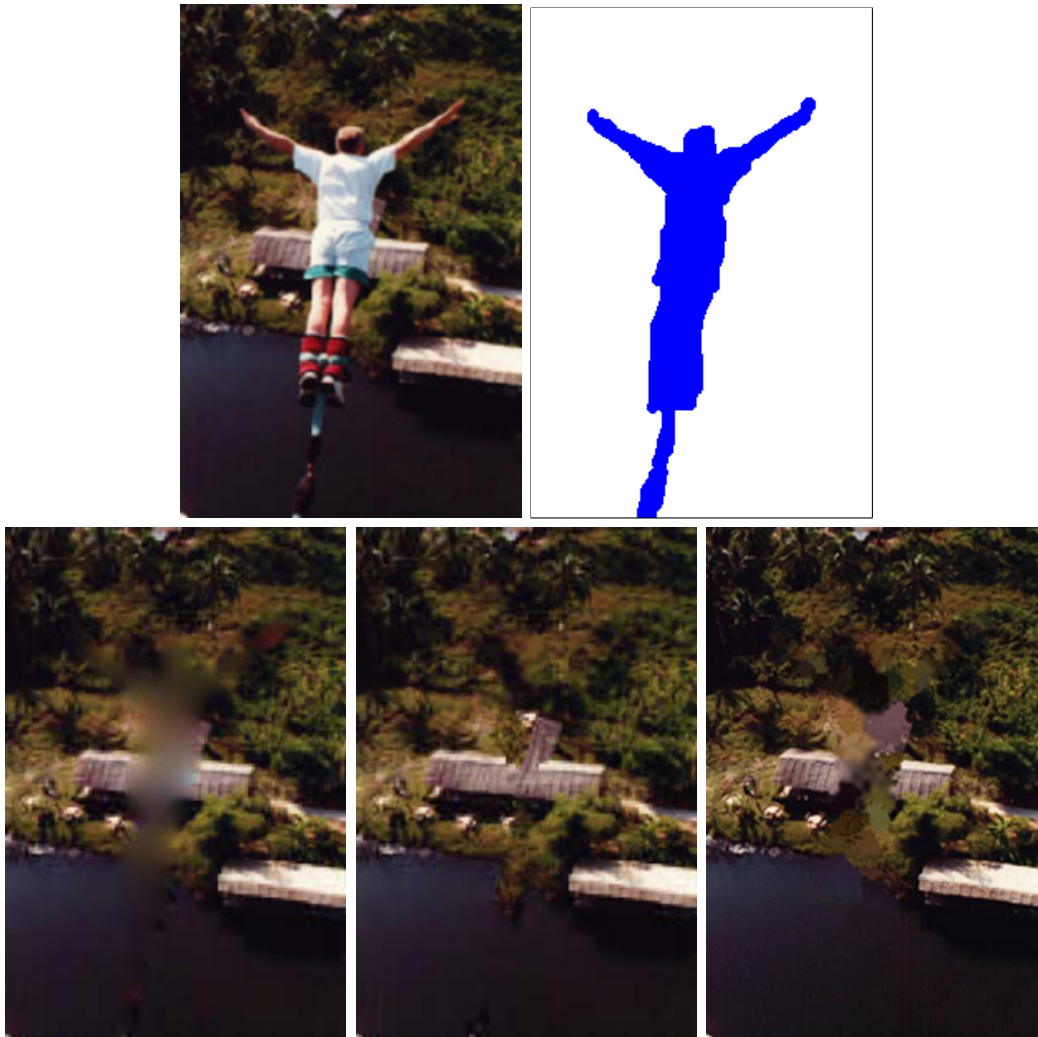


Figure A.3: Bungee - Top: Original image and inpainting mask; Bottom: (left to right) inpainting results from structure propagation based method [6], exemplar based method [24] and our proposed method.

the texture feature (LBP) used in our method is limited to a 3×3 neighbourhood. Furthermore, our approach is designed to propagate spatiotemporal neighbouring pixel-exemplars into the missing region where the temporal neighbours play a more important role in restoring the texture information while spatial neighbours mostly help propagate image structures into the region.

Video Inpainting

Our proposed defect removal method can be directly applied to video inpainting by using the handlabelled masks instead. Figure A.4 and A.5 illustrate two examples with five consecutive frames extracted from the inpainting image sequences. In a similar sense as the applications demonstrated here, the proposed method may also be useful in some film post production processes, *e.g.* removing distractions or any unwanted objects.

