ÉCOLE CENTRALE DES ARTS
ET MANUFACTURES
« ÉCOLE CENTRALE PARIS »

Thèse

présentée par

Raphaël BORNARD

pour obtenir le grade de

Docteur de l'École Centrale Paris

Spécialité : Mathématiques Appliquées

APPROCHES PROBABILISTES APPLIQUÉES À LA RESTAURATION NUMÉRIQUE D'ARCHIVES TÉLÉVISÉES

soutenue le 27 novembre 2002, devant le jury composé de :

M. Rachid	DERICHE	INRIA Sophia-Antipolis	Président
M. Patrick	BOUTHEMY	IRISA/INRIA Rennes	Rapporteurs
M. Anil	KOKARAM	Trinity College Dublin	
M. Jean-Hugues	CHENOT	INA	Examinateurs
M. Patrick	PÉREZ	Microsoft Research Cambridge	
M. Christian	SAGUEZ	École Centrale Paris	
M. Bruno	DESPAS	Centrimage	Membre invité

2002 - 32

ÉCOLE CENTRALE DES ARTS
ET MANUFACTURES
« ÉCOLE CENTRALE PARIS »

Thèse

présentée par

Raphaël BORNARD

pour obtenir le grade de

Docteur de l'École Centrale Paris

Spécialité : Mathématiques Appliquées

APPROCHES PROBABILISTES APPLIQUÉES À LA RESTAURATION NUMÉRIQUE D'ARCHIVES TÉLÉVISÉES

soutenue le 27 novembre 2002, devant le jury composé de :

M. Rachid	DERICHE	INRIA Sophia-Antipolis	Président
M. Patrick	BOUTHEMY	IRISA/INRIA Rennes	Rapporteurs
M. Anil	KOKARAM	Trinity College Dublin	
M. Jean-Hugues	CHENOT	INA	Examinateurs
M. Patrick	PÉREZ	Microsoft Research Cambridge	
M. Christian	SAGUEZ	École Centrale Paris	
M. Bruno	DESPAS	Centrimage	Membre invité

2002 - 32

ÉCOLE CENTRALE DES ARTS
ET MANUFACTURES
“ÉCOLE CENTRALE PARIS”

Thesis

submitted by

Raphaël BORNARD

for the degree of

Doctor of Philosophy of École Centrale Paris

Speciality: Applied Mathematics

PROBABILISTIC APPROACHES FOR THE DIGITAL RESTORATION OF TELEVISION ARCHIVES

defended on November 27th, 2002, before the jury composed of:

Dr. Rachid	DERICHE	INRIA Sophia-Antipolis	President
Dr. Patrick	BOUTHEMY	IRISA/INRIA Rennes	Assessors
Dr. Anil	KOKARAM	Trinity College Dublin	
Mr. Jean-Hugues	CHENOT	INA	Examiners
Dr. Patrick	PÉREZ	Microsoft Research Cambridge	
Dr. Christian	SAGUEZ	École Centrale Paris	
Mr. Bruno	DESPAS	Centrimage	Invited member

2002 - 32

Remerciements

Le travail de thèse présenté ici a été financé par l'INA et l'ANRT (Association Nationale de la Recherche Technique), que je remercie, dans le cadre d'une convention CIFRE (Convention Industrielle de Formation par la Recherche en Entreprise).

Je tiens à exprimer tous mes remerciements aux membres du jury de soutenance :

Rachid Deriche pour m'avoir fait l'honneur de présider ce jury,

Patrick Bouthemy pour l'intérêt qu'il a porté à mon travail et pour avoir accepté la tâche de rapporteur,

Anil Kokaram pour avoir bien voulu expérimenter le système doctoral français et pour le réel plaisir que ce fut de travailler avec lui dans le cadre du projet BRAVA,

Jean-Hugues Chenot pour sa disponibilité et son précieux soutien sur tous les aspects (tant scientifique et technique qu'administratif et informatique) durant ces trois années, pour son temps consenti à de nombreuses relectures et pour avoir toujours su suggérer (judicieusement) sans imposer,

Patrick Pérez pour son enthousiasme communicatif, ses relectures rigoureuses et les nombreux échanges enrichissants et stimulants dont je lui suis redevable,

Christian Saguez pour avoir accepté d'être mon directeur de thèse et pour ses conseils,

Bruno Despas pour sa participation enthousiaste et éclairée au jury.

J'adresse également mes plus vifs remerciements pour leur aide à Louis Laborelli (INA) pour son encadrement scientifique et nos nombreuses discussions sur des sujets divers et variés, et à Emmanuelle Lecan pour sa contribution sur l'étape de correction.

Outre les personnes mentionnées précédemment, je suis reconnaissant à tous ceux qui m'ont fait également des relectures attentives et des suggestions utiles, en particulier Philippe Poncin, Laurent Vinet, Frédéric Dumas, Olivier Buisson, Guillaume Picard (INA), Paul-Henry Cournède (ECP), sans oublier la relecture complète de Pierre Bornard. Tous mes remerciements également à Alain Perrier (INA) pour son aide pour la réalisation des MPEG, ainsi qu'à Richard Wright et à la BBC pour l'autorisation d'utiliser bon nombre d'images et de séquences.

J'aimerais enfin saluer amicalement mon collègue de bureau Guillaume Picard, Jean-Etienne Noiré, Matthieu Fleurmont et mes collègues de l'INA, Justine Grave, Andrei Rareş (TU Delft), Denis Sidorov (Trinity College Dublin et Irkutsk State University), Julien Buffet et Lionel Gabet (ECP).

Acknowledgements

The thesis work presented here has been funded thanks to INA and ANRT (National Association for Technical Research) through a CIFRE grant (Industrial Convention for Education through Corporate Research).

I would like to express all my gratitude to the members of the jury:

Rachid Deriche for doing me the honour to serve as president of this jury,

Patrick Bouthemy for the interest he has shown in my work and for having accepted the task of assessing it,

Anil Kokaram for his willingness in experimenting the French doctorate system and for the genuine pleasure it has been to work with him within the framework of the BRAVA project,

Jean-Hugues Chenot for his availability and his valuable support on all aspects (scientific and technical as well as on administrative and computing issues) during these three years, for the time he devoted to much proof-reading and for having always been able to (judiciously) suggest without imposing,

Patrick Pérez for his infectious enthusiasm, his rigorous proof-reading and the many enriching and thought-provoking exchanges for which I am indebted to him,

Christian Saguez for having accepted to be my academic supervisor and for his advice,

Bruno Despas for his enthusiastic and enlightened participation to the jury.

I also owe a great deal for their help to Louis Laborelli (INA) for his scientific supervision and for many discussions on various subjects, and to Emmanuelle Lecan for her contribution on the correction step.

In addition to all persons previously mentioned, I am grateful to all those who helped me with careful proof-reading and useful suggestions, in particular Philippe Poncin, Laurent Vinet, Frédéric Dumas, Olivier Buisson, Guillaume Picard (INA), Paul-Henry Cournède (ECP), not to forget Pierre Bornard. All my thanks as well to Alain Perrier (INA) for his help on making the MPEGs and to Richard Wright and the BBC for giving me the authorisation to use a number of images and sequences.

I would finally like to send my friendly greetings to my roommate Guillaume Picard, Jean-Etienne Noiré, Matthieu Fleurmont and my colleagues from INA, Justine Grave, Andrei Rareş (TU Delft), Denis Sidorov (Trinity College Dublin and Irkutsk State University), Julien Buffet and Lionel Gabet (ECP).

Contents

List of Abbreviations

2AFC	Two-Alternative Forced Choice
AR	Autoregressive
BBC	British Broadcasting Corporation
BN	Bayesian/Belief Network
CCIR	Comité Consultatif International des Radiocommunications
CSF	Contrast Sensitivity Function
DCT	Discrete Cosine Transform
DFD	Displaced Frame Difference
DOC	DropOut Compensator
DSCQS	Double Stimulus Continuous Quality Scale
DV	Digital Video
GRF	Gibbs Random Field
HD	High Definition
HVS	Human Visual System
ICM	Iterated Conditional Modes
INA	Institut National de l'Audiovisuel
ITU	International Telecommunication Union
JOMBADI	JOint Model BAsed Detection and Interpolation
MAP	Maximum A Posteriori
MCMC	Markov Chain Monte Carlo
MF	Mean Field
MMF	Multistage Median Filter
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
MPM	Maximizer of the Posterior Marginals
MRF	Markov Random Field
MSE	Mean Squared Error
NTSC	National Television System Committee
OS	Order Statistic
PAL	Phase Alternating Line
PDE	Partial Differential Equation
pdf	probability density function
PM	Pathological Motion
PSNR	Peak Signal-to-Noise Ratio
RF	Random Field
RGB	Red Green Blue
RMSE	Root Mean Squared Error
ROC	Receiver Operating Characteristic
RT	Reaction Time
SA	Simulated Annealing
SDI	Spike Detection Index
SECAM	SEquentiel Couleur A Mémoire
TBC	Time-Base Corrector
TV	Television
VQEG	Video Quality Experts Group
VTR	Video Tape Recorder

Chapter 1

Introduction

1.1 Context

Beyond their historical or cultural interest, television archives are potentially valuable through commercial re-exploitation by today's broadcasters. The demand in this direction has noticeably increased with the advent of new digital media (cable, satellite, DVD) related to the MPEG-2 standard. But it is coupled with higher and higher expectations concerning the visual quality of the documents.

However, due to ageing and/or to early technical limitations, archives are generally affected by a number of deteriorations and are not in a condition that enables to re-use them directly for broadcast. Therefore after being transferred onto a digital medium, they need to be restored. This process is long and costly and may take up to several tens of hours of work per hour of programme. That is why more **automation** is essential for significant cost reduction. This automation is to be partly achieved by the development of specific processing algorithms.

Whereas in most countries, individual broadcasters are in charge of their own archives, this task has been assigned in France to a state-owned company (or Etablissement Public à caractère Industriel et Commercial, EPIC) originally created for this purpose, namely Institut National de l'Audiovisuel (INA). Set up in 1975, INA is one of the components resulting from the splitting of the former state broadcast monopoly. Among other missions established by law, it is responsible for the collection, preservation and re-exploitation of the national audiovisual heritage. INA holds in particular one of the largest TV archives collections in the world with nearly 500,000 hours of programmes at the end of 2001. As such, it is daily faced with large-scale restoration issues.

Despite earlier isolated works, archives restoration has only started to be acknowledged as a research field of its own since the pioneering work of Kokaram in 1993 [Kok 93]. Because archives are its core business, INA has invested a significant research effort in this domain during the last decade. Collaboration with other partners led to the

Chapitre 1

Introduction

1.1 Contexte

Au-delà de leur intérêt historique ou culturel, les archives télévisées ont potentiellement une grande valeur liée à leur ré-exploitation commerciale par les diffuseurs. La demande en ce sens s'est particulièrement accrue avec l'apparition des nouveaux médias numériques (câble, satellite, DVD) issus de la norme MPEG-2. Mais elle s'accompagne d'exigences de plus en plus élevées quant à la qualité visuelle des documents.

Or, du fait de leur vieillissement et/ou des limitations techniques de l'époque, les archives sont généralement affectées par un certain nombre de dégradations et ne sont pas diffusables en l'état. Elles doivent donc passer, après leur transfert sur support numérique, par une étape de restauration. Ce processus est lent et coûteux et peut prendre jusqu'à plusieurs dizaines d'heures de travail par heure de programme. C'est pourquoi une plus grande **automatisation** est indispensable pour parvenir à une réduction significative des coûts. Cette automatisation passe en partie par le développement d'algorithmes de traitement spécifiques.

Alors que dans la plupart des pays, les diffuseurs sont eux-mêmes responsables de leurs propres archives, cette tâche a été confiée en France à un Etablissement Public à caractère Industriel et Commercial (EPIC) créé à l'origine dans ce but, à savoir l'Institut National de l'Audiovisuel (INA). Né en 1975, l'INA est une des sociétés issues du démantèlement de l'ORTF. Entre autres missions fixées par la loi, l'Institut est chargé de la collecte, de la conservation et de la ré-exploitation du patrimoine audiovisuel national. L'INA détient en particulier l'un des principaux fonds d'archives télévisées au monde avec près de 500.000 heures fin 2001. En tant que tel, il est confronté quotidiennement aux problèmes de restauration à grande échelle.

Malgré quelques travaux ponctuels antérieurs, la restauration d'archives a réellement commencé à s'imposer comme un domaine de recherche à part entière depuis les travaux précurseurs d'Anil Kokaram en 1993 [Kok 93]. Dans la mesure où les archives constituent son cœur de métier, l'INA a consenti des efforts de recherche conséquents dans

European Union's AURORA project (AUtomated Restoration of ORiginal film and video Archives) between 1995 and 1999 [Che 98] and more recently to the BRAVA project (BRoadcast Archives restoration through Video Analysis) from 2000 to 2002. This thesis has partly taken place within the framework of this latter project.

1.2 Contribution of the thesis

Among the wide variety of impairments that can affect archived material, we have chosen to focus on **impulsive defects**. This problem occurs very frequently on film documents and appears as flashing blotches (dirt or gelatine sparkle). Similar artifacts are found to a lesser extent in video: what is referred to as dropout is when one or more lines cannot be read properly. As these artifacts are essentially local both in time and space, they are removed in two steps: **detection** of the damaged pixels and their ensuing **correction** by interpolating missing data. These two steps heavily rely on motion estimation between images in order to take advantage of the temporal redundancy.

In practice, existing detection methods have a major limitation: they are very sensitive to motion estimation failures, which are the source of false alarms. These false alarms have a critical impact on the performance of the overall system: when they are combined with perfectible correction methods which do not always wisely incorporate temporal information, they give rise to the creation of visually disturbing artifacts in regions which were initially free from trouble. A manual intervention is then necessary to avoid damage and the resulting degree of automation is largely insufficient.

To overcome these problems, the key innovation in our approach is to take into account these undesirable failures of motion estimation. They are due to complex natural events which are an integral part of the original document and should therefore be preserved, and yet act as a disturbance for the restoration process. This notion of what we shall call **pathological motion** is at the heart of this thesis. On the one hand, we incorporate it in our detection model by involving a larger temporal window than the usual three frames; this shall allow to distinguish defects from pathological motion, which persist on a longer duration. On the other hand, the proposed correction scheme aims at performing well even in the presence of pathological motion: it attempts to make an "intelligent" use of temporal information and to be able to do without it when it cannot be used reliably. For these two steps, specific attention is devoted to the fulfilment of requirements of efficiency, genericity, robustness, automation and computational speed.

The tools investigated in this thesis to achieve our aims are probabilistic models and more specifically **Markov random fields**. They consist in specifying local probabilistic interactions between neighbouring pixels. They are used explicitly (parametric models) for the proposed detection method within the framework of the Bayesian theory of estimation; they are involved as heuristics derived from the Markovian "philosophy" (non-parametric models) and inspired by recent works on texture synthesis for the developed correction technique. In both cases, the proposed algorithms are compared with the most relevant works in the literature. The overall work detailed in this thesis has been validated ce domaine durant les dix dernières années. La collaboration avec d'autres partenaires a conduit aux projets AURORA (*AUtomated Restoration of ORiginal film and video Archives*) entre 1995 et 1999 [Che 98], puis BRAVA (*BRoadcast Archives restoration through Video Analysis*) de 2000 à 2002, tous deux soutenus par l'Union Européenne. Cette thèse s'est en partie déroulée dans le cadre de ce dernier projet.

1.2 Contribution de la thèse

Parmi la grande variété de défauts qui peuvent affecter les documents archivés, nous avons choisi de nous attaquer aux **défauts impulsifs**. Ce problème est très fréquent sur les documents film et se manifeste par l'apparition fugitive de taches (salissures ou éclats de gélatine). On retrouve dans une moindre mesure des artefacts similaires en vidéo : on parle de "dropout" lorsqu'une ou plusieurs lignes sont mal lues. Dans la mesure où ces artefacts sont très localisés à la fois spatialement et temporellement, leur élimination s'effectue en deux étapes : la **détection** des pixels corrompus, puis leur **correction** en régénérant les données manquantes. Ces deux étapes s'appuient fortement sur l'estimation de mouvement entre les images afin d'exploiter la redondance temporelle.

En pratique, les méthodes de détection existantes présentent une limitation majeure : elles sont particulièrement sensibles aux erreurs d'estimation de mouvement, qui génèrent des fausses alarmes. Ces fausses alarmes ont un impact critique sur les performances du système complet : lorsqu'elles sont combinées avec des méthodes de correction imparfaites, qui ne font pas toujours une utilisation prudente de l'information temporelle, on aboutit à la création d'artefacts visuellement gênants dans des zones qui en étaient initialement dépourvues. Une intervention manuelle est alors nécessaire pour éviter les dégâts et le degré d'automatisation final est largement insuffisant.

Pour surmonter ces problèmes, l'élément clé de notre approche est la prise en considération de ces défaillances indésirables de l'estimation de mouvement. Elles sont dues à des phénomènes naturels complexes qui font partie intégrante du document d'origine et doivent donc être conservés, et qui agissent cependant comme une perturbation vis-à-vis du processus de restauration. Cette notion de ce que nous qualifierons de **mouvement pathologique** est au cœur de cette thèse. D'une part, nous l'incorporons dans notre modèle de détection en prenant en compte une fenêtre temporelle plus large que les trois images habituelles ; on peut ainsi espérer mieux distinguer les défauts des mouvements pathologiques, qui persistent sur une durée plus importante. D'autre part, la stratégie de correction proposée vise à donner des résultats satisfaisants y compris en présence de mouvement pathologique : elle s'efforce de faire une utilisation "intelligente" de l'information temporelle et de savoir s'en passer lorsque celle-ci ne peut pas être utilisée de manière fiable. Pour ces deux étapes, nous portons une attention toute particulière à la satisfaction d'objectifs d'efficacité, de généricité, de robustesse, d'automatisation et de faible temps de calcul.

Les outils explorés dans cette thèse pour parvenir à nos fins sont des modèles probabilistes et plus spécifiquement les **champs de Markov**. Ils consistent en la spécification by the implementation from A to Z of a software prototype for the concealment of impulsive defects.

1.3 Thesis outline

Chapter 2: Problem to be addressed
The thesis begins by providing the basic concepts necessary to understand the problem. We then focus on the restoration step and after an overview of the main impairments, on impulsive defects.

Chapter 3: Existing approaches for the concealment of impulsive defects
After having described how detection and correction algorithms are currently evaluated, this chapter reviews the state of the art of existing methods for these two steps. It highlights the major limitations that prevent them to meet their expectations.

Chapter 4: Probabilistic tools in image analysis
We introduce in this chapter the theoretical tools that are used in the remainder of the thesis, namely Markov random fields, Bayesian estimation and their combination.

Chapter 5: Impulsive defect detection and pathological motion
The beginning of this chapter is devoted to the definition of what we point out as pathological motion. Our detection model enabling to distinguish it from impulsive defects is exposed. It is validated on real sequences chosen for their complexity in terms of motion.

Chapter 6: Correction in missing data areas
We then describe and experiment the algorithm developed to resynthesize information in damaged regions, which can be applied to archives restoration as well as to many other domains (suppression of superimposed logos or subtitles, image retouching or special effects).

Chapter 7: Experimentation of the complete prototype
After having validated separately our proposals for detection and correction, we stress here on the combination of these two steps to create a complete prototype for the concealment of impulsive defects.

Chapter 8: Conclusions and further research
The final chapter highlights the achievements of the thesis and discusses the four main points that are seen as interesting directions for future work.

d'interactions locales probabilistes entre pixels voisins. Nous les utilisons de manière explicite (modèles paramétriques) pour la méthode de détection proposée, dans le cadre de la théorie bayesienne de l'estimation ; ils interviennent sous la forme d'heuristiques dérivées de la “philosophie” markovienne (modèles non-paramétriques) et inspirées par les récents travaux sur la synthèse de texture pour la technique de correction développée. Dans les deux cas, les algorithmes proposés sont comparés aux principaux travaux du domaine. La démarche complète détaillée dans cette thèse a été validée par l'implémentation de A à Z d'un prototype logiciel de suppression des défauts impulsifs.

1.3 Organisation de la thèse

Chapitre 2 : Problématique traitée
La thèse commence par présenter les concepts essentiels à la bonne compréhension du problème. Nous nous focalisons ensuite sur l'étape de restauration et, après un survol des principales détériorations, sur les défauts impulsifs.

Chapitre 3 : Approches existantes pour la suppression des défauts impulsifs
Après avoir décrit la manière dont sont actuellement évalués les algorithmes de détection et de correction, ce chapitre dresse l'état de l'art des méthodes existantes pour ces deux étapes. Il met en lumière les limitations majeures à cause desquelles elles ne donnent pas satisfaction.

Chapitre 4 : Outils probabilistes en analyse d'images
Nous présentons dans ce chapitre de manière détaillée les outils théoriques utilisés dans la suite de la thèse, à savoir les champs de Markov, l'estimation bayesienne et leur combinaison.

Chapitre 5 : Détection de défauts impulsifs et mouvement pathologique
Le début de ce chapitre est consacré à la définition de ce que nous qualifions de mouvement pathologique. Il expose notre modèle de détection permettant de les distinguer des défauts impulsifs. Ce modèle est validé sur des séquences réelles choisies pour leur complexité en termes de mouvement.

Chapitre 6 : Correction dans les zones d'information manquante
Nous décrivons et testons ensuite l'algorithme développé pour la resynthèse d'information dans les zones corrompues, applicable à la restauration d'archives comme à beaucoup d'autres domaines (suppression de logos ou sous-titres, retouche d'images ou effets spéciaux).

Chapitre 7 : Expérimentation du prototype complet
Après avoir validé séparément nos propositions pour la détection et la correction, nous mettons l'accent ici sur la mise bout à bout de ces deux étapes pour constituer un prototype complet de suppression des défauts impulsifs.

Chapitre 8 : Conclusions et perspectives
Le chapitre final résume les apports de la thèse et discute les quatre principaux points considérés comme des axes prometteurs de recherches futures.

Chapter 2

Problem to be addressed

This chapter introduces the practical problem which is the subject of our concern in the thesis. As a preliminary point, we first explain what television archives are, i.e. film or video material. The re-exploitation workflow is then described before focusing on the restoration step. After an overview of the wide variety of impairments that can affect archived documents, impulsive defects which are at the heart of this thesis are detailed. Finally we list the objectives that are targeted for the concealment of these defects.

2.1 Introduction to film and video

We start by giving a quick overview of the technical aspects involved to represent animated images in the broadcast industry. This is necessary before introducing restoration issues, since a basic understanding of film and video technologies is essential to fully apprehend how defects can occur. Most information in this section is based either on INA's internal expertise or on field knowledge gathered from various sources.

2.1.1 Film

Photographic film, especially 35mm, is regarded as the excellence for animated image acquisition. It is recognized not only for its very high resolution qualities but also for its ability to accurately represent the full tonal range and texture of any scene. The use of film is not limited to the motion picture industry: a huge amount of material has been recorded on this medium specifically for TV broadcast without any exploitation in cinema theatres. This support is still used nowadays in the broadcast industry for high-end productions.

A film is composed of different layers (figure 2.1):

- a *protective overcoating* made of a thin layer of gelatine to protect the film from abrasion;

- the *emulsion layer* which includes a suspension of light-sensitive material in gelatine. This consists of silver halides for black and white film or successive layers of colour-sensitive dyes separated by filter layers for colour film;
- a transparent *base* or *support layer*. Originally, film base was made of cellulose nitrate but the manufacturing of this type of film, nicknamed *film-flamme* in French, was stopped in the early 50's because of its extreme flammability and its chemical instability. Cellulose nitrate has been replaced since then by cellulose triacetate and more recently polyester;
- an *anti-halation and anti-curl backing* preventing light reflections and curling caused by humidity.

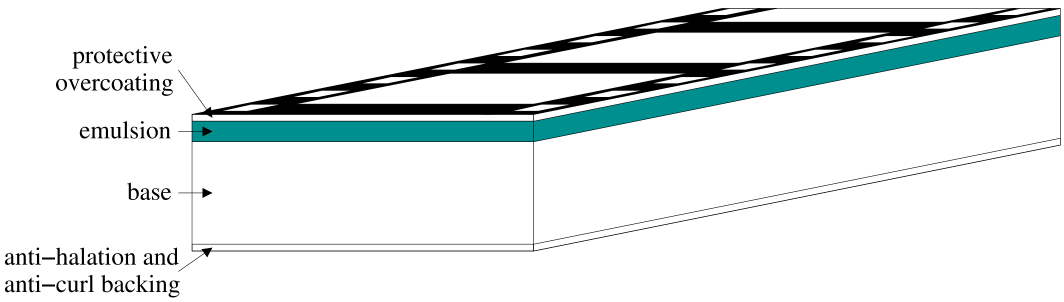


Figure 2.1: cross-section of film

Film is universally recorded at the speed of 24 frames per second. Depending on the period and the type of programme, film documents can be found in 35mm or 16mm formats, usually as negative or reversal originals and more rarely as intermediate prints (inter-negative, inter-positive). Beside documents that have been originally produced on film support, archives also contain film documents known as "kinescopes". The kinescope process consisted in recording on film (usually reversal 16mm or 35mm) the image displayed on a specially designed TV monitor. This technique was in widespread use from the 50's to the beginning of the 70's. At this time, it was the only way to record live programmes. This transfer process usually results in a poor document quality.

Because it preceded the invention of video and because of its intrinsic qualities, film material is very commonly found in broadcast archives. In the case of French TV archives, an estimated 40 to 50% of INA's collection is on film support.

2.1.2 Video

Unlike film, video is characterised by a large heterogeneity of formats that have coexisted and superseded each other. Videotape consists of a magnetic layer supported by a polyester substrate. This magnetic layer, made of magnetic pigments suspended within a polymer binder, is capable of recording a magnetic signal. A common feature of all the different formats is that the video signal is stored on the magnetic layer following a pattern called helical scanning (see figure 2.2): the video tracks are recorded diagonally on the tape.

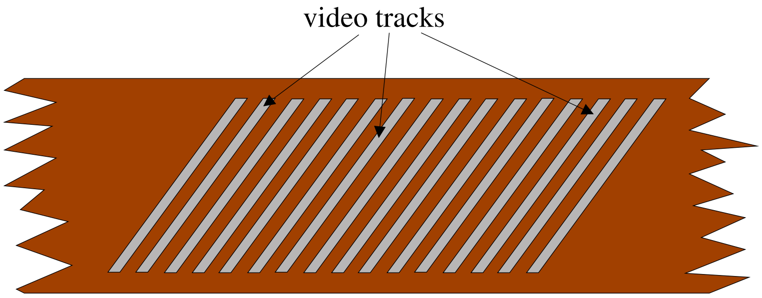


Figure 2.2: helical scan recording on a videotape

A video frame is composed of two “half-images” or *fields*. The odd and even fields respectively regroup all the odd-numbered and even-numbered lines. Each video frame is thus recorded or displayed in two steps: in a first step, one half of the lines (first field) is transmitted and in a second step, the other half of the lines (second field) is transmitted in turn. This scanning process is known as *2:1 interlaced scanning*. It was originally introduced to trade off vertical resolution against temporal resolution: achieving a rate on a frame basis that is high enough to comply with the human persistence of vision would have required too much bandwidth. A consequence of interlaced scanning is that for a document originally recorded on video, the two fields of the same frame capture two distinct moments of time separated by a very short time interval.

Another point that is common to all video formats is the colour space chosen to express colour information. Whereas the RGB colour space is ubiquitous in the computer (and film post-production) industry, colour is represented in video by one *luminance* component coupled with two colour differences (or *chrominance*) components. This colour space is found in slightly different but equivalent flavours, often called YIQ for analog NTSC, YUV for analog PAL, YDbDr for analog SECAM and YCbCr for digital video (see appendix A). A major reason for the choice of this colour space was the constraint of compatibility with monochrome television. The other reason in favour of this colour space is the fact, demonstrated by psychovisual studies, that the human eye is much more sensitive to small variations in black and white than to colour difference details. This allows subsequent bandwidth reduction on the chrominance components.

2.1.2.1 Analog video systems

Broadly speaking, there are different ways to handle colour in the video signal. The most commonly found are known as composite and component video. In *component video*, the three luminance and chrominance signals are conveyed separately. In *composite video*, the chrominance components are encoded on top of the luminance signal to form a single signal. This is the encoding exploited in analog video systems to store a single continuous video signal on the magnetic tape. Three different and incompatible analog systems are in use throughout the world: NTSC, PAL and SECAM.

NTSC (National Television System Committee) [Duv 96] was adopted in 1954 in the United States. It is mainly used in North America, Japan, Central America and one third of South America. The NTSC system is also commonly referred to as a *525-line* system because this is the total number of lines in each frame (among which active lines of video, horizontal and vertical blanking intervals and synchronisation pulses). The field frequency is 59.94 Hz and the frame frequency 29.97 Hz. The technique chosen to embed the chrominance signals in the luminance signal is known as Vestigial SideBand Amplitude Modulation (VSB-AM) or *double Modulation d'Amplitude à Porteuse Supprimée (MAPS) en quadrature*.

On the other side, PAL and SECAM systems show much similarity. They are commonly known as *625-line* systems. Both of them have a field frequency of 50 Hz and a frame frequency of 25 Hz. Therefore, they have traded off a higher vertical resolution than the NTSC system against a lower frame rate. The main difference between PAL and SECAM resides in the way colour information is conveyed.

PAL (Phase Alternating Line) [Duv 95] was initially developed in Germany as an improvement of the NTSC system and was finalized in 1967. It is the predominant television system in the world and is the standard in around 60% of the world countries. PAL is commonly used in most Western Europe (France and Greece being notable exceptions), most Asian and non-French speaking African countries, part of the Middle-East, Pacific countries. Local variants of the PAL system are also used in Brazil (PAL-M), Argentina, Uruguay and Paraguay (PAL-N). As for NTSC, colour is encoded using VSB-AM, but with the phase of the subcarrier inverted every other line, which makes it significantly more robust to colour shifts.

SECAM (SEquentiel Couleur A Mémoire) [Duv 86] was a French initiative started in 1953 and adopted in 1967. This is the system used in France as well as in Greece, Eastern Europe, Russia, French-speaking African countries and part of the Middle-East. Unlike PAL and NTSC colour encodings which are based on amplitude modulation, SECAM uses a Sequential Frequency Modulation with the separation of the two chrominance components: in each line is alternatively encoded either Db or Dr.

2.1.2.2 Digital video

Digital video follows a universally acknowledged standard issued by the ITU-R (International Telecommunication Union - Radiocommunication sector), formerly known as CCIR (Comité Consultatif International des Radiocommunications). This standard initiated in 1982 is known as Recommendation *CCIR 601* or *ITU-R 601* or is often simply nicknamed *4:2:2* [ITU 95]. It defines a digital video coding for both 525-line and 625-line systems with a common sampling frequency of the luminance signal equal to 13.5 MHz. The samples can be encoded either on 8 bits or on 10 bits as allowed by the standard. 8-bit accuracy is often considered to be sufficient in many contexts, including archives restoration.

In this coding standard, video is considered as a component signal, i.e. the luminance Y and the colour differences Cb and Cr are handled separately. As composite colour encoding was the major difference between analog PAL and SECAM, this difference disappears in the digital world, leaving only two systems. The fundamental characteristics of these two systems are summarized in table 2.1.

System	Resolution	Frame rate	Field rate
ITU-R 601 NTSC	720×480 pixels	29.97 Hz	59.94 Hz
ITU-R 601 PAL/SECAM	720×576 pixels	25 Hz	50 Hz

Table 2.1: fundamental characteristics of ITU-R 601

As the human vision is less sensitive to chrominance details, Cb and Cr signals are sampled with half of the sampling frequency of the luminance signal Y, i.e. 6.75 MHz. This chrominance subsampling by a factor of two is known as 4:2:2. Figure 2.3 shows the spatial arrangement given by 4:2:2 sampling: chrominance samples are located on odd-numbered columns.

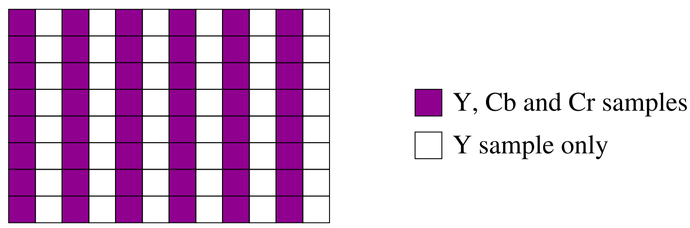


Figure 2.3: 4:2:2 sampling

The data rate of digital video is the same for 525-line and 625-line systems and is about 166 Mbits/s or 20.7 MBytes/s for 8-bit samples. The huge amount of data is one of the inherent difficulties in video processing: as a comparison point, the data rate of a studio-quality (48 kHz, 16 bits) stereo soundtrack is only 1540 Kbits/s or 188 KBytes/s.

2.1.2.3 Video formats

Many different formats for storing video have emerged and faded from the scene throughout television history. Some of these formats have coexisted at the same period but were purposefully used for different types of programmes: some were considered as the best, studio-quality of their time and were intended for high-end programmes; others were considered as a better balance between cost and quality and were restricted to programmes for which quality expectations are less stringent (typically news programmes). Other formats never managed to break through and never attained commercial success. The following list is restricted to the main professional video formats that have gained wide acceptance along with their period of full exploitation:

- *2-inch* tapes, which was the first video recording format, introduced in 1956 and in widespread use from the 60's to the beginning of the 80's.
- *1-inch B* and *1-inch C* tapes respectively introduced in 1977 and 1979 and exploited until the very beginning of the 90's.
- *U-matic* and its variants launched in 1971 and fully exploited from the middle of the 70's to the beginning of the 80's.
- *BVU* (Broadcast Video U-matic), also known as U-matic High Band, which was an improvement of the U-matic format introduced in 1978 and used from the beginning of the 80's until the very beginning of the 90's.
- *Betacam* tapes introduced in 1982 and soon superseded by the backward compatible *Betacam SP* in 1986. This format, although in decline, is still in use today.
- *Digital Betacam* introduced in 1993. It is the first digital format that became mainstream. The compression is nearly lossless with a ratio of 2:1 and involves intraframe DCT-based data reduction. It is currently the *de facto* standard in the broadcast industry.

Besides Digital Betacam which will probably be around for many years, several digital video formats with higher compression ratios are currently struggling for a share of the market. Whereas it is too early to make any prediction, some of them may become mainstream in the future and will therefore join the long list of formats that archives holders have to cope with. These formats are based either on the DV (such as DVCAM, DVCPRO25, DVCPRO50) or on the MPEG-2 (such as Betacam SX or IMX) compression standards.

2.1.3 Film-to-video conversion

In this section, only problems related to frame rate conversion will be described as this has an influence on how film documents converted to video are processed afterwards. However, the reader should be aware that the film-to-video conversion process involves other issues such as soundtrack synchronization or aspect ratio conversion (e.g. from Cinemascope 2.35:1 to 4/3 or 16/9).

Film is converted to video during what is called the *telecine* process. A telecine device has the capability to scan in real time the film frames and convert them into a video signal. The conversion from film to PAL/SECAM video is quite straightforward: as the film 24 frames/s is close to the PAL/SECAM 25 frames/s, two video fields (i.e. one video frame) are generated from one single film frame. This amounts to a slight acceleration which is usually not noticeable. With NTSC, conversion is a bit more complicated. As the NTSC field rate is approximately 60 fields/s, that is 2.5 times the film frame rate, one out of two film frames is scanned to generate 3 fields while the other one is scanned to generate 2 fields (see figure 2.4). Therefore, 5 NTSC video frames are generated from 4 film frames. This conversion process is referred to as *3:2 pulldown*.

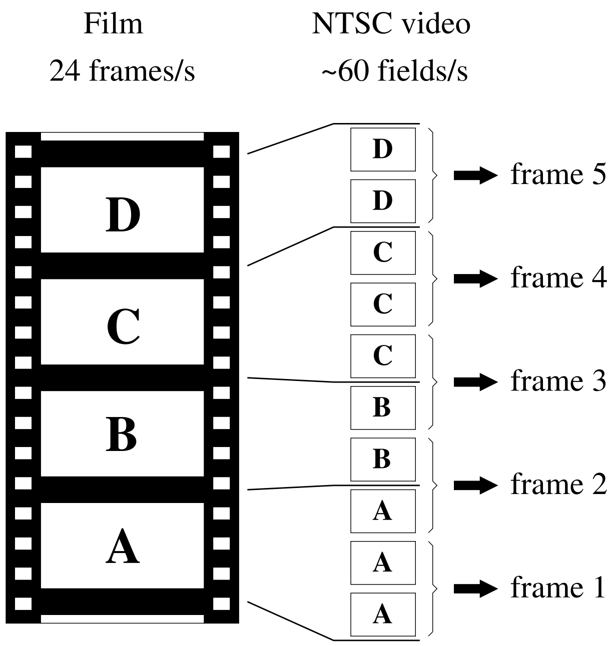


Figure 2.4: 3:2 pulldown

A consequence of this conversion on restoration and all subsequent image and video processing operations is that a document should be processed differently according to its origin. When the original document is video, all fields capture distinct moments of time and the digitized video should be processed on a field-by-field basis. When the original document is film, there must be a pre-processing step to recover the original film frames at video resolution from the digitized video. Processing algorithms can then proceed on a frame-by-frame basis.

2.2 Archives re-exploitation

The business of a TV archivist at the head of huge holdings of film and video documents can be mainly divided into three parts:

- the *collection* of programmes, including their identification and documentation in a database in order to allow requests to be made;
- the *preservation* of the archives. This encompasses preventing the physical deterioration of the supports by appropriate storage and handling conditions, but also

keeping in working order old playback devices and maintaining the associated human skills;

- the *re-exploitation* of the archives for the creation of new programmes.

Our attention will be more specifically focused on this last task. Collection and preservation tend to the same final aim: enable the audiovisual professionals (broadcasters, producers, editors) to re-use archived material in a new context for their own needs. The corresponding demand has noticeably increased in the last few years with the multiplication of available media for distribution (cable, satellite, DVD). There are often two slightly different procedures depending on whether the request concerns excerpts or integral works. We now concentrate on the technical steps involved in the re-exploitation regardless of the legal issues that are often very intricate.

Whatever the type of document, the first step is the *transfer* from the original support to digital video. The original document is first cleaned on a specific device. Then in the case of film, it is transferred to video by the telecine process, preceded if necessary by a physical restoration (e.g. fixing of the damaged perforations); if the original document is video, it is played in the best possible conditions on the appropriate VTR and copied to a digital video format. The goal of the transfer step is only to get a copy that is as close as possible to the original. This digital copy, currently a Digital Betacam tape for most archivists, is called the *preservation master*.

When the request concerns excerpts, a copy of the preservation master is usually directly delivered to the customer. This is possible because quality expectations for short duration sequences are much less stringent than for long programmes. Moreover, especially for excerpts intended for TV news programmes, deliveries at very short notice are required and this leaves no spare time for further processing. For integral works on the other hand, there is often no emergency and the expectations concerning the visual quality of the documents are much higher and keep increasing with broadcast technology advances. However, due to ageing and/or to early technical limitations, archives are generally affected by a number of deteriorations and are not in a condition that enables to re-use long sequences directly for broadcast. Examples of such deteriorations will be detailed in the next section. For this reason, after the transfer onto a digital medium, a step of *digital restoration* is usually necessary (figure 2.5). The restoration leads to a *"commercial" master* (or PAD for Prêt A Diffuser) that is used for exploitation.

The video-to-video digital restoration is performed by an expert operator who takes the decisions and makes all required adjustments. With present tools, this process is long and may take up to several tens of hours of work per hour of programme. The problem is the associated cost, essentially in human resources, which can be as high as many thousands of euros per hour of programme.

This is why more *automation* is essential for significant cost reduction. This automation is to be partly achieved by the development of specific processing algorithms. The aim is to relieve the operator from the most tedious tasks and to enable him to concentrate on high-level issues and supervision. Restoration time is thus expected to drop to at most

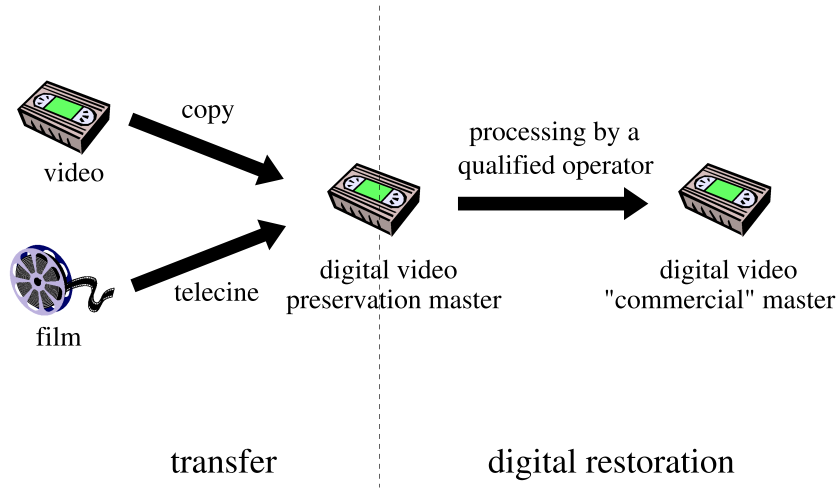


Figure 2.5: re-exploitation workflow

a few hours per hour of programme.

2.3 Typology of the main impairments

A very wide diversity of visual impairments can be encountered by archivists. They may have occurred at all the different steps of the lifecycle of the programme (shooting, editing, post-production, transmission, storage, encoding, systems conversion, film $\leftrightarrow$ video transfer, playback, etc). We now provide a brief description of the main defects without aspiring to exhaustivity: this list is far from including all known artifacts, but is rather meant to give an idea of the variety of the problems and their origins.

- *Flicker* (*battement d'intensité* or *pompage*) is defined as unnatural temporal fluctuations in intensity from frame to frame. This can have many origins, the most common of which are variations in the shutter time of early cameras causing variations in exposure time, inhomogeneous ageing or degradation of the film support or problems in lighting synchronization. This impairment can be a spatially localized effect. Efficient solutions have been proposed to tackle this problem, see [Roo 99b] and [Roo 99a], chap. 3.

- *Unsteadiness* (*instabilité*) appears as unwanted fast shaking. For film documents, this global frame-to-frame displacement can be caused by a lack of reliability of the film transport system in the camera or the telecine or by damaged perforations along the support. More generally, it can also be due to bad shooting conditions. This artifact is easily removed by compensating high-frequency global displacements (see [Uom 90], [Kin 90] among the earliest works). Unsteadiness suppression technology is even readily available to the consumer market as many hand-held cameras now incorporate dedicated real-time hardware.

- *Twin-lens telecine flicker* (*battement de téléciné à double trajet optique*) is closely related to unsteadiness. In some early telecine devices, the optical paths for the generation of the two video fields were different. When these devices were not properly calibrated, there could be an alternate displacement between each recorded field, appearing as a kind of "flicker" effect, although it is not due to the change of light intensity between fields. Figure 2.6 shows the odd and even field of a video document affected by this artifact. This issue has been addressed in [Vla 96].

- *Dirty splices* (*collures sales*) are frequently found in film documents. During film editing, the two pieces of film are joined together with scotch tape. With ageing these joins can accumulate dirt and become visible (figure 2.7). For such documents, frames right before and after each shot change have to be retouched or replaced.

- *Line scratches* (*rayures film*) are defects typically related to film which are relatively common. They are due to the abrasion of the film in a direction parallel to the direction of film transport by a particle caught in the mechanism. Line scratches can occur either on the base side or on the emulsion side, in which case they can partially or completely damage the light-sensitive material. They are visible as usually bright or dark vertical lines (figure 2.8), and can occasionally have a specific colour when only part of the emulsion has been damaged. They usually have the same or nearly same location in consecutive frames. One of the most successful approaches to date considers an additive model for these scratches [SW 01]. Alternative techniques include [Kok 98] chap. 9, [Dec 97] chap. 5, [Joy 99], [Joy 00] chap. 3 and 4.

- *Video scratches* (*rayures vidéo*) are the video counterparts of line scratches, with a similar cause but a completely different look; they are however less frequent. In the video case, horizontal scratches on the physical medium disturb the magnetic information stored on the tape. As a result of the helical scanning, the heads of the VTR sweep over the scratch periodically as they rotate. The consequence on the image is the presence of horizontal pulses with a comet-like tail to the right following a regular pattern (figure 2.9). Few works exist on this problem [Har 97], [Arm 99] chap. 4.

- *Noise* (*bruit*) is a very general problem in all recorded signals. Film and video are not exceptions and can show various types of noise. Noise removal has been extensively studied in many contexts including animated images. Among others, [Bra 95], [Dek 01], [Roo 99a] chap. 5 are interesting reviews of the subject. *Film grain* (*grain film*) is a specific kind of noise due to the individual light-sensitive elements in the film emulsion and has the particularity to be not uncorrelated from pixel to pixel and to be signal-dependent. Its level increases with the physical degradation of the film (figure 2.10). Whereas it should be reduced in some cases, a complete removal of this kind of noise is usually not desirable as it significantly contributes to the film "feel".

- *Colour fading* (*virage colorimétrique*) occurs when one or several dye layers used in colour films degrade over time (figure 2.11). The individual layers often fade unevenly, at different rates resulting in colour variations.