Figure A.4 shows sample frames from *Container*, which is a real sequence with a static background and slow motions of the container ship in the front. Our target here is to remove distractions (two swiftly crossing birds). The proposed method is able to properly restore the texture details (sea waves) and the occluded object (the flagpole in the last frame) on the missing regions left from removing the birds.

In Figure A.5, we illustrate a more challenging example of video inpainting with sample frames from an image sequence we call *stadium*. Three layers of motions can be observed in *stadium*, *i.e.* slow motions in the background (the camera pans towards left), relatively faster motions of the fence and fast motions of the athlete. In this case, we would like to remove the running athlete from the scene to draw more attention on the people sitting on the bench. The proposed approach provides reasonable restoration of the inpainting regions but some blurring-like artifacts are also produced over the fence (in the first frame) and man’s head (in the last three frames). Due to the complex motions in this image sequences, incorrectly restored motion information further causes these blurring-like artifacts.

In this appendix, we illustrated the application of our proposed defect removal approach in inpainting. When applied in image inpainting, our method was not good

at propagating image structure information into large missing regions, which leads to a segmentation-like side effect in the inpainted regions. It also indicated that the proposed defect removal method mostly relied on the temporal information that existed in the image sequences.

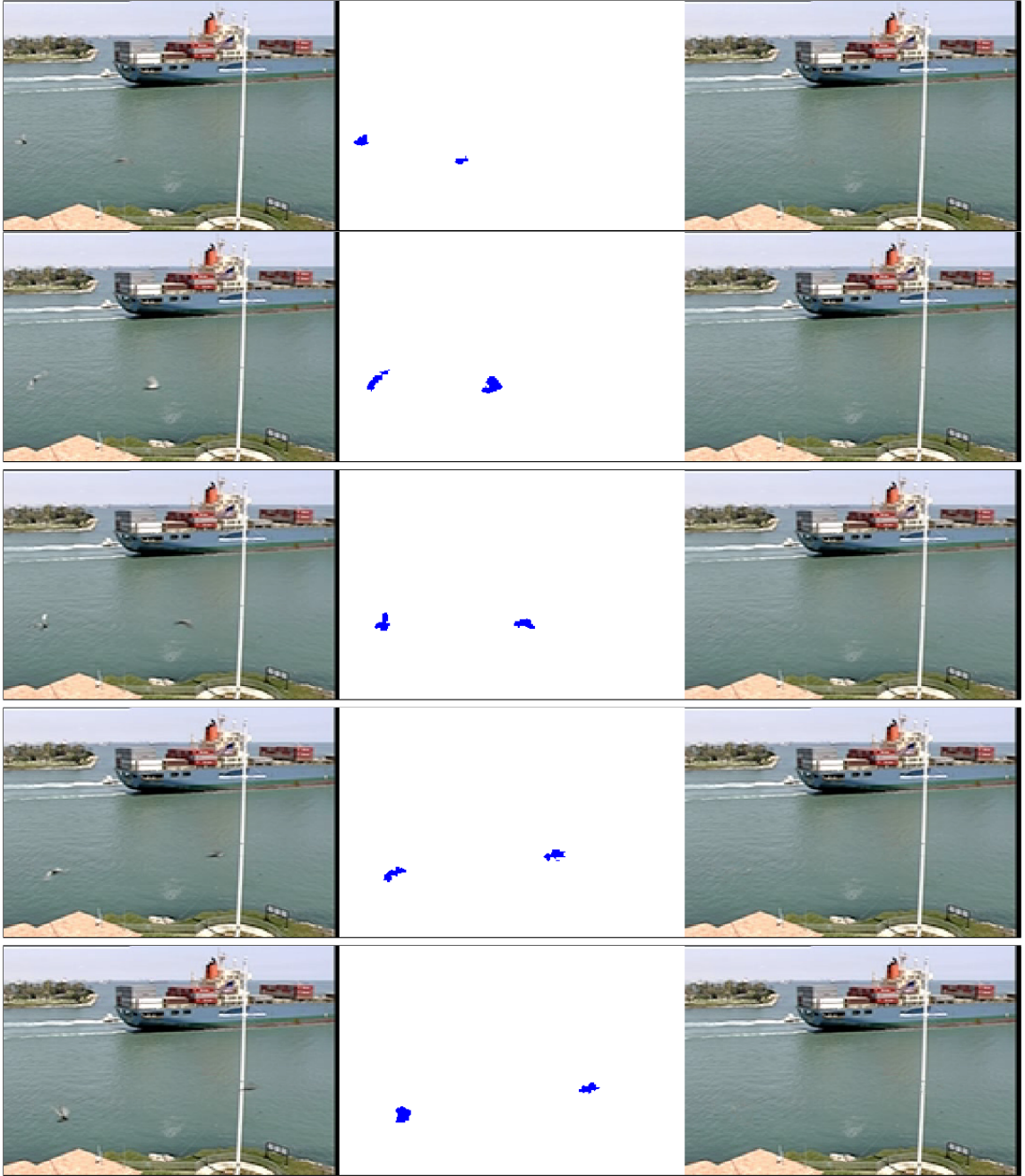


Figure A.4: Container - Inpainting results from 5 consecutive frames from top to bottom. Original frames are shown on the left, mask in the middle and results on the right.