- *Tear damage* (*déchirures*) can occasionally affect one or several consecutive frames of a film document (figure 2.12), which must then be edited.
- *Line jitter* (*jitter ligne*) is a video problem of synchronization between lines: they appear to have been displaced by a pseudo-random horizontal shift (figure 2.13). Line jitter can have various origins such as interferences with an electrical signal or a failure of the Time-Base Corrector (TBC) of the VTR. This is not a very common impairment, but the resulting visual discomfort is especially severe. Relatively little work has been devoted to this problem, see [Kok 98], chap. 5 and [Kok 97].
- *Moiré* is an artifact found on kinescope documents once they are transferred back to video by the telecine process. As the geometry, orientation and spacing of the scan lines of the TV monitor recorded on film differ slightly from those used by the telecine device to scan the film, aliasing appears in the resulting signal. This aliasing is visible as periodic dark rings that can be curved near the extremities of the image (figure 2.14). Moiré is a very difficult problem that is just beginning to be studied in the context of television [Sid 02].
- The *vinegar syndrome* (*syndrome du vinaigre*) specifically affects film material on a cellulose triacetate base. When exposed to improper ambient storage conditions, such film can undergo a chemical hydrolysis reaction which is not reversible. This slow degradation causes breakdowns of gelatine that will eventually be visible (figure 2.15). Vinegar syndrome owes its name to the very characteristic odour of acetic acid that emanates from contaminated films.
- *Blotches* (*taches*) are at the heart of this thesis. This impairment will be detailed in the next section.
- *Video dropouts* (*"dropouts" vidéo*) will similarly be detailed in the next section.

Figure 2.6: twin-lens telecine flicker

Figure 2.7: dirty scotch tape

Figure 2.8: film scratches (image by courtesy of the BBC)

Figure 2.9: video scratches, here on a 2-inch tape (image by courtesy of the BBC)

Figure 2.10: film grain (image by courtesy of the BBC)

Figure 2.11: colour fading

Figure 2.12: film tear damage (image by courtesy of the BBC)

Figure 2.13: line jitter

Figure 2.14: kinescope moiré

Figure 2.15: vinegar syndrome

2.4 Impulsive defects

2.4.1 Description

Among all types of impairments that can affect archived material, so-called *impulsive defects* are among the most frequent. They embrace two distinct types of artifacts, namely *blotches* and *video dropouts*.

What is generically called *blotches* is a film defect encompassing two different physical processes: the loss of pieces of gelatine constituting the film (*sparkle*) or the electrostatic adhesion of various particles on the film (*dirt*), such as dust, hair, pieces of cloth, glue or solvent spots, etc. Both kinds of corruption appear as flashing blotches on the images after transfer to video (figure 2.16). This problem occurs very frequently and affects to a certain extent almost all film archives.

Similar artifacts are found to a lesser extent in video: what is referred to as *dropout* is when one or more lines cannot be read properly during the tape playback. Some VTRs do include a DropOut Compensator (DOC) which replaces the missing signal by samples of a previous signal properly read. Others do not perform any compensation. These losses of signal result in the replacement of portions of lines or entire lines by a uniform colour or by a replication of one of the preceding lines, with colours often inverted or strongly distorted (figure 2.17).

These defects have in common their impulsive nature, either on a frame-by-frame basis for blotches or on a field-by-field basis for dropouts. Our thesis is more specifically focused on these impairments. These defects, especially blotches, often occupy a significant portion of the operators' time in typical restorations. There is therefore a strong operational demand for effective solutions and previous works on these artifacts, which will be detailed in the next chapter, have but partly come up to these expectations.

2.4.2 Experimental observations

Extensive experimental observations of corrupted documents yield the following remarks on blotches:

- They can be opaque as they can be transparent (glue spots for example) causing a complete or partial loss of information.
- They are often rather bright or dark but the whole spectrum of intermediate cases can also be found.
- Their size can range from a few pixels to a significant portion of the image.
- Their shape can be extremely varied, whether in terms of topology (presence or absence of "internal holes" inside the blotch), compacity (regularity of the shape) or elongation (ratio of the largest dimension over the smallest).

(a)

(b)

Figure 2.16: blotches (image (b) by courtesy of the BBC)

- Blotches often look similar in the same sequence even though it is absolutely not incompatible with the occasional appearance of “atypical” blotches within this very sequence.

Similarly, the following observations can be made on dropouts:

Figure 2.17: video dropout, here for a SECAM Beta SP tape (resized field)

- The underlying information is completely lost.
- They appear differently according to the involved video technology: for example there can be a copy of one of the last two preceding lines or the line before, with or without horizontal shifts, with or without colour inversion, or simply no copy at all and the replacement by a uniform signal. This depends on whether the system is PAL, SECAM or NTSC as well as on the video format (e.g. 2-inch tape, U-matic, Beta SP, ...).
- They can affect small portions of a line (a few pixels) or huge blocks of entire lines (up to one half of a field).
- They naturally show a strong horizontal orientation, since they are a disruption of the line scanning process.
- In the same sequence are often found many small dropouts and/or less frequently a few large ones.

2.4.3 Detection and correction steps

A global processing of the whole image is not reasonable in so far as we are dealing with artifacts which are essentially local both in time and space. The removal of impulsive defects is therefore performed in two steps:

- *detection* which consists of flagging pixels that are considered as damaged;

- *correction* which consists of resynthesizing missing information in the flagged areas.

2.5 Target objectives

In order to have a practical interest for real-world restoration applications, algorithms for the detection and correction of impulsive defects should target several objectives:

- *efficiency* is undoubtedly the primary objective
- *genericity*, i.e. no specialization toward a very specific category of impulsive defect and relative independence with respect to the other "blocks" of a restoration system
- *robustness*, or the ability to be efficient on a wide variety of documents with different contents, levels of activity and states of degradation
- *automation*: there should be a very restricted number of parameters that would need to be hand-tuned to maintain efficiency in different contexts. This can be either because the algorithms involve few parameters, or because most parameters do not need to be changed.
- *computational speed*: ideally, the restoration operator should have a real-time feedback from his actions. However, because of the huge amount of data, software processing of uncompressed video in real time is currently hardly feasible. Algorithms should nevertheless be as fast as possible: in a first step, this makes a real-time hardware or semi-hardware implementation conceivable; in a more remote prospect, it leaves the door open to a real-time software implementation in the near future. This objective has a significant importance even at the research stage as it is essential to enable experiments on large enough testing datasets.

Chapter 3

Existing approaches for the concealment of impulsive defects

Before getting deeper into algorithmic details, we first discuss in this chapter how detection and correction methods are currently evaluated. We then review the main families of existing methods for these two steps. We finally highlight in the last section the major limitations that prevent them to come up to expectations and the main orientations which we decided on to alleviate these shortcomings.

3.1 Evaluation issues

The most obvious way to evaluate how well an image processing algorithm fulfils its goal is often to simply look at its output. This especially makes sense for restoration algorithms as they are ultimately intended for the television viewers at the very end of the process. However, beside individual subjective evaluation, it can be desirable to be able to put figures over the performance of different algorithms. Although the number and complexity of algorithms developed in the field of restoration has grown, relatively little attention has been devoted in comparison to this problem of quantitative assessment. This is all the more true for such a specific defect as blotches or dropouts. Evaluation in such a case is an intrinsically difficult problem as human perception is involved to some extent. This section discusses the related issues and analyses the main trends found in the literature.

3.1.1 Interest of quantitative evaluation

The ability to quantitatively assess the quality of a performed detection or correction is interesting for mainly two reasons:

- Firstly, it can be employed to *choose the optimal parameter settings* of a given algorithm. The parameters of a detection or correction algorithm can thus be tuned such as the measured quality of the output is optimal.

- Secondly, it can be used to *compare the performance of different algorithms*. Once a testing dataset has been carefully chosen, several algorithms can be benchmarked on this corpus and this would be very helpful to determine which one produces the best results.

3.1.2 The testing dataset

A prerequisite for the quantitative assessment of detection and correction algorithms is the availability of sequences for which the expected output is known. This is commonly referred to as *full-reference* evaluation. Ideally, there should exist a reference dataset which is shared among the research community.

For blotch detection, the reference dataset could consist of "clean" sequences along with the same sequences corrupted by blotches. Such a dataset is not yet a reality: there is currently no well-accepted reference dataset that can be used for evaluation as this exists for other applications, such as the Yosemite sequence for motion estimation, the FERET database for face recognition, the Brodatz and VisTex datasets for texture analysis and synthesis and many others. This is mainly due to the fact that archives restoration has traditionally attracted less attention than other domains and also partly to the inherent difficulties of copyright issues. As there is no such dataset, researchers in this field usually experiment their algorithms on the sequences they have at hand. These sequences often contain very little motion and are usually very short, sometimes as short as a few frames. Artificial blotches, often consisting of uniform patches, are added to these sequences. For missing data correction, what should be known for full-reference evaluation is simply the underlying original information. The variety of images and sequences used by researchers for testing is even wider.

3.1.3 Quantitative evaluation for detection

Blotch detectors are usually compared in the literature through their *Receiver Operating Characteristics* (ROC) ([Kok 98], chap. 6.4, [Roo 99a], chap. 4.2.1). An ROC plots the *correct detection rate* versus the *false alarm rate* for all possible variations of a parameter, as shown in figure 3.1. The correct detection rate is defined as the number of pixels correctly flagged as corrupted divided by the total number of corrupted pixels. The false alarm rate is defined as the number of pixels incorrectly flagged as corrupted divided by the total number of clean pixels. Usually, as many detectors involve more than one parameter, some of them are arbitrarily set or tied together so that we are left with a single value to tune. Another possibility is to test all possible parameter settings and keep the ROC which corresponds to the best performance of the algorithm. The ROC averaged on whole test sequences is usually plotted.

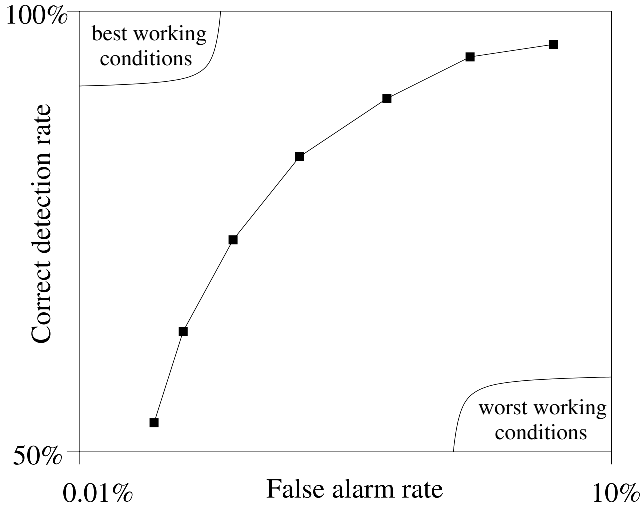


Figure 3.1: Receiver Operating Characteristic

3.1.4 Quantitative evaluation for correction

The best way to assess the quality of a restoration is certainly to involve human viewers as they are the ones that should ultimately benefit from the whole process. This is known as *subjective* quality evaluation. One of the most widely used methodologies is the Double Stimulus Continuous Quality Scale (DSCQS) protocol of Recommendation ITU-R 500 [ITU 00] and the associated Mean Opinion Score (MOS). Several other testing methodologies exist, among which the Two-Alternative Forced Choice (2AFC) or the Reaction Time (RT) method [Yeh 98] which accounts for sensibilities well above the visibility threshold. Because they must involve a significant number of human observers, they have the major drawback to be costly, time-consuming and very impractical for use during a research cycle in progress. For these reasons, much work has been devoted to the design of metrics that could successfully replace the use of subjective assessment.

The most popular metrics in missing data correction ([Roo 99a], chap. 4.2.2) and used much more widely are the Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). These simple, mathematically defined measures are completely equivalent. The MSE between original image I_{orig} and corrected image I_{corr} over N indexed, replaced pixels is defined as

$$\text{MSE} = \frac{1}{N}\sum_{n=1}^{N}[I_{\text{orig}}(n) - I_{\text{corr}}(n)]^2 \tag{3.1}$$

The Root Mean Squared Error (RMSE) is simply the square root of the MSE. PSNR is measured in decibels (dB) and is defined for an 8-bit image as

$$\text{PSNR} = -20\log_{10}\frac{RMSE}{255} \tag{3.2}$$

These two measures are in overwhelming use in archives restoration as well as in many other contexts.

Many much more sophisticated measures have been proposed that attempt to model to some extent the Human Visual System (HVS), among which [Dal 93, Teo 94, Lub 95]. They try to incorporate known psychovisual effects such as *contrast sensitivity* (also called luminance sensitivity) expressed by the Weber-Fechner law, *spatial frequency sensitivity* or *pattern sensitivity* typically described by a Contrast Sensitivity Function (CSF) and *masking effects* [Dal 93, Vle 02]. Masking effects can be defined as the reduction of visibility of a signal by the presence of another signal (spatial or edge masking, contrast or pattern masking, activity masking, etc) and remain very complicated and largely ill-known phenomena. It must be noted that very few of these measures incorporate motion and/or colour. Some of them are restricted to very specific types of distortions, such as compression artifacts which attracted much interest [Yu 02].

3.2 Detection

The detection step is concerned with flagging pixels that are damaged. Blotch detection methods developed so far mostly rely on the following principle [Kok 95a]: when motion estimation is performed, blotches are the elements that are ill-matched both toward the previous and the next frame. For this reason, motion estimation plays a very important role in this process.

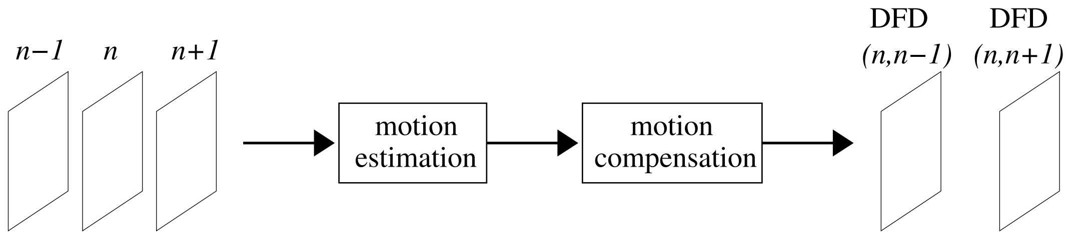


Figure 3.2: principle of detection

We will distinguish three main categories of works for the detection of impulsive defects: heuristic methods, methods involving mathematical morphology and methods based on probabilistic models within a Bayesian framework. All these methods only incorporate three frames in the detection process.

3.2.1 Heuristic methods

Heuristic methods follow the simplest strategy, which consists in directly comparing a pixel with the corresponding pixels in the motion-compensated previous and next frame. The first algorithm based on this idea is known as SDIa (Spike Detection Index - a): it has been introduced by the BBC [Sto 85, BBC 84], although the original algorithm did not involve motion compensation at the time. For each pixel p in image n, the backward and forward Displaced Frame Differences (DFD) are computed:

$$\begin{aligned}\text{DFD}_b(p) &= I_n(p) - I_{n-1}^{(mc)}(p)\\ \text{DFD}_f(p) &= I_n(p) - I_{n+1}^{(mc)}(p)\end{aligned}$$

where $I^{(mc)}$ denotes motion-compensated intensities.

A backward or forward discontinuity is set when these differences are over a selected threshold T_h:

$$b_b(p) = \begin{cases} 1 & \text{if } |\text{DFD}_b(p)| > T_h \\ 0 & \text{otherwise} \end{cases}$$

$$b_f(p) = \begin{cases} 1 & \text{if } |\text{DFD}_f(p)| > T_h \\ 0 & \text{otherwise} \end{cases}$$

The pixel is then considered as corrupted if it supports both a backward and forward temporal discontinuity:

$$b_{\text{SDIa}}(p) = \begin{cases} 1 & \text{if } b_b(p) = 1 \text{ and } b_f(p) = 1 \\ 0 & \text{otherwise} \end{cases}$$

3.2.2 Mathematical morphology

Other works [Bui 97, Dec 97] rely on mathematical morphology to perform the detection. As their name suggests, morphological operations do have a reliance on shapes and are based on connectivity properties. Opening and closing operations, controlled by the choice of a structuring element, are used to build a detector of local intensity extrema. These tools are then combined with an analysis of temporal continuity to locate the artifacts. However, the size of the structuring element and the assumption of local extrema restrict this kind of detector to impulsive defects having a very specific profile.

3.2.3 Bayesian framework and Markov Random Field models

The last category of work deals with probabilistic approaches developed within the framework of Bayesian theory [Gem 92, Mor 95, Cho 97, Kok 98]. Bayesian estimation provides a probabilistic framework to infer the values of unknowns X from the values of observations Y. A *prior distribution* $P(X = x)$ is defined for the unknowns as well as the *likelihood*, i.e. the conditional distribution $P(Y = y|X = x)$ linking the unknowns to the observations. A *posterior distribution* $P(X = x|Y = y)$ is then computed thanks to Bayes' rule. From this posterior distribution, an estimator is chosen to infer the values of the unknowns. All this theoretical framework will be detailed in chapter 4.

Here, the unknown variables are typically temporal discontinuities between two images and distributions are modelled by Markov Random Fields (MRF). The scheme developed in [Mor 95] can be seen as the probabilistic equivalent of SDIa (figure 3.3): the unknown variables are the backward discontinuities B_b or forward discontinuities B_f and the observations are the motion-compensated intensities:

$$\left\{\begin{array}{ll} X & = B_b \\ Y & = (I_n, I_{n-1}^{(mc)}) \end{array}\right. \quad \text{or} \quad \left\{\begin{array}{ll} X & = B_f \\ Y & = (I_n, I_{n+1}^{(mc)}) \end{array}\right.$$

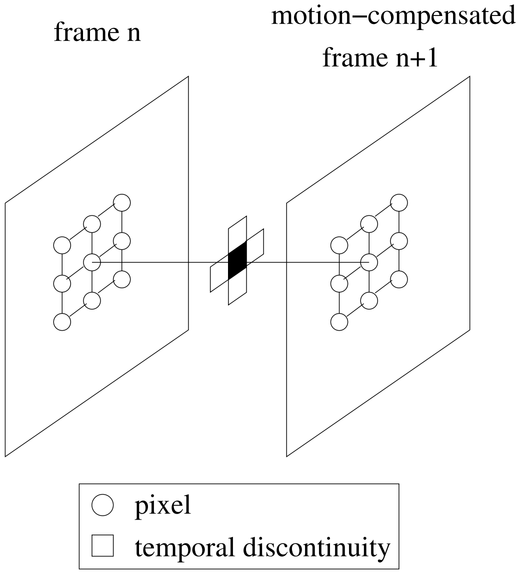


Figure 3.3: Morris' temporal discontinuity detection

But a more global strategy can also be pursued such as Kokaram's JOMBADI algorithm (JOint Model BAsed Detection and Interpolation) [Kok 98], chap. 7. In this technique, Markovian and autoregressive models (introduced in section 3.3.2.3) are made to cooperate together within a common Bayesian framework and the detection mask, the motion fields and the corrected intensities are simultaneously estimated, at the cost however of a significant computational complexity.

3.3 Correction

During the correction step, we aim at automatically interpolating missing information in the damaged regions from the surrounding. The ultimate goal is that a person viewing only the corrected document should not be able to realize that it may have undergone changes. Existing works in this field are not restricted to restoration as many other applications have the same requirement of being able to resynthesize information in whole image regions.

We distinguish two categories of works specifically targeted at missing data correction, depending on whether they deal with still images or image sequences. We also describe a third complementary group of techniques which is not directly related to our problem but which has been very influential for our work: texture synthesis.

3.3.1 Still image correction

The first category is focused on still images and involves the spatial surrounding. For all techniques belonging to this category, the extension to several frames is far from straightforward.

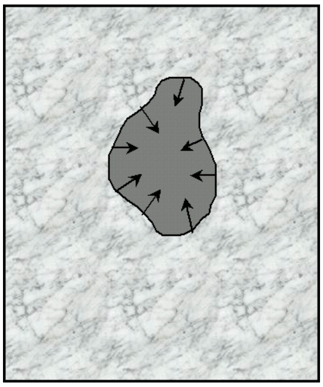


Figure 3.4: spatial correction

3.3.1.1 Frequency-based techniques

A possible approach is to correct missing areas by reshaping the Fourier spectrum of the regions of interest according to a reference spectrum. This is the correction strategy developed in [Hir 96], expressed in the general framework of Projection Onto Convex Sets (POCS): the reference spectrum is computed from a sample subimage which must be very carefully selected by the user. This is also the approach chosen in [Bui 97] and [Joy 00] by reconstructing the low frequencies with polynomial interpolation and the high frequencies with Fourier series. This is however applicable only for small deteriorations.

3.3.1.2 Partial Differential Equations

Other approaches based on the use of Partial Differential Equations (PDE) [Mas 98], [Ber 00], [Ber 01] show impressive results. The idea is to extend inward the lines arriving at the hole boundaries.

In particular, for the *image inpainting* algorithm ([Ber 00], [Ber 01] chap. 3), the iterative process can be written as

$$I^{k+1}(p) = I^k(p) + \Delta t I_t^k(p) \tag{3.3}$$

where t denotes the inpainting "time" and k the iteration index. The update $I_t^k(p)$ is given by

$$I_t^k(p) = \delta \vec{L}^k(p) \cdot \vec{N}^k(p) \tag{3.4}$$

where L is the information being propagated and $\vec{N}$ is the direction of propagation. L is chosen to be the intensity Laplacian, which can be considered as a "smoothness estimator". $\vec{N}$ should be the direction of the isophotes (lines of equal grey values) and is given by the perpendicular to the gradient vector. This propagation process is interleaved with anisotropic diffusion.

This approach performs well on a wide variety of examples. However these results are obtained at a high computational expense and the lack of textural information in reconstructed regions can be visible, especially when these regions are large. It should be noted that recent techniques are also related to this approach although they are not based on PDEs [Rar 02]. They can be considered as simplified versions restricted to structured areas: their principle is to prolong and join disrupted edges for faster reconstructions.

3.3.2 Image sequence correction

The second group of works deals with missing data in image sequences and with how to extract information from the previous and next frames. These techniques cannot in general be applied to still images and cannot be used when the region to be corrected runs across several frames. They require accurate motion vectors to be efficient and a preliminary step of motion vector repair is therefore usually necessary.

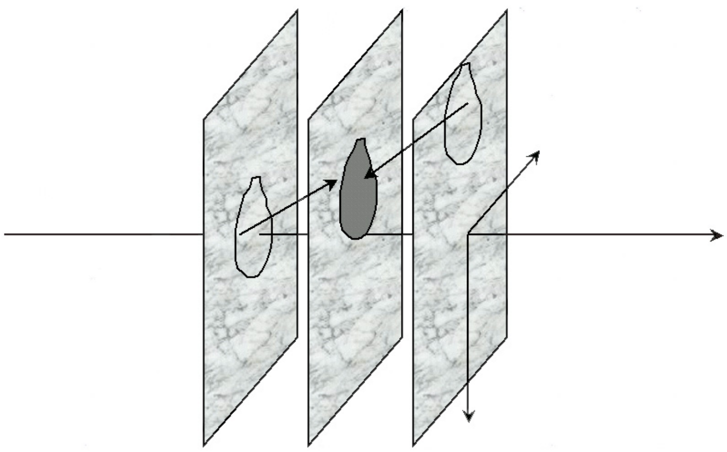


Figure 3.5: spatio-temporal correction

3.3.2.1 Preliminary motion vector repair

The presence of corrupted regions make some motion vectors unreliable and likely to be completely wrong (cf. [Kok 98], chap. 6.6). For this reason, a step of motion vector interpolation can be required to avoid extracting erroneous data from the previous and next frames: this is actually the case for all correction methods using motion-compensated frames. Two possibilities can be considered for motion vector repair: rely on the intensities of the surrounding pixels or rely on the neighbouring motion vectors. Several methods have been developed for this purpose (see [Kok 98], chap. 6.6 and 8, [Roo 99a], chap. 4.2.2). This remains however a delicate issue as interpolating wrong motion vectors or missing pixels are challenges of the same order of difficulty.

3.3.2.2 Rank-order filters

A rank-order filter consists of a sliding window the output of which is chosen as one of its inputs on the basis of a rank-ordering of those inputs. These filters are included in the more general class of order statistic (OS) filters, which take as their output a linear combination of its rank-ordered input values.

Median-type filters are the most famous examples of rank-order filters. Originally introduced for noise suppression [Arc 91], Multistage Median Filters (MMF) have been proposed for blotch correction, such as Kokaram's ML3Dex algorithm ([Kok 98], chap. 6.5 and 6.6, [Kok 95b]). ML3Dex first applies five sub-filters shown in figure 3.6. In this figure, the top plane of each sub-filter represents the motion-compensated next frame, the centre plane represents the current frame and the bottom plane represents the motion-compensated previous frame. The central pixel refers to the pixel being processed. The final output of the ML3Dex filter is defined as

$$\begin{aligned} z_l &= \text{median}[W_l] \quad \text{for } 1 \leq l \leq 5 \\ \text{ML3Dex} &= \text{median}[z_1, z_2, z_3, z_4, z_5] \end{aligned}$$

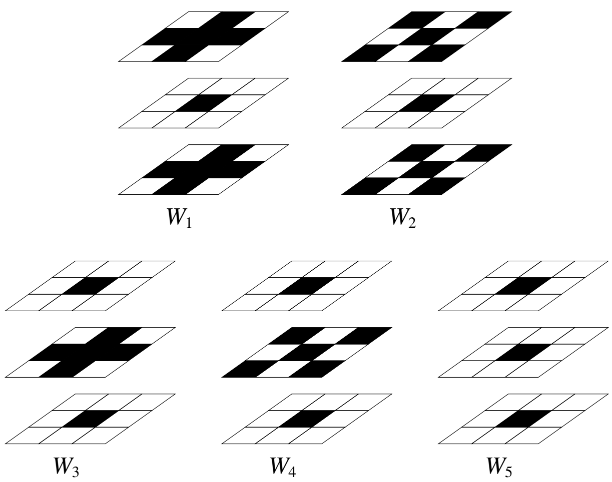


Figure 3.6: Kokaram's ML3Dex sub-filter masks

3.3.2.3 Autoregressive models

A spatio-temporal autoregressive (AR) model defines a pixel as a weighted combination of the surrounding pixels ([Kok 98], chap. 2.3 and 3). The intensity of a pixel p is then given by

$$I_n(p) = \sum_{q=1}^{N(n)} a_q I_n(q) + \sum_{f=1}^{F^+} \sum_{q=1}^{N(n+f)} a_q I_{n+f}^{(mc)}(q) + \sum_{f=1}^{F^-} \sum_{q=1}^{N(n-f)} a_q I_{n-f}^{(mc)}(q) \tag{3.5}$$

where f is the frame offset and F^+ and F^- are the maximum frame offsets in the forward and backward directions,

q is the index of a pixel in the AR support and $N(k)$ is the total number of pixels in the AR support in frame k.

Usually, the AR coefficients a_q are assumed to be the same for all pixels in a given block. The optimal coefficients in the least-square sense can then be estimated for each block. Figure 3.7 shows an example of a causal AR model with a support of nine pixels in the previous frame. It should be noted that simple temporal frame averaging $I_n(p) = (I_{n-1}^{(mc)}(p) + I_{n+1}^{(mc)}(p))/2$ can be seen as a special case of AR models with a support of one single pixel in the previous and in the next frame and no estimation of the optimal coefficients.

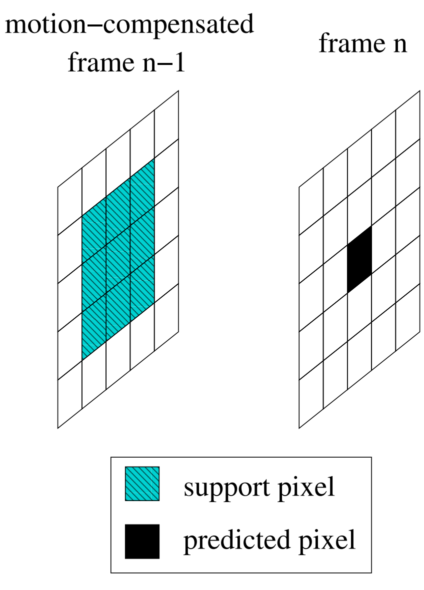


Figure 3.7: example of an AR model

The use of AR models for spatio-temporal correction is developed in [Kok 95b], [Kal 97] and [Kok 02]. The main shortcoming of the AR assumption is that its validity is practically restricted to highly textured regions.

3.3.2.4 Parametric Markov random fields

Parametric probabilistic models can also be introduced to interpolate missing data. The original sequence is then modelled as a spatio-temporal Markov random field and the corrupted pixels are corrected by drawing a representative sample from the Markovian distribution. The specified Markovian distribution usually encourages spatial and temporal smoothness between neighbouring pixels. Examples of explicit Markovian distributions for this purpose can be found in [Mor 95, Kok 95b]. For computational tractability, these models involve very small neighbourhoods.

3.3.3 Texture synthesis

Complementary to these works on correction are recent works in texture synthesis. The main goal is here to generate from a sample texture large and/or tileable texture patches that are suitable for mapping (figure 3.8). While a variety of approaches had been proposed including the use of Markov random fields parametrized by filter responses [Zhu 98], this domain has known a renewed interest with the insightful heuristic technique introduced in [Efr 99]. In this algorithm as well as in subsequent works derived from the same idea [Wei 00, Ash 01, Efr 01, Har 01], the synthesis is based on a non-parametric Markovian model. Unlike [Zhu 98], the probability distribution is not constructed explicitly, it is rather directly approximated from the reference sample texture. The corresponding improvement in terms of quality is very significant.

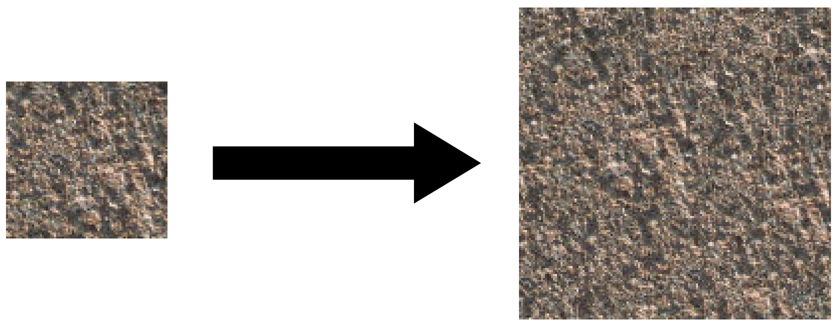


Figure 3.8: texture synthesis

Rare attempts have been made to apply texture synthesis to image correction [Ige 97]. Although these techniques generate high-quality results, their efficiency is limited to texture, i.e. stationary patterns and often to some specific classes of texture (e.g. stochastic textures).

3.4 Limitations and proposed strategy

Beyond shortcomings that can be specific to a given group of techniques, all the previously proposed detection methods have a major practical limitation: they are very sensitive to motion estimation failures, which are the source of false alarms. Unfortunately, it is precisely in these situations that the efficiency of existing spatio-temporal correction methods declines dramatically: they incorporate from the previous and next frames information which is not relevant. The combination of the false alarms and the erroneous use of temporal information gives rise to the creation of visually disturbing artifacts in regions which were initially free from trouble. As this makes a manual intervention necessary, it has a critical impact on the performance of the overall system.

From a careful analysis of these failures arises the awareness that they have a common origin. They are due to natural events perceived as complex, which are not artifacts and yet represent violations of the motion estimation model.

The key element in our strategy is therefore to take into account what we shall call *"pathological" motion* in this thesis. A reliable detection scheme should be able to distinguish impulsive defects from pathological motion. This could be achieved by incorporating a larger temporal aperture. Similarly, an ideal correction technique should be wise enough to use very little temporal information in the presence of pathological motion. A mechanism of fallback on spatial information should be provided in this case.

The other important point in our strategy is the use of *probabilistic models* such as Markov random fields. They indeed provide enough flexibility to express implicitly or explicitly what we wish about pathological motion. The proposed detection method belongs to the category of Markovian models within a Bayesian framework. Computational efficiency is sought by the incorporation of multiscale techniques. This is described along with our definition of pathological motion in chapter 5. For the correction step, the unknowns are not binary any more and a larger spatial neighbourhood should be involved to model the underlying original sequence. We therefore turn for our correction scheme to non-parametric Markov random fields inspired by texture synthesis approaches. This scheme is presented in chapter 6.

Before detailing the proposed techniques, we now describe the probabilistic tools that will be used in the remainder of the thesis.

Chapter 4

Probabilistic tools in image analysis

This chapter aims at giving an overview of the probabilistic tools commonly used in image processing and in particular Markovian models, which allow to express non-linear probabilistic interactions. These models, known for long in statistical physics, gained wide popularity among the signal processing and computer vision communities thanks to the pioneering article from Geman and Geman [Gem 84]. They have since been used in a variety of problems.

This chapter reviews the main definitions and algorithms assuming that the reader is familiar with the basic concepts of probability theory. For an in-depth introduction to these concepts, see for example [Sap 90], [All 90] or [Bre 99].

4.1 Markov Random Fields

4.1.1 Basic introduction to graph theory

Graph theory is a field of mathematics which has applications in a wide variety of domains [Ber 91] and is especially popular in artificial intelligence and for network modelling. A graph can usefully represent structure and connections in a generic way and express relations and dependencies between different elements.

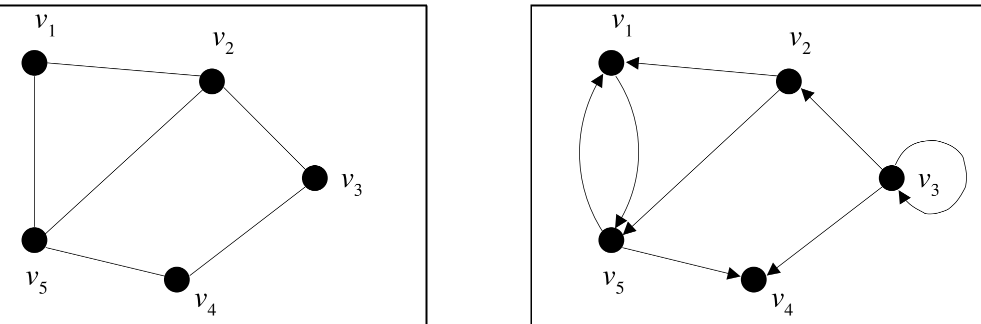
4.1.1.1 Definition and representation of a graph

A graph $G = (V, E)$ is defined by:

- a finite or denumerable set $V = \{v_1, v_2, ...\}$, the elements of which are called *vertices*, *nodes* or *sites*. In most applications, V is a finite set comprised of $N = |V|$ elements;

- a set E of pairs of elements from V called either *edges* or *arcs*.

A graph is said to be *directed* (or called a *digraph*) if elements of E are considered as *ordered* pairs. In this case, elements of E are called arcs. The graph is referred to as *undirected* if E is considered to be a set of *unordered* pairs. These pairs are then called edges. A graph can be graphically represented by dots connected by lines (for undirected graphs) or by arrows (for directed graphs). Figure 4.1 shows examples of such graphical representations.



$$V = \{v_1, v_2, v_3, v_4, v_5\}$$

$$E = \big\{\{v_1, v_2\}, \{v_2, v_3\}, \{v_2, v_5\}, \{v_3, v_4\}, \{v_4, v_5\}, \{v_1, v_5\}\big\}$$

(a) Undirected graph defined by (V, E)

$$V = \{v_1, v_2, v_3, v_4, v_5\}$$

$$E = \big\{\{v_2, v_1\}, \{v_3, v_2\}, \{v_2, v_5\}, \{v_3, v_4\}, \{v_5, v_4\}, \{v_1, v_5\}, \{v_5, v_1\}, \{v_3, v_3\}\big\}$$

(b) Directed graph defined by (V, E)

Figure 4.1: examples of undirected and directed graphs

4.1.1.2 Multigraphs and simple graphs

A *multigraph* is a graph which can contain the same arc or edge more than once. A *loop* is a node connected with itself, i.e. an element of E of the form $\{v, v\}, v \in V$. A graph is said to be *simple* if:

- it does not contain multiple edges or arcs,
- it does not contain loops.

As most applications involve simple graphs, we shall restrict in the following to this category.

4.1.1.3 Neighbourhood system

For undirected graphs, two nodes v and w are said to be neighbours if $\{v, w\} \in E$. The set of edges E defines a *neighbourhood system* $\mathcal{N}$ on V:

$$\begin{aligned} \mathcal{N} : V &\longrightarrow \mathcal{P}(V) \\ v &\longrightarrow \mathcal{N}_v = \big\{w \in V | \{v, w\} \in E\big\} \end{aligned}$$

The subset $\mathcal{N}_v$ is called the neighbourhood of node v. Equivalently, any family $\mathcal{N} = \{\mathcal{N}_v\}_{v \in V}$ of subsets of V having the following properties:

- $\forall v \in V, v \notin \mathcal{N}_v$ (4.1)
- $\forall (v, w) \in V \times V,\ w \in \mathcal{N}_v \iff v \in \mathcal{N}_w$ (4.2)

uniquely defines a simple undirected graph on V. The graph $G = (V, E)$ can thus equivalently be noted $G = (V, \mathcal{N})$. The graph is said to be *complete* if all nodes are mutually neighbours of each other, i.e. $\forall (v, w) \in V \times V,\ w \neq v \Rightarrow w \in \mathcal{N}_v$.

For directed graphs, we can very similarly define the notions of *in-neighbourhood* and *out-neighbourhood*.

4.1.1.4 Neighbourhood degree

The degree $d(v)$ of a node v in an undirected graph is the number of its neighbours:

$$d(v) = |\mathcal{N}_v| \tag{4.3}$$

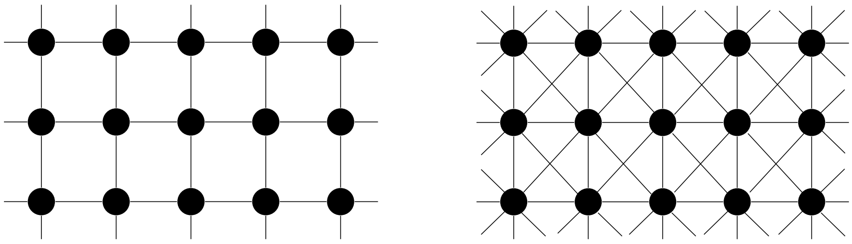
Similarly, the in-degree and out-degree of each node can be defined for directed graphs.

4.1.1.5 A particular type of graph: grids

Grids are a very important type of graph in image and video processing. A grid of dimension d (usually 2 ou 3) is a graph such as V is a part of $\mathbb{Z}^d$ and for which each node can consequently be identified with the vector of its integer coordinates. This allows to associate a distance function, usually the L^2 distance, to the considered grid. It is then possible to define neighbourhood systems from this distance as “successive layers” of nearest nodes: the first-order neighbours of a node are all surrounding nodes at the smallest distance; the k-order neighbours are recursively defined as the union of the $(k-1)$-order neighbours with all the nearest nodes that are not among the $(k-1)$-order neighbours. The most widely used neighbourhood systems on grids are the first-order and second-order neighbourhoods for the L^2 distance. In the case of 2-D grids, they are commonly referred to as 4-neighbourhood and 8-neighbourhood systems for obvious reasons (see figure 4.2).

4.1.1.6 Subgraphs

Let V_s denote a subset of V. $G_s = (V_s, E_s)$ is said to be the *subgraph* of $G = (V, E)$ generated by V_s if $E_s = (V_s \times V_s) \cap E$. G_s is simply the graph the nodes of which are elements of V_s and the edges (or arcs) of which are the edges (or arcs) of G having their two ends in V_s.



(a) (b)

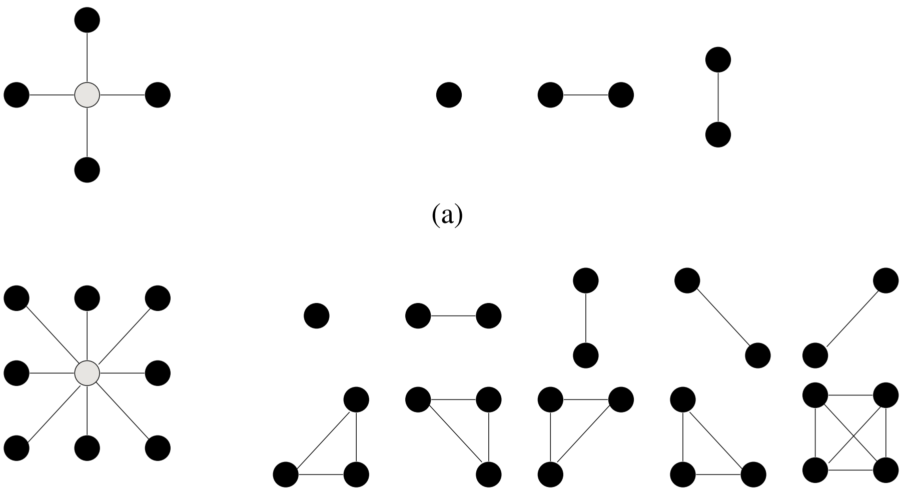
Figure 4.2: 2-D grids with (a) 4- and (b) 8-neighbourhood systems

4.1.1.7 Cliques

A clique c for the graph $G = (V, \mathcal{N})$ is a subset of V such as:

- either c consists of a single site,
- or all pairs of sites in c are mutual neighbours, i.e. $\forall \{v, w\} \subset c, w \neq v \Rightarrow w \in \mathcal{N}_v$.

In other words, c is a clique if the subgraph generated by c is complete. A clique c is called *maximal* if for any site $v \notin c$, $c \cup \{v\}$ is not a clique. The set of all cliques of graph G is denoted by $\mathcal{C}$. Figure 4.3 shows the cliques associated to the 4- and 8-neighbourhood systems in 2-D grids. As will be seen later, this notion of clique is very important in Markovian models.



(a)

(b)

Figure 4.3: cliques associated to (a) 4- and (b) 8-neighbourhood systems

4.1.2 Random fields

In the following, given definitions will be "weak" definitions, in the sense that they will be expressed without summoning explicitly notions of measure theory such as σ-algebras (also called σ-fields or tribes), measurable spaces and probability measures. For mathematically accurate definitions involving these notions of measure theory, the reader is referred to [Per 93], chap. 1.

Let $V = \{v_1, v_2, ...\}$ be a finite or denumerable set. Let us associate to each element $v \in V$ a random variable X_v taking its values in a set Λ_v. A *Random Field* (RF) on V is a collection $X = \{X_v\}_{v \in V}$ of such random variables. The sets Λ_v are in practice often taken to be the same set Λ called the *state space*. A random field is said to be discrete-valued or continuous-valued, scalar or vector depending on the nature of the state space Λ. Typical state spaces include:

- $\Lambda = \{0, 1\}$ for binary fields,
- $\Lambda = \{1, ..., M\}$ for labels involving M possible classes,
- $\Lambda = \{0, ..., 255\} \subset \mathbb{N}$ for greyscale quantized images,
- $\Lambda = [-D_{\max}, D_{\max}] \times [-D_{\max}, D_{\max}] \subset \mathbb{R}^2$ for motion fields.

A sample realisation $x = (x_v)_{v \in V}$ where $\forall v \in V, x_v \in \Lambda$ is said to be a *configuration* of the random field X. The set $\Omega = \Lambda^{|V|}$ of all possible configurations is called the *configuration space*.

4.1.3 Graphical models

Graphical models can be seen as a marriage between probability theory and graph theory [Mur 01]. A graphical model is a simple graph (i.e. with no loops and no multiple edges or arcs) in which:

- the set of nodes (often called sites in this case) represents a random field,
- the lack of arcs or edges between nodes represents conditional independence assumptions.

Graphical models fall into two main classes, those based on directed graphs and those based on undirected graphs. Directed graphical models are commonly known as *Bayesian Networks* or *Belief Networks* (BN). They are used to express causal relations represented by the direction of the arcs. It should be noted that despite their names, Bayesian Networks do not necessarily have any direct relation with the notion of Bayesian inference that will be defined later. Graphical models based on undirected graphs are usually known as *Markov Random Fields* (MRF).

4.1.4 Markov Random Fields and Gibbs Random Fields

The development of Markov random fields historically owes a lot to statistical physics (i.e. the science of the transition from local interactions between particles and molecules to the macroscopic properties of matter). For this reason, much of the vocabulary used originates from this domain. Markov random fields can be seen as a generalization of 1-D causal Markov chains based on the same notion of *conditional independence*. The key difference however is that the principle of causality is lost.

For the sake of clarity, all definitions and properties will be expressed in the following for discrete-valued random fields. They all extend to the case of continuous variables by replacing summations by integrals and probability masses by probability density functions (pdf).

4.1.4.1 Markov Random Fields

The random field $X = \{X_v\}_{v \in V}$ is called a *Markov Random Field* (MRF) with respect to the neighbourhood system $\mathcal{N} = \{\mathcal{N}_v\}_{v \in V}$ if

$$P(X_v = x_v | X_w = x_w, w \neq v) = P(X_v = x_v | X_w = x_w, w \in \mathcal{N}_v) \tag{4.4}$$

for all $v \in V$ and $x = (x_v)_{v \in V} \in \Omega$.