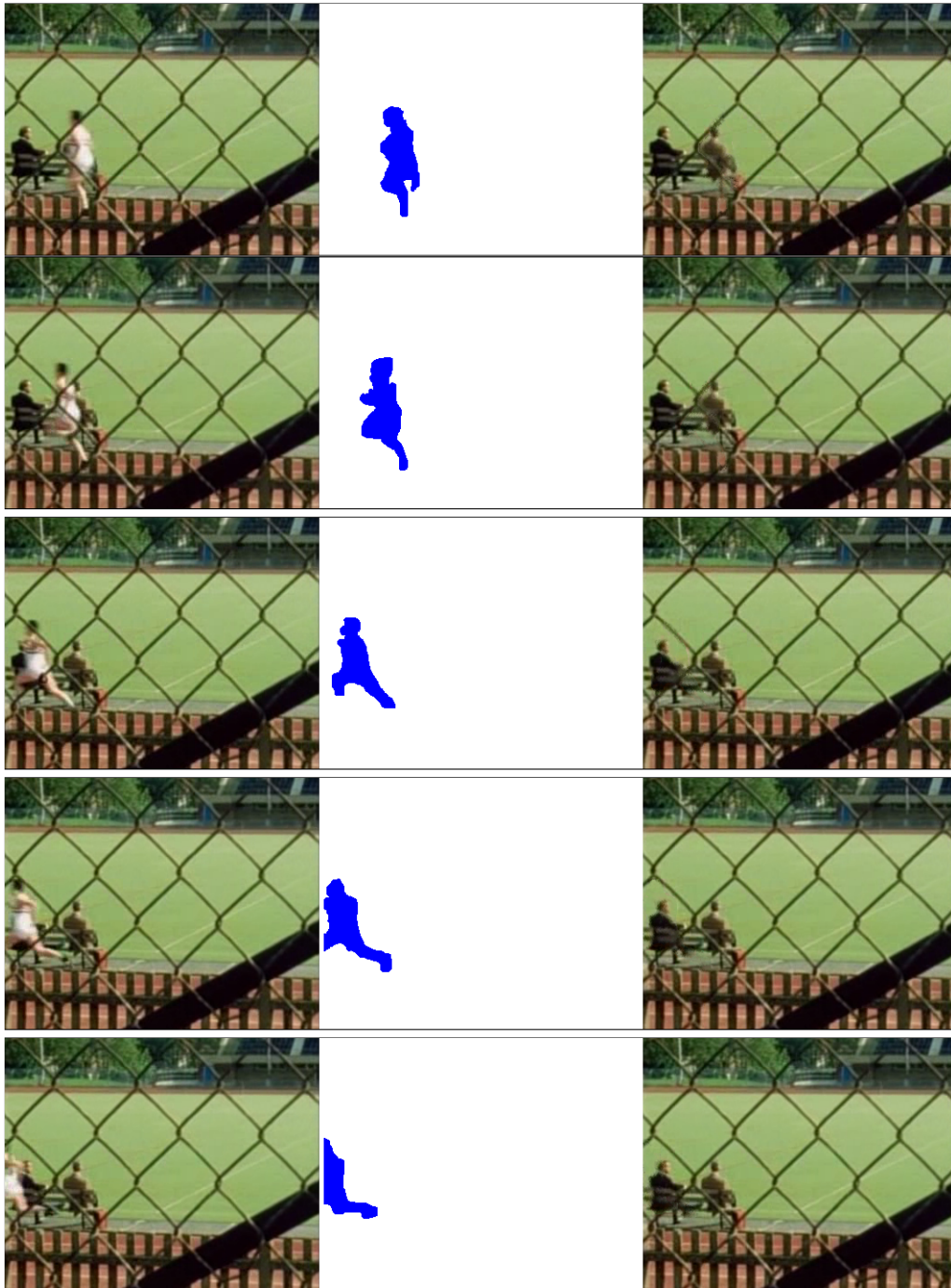


Figure A.5: Stadium - Inpainting results from 5 consecutive frames from top to bottom. Original frames are shown on the left, mask in the middle and results on the right.