The Markovian property (4.4) states that the probability distribution for one site given the value of its neighbours is independent of the rest of the field. It expresses the fact that all the information about one variable is carried by the value of its neighbours. This does not mean that two variables on sites that are not neighbours are independent of each other: all variables are in general mutually dependent, but only through the combination of successive *local interactions*. These local interactions are specified by the *local conditional probabilities* $P(X_v = x_v | X_w = x_w, w \in \mathcal{N}_v)$. It should be noted that any random field is Markovian with respect to the trivial complete topology, where $\forall v \in V, \mathcal{N}_v = V \setminus \{v\}$. However, interesting MRFs in practice are those with very small neighbourhoods for computational reasons that will be detailed later.

The specification of an MRF from this definition is far from convenient: there is no obvious intuitive relation between the local conditional probabilities and the joint global probability $P(X = x)$ governing the behaviour of the whole field. This is addressed by the Hammersley-Clifford theorem as will be seen later.

4.1.4.2 Positivity condition

An MRF X is said to satisfy the *positivity condition* if all possible configurations have a non-zero probability:

$$\forall x \in \Omega, \quad P(X = x) > 0 \tag{4.5}$$

This condition is sometimes considered to be an integral part of the definition of an MRF.

4.1.4.3 Gibbs Random Fields

The random field $X = \{X_v\}_{v \in V}$ is said to be a *Gibbs Random Field* (GRF) with respect to the neighbourhood system $\mathcal{N} = \{\mathcal{N}_v\}_{v \in V}$ if the global probability distribution of X can be expressed as

$$P(X = x) = \frac{1}{Z} \exp\Big(- \sum_{c \in \mathcal{C}} V_c(x) \Big) \tag{4.6}$$

where $\mathcal{C}$ is the set of all cliques associated to $\mathcal{N}$ and the collection of functions $\{V_c\}_{c \in \mathcal{C}}$, called *potentials*, is such as each V_c depends only on the values at sites $v \in c$.

The form of the distribution in (4.6) is called a *Gibbs distribution*. Z is a normalizing constant known as the *partition function*:

$$Z = \sum_{x \in \Omega} \exp\Big(- \sum_{c \in \mathcal{C}} V_c(x) \Big) \tag{4.7}$$

The function U defined for any configuration in Ω as

$$U(x) = \sum_{c \in \mathcal{C}} V_c(x) \tag{4.8}$$

is the *energy function* associated to the potentials $\{V_c\}_{c \in \mathcal{C}}$. For a GRF, the most likely configuration is the one having the lowest energy, while low probabilities are associated to configurations with high energies.

4.1.4.4 Hammersley-Clifford Theorem

Let $X = \{X_v\}_{v \in V}$ be a random field on a finite set V. X is a Markov random field on $G = (V, \mathcal{N})$ verifying the positivity condition (4.5) if and only if X is a Gibbs random field on G.

This important theorem states the *equivalence between the concepts of Markov and Gibbs random fields*. The major practical consequence of this theorem is that an MRF is completely specified by:

- its neighbourhood system,
- a family of potentials over the cliques of this neighbourhood.

The specification of these *local characteristics* is sufficient to completely determine its *global behaviour*, characterised by a Gibbs distribution. The choice of the potential functions should be practically based on intuition of the desirable properties. It can be shown that the local conditional probabilities derived from these potentials are

$$P(X_v = x_v | X_w = x_w, w \in \mathcal{N}_v) = \frac{1}{Z(x_{\mathcal{N}_v})} \exp\Big(- \sum_{c \in \mathcal{C} | v \in c} V_c(x) \Big) \tag{4.9}$$

where

$$Z(x_{\mathcal{N}_v}) = \sum_{x_v \in \Lambda} \exp\Big(- \sum_{c \in \mathcal{C} | v \in c} V_c(x) \Big) \tag{4.10}$$

This important theoretical theorem was originally proved in [Bes 74] based on Hammersley and Clifford's unpublished paper of 1971. Alternative proofs can be found for example in [Bre 99], chap. 7.2 or [Per 93], appendix B for the case of a finite state space Λ.

4.1.4.5 Complementary definitions

An MRF is said to be *stationary* or *homogeneous* if the local conditional probabilities $P(X_v = x_v | X_w = x_w, w \in \mathcal{N}_v)$ are independent of the considered site v. This implies in particular that all sites must have the same neighbourhood degree. For such a field, only a limited number of clique potentials needs to be defined. A stationary MRF can be seen as "translation-invariant". MRFs defined on finite grids are also commonly termed stationary when this property is true except at the grid boundaries, even though this does not strictly comply with the definition of stationarity.

An MRF is said to be *isotropic* if the potentials $\{V_c\}_{c \in \mathcal{C} | v \in c}$ for cliques containing any given site v only depend on the total number of sites $|c|$ in these cliques. For example, an MRF on a 2-D grid with a first-order neighbourhood system (see figure 4.3(a)) is isotropic if the potential functions associated to the horizontal and vertical pair cliques are the same. This can be similarly seen as a property of "rotational invariance".

4.1.4.6 Local smoothness, the example of the Ising model

The *Ising model* is a very standard example which gives an idea of the kind of local interaction that can be expressed with MRFs. It was introduced in the twenties in the context of ferromagnetism. Each site can take a binary value in the state space $\Lambda = \{-1, 1\}$ which models the orientation of an elementary magnetic dipole. In the absence of an external magnetic field, the global energy assigned to each configuration is:

$$U(x) = -J \sum_{\{v,w\} \in \mathcal{C}} x_v x_w \quad \text{with } J > 0$$

For each pair of neighbouring sites, the potential is equal to J if the two spins have the same orientation and to $-J$ if they have a different orientation. As $J > 0$, the interactions statistically favour identical neighbours. On a more general basis, this kind of interaction implying that two neighbouring sites should usually have close values can be seen as *local smoothness* interactions. They are commonly employed for any type of state space Λ and can be enforced by choosing potential functions on pair cliques that are distance functions.

4.1.5 Monte Carlo simulation techniques

4.1.5.1 Sampling from a Gibbs distribution

Once an MRF model is defined, a key necessity for its practical use is the ability to draw samples from its Gibbs distribution $P(X = x)$. Unfortunately, the very high dimensionality of Ω in typical problems makes the expression (4.6) of $P(X = x)$ very difficult to handle. In particular, computing the normalizing constant Z in (4.7) would require an exhaustive visit of Ω and is therefore practically completely out of reach. On the other hand, the local conditional probabilities (4.9) are much easier to handle since they involve summations over Λ. For this reason, techniques developed to sample successfully from $P(X = x)$ rely on the locality of the model. These simulation methods belong to the class of *Markov Chain Monte Carlo (MCMC)* techniques.

4.1.5.2 Principle of Monte Carlo simulation

The principle of MCMC simulation is to construct a Markov chain admitting $\pi(x) = P(X = x)$ as a stationary distribution (for a review of the concept of Markov chain, the reader is referred to [Res 92] or [Bre 99]). Each state of the constructed Markov chain corresponds to a global configuration of our MRF. At each transition, the value of a single site of the field is changed and this change is based on the local conditional probabilities. An MCMC simulation algorithm will thus be characterised by:

- an update algorithm, i.e. how the transition of a site will be decided according to the local conditional probabilities,
- a scanning strategy, i.e. in what order the sites will be visited.

If these characteristics are well-chosen, it can be shown that the constructed Markov chain $\{X(t)\}_{t\in\mathbb{N}}$ is irreducible, aperiodic and admits the target distribution π as its stationary distribution independently of the initial configuration x_0:

$$\lim_{t\to+\infty} P(X(t) = x | X(0) = x_0) = \pi(x) \tag{4.11}$$

Practically, the simulation has to be stopped after a finite number of runs which should be large enough. Deciding whether we are "close enough" to the equilibrium distribution from a numerical point of view is not a simple issue.

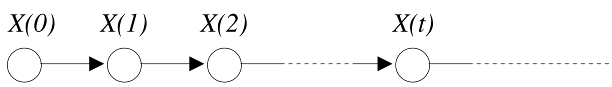


Figure 4.4: Markov chain constructed for Monte Carlo simulation

For the scanning strategy, there are many possibilities. Theoretical convergence is guaranteed if each site is visited "infinitely often". The most common strategies are random visiting and raster scan visiting, with a forward pass followed by a backward pass in order to reduce the bias that could be introduced by a specific propagation direction. Two very common update algorithms are now presented without detailing the proof of the desirable convergence of the associated Markov chain: the *Gibbs sampler* and the *Metropolis algorithm*. A more general description of these methods can be found in [Rua 96].

4.1.5.3 Gibbs sampler

The Gibbs sampler, introduced in [Gem 84] is the most popular algorithm for drawing samples from an MRF. This algorithm is applicable when the state space Λ is a finite set. It can be readily extended to continuous-valued fields if samples can be drawn directly from the local conditional distribution or by discretizing the continuous state space. The transition for each site is here directly based on the local conditional distribution expressed by equation (4.9):

$$P(X_v = x_v | X_w = x_w, w \in \mathcal{N}_v) = \frac{1}{Z(x_{\mathcal{N}_v})} \exp\Big(- \sum_{c \in \mathcal{C} | v \in c} V_c(x) \Big) \tag{4.12}$$

Let us assume that we are in configuration $X(t) = x$ after t iterations and that v is the site being updated at iteration $t+1$. The update is conducted as follows:

- for all possible values $\lambda \in \Lambda$, $P(X_v = \lambda | X_w = x_w, w \in \mathcal{N}_v)$ is computed;
- the new value λ_a for site v is drawn from this distribution. The new configuration $X(t+1)$ is therefore such as $X_v(t+1) = \lambda_a$ and $\forall w \neq v, X_w(t+1) = X_w(t)$.

This second step is practically carried out by dividing the real interval [0,1] into $|\Lambda|$ segments, each of which having a length proportional to $P(X_v = \lambda | X_w = x_w, w \in \mathcal{N}_v)$. A number a is then randomly drawn between 0 and 1 according to a uniform distribution and the new value λ_a is defined by the segment to which a belongs.

4.1.5.4 Metropolis algorithm

For the Metropolis (also known as Metropolis-Hastings) algorithm, unlike what is the case for the Gibbs sampler, the transition of the site being updated depends on its current value. As previously, we consider the global configuration $X(t)$ after t iterations and update of site v at iteration $t+1$. A value $\lambda_a \in \Lambda$ is drawn from a uniform distribution. The new proposed configuration is x' such as $x'_v = \lambda_a$ and $\forall w \neq v$, $x'_w = x_w$. This proposed configuration is accepted or rejected depending on the energy difference:

$$\Delta U = U(x') - U(x) = \sum_{c \in C | v \in c} [V_c(x') - V_c(x)] \tag{4.13}$$

We are faced with two possibilities:

- if $\Delta U \leq 0$, then $X(t+1) = x'$, the proposed configuration is accepted;

- if $\Delta U > 0$, then $\begin{cases} X(t+1) = x' & \text{with probability } \exp(-\Delta U) \\ X(t+1) = x & \text{with probability } 1 - \exp(-\Delta U) \end{cases}$

When we are in the latter case ($\Delta U > 0$), another random number is drawn between 0 and 1 to decide whether the new state is accepted or rejected.

As in the case of the Gibbs sampler, an increase in energy is possible at each transition. The computational cost of both algorithms is proportional to the number of cliques containing site v. This explains why computationally tractable MRFs are those involving small neighbourhoods: the number of cliques increases considerably with the degree of the graph supporting the MRF. Both algorithms are suited to massive parallelization: many sites can be synchronously updated provided they are not neighbours of each other.

4.2 Bayesian theory of estimation

4.2.1 Inverse problems and statistical inference

In many computer vision and signal processing problems, we are interested the physical properties of a given system. This system is indirectly analysed by means of sensors which record data. The nature of the recorded information can vary very widely depending on the specific problem: it can be for example visible light for photography or video, other electromagnetic waves in medical imaging, remote sensing, astrophysics or microscopy, air vibrations for sound, seismic waves in geophysics and many others. All these problems can be seen in a unifying perspective as attempting to extract information about unknown properties of the system from the set of observed data. This type of problem is known as an *inverse problem*.

An analysis of the physical phenomena involved allows to define a *model* describing the formation process of the observations and how they are influenced by the unknown physical properties. When the phenomena are modelled as a stochastic process, recovering information about the unknowns from the set of observations is known as *statistical inference* (see figure 4.5). Unfortunately, because the nature of the sought properties is often more complex than what is measured, the observation process is in general associated with a *considerable loss of information*, which is the source of uncertainty. In particular, this is typically the case when we are interested in the 3-D characteristics of a world recorded on a 2-D surface. In addition, the sensors themselves and the subsequent recording process may introduce distortions of the observed data, e.g. noise. For these reasons, the sole model of the link between the unknowns and the observations, either deterministic or probabilistic, is usually not sufficient to fully extract the desired information. We are faced with what is called *ill-posed problems*. In this case, it becomes necessary to incorporate additional assumptions and knowledge about the unknown properties themselves in order to remove the ambiguities. The integration of such constraints in the information extraction process is called *regularization*.

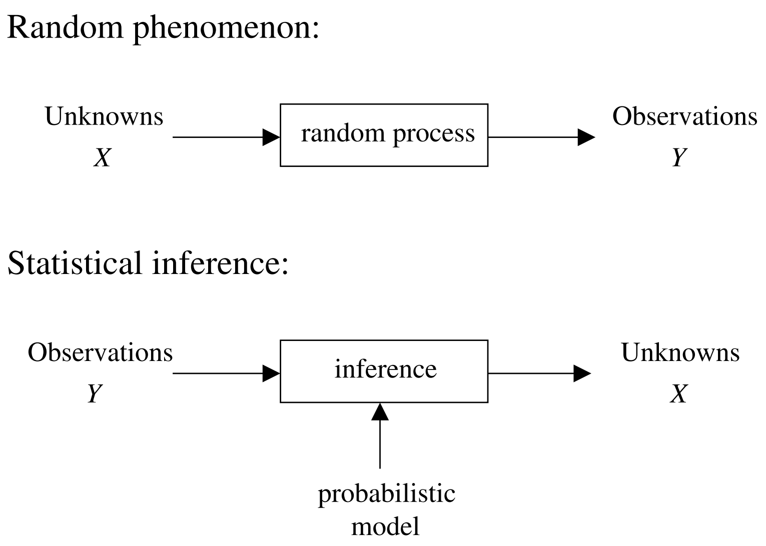


Figure 4.5: principle of statistical inference

4.2.2 Bayesian inference

As stated previously, the general purpose of statistical inference is to extract information about the unknown properties when our model of the cause-effect relationship is probabilistic. *Bayesian inference* is specifically based on the concept of the *inversion of probabilities*.

In the Bayesian framework, we perform a semantic drift: our point of view is changed from a fixed but unknown cause to a random and "probabilized" cause. The regularization is then introduced by defining a prior probability for these unknown properties. Bayesian inference therefore revolves around two points:

- the probabilistic model linking the unknowns to the observations. It is expressed as the conditional distribution of the observations given the unknowns $P(Y = y|X = x)$. This conditional probability is commonly referred to as the *likelihood* of the observations;

- the introduction of prior knowledge about the unknowns for regularization purpose. This is done by specifying a *prior distribution* on the unknowns $P(X = x)$. This prior distribution is designed to express in a generic way what types of configurations are intuitively more likely than others. In the worst case where we have absolutely no prior expectation, it can be chosen as the uniform distribution when possible.

From these two distributions and from the well-known Bayes' theorem, we are able to derive the *posterior* distribution of the unknowns conditional on the observations:

$$\begin{aligned} P(X=x|Y=y) &= \frac{P(X=x,Y=y)}{P(Y=y)} \\ &= \frac{P(Y=y|X=x)P(X=x)}{P(Y=y)} \end{aligned}$$

which can be written as

$$P(X=x|Y=y) = \frac{1}{Z(y)}P(Y=y|X=x)P(X=x) \tag{4.14}$$

4.2.3 Optimal Bayesian estimators

In the great majority of cases, the information we want to extract about the unknowns is no less than their numerical value. This is called *estimation* and more specifically point estimation. An estimator $\hat{x}$ is a function which associates to a given set of observations y an estimated value $\hat{x}(y)$ of the unknowns. In other words, it can be seen as a "guess" of the value of the unknowns.

How should this estimator be chosen, based on the posterior distribution, in order to be the "best possible"? An optimal estimator can only be defined in conjunction with what we consider as "small and large errors" between the estimated value and the actual value. This is specified by a *cost function.*

4.2.3.1 Cost function

We will here consider the most general case where X is a random field on a set V with the notations defined in section 4.1.2. Y is a random field on the same or a different set of sites, with a configuration space denoted Ω_{obs}.

A cost function C is a distance defined on Ω. The performance of an estimator $\hat{x}$ can then be measured by $C(x,\hat{x})$. The risk associated to a given estimator $\hat{x}$ is defined as the average cost over all possible observations and unknowns:

$$\begin{aligned} R(\hat{x}) &= \sum_{y\in\Omega_{\text{obs}}}\sum_{x\in\Omega} C(x,\hat{x}(y))P(X=x,Y=y) \\ &= \sum_{y\in\Omega_{\text{obs}}}\left(\sum_{x\in\Omega} C(x,\hat{x}(y))P(X=x|Y=y)\right)P(Y=y) \end{aligned}$$

The *optimal Bayesian estimator* $\hat{x}^*$ with respect to the cost function C is defined as the estimator which minimizes the risk R. From the formulation above, it can be easily seen that such an estimator must minimize the marginal expected cost for each possible observation:

$$\hat{x}^*(y) = \arg\min_{x' \in \Omega} \left(\sum_{x \in \Omega} C(x, x') P(X = x | Y = y) \right) \tag{4.15}$$

Several cost functions are in common use in the literature. Three popular examples are given here along with the corresponding optimal Bayesian estimators. For the derivation of these estimators from equation (4.15), see [Per 93], chap. 2.2 and [Mar 85], chap. 3.3 and 3.4 for more details.

4.2.3.2 Maximum A Posteriori (MAP)

Let us first consider the "hit-or-miss" cost function:

$$C(x, x') = 1 - \delta(x - x') \tag{4.16}$$

where

$$\delta(a) = \begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$$

This cost function is rather crude as it makes no distinction between all configurations x' different from x. It can be shown that the corresponding optimal Bayesian estimator is the maximizer of the posterior distribution:

$$\hat{x}^{\mathrm{MAP}}(y) = \arg\max_{x \in \Omega} P(X = x | Y = y) \tag{4.17}$$

This estimator, known as the *Maximum A Posteriori* (MAP) estimator, selects the most "likely" configuration given the observations y. Despite the "brutality" of the cost function, it remains the most employed in practice for computational reasons that will be detailed later.

4.2.3.3 Maximizer of the Posterior Marginals (MPM)

The slightly more sophisticated cost function:

$$C(x, x') = \sum_{v \in V} \left(1 - \delta(x_v - {x'}_v)\right) \tag{4.18}$$

counts the number of sites on which x and x' are different. The associated optimal estimator, the *Maximizer of the Posterior Marginals* (MPM) is defined by:

$$\forall v \in V, \quad \hat{x}_v^{\mathrm{MPM}}(y) = \arg\max_{x_v \in \Lambda} P(X_v = x_v | Y = y) \tag{4.19}$$

This estimator was first introduced in [Mar 85].

4.2.3.4 Mean Field (MF)

Let us finally consider the quadratic cost function:

$$C(x, x') = \sum_{v \in V} (x_v - {x'}_v)^2 \tag{4.20}$$

The optimal estimator in this case can be shown to be

$$\forall v \in V, \quad \hat{x}_v^{\mathrm{MF}}(y) = \sum_{x \in \Omega} x_v P(X = x | Y = y) \tag{4.21}$$

As its name implies, this *Mean Field* (MF) estimator is the expectation of X conditional on the observations $Y = y$.

4.2.4 Summary

The different steps of Bayesian estimation explained in section 4.2 are summarized in figure 4.6.

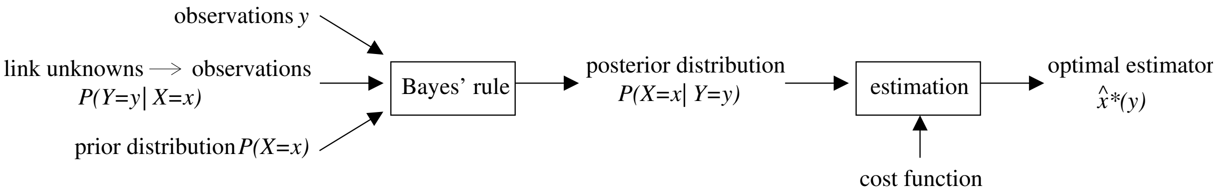


Figure 4.6: overview of Bayesian estimation

4.3 Markovian models within a Bayesian framework

In the previous section, we have reviewed the general principle of Bayesian estimation. The combination of this framework with Markovian models proves very fruitful: as will be shown now, the posterior distribution has in this case the desirable property to follow as well a Gibbs distribution. It is then possible to take advantage of this specific form and to rely on MCMC sampling techniques to compute the optimal Bayesian estimators.

4.3.1 Nature of the posterior probability

Let us consider that the field X is a Markov random field. The prior distribution can then be expressed as a Gibbs distribution (4.6):

$$P(X = x) = \frac{1}{Z_0} \exp\Big(- \sum_{c \in \mathcal{C}} V_c(x) \Big) \tag{4.22}$$

For the sake of simplicity, we shall consider that the observations are located on the same set of sites V and that during the formation of the observations, observation y_v on site v is only influenced by unknown x_v on the very same site. However, the following reasoning can be generalized to more complex cases: observations and unknowns can be located on different sites and a single observation can be affected by a subset of V not restricted to a single site (e.g. a blurring process) [Gem 84]. When assumed to be Markovian, the link between the observations and the unknowns can in our simple case be defined as

$$\begin{aligned} P(Y=y|X=x) &= \prod_{v\in V} P(Y_v=y_v|X=x) \\ &= \prod_{v\in V} P(Y_v=y_v|X_v=x_v) \\ &= \exp\Big(-\sum_{v\in V} W_v(x_v,y_v)\Big) \end{aligned} \tag{4.23}$$

with $W_v(x_v,y_v) = -\ln P(Y_v=y_v|X_v=x_v)$

Using Bayes' rule (4.14) with (4.22) and (4.23), the posterior distribution can be written as

$$\begin{aligned} P(X=x|Y=y) &= \frac{1}{Z_y} P(Y=y|X=x)P(X=x) \\ &= \frac{1}{Z_y}\frac{1}{Z_0}\exp\Big(-\sum_{c\in\mathcal{C}} V_c(x) - \sum_{v\in V} W_v(x_v,y_v)\Big) \\ &= \frac{1}{Z}\exp\Big(-U(x,y)\Big) \end{aligned} \tag{4.24}$$

with

$$U(x,y) = \sum_{c\in\mathcal{C}} V_c(x) + \sum_{v\in V} W_v(x_v,y_v) \tag{4.25}$$

This posterior distribution has the form of a Gibbs distribution with respect to the prior neighbourhood system of field X. Therefore, *the field of the unknowns conditional on the observations $Y=y$ is a Markov random field* with respect to this neighbourhood system. When a single observation is influenced by several unknowns at different sites, the posterior neighbourhood system is bigger than the prior one.

4.3.2 MAP optimization

As we are faced with a Markov random field, MCMC techniques can be used to draw samples from this posterior distribution. Let us consider m samples $x^1, x^2, ..., x^m$, not necessarily consecutive, drawn from the Markov chain constructed for this purpose, e.g. with the Metropolis algorithm or the Gibbs sampler. Because of the ergodicity of the underlying Markov chain, the MF estimator (4.21) can then be approximated by

$$\forall v \in V, \quad \hat{x}_v^{\mathrm{MF}}(y) \approx \frac{1}{m}\sum_{i=1}^{m} x_v^i \tag{4.26}$$

and the MPM estimator (4.19) is approximated at each site by the value appearing the most frequently:

$$\forall v \in V, \quad \hat{x}_v^{\mathrm{MPM}}(y) \approx \arg\max_{x_v \in \Lambda} \sum_{i=1}^{m} \delta(x_v^i - x_v) \tag{4.27}$$

As these estimators require drawing a large number of samples from the posterior Gibbs distribution, the MAP estimator is usually preferred.

As seen in equation (4.17), the MAP estimator is the configuration that maximizes the posterior probability. In this case, it is equivalent to minimizing the energy function (4.25) on Ω. As this energy function $U(x, y)$ usually has many local minima, this is a difficult optimization problem, complicated further by the very high dimensionality of Ω. For this reason, iterative algorithms updating one site at a time based on the local variations of the energy function must be used. They fall into two categories: *stochastic methods* based on simulated annealing combined with MCMC sampling techniques or *deterministic methods* that reach a local minimum depending on their initialization.

4.3.2.1 Simulated Annealing (SA)

The principle of Simulated Annealing (SA) is to modify a standard Monte Carlo sampling procedure (e.g. the Metropolis algorithm or the Gibbs sampler) in order to converge to the desired minimum of energy. To that end, a *temperature parameter* T is introduced in the target Gibbs distribution:

$$\pi_{T,y}(x) = \frac{1}{Z_T(y)} \exp\left(- \frac{U(x, y)}{T} \right) \tag{4.28}$$

When $T = 1$, this is the posterior probability (4.24). When $T \to 0$, it can be shown that $\pi_{T,y}$ converges in probability to an impulse at the configuration(s) of minimum energy. The idea behind simulated annealing is to decrease progressively the temperature while we are sampling from the modified Gibbs distribution. An *annealing schedule* is defined as a series of temperatures $(T(t))_{t \in \mathbb{N}}$ such as $\forall t \in \mathbb{N}, T(t+1) \leq T(t)$ and $\lim_{t \to +\infty} T(t) = 0$. This annealing schedule is combined with the chosen MCMC technique: at iteration t, the site under consideration is updated for the Gibbs distribution (4.28) with temperature $T(t)$. Intuitively, this can be understood as progressively "stretching" the original energy and amplifying the differences between energy "valleys" and "peaks" in order to ensure that the drawn samples will get "trapped" in the global minimum.

[Gem 84] shows that if (i) there is an upper bound on the time interval between two consecutive visits of any site and (ii) the cooling rate is logarithmic, i.e. $T = T_0 / \ln(t + \mathrm{e})$ with T_0 sufficiently large, then the convergence to a configuration minimizing the original energy is guaranteed. Condition (i) imposes no limitation in practice: sites are usually visited in a forward-backward raster scan ordering ensuring that each site has been updated twice after $2|V|$ iterations. Very often, the temperature is kept to a constant value T_n during the whole sweep n before being decreased for the following sweep. Unfortunately, condition (ii) is too restrictive: the logarithmic rate is much too slow to be followed in practice. Instead, an exponential cooling schedule is most of the time preferred, with $T_n = T_0 a^n$ and $0 < a < 1$. Although simulated annealing algorithms with this kind of cooling schedule are outside the scope of the theoretical conditions of convergence, they usually give very good results that are largely insensitive to the initial configuration. However, they remain rather slow.

4.3.2.2 Iterated Conditional Modes (ICM)

In order to reduce the amount of computation, deterministic methods rely on transitions between configurations that do not involve random sampling. They systematically decrease the global energy at each update. The ICM algorithm (for Iterated Conditional Modes) introduced by Besag [Bes 86] and its variants are the most widespread deterministic methods. They perform a succession of local optimizations in replacement of the global optimization problem.

In the basic ICM algorithm, the new value for visited site v is the value which maximizes the local conditional probability

$$\begin{aligned} P(X_v = x_v | Y = y, X_w = x_w, w \in \mathcal{N}_v) &= \frac{1}{Z(x_{\mathcal{N}_v}, y)} \exp\Big(- \sum_{c \in \mathcal{C} | v \in c} V_c(x) - W_v(x_v, y_v)\Big) \\ &= \frac{1}{Z(x_{\mathcal{N}_v}, y)} \exp\Big(- U_v(x, y)\Big) \end{aligned}$$

or equivalently the value which minimizes the local energy

$$U_v(x, y) = \sum_{c \in \mathcal{C} | v \in c} V_c(x) + W_v(x_v, y_v) \tag{4.29}$$

The algorithm therefore performs the update as follows:

- for all possible values $\lambda \in \Lambda$, $U_v(x^{\lambda,v}, y)$ is computed where $x_v^{\lambda,v} = \lambda$ and $\forall w \neq v, x_w^{\lambda,v} = x_w$;
- the new value λ_a for site v is such as $\lambda_a = \arg\min_{\lambda \in \Lambda} U_v(x^{\lambda,v}, y)$.

It should be noted that this algorithm implies exploring the whole state space Λ at each iteration, as is also the case for simulated annealing coupled with Gibbs sampling. As the global energy is systematically decreased, convergence is reached after a finite number of iterations. However, the configuration at convergence corresponds to a local minimum of energy in the sense that we cannot decrease the energy any more by changing only a single site. The local minimum reached depends on the initial configuration and there is absolutely no guarantee that it coincides with the global minimum.

Chapter 5

Impulsive defect detection and pathological motion

As highlighted in section 3.4, the major limitation of detection methods developed so far is that they are very sensitive to motion estimation failures, which are the source of false alarms.

To overcome this problem, one possible approach is to improve further the models used for motion estimation and make them even more realistic. A great deal of effort has been devoted to this task for twenty years and a wealth of motion estimation algorithms have been and are still proposed (see notably [Sti 99], [Bar 94], [Bea 95] and [Tek 95] chap. 5 to 8 for a description of the main ones). However, motion estimation remains today a difficult and largely open problem: as acknowledged in the conclusion of [Sti 99] p. 89, “we are still far from generic, robust, real-time motion-estimation algorithms”. Each method has its weaknesses that are bound to appear on real sequences showing a bit of activity. We shall term *“pathological” motion* (PM) the events which are the source of such weaknesses and that we shall detail in the next section.

The other approach that is followed in this thesis is to take into account within the detection the undesirable failures of motion estimation, rather than attempting to get rid of them. In other words, we overcome these failures by integrating them in the detection model. This implies in particular considering a *larger temporal window* than the usual three frames as suggested in [Roo 99a], chap. 4.4.

5.1 Definition of pathological motion

Motion estimation failures are due either to natural events which represent violations of the model or to artifacts. It is precisely on this characteristic of impulsive defects that their detection relies. We therefore define pathological motion as the motion estimation failures that are not due to impulsive defects. This pathological motion is an integral part of the original document and acts as a disturbance with respect to the detection as it is the source of false alarms.

What this notion precisely encompasses depends on the considered motion estimation technique: a given motion will be pathological for some motion estimators whereas it may not be for others. However, some natural events are the source of problems for the great majority of motion estimators. Based on experimental observations and on the classification initiated in [Rar 01], we shall in particular distinguish the following events:

- *occlusion* and *uncovering*
- *intermittent motion* is a somewhat repetitive motion with a high frequency. Helicopter blades, plane propellers, flapping wings of a bird, a blinking light are typical examples of intermittent motion.
- *erratic motion*, which can be described as a fast and highly irregular motion. Flames, swirls, hair during a sudden head movement, leaves, flags or clothes swept by the wind are examples of this kind of motion.
- *motion blur* affects the appearance of objects which move too fast compared to the exposure time. This can even affect the entire scene during a very quick pan.
- *large displacements*, larger than the maximum displacement that the motion estimator is able to track. This limit depends a lot on the specific motion estimator and its parameter settings.
- *changing lighting effects*: transparency (typically with windows or glass doors), reflections (e.g. on the body of a car or on the water surface), or shadows and displacements of light sources

Our purpose is not to model each type of PM precisely, but only to find a common characteristic that allows to distinguish them from impulsive defects. We shall leave aside occlusion and uncovering which are simple isolated temporal discontinuities and present no risk to be mistaken for blotches. The principle on which we rely is that other types of pathological motion have a temporal persistence of several images. A blotch is then assimilated to a double temporal discontinuity, whereas pathological motion is considered to be a discontinuity repeated on a larger window.

The method proposed in this chapter belongs to the category of probabilistic approaches within a Bayesian framework and is more specifically inspired by Morris' work [Mor 95]. In addition to involving a larger temporal aperture, we do not consider any more that a pixel can have only two possible states, i.e. corrupted or non-corrupted but we also consider that it can be affected by pathological motion.

5.2 Proposed model

The proposed method is composed of two steps: the *localization of temporal discontinuities* on one hand, followed by their *interpretation in terms of blotches or pathological motion* on the other hand.

5.2.1 Localization of the temporal discontinuities

The graph A set of 5 frames is considered around the current frame n: this is probably the smallest temporal aperture which allows to distinguish between blotches and pathological motion. A 3-D grid is defined on this set with each site representing a pixel. We associate to this grid the first-order neighbourhood for the L^2 distance: each site which is not on the grid boundary has 4 spatial and 2 temporal neighbours. The choice of this very small neighbourhood is dictated by computational reasons: a second-order neighbourhood would certainly improve the model, but at the price of a significant computational overhead. The graph thus defined is coupled with its dual edge graph, i.e. the graph the nodes of which represent the arcs between pixels (figure 5.1). The nodes of the edge graph will be the supports for possible binary spatial and temporal discontinuities between pixels.

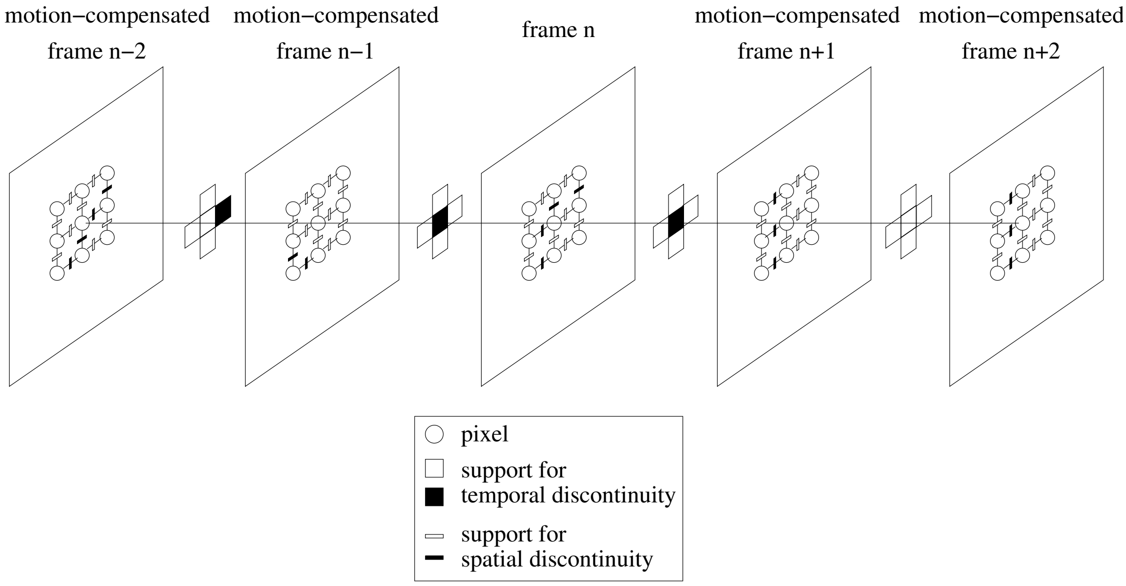


Figure 5.1: graph defined on the set of 5 frames

The Bayesian framework We consider a Bayesian framework: the unknowns in which we are interested are the temporal discontinuities B. The observations are the binary spatial discontinuities L and the greyscale intensities I motion-compensated with respect to the current frame n, i.e. simple compensation for frames $n-1$ and $n+1$, double compensation for frames $n-2$ and $n+2$. This means that we consider at a spatial location x the intensities $I_{n-2}(x + d_{n,n-1} + d_{n-1,n-2})$, $I_{n-1}(x + d_{n,n-1})$, $I_n(x)$, $I_{n+1}(x + d_{n,n+1})$ and $I_{n+2}(x + d_{n,n+1} + d_{n+1,n+2})$, where $d_{k,l}$ is the motion field from frame k to frame l. As for the observed spatial discontinuities, they can be obtained from a classic edge detector, such as [Can 86, Der 87]. We consider here the most general model, although a simpler model without any spatial discontinuity has been preferred for the experiments. With the notations of section 4.2, we then have $X = B$ and $Y = (I, L)$. The associated state spaces are respectively $\Lambda_B = \{0, 1\}$, $\Lambda_I = \{0, ..., 255\}$ and $\Lambda_L = \{0, 1\}$.

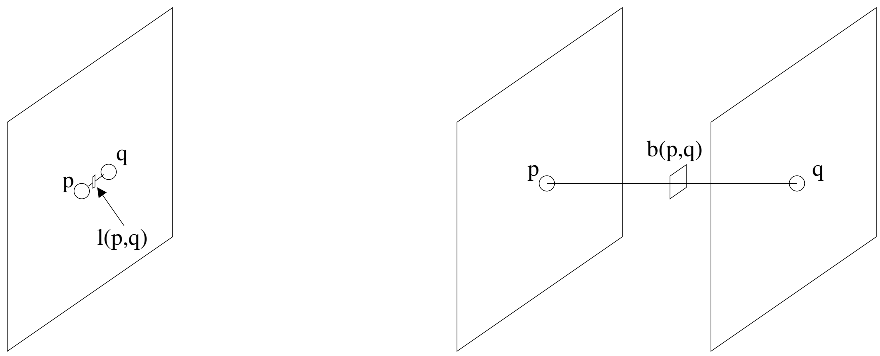
The likelihood The likelihood and the prior distribution are modelled as *stationary* and *spatially isotropic* Markov random fields (see section 4.1.4.5). The likelihood of the motion-compensated intensities knowing the discontinuities is defined by:

$$
\begin{aligned}
P(I = i|B = b, L = l) = \\
\frac{1}{Z_1(b,l)} \exp\Bigg[-\sum_{p\in V}\Bigg(\alpha'_1 \sum_{q\in\mathcal{S}(p)} (i(p)-i(q))^2(1-l(p,q)) \\
+\alpha'_2 \sum_{q\in\mathcal{T}(p)} (i(p)-i(q))^2(1-b(p,q))\Bigg)\Bigg]
\end{aligned}
\tag{5.1}
$$

with the following notations:

V is the set of pixel sites,
$\mathcal{S}(p)$ is the spatial neighbourhood of pixel p,
$\mathcal{T}(p)$ is the temporal neighbourhood of pixel p,
$l(p,q)$ and $b(p,q)$ are respectively the spatial and temporal discontinuities between sites p and q.

This distribution involves two local interactions. The first term $\alpha'_1 \sum_{q\in\mathcal{S}(p)}(i(p) - i(q))^2(1-l(p,q))$ expresses a constraint of spatial smoothness on the intensities, switched off if there is explicitly a spatial discontinuity. Similarly, the second term $\alpha'_2 \sum_{q\in\mathcal{T}(p)}(i(p) - i(q))^2(1-b(p,q))$ is its temporal counterpart and introduces temporal smoothness. Positive parameters α'_1 and α'_2 determine the relative weights of these interactions. It should be noted that the first term does not depend on the unknowns B and only involves observations.



(a) spatial smoothness weighted by α'_1 (b) temporal smoothness weighted by α'_2

Figure 5.2: local interactions in the likelihood

The prior distribution Similarly, the prior distribution on the temporal discontinuities is defined by:

$$P(B=b|L=l)=$$

$$\frac{1}{Z_2(l)}\exp\left[-\sum_{v\in B(V)}\left(-\beta'_1\sum_{w\in\mathcal{S}(v)}(b(v)-1/2)(b(w)-1/2)(1-l(p(v),q(w)))\right.\right.$$
$$\left.\left.+\beta'_2 b(v)+\beta'_3 b(v)\sum_{w\in\mathcal{T}(v)}b(w)\right)\right] \quad (5.2)$$

with the following notations:

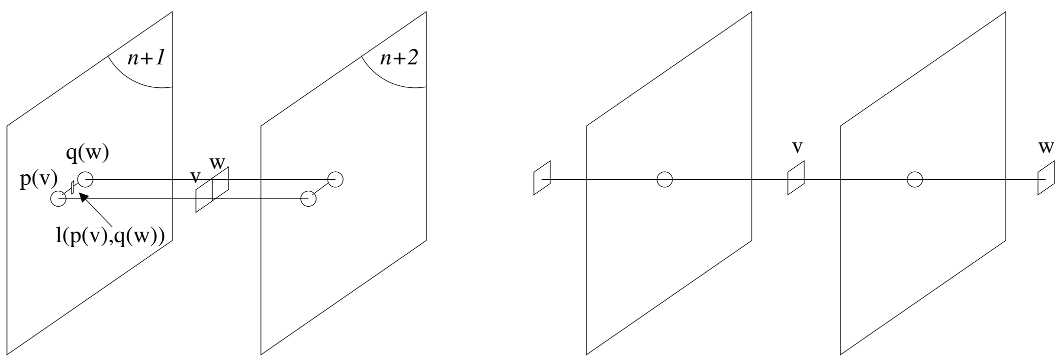
$B(V)$ is the set of temporal discontinuity sites,

$\mathcal{S}(v)$ is the spatial neighbourhood of discontinuity v, containing four sites except at the grid boundaries,

$\mathcal{T}(v)$ is the temporal neighbourhood of discontinuity v, containing either two sites if v is between frames $n-1$ and n or n and $n+1$, or one single site if v is between frames $n-2$ and $n-1$ or $n+1$ and $n+2$,

$p(v)$ and $q(w)$ are the pixel sites at the same spatial locations as v and w, belonging to the surrounding frame closest to frame n (e.g. frame $n+1$ if v and w are between frame $n+1$ and frame $n+2$).

The term $-\beta'_1\sum_{w\in\mathcal{S}(v)}(b(v)-1/2)(b(w)-1/2)(1-l(p(v),q(w)))$ expresses a constraint of spatial smoothness on the temporal discontinuities, again switched off if necessary. It is a negative quantity if both discontinuities are set to 0 or 1 and positive if they are different. $\beta'_2 b(v)$ is a bias toward the absence of discontinuities. The term $\beta'_3 b(v)\sum_{w\in\mathcal{T}(v)}b(w)$ modulates this bias on multiple discontinuities. The global penalty for a single temporal discontinuity is thus β'_2, it is $2\beta'_2+2\beta'_3$ for two adjacent discontinuities, $3\beta'_2+4\beta'_3$ for three adjacent discontinuities and $4\beta'_2+6\beta'_3$ for four adjacent discontinuities. Unlike the other weighting parameters, β'_3 is not necessarily positive. Without this parameter, the four temporal discontinuity fields would be independent of each other.



(a) spatial smoothness weighted by β'_1

(b) bias modulation on multiple discontinuities weighted by β'_3

Figure 5.3: local interactions in the prior distribution

The posterior distribution From the prior distribution on the temporal discontinuities and the likelihood of the motion-compensated intensities knowing the discontinuities, the posterior probability would be given by:

$$\begin{aligned} P(B =& b|L = l, I = i) = \\ &\frac{1}{Z_3(l,i)} P(I = i|B = b, L = l) P(B = b|L = l) \end{aligned} \tag{5.3}$$

After reorganising the sums and changing the weight notations (to remove factors of 2 induced by sum modifications), based on (5.3) or alternatively by specifying it directly, the posterior distribution is chosen as:

$$\begin{aligned} P(B = b|L = l, I = i) = \frac{1}{Z} \exp \Bigg[-\Bigg(& \alpha \sum_{v \in B(V)} (1 - b(v))(i(v^+) - i(v^-))^2 \\ & - \beta_1 \sum_{\{v,w\} \in \mathcal{C}_{\text{spat}}} (b(v) - 1/2)(b(w) - 1/2)(1 - l(p(v), q(w))) \\ & + \beta_2 \sum_{v \in B(V)} b(v) + \beta_3 \sum_{\{v,w\} \in \mathcal{C}_{\text{temp}}} b(v) b(w) \Bigg) \Bigg] \end{aligned} \tag{5.4}$$

where v^+ and v^- are the two pixel sites between which temporal discontinuity site v is located, $\mathcal{C}_{\text{spat}}$ and $\mathcal{C}_{\text{temp}}$ are respectively the sets of all spatial and temporal pair cliques in $B(V)$. As the first term weighted by α'_1 in (5.1) did not involve temporal discontinuities, it has been be isolated and included in the normalizing constant Z. The weighting parameter α thus stands only for the former α'_2.

The estimator From this posterior distribution, the estimator chosen to infer the value of the unknowns is the *Maximum A Posteriori* (see section 4.2.3.2). This is here the configuration on the four temporal discontinuity fields which maximizes the posterior distribution. This configuration depends on parameters α, β_1, β_2 and β_3 but is unchanged if they are multiplied by a scaling factor. It then actually depends on 3 independent parameters.

Extension to dropouts This model has been primarily designed for blotches because this artifact is very common whereas dropouts are less frequent in practice. However, it can be also applied to dropout detection with minor modifications. The first difference is that we should use in this case motion-compensated fields instead of motion-compensated frames. The second difference is that the MRFs should not be spatially isotropic any more for dropouts since they naturally show a strong horizontal orientation. It can be done by weighting differently horizontal and vertical spatial pair cliques and thus replacing the parameter β_1 by two parameters β_{1h} and β_{1v}, with $\beta_{1h} \gg \beta_{1v}$.

5.2.2 Interpretation

Once temporal discontinuities have been located, we proceed with an interpretation step to determine the state of each pixel belonging to the current frame. Pixels in frame n that are between two discontinuities with frames $n-1$ and $n+1$ are examined more closely, the others being considered as non-problematic. For these pixels, we need to take a closer look at the discontinuity fields between $n-2$ and $n-1$ and between $n+1$ and $n+2$ in a large neighbourhood. In each of these two fields, the number of discontinuities in the vicinity of the spatial location of interest is counted in a radius r (figure 5.4). There are two possible cases: if the surrounding contains few discontinuities, say below a threshold M, the double discontinuity can be seen as isolated and is consequently considered as a blotch; if the surrounding contains more than M discontinuities, the nearby repetition of temporal disruptions is considered to be the symptom of pathological motion.

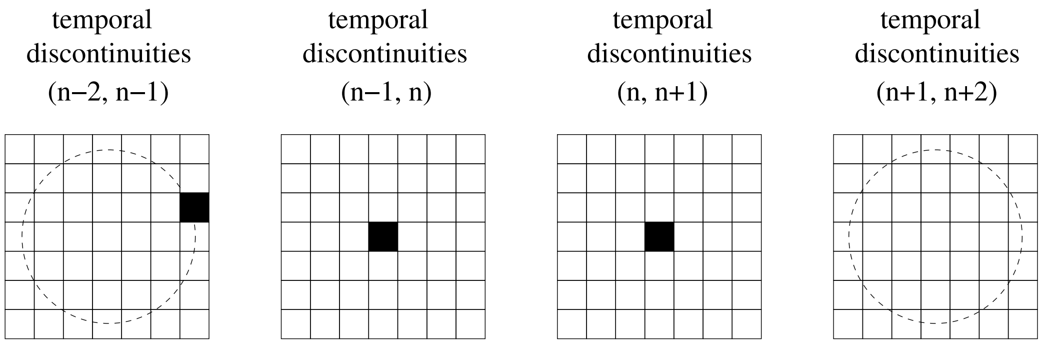


Figure 5.4: interpretation of the temporal discontinuities

This interpretation step involves long-range interactions between temporal discontinuities. These interactions could in theory be integrated to the Markovian model by choosing a neighbourhood system with a sufficient size. The potential associated to each pair clique could be weighted by a constant parameter or by a function of the distance between the two sites. Unfortunately, the extension of the neighbourhood size would greatly increase the computational complexity and neighbourhoods with an order higher than the first or second order would be intractable in practice. It is therefore mainly for computational efficiency reasons that these constraints have been expressed in a purely deterministic way.

5.3 Multiscale optimization for MAP estimation

The most tricky point in the proposed method is clearly the localization of the temporal discontinuities and more precisely the MAP estimation. MAP computation is a difficult optimization problem because of the huge number of possible configurations: the set Ω of all possible global configurations of the unknowns has a dimension which is here $2^{|B(V)|}$. As explained in section 4.3.2, algorithms to approximate the MAP estimator can be either stochastic or deterministic. Stochastic algorithms, the most employed of which is Gibbs sampling coupled with Simulated Annealing (section 4.3.2.1), converge to the global maximum whatever the initialization, but are extremely slow. Deterministic algorithms, such as ICM (section 4.3.2.2), are much faster, but reach local maxima that are highly dependent on the initial configuration.

To alleviate these shortcomings, hierarchical approaches have been developed actively during the last decade. As classified in [Gra 95], hierarchical approaches can be divided between *explicit hierarchy* and *induced hierarchy*. In the former case, the hierarchy is directly integrated in the model definition; in the latter case, the initial problem is transformed into a succession of simpler optimization problems. As the graph defined for our model is not intrinsically hierarchical, we turn to the family of induced hierarchy approaches. Besides renormalization group methods that can only be applied to very specific classes of MRFs, Pérez's multiscale approach [Per 93, Hei 94] gives a rigorous and efficient framework inspired by multigrid methods developed for the numerical analysis of partial differential equations. This is the approach that we implemented to get fast computation times without sacrificing the quality of the final result. The principle is to define hierarchised subsets of the set of all possible configurations Ω:

$$\Omega^K \subset \Omega^{K-1} \subset \ldots \subset \Omega^k \subset \ldots \subset \Omega^0 = \Omega \tag{5.5}$$

As k is increased, these correspond to sets of all configurations that are constant on "bigger and bigger" parts of the set $B(V)$ of temporal discontinuity sites. We therefore define a hierarchised set of cells that form connected partitions $B^k(V)$ of the set of all sites $B(V) = B^0(V)$. At each level k, Ω^k is the set of all configurations that are constant on every cell of $B^k(V)$. The choice of square cells containing $2^k \times 2^k$ sites is very common and it is what we implemented in our case (see figure 5.5).

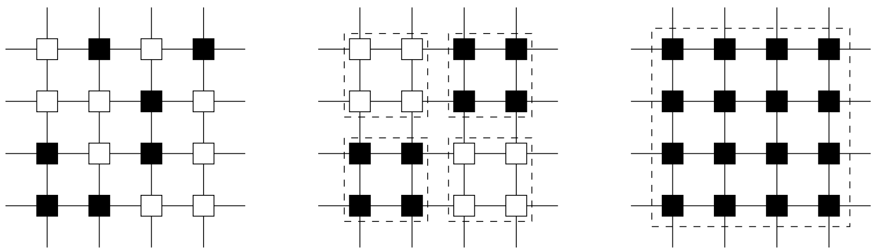


Figure 5.5: example of allowed configurations with associated square cells

At each level k, the initial problem, i.e. the minimization of the global energy over Ω is replaced by the minimization of the global energy over the smaller set Ω^k. In the following, we will leave aside spatial edges for simplification purpose. The global energy corresponding to (5.4) can be written as:

$$\begin{aligned} U(b,i) = \quad & \alpha \sum_{v \in B(V)} (1-b(v))(i(v^+) - i(v^-))^2 - \beta_1 \sum_{\{v,w\} \in \mathcal{C}_{\text{spat}}} (b(v)-1/2)(b(w)-1/2) \\ & + \beta_2 \sum_{v \in B(V)} b(v) + \beta_3 \sum_{\{v,w\} \in \mathcal{C}_{\text{temp}}} b(v)b(w) \end{aligned}$$

The global energy can be divided into two terms: a first term $U_1(b)$ which is known as the *contextual energy* and is only related to the unknowns and a second term $U_2(b,i)$ which is the *link to the data* and also involves the observations. Both of these energies can be re-expressed on the set Ω^k in order to exploit the constraint that all configurations must now be constant on each cell. This will be explained for our specific energy function, but a more general-purpose description can be found in [Per 93, Hei 94].

For the contextual energy, terms that involve a single site can be grouped by cells. As for terms involving pair cliques, we will group separately cliques that are within the same cell and cliques that link two sites belonging to two different cells:

$$\begin{aligned} U_1(b) =& -\beta_1 \sum_{\{v,w\}\in\mathcal{C}_{\text{spat}}} (b(v)-1/2)(b(w)-1/2) + \beta_2 \sum_{v\in B(V)} b(v) + \beta_3 \sum_{\{v,w\}\in\mathcal{C}_{\text{temp}}} b(v)b(w) \\ =& -\beta_1 \times 1/4 \times \sum_{v\in B^k(V)} d_v^k - \beta_1 \sum_{\{v,w\}\in\mathcal{C}^k_{\text{spat}}} l^k_{\{v,w\}}(b(v)-1/2)(b(w)-1/2) \\ & + \beta_2 \sum_{v\in B^k(V)} c_v^k b(v) + \beta_3 \sum_{\{v,w\}\in\mathcal{C}^k_{\text{temp}}} c_v^k b(v)b(w) \end{aligned} \tag{5.6}$$

where $d_v^k = 2\times 2^k(2^k-1)$ is the number of pair cliques within cell v,
$l^k_{\{v,w\}} = 2^k$ is the number of pair cliques linking spatially neighbouring cells v and w or the "spatial contact surface",
$c_v^k = 2^k \times 2^k$ is the cardinal of cell v. It can also be seen as a "temporal contact surface", since it is the number of pair cliques linking two temporally neighbouring cells.

For the link to the data, we can perform similar groupings:

$$\begin{aligned} U_2(b,i) =& \ \alpha \sum_{v\in B(V)} (1-b(v))(i(v^+)-i(v^-))^2 \\ =& \ \alpha \sum_{v\in B^k(V)} (1-b(v)) \sum_{w\in v}(i(w^+)-i(w^-))^2 \end{aligned}$$

As both of these energies can be expressed as a sum of local interactions over single cells and pair cliques of cells, it is natural to assimilate each cell with a single site belonging to a new graph (see figure 5.6). On this graph is defined a new MRF with an associated energy very similar to the original one: there are simply additional weights depending on the cell size for the contextual energy and sum of square differences instead of square differences for the link to the data. The characteristics of this field are thus completely defined from those of the original model.

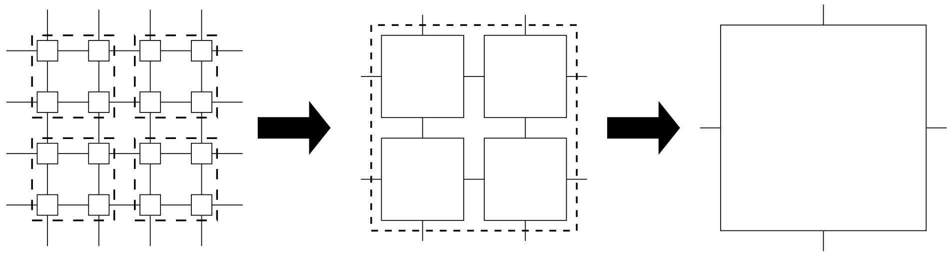


Figure 5.6: natural pyramidal structuration

The multiscale algorithm sets a cooperation strategy between the $K+1$ optimization problems. Because of the inclusion $\Omega^{k+1} \subset \Omega^k$, the algorithm is the following:

- optimization is first performed at the coarsest level,
- optimization at level k is initialized by the result obtained at level $k+1$,
- at level $k=0$, the final result is the solution of the initially considered problem.

This multiscale algorithm actually explores the complete set Ω in an implicit manner. The power of the algorithm resides in the fact that a single transition at level k is equivalent to many simultaneous transitions at level 0. This results in an increase in the convergence speed. In the deterministic case, as each optimization is started with a "good" initialization from the previous level, better local minima are usually reached.

5.4 Experimental comparison of optimization algorithms

Optimization algorithms were evaluated for a simple model of an isolated temporal discontinuity field between two images without spatial edges. The two images chosen to compare the optimization algorithms are shown on figure 5.7. These are two consecutive frames of the same PAL sequence (720×576 pixels) with a real transparent blotch and nearly no motion; motion estimation and compensation are thus not necessary here and do not bias the comparison.

For all algorithms, the sites are visited in a forward-backward raster scan ordering. Convergence is assumed and optimization is consequently stopped when no single discontinuity has been changed at the end of a forward-backward sweep (considered as one iteration). For stochastic algorithms, specific care must be taken in the design of the random number generator: its pseudo-period should be much larger than the total number of sites. Unfortunately, this is not the case with random number generators provided in the standard library of most programming languages. We therefore implemented Park and Miller's multiplicative linear congruential generator for this purpose [Par 88]. For the exponential cooling schedule employed in stochastic algorithms, we choose $T_0 = 50$ and

(a) Frame $n-1$

(b) Frame n with circled blotch

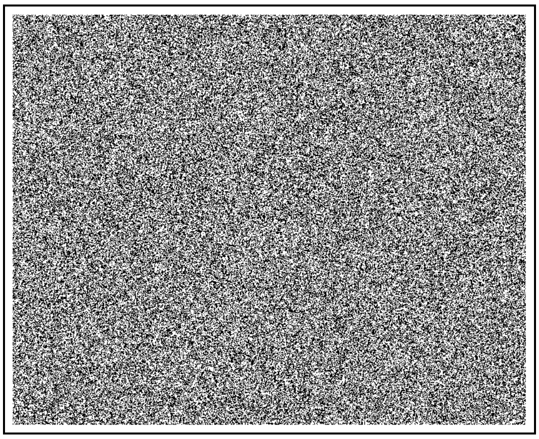
Figure 5.7: test images for the comparison of optimization algorithms

$a = 0.95$. In the multiscale version of stochastic and deterministic algorithms, we use a total of $K = 7$ levels, which yields cells of size 64×64 at the highest level.

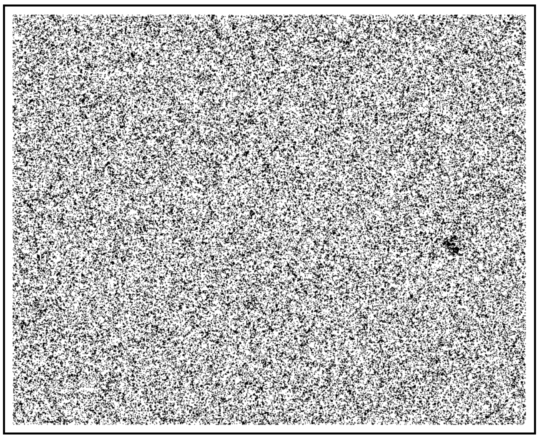
In the case of stochastic optimization, there is much random agitation at the beginning and convergence is relatively slow (see figure 5.8). More than a hundred iterations are necessary to reach the maximum. A comparable maximum is reached for multiscale stochastic optimization with roughly the same number of iterations, but as many of them are performed at a higher level, they have a significantly reduced computational cost. However, the gain is not as large as one could expect. The optimization process for the multiscale stochastic algorithm is shown on figure 5.9.

In the case of deterministic optimization, a local maximum is reached very fast with very few iterations but this maximum highly depends on the initial configuration as illustrated by figure 5.10. The incorporation within a multiscale scheme makes the result much less sensitive to initial conditions while preserving the speed of the algorithm. Multiscale deterministic optimization is shown on figure 5.11.

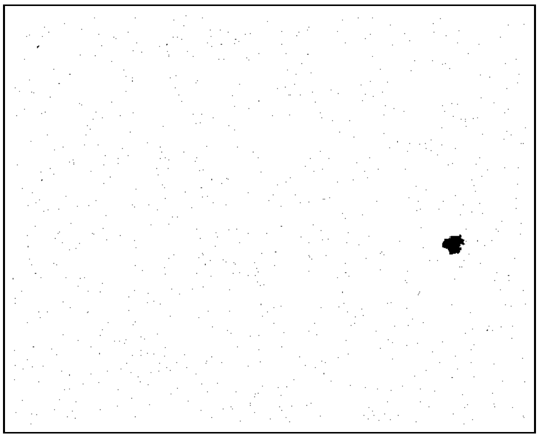
All the results are summarized in figure 5.12 where the result for ICM (figure 5.12(c)) was computed from an initialization with the thresholded DFD at $T_h = 30$ (see figure 5.10(c)). The corresponding computation times using C++ code on a PC with a 1.5 GHz Pentium 4 processor can be found in table 5.1. For multiscale algorithms, the total number of iterations in this table includes iterations at full resolution as well as iterations at coarser levels. These tests lead to the selection of the multiscale ICM algorithm for further experiments: it provides satisfactory results while being very efficient in terms of computation time.



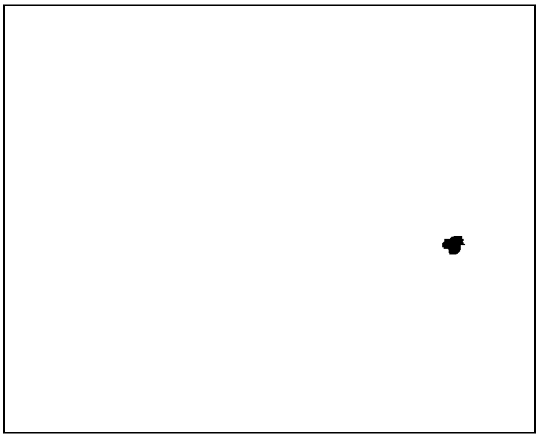
(a) After iteration 1



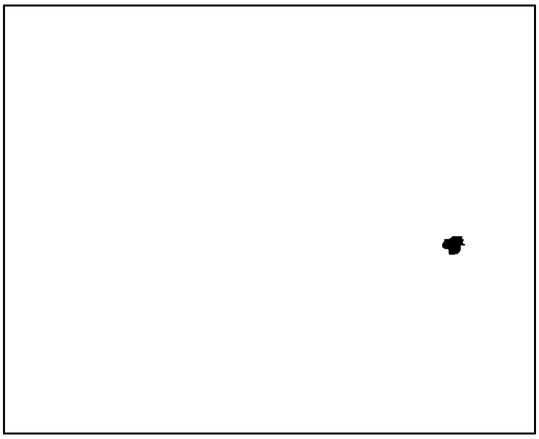
(b) After iteration 25



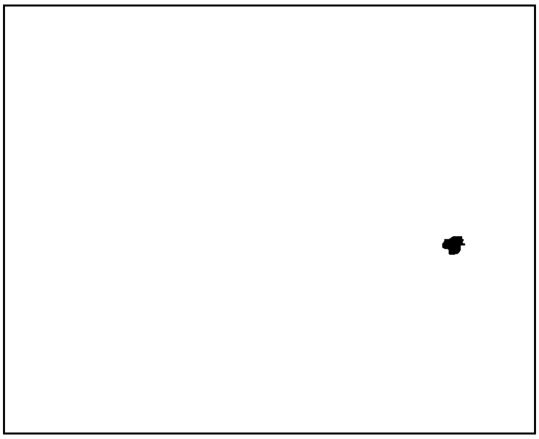
(c) After iteration 50



(d) After iteration 75

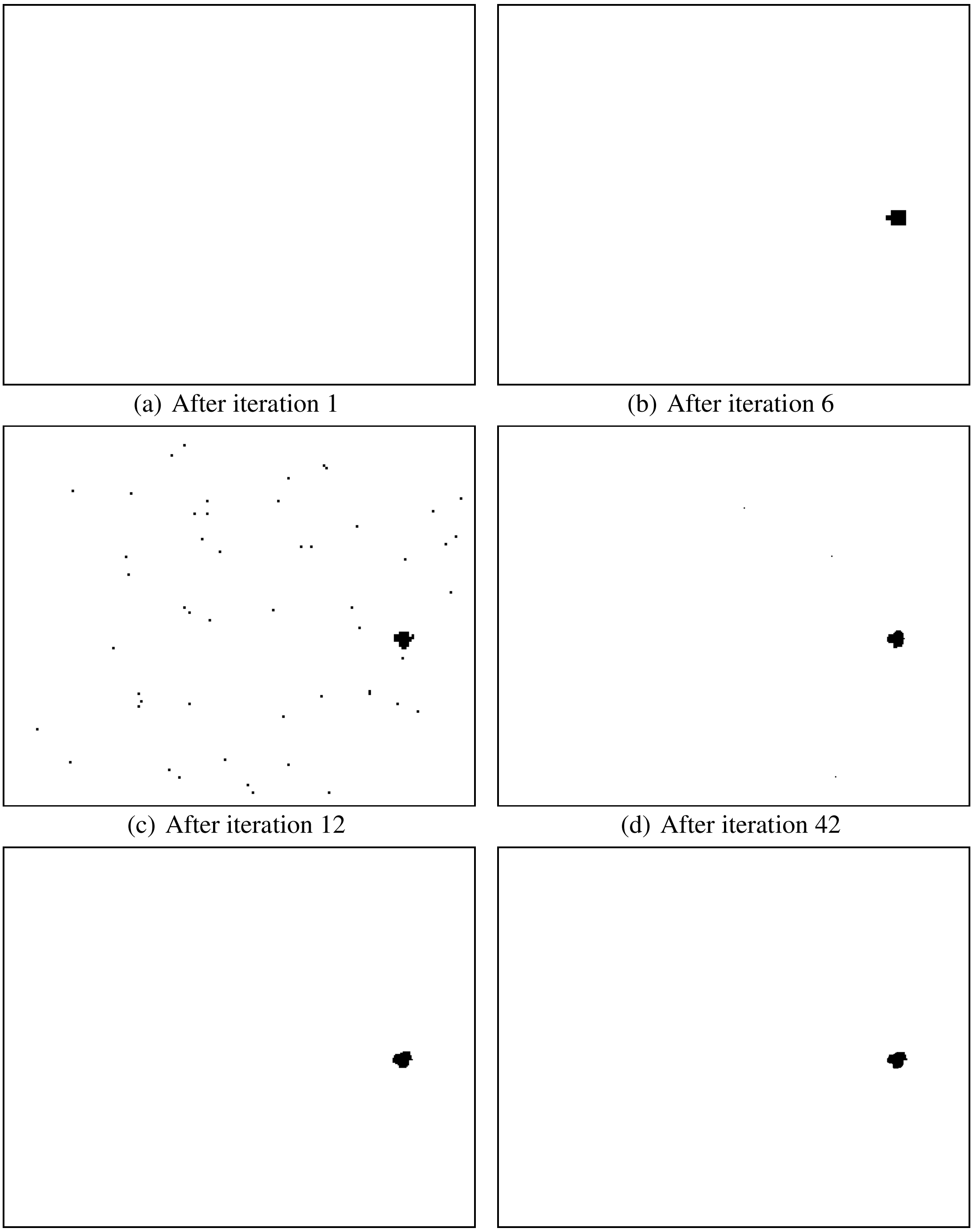


(e) After iteration 100



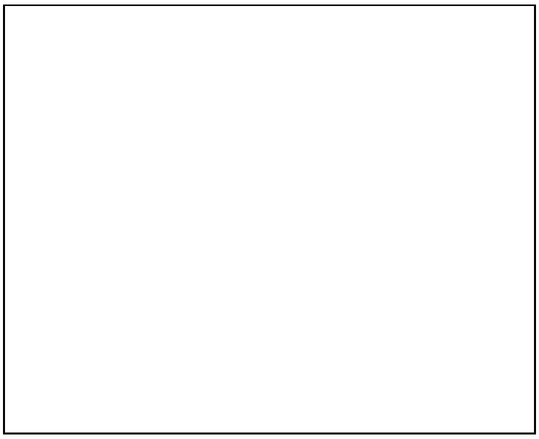
(f) After final iteration 119

Figure 5.8: stochastic optimization

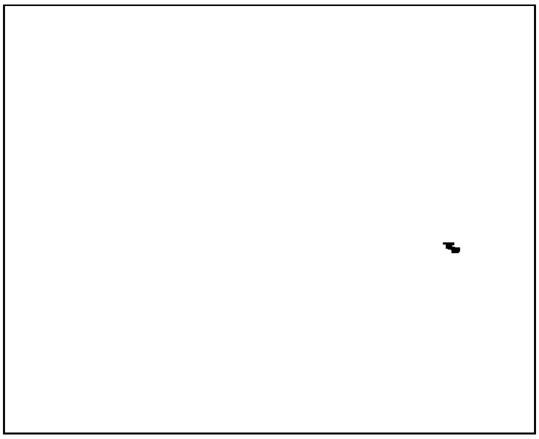


(a) After iteration 1
(b) After iteration 6
(c) After iteration 12
(d) After iteration 42
(e) After iteration 82
(f) After final iteration 112

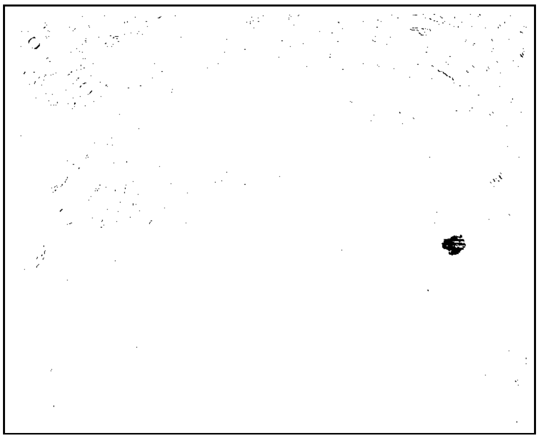
Figure 5.9: multiscale stochastic optimization



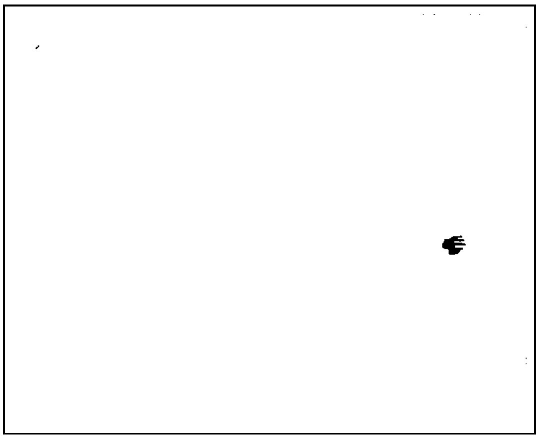
(a) Initial configuration 1 (all discontinuities set off)



(b) Deterministic optimization with initial configuration 1

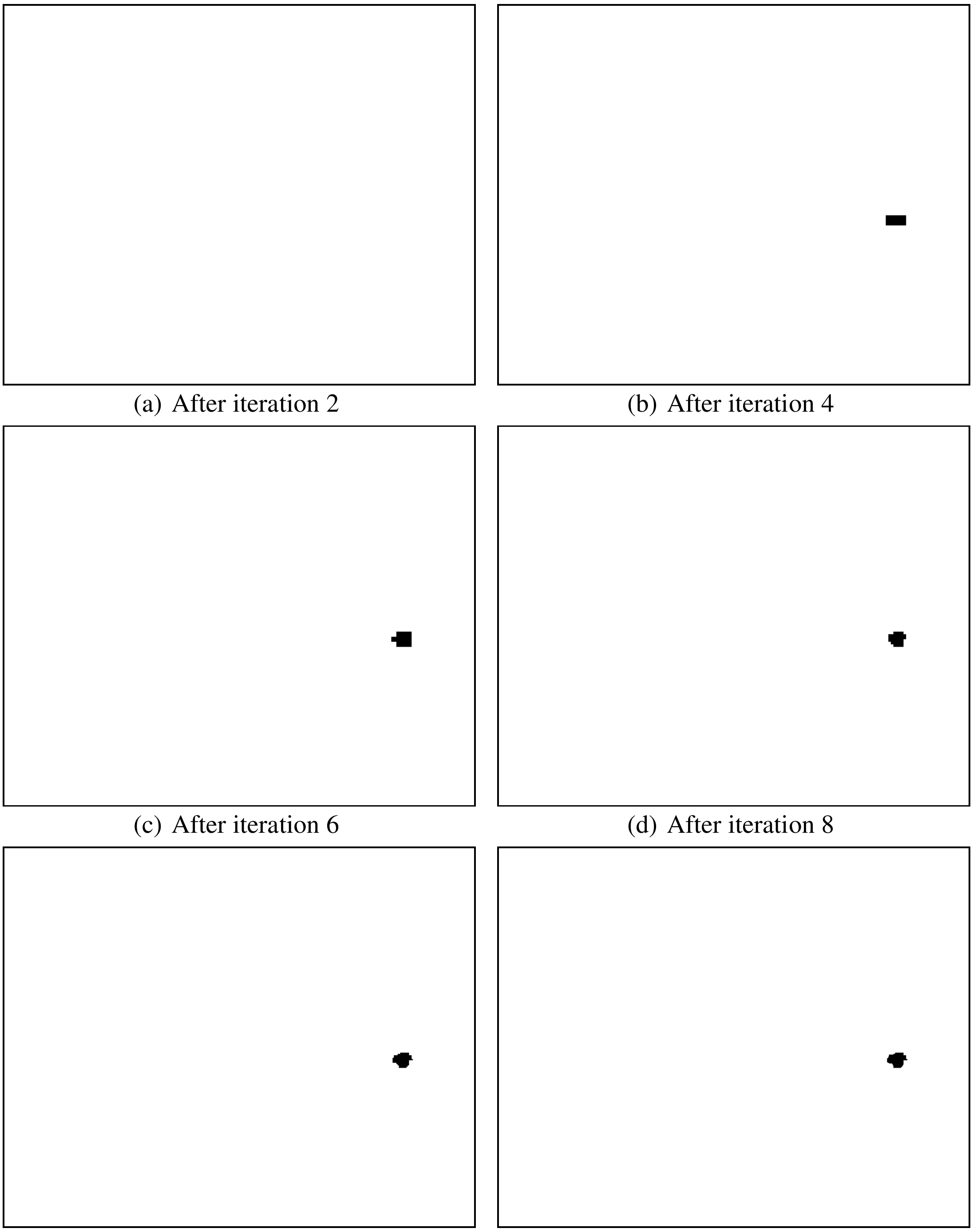


(c) Initial configuration 2 (thresholded DFD with $T_h = 30$)



(d) Deterministic optimization with initial configuration 2

Figure 5.10: deterministic optimization

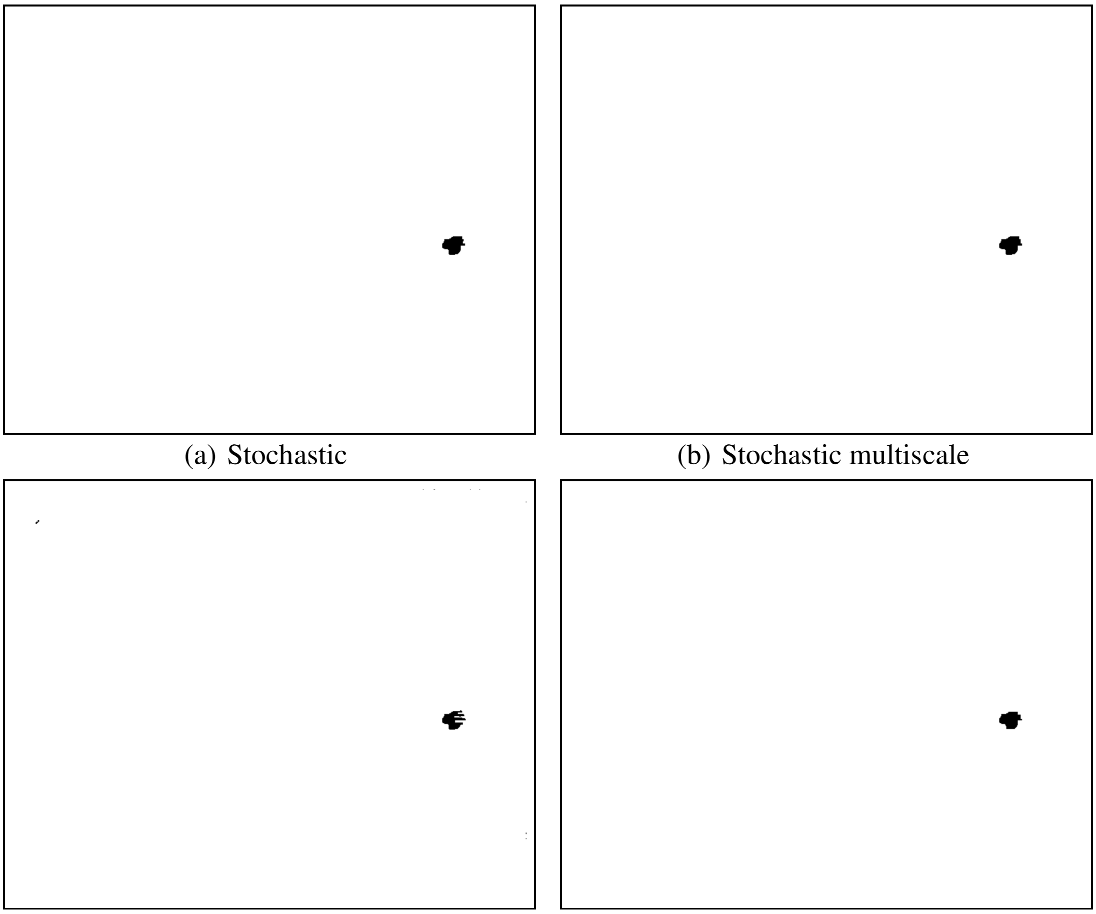


(a) After iteration 2 (b) After iteration 4

(c) After iteration 6 (d) After iteration 8

(e) After iteration 10 (f) After final iteration 12

Figure 5.11: multiscale deterministic optimization



(a) Stochastic (b) Stochastic multiscale

(c) Deterministic (d) Deterministic multiscale

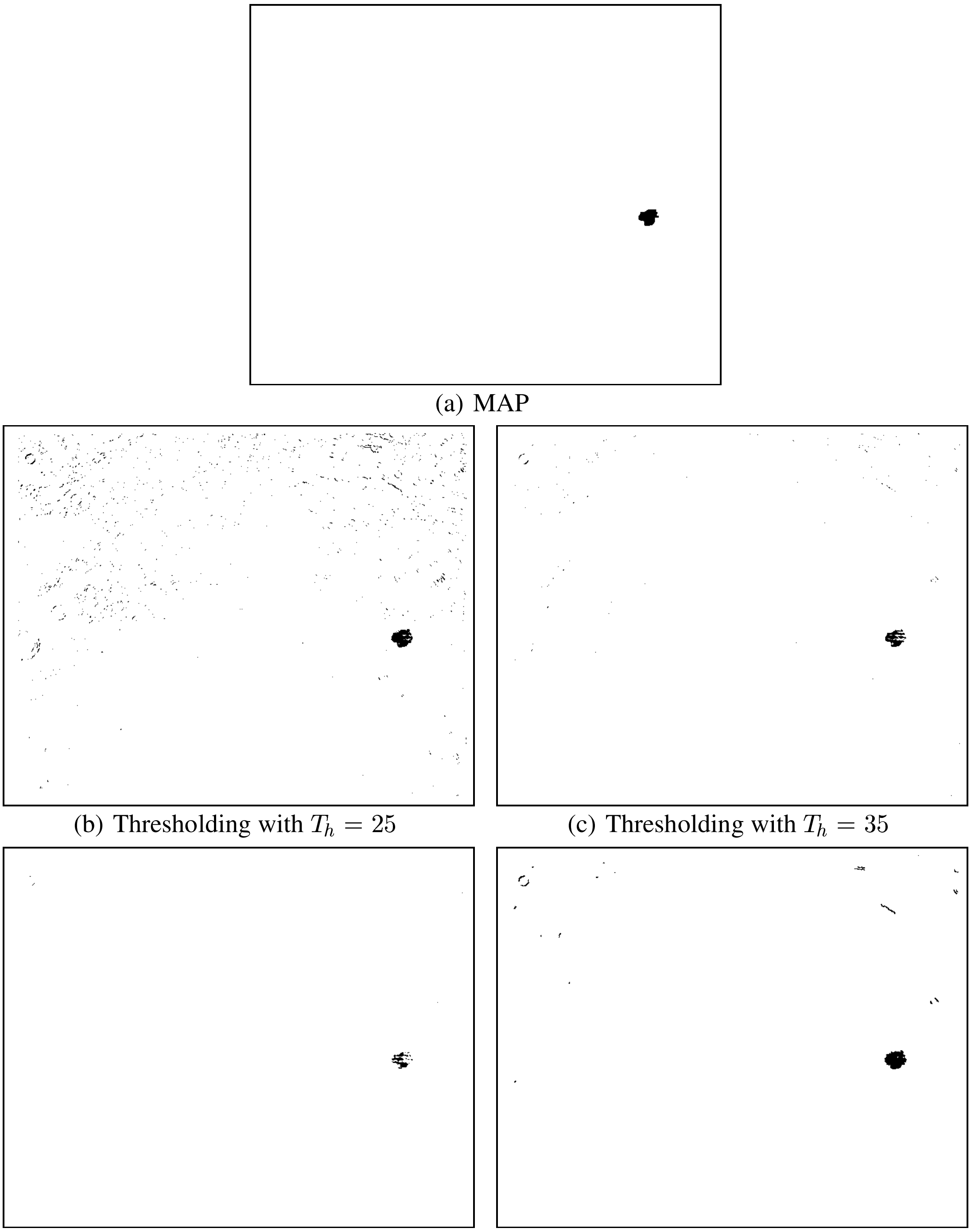
Figure 5.12: comparison of maxima reached

	stochastic	**stochastic multiscale**	**deterministic**	**deterministic multiscale**
Algorithm	SA coupled with Gibbs sampling	multiscale SA coupled with Gibbs sampling	ICM	multiscale ICM
Total number of iterations	119	112	4	12
Computation time	1mn 29s	29s	0.4s	0.4s

Table 5.1: comparison of computation times (1.5 GHz Pentium 4 processor)

5.5 Interest of the Markovian model

On the same test images, the comparison of the estimated *Maximum A Posteriori* with simple thresholdings gives a good insight into the interest of Markovian models (figure 5.13). Among other interactions, the proposed model includes spatial interactions between pixels in a flexible and natural way. For this reason, it clearly gives much less spurious responses than basic thresholding (figures 5.13(b), 5.13(c) and 5.13(d)) or even hysteresis thresholding (figure 5.13(e)). It is in particular much less sensitive to noise or film grain.



(a) MAP

(b) Thresholding with $T_h = 25$

(c) Thresholding with $T_h = 35$

(d) Thresholding with $T_h = 45$

(e) Hysteresis thresholding with $T_{\text{high}} = 40$ and $T_{\text{low}} = 20$

Figure 5.13: comparison of MAP with thresholding

5.6 Discussion on evaluation

The performance of a detection algorithm is typically evaluated with an ROC curve plotting the correct detection rate versus the false alarm rate as explained in section 3.1.3. This type of evaluation is unsatisfactory because it counts similarly missed detections and false alarms without regard to their subsequent effects. The consequence of a missed detection is that a blotched pixel will not be removed from the document. This will result in a certain visual discomfort that can be approximated to be equivalent for all non-removed blotched pixels. The way the missed detection rate (or equivalently the correct detection rate) is evaluated is therefore satisfactory. However, the situation is completely different for false alarms. The consequence of a false alarm will be the unnecessary interpolation of a “clean” pixel. The visual discomfort resulting from this mistake can be dramatically different depending on the performance of the correction scheme in the specific situation. This discomfort can vary from null to highly annoying and is quite often much more disturbing than the visual impact of a non-removed blotch. It is therefore highly undesirable to count with the same weight all pixels in false alarm. To incorporate this fact, the question should rather be “how much” the correction of each false alarm will be visible. This shifts the problem of quantitative evaluation of detection to the problem of perceived quality measurement.

These remarks lead to the conclusion that an impulsive defect detector cannot be properly evaluated independently of the subsequent correction step. More precisely, it is in the counting of false alarms that the consequences of correction must be accounted for, either implicitly or explicitly.

If correction is explicitly taken into consideration, then for each false alarm, the corresponding correction should be temporarily performed with the chosen technique and its quality should be measured. Once these measures have been summed over the image, we should be able to plot ROCs with the correct detection rate versus the introduced distortion rate. This is deferred to the existence of a suitable objective metric for video quality. This is believed to be a long-term prospect and will be discussed in section 6.2.

Another possibility is to account implicitly for the correction and thus simplify the problem by stating that some false alarms will be usually more dangerous than others for most correction algorithms. This classification between problematic and non-problematic false alarms could revolve around the notion of pathological motion: we could for each pixel in false alarm wonder whether this pixel is *within* (usually implying serious repercussions) or *outside* (implying a likely invisible correction) areas of PM. We could then count differently these two kinds of false alarms by penalizing them differently: a false alarm within PM would be somewhat arbitrarily considered κ times more dangerous than a false alarm outside PM. In order to be able to make this distinction, a pathological motion detector would be necessary. This is a problem as we are aware of no other published work on the identification of PM. The use of our PM masks to evaluate detection methods including ours would certainly introduce a bias. In the absence of an independent technique for this purpose, we consider that a visual evaluation of the binary masks is the best way to grasp the benefits of our approach.

5.7 Experimental results

A simplified version of the described method has been tested: experiments were performed without the incorporation of spatial edges. Our model is combined with a software phase-correlation motion estimator with subpixel accuracy (see appendix B) and bilinear motion compensation. This motion estimator was specifically developed to test the validity of our approach and this family of motion estimators was mainly chosen for its ease of implementation. However, it must be kept in mind that this approach can be used with any motion estimator and will give results all the more impressive as the chosen motion estimator is efficient and accurate.

5.7.1 Comparison with SDIa and Morris' method

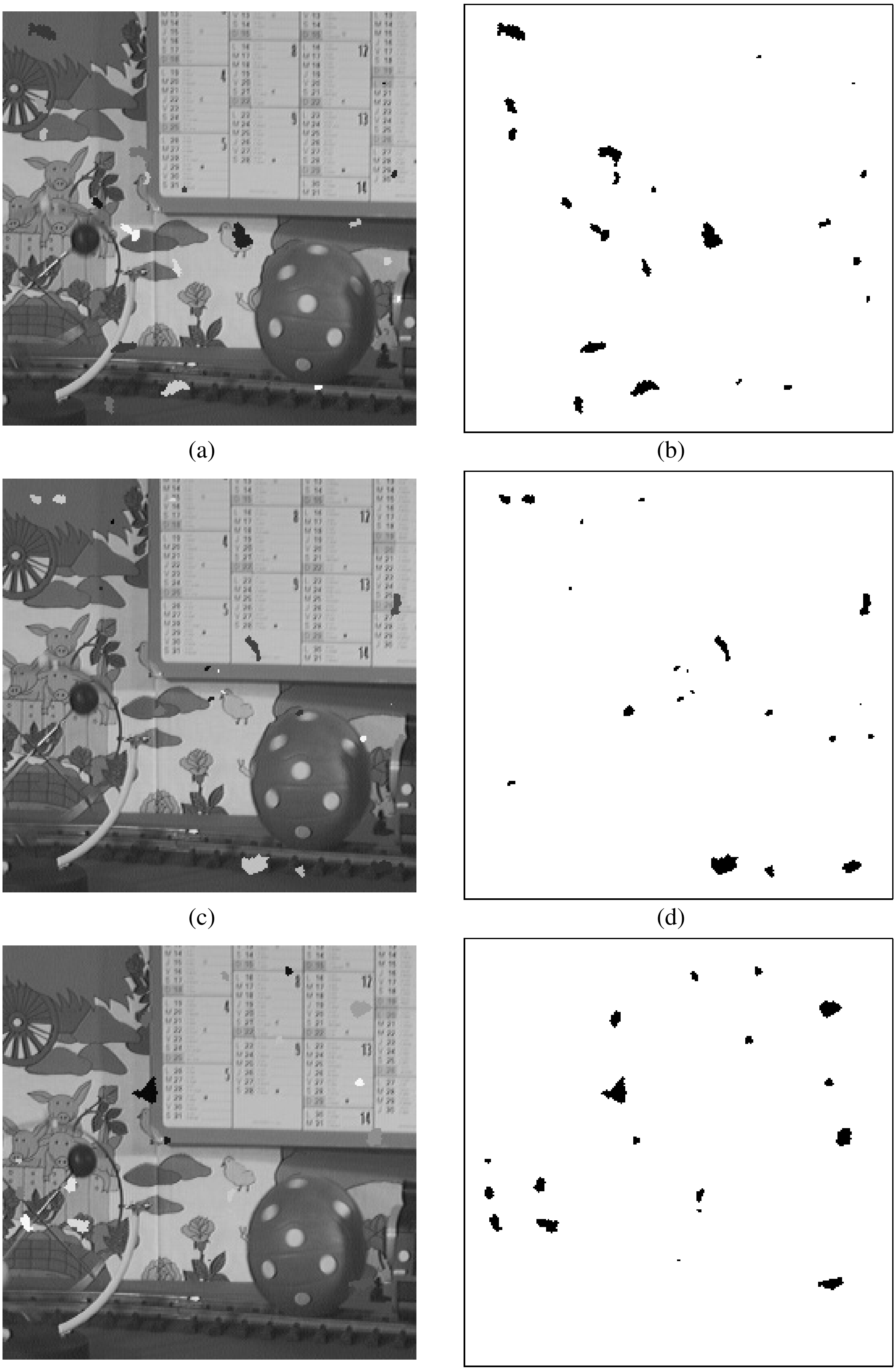
As an illustration of the benefit of this approach, our method is compared to Morris' algorithm which is the closest to ours (section 3.2.3). This algorithm consists in computing the blotch mask on 3 images based on independent Markovian models on backward and forward discontinuities. These models are intrinsically 2-D and do not involve any temporal neighbourhood on the discontinuities. As a reference point, comparison with SDIa based on simple thresholding (section 3.2.1) is also given.

5.7.1.1 ROC comparison

Despite the limited interest of conventional ROC plots as explained in previous section 5.6, this type of curve is provided here, if only to give a common point to compare with existing evaluations. These measurements are performed on the artificially degraded "Mobile and Calendar" sequence[1] used in previous work ([Kok 98] chap. 6.4 and 7.9). This very short sequence (1s, 25 frames) consists in 256×256 portions of the full frame size. Its activity in terms of motion can be described as moderately difficult. This sequence has been artificially corrupted with completely opaque and uniform blotches, i.e. having a flat intensity profile set at a random grey level (see figure 5.14). Because these artificial blotches are rather unrealistic, the sequence should be used with much care. In particular, it should be noted that any detection algorithm which would explicitly model blotches as uniform would give exceptionally good results on this sequence regardless of its possible good or bad performance on real sequences. This is not anyway the case for any of the three algorithms tested here. Compared to what can be commonly seen in real archives, the sequence can be considered as very heavily corrupted.

For the three algorithms, the average correct detection and false alarm rates, as defined in section 3.1.3, are computed. For SDIa, the curve corresponds to all possible variations of parameter T_h. For Morris' method and our proposed method, which both include more than one single parameter, many different parameter settings are tested in order to cover

[1] The sequence is available on the CD-ROM provided with [Kok 98].



(a) (b)

(c) (d)

(e) (f)

Figure 5.14: Degraded "Mobile and Calendar" sequence (Origin: CCETT, artificial corruptions by Kokaram). Left: Frames 2, 3 and 4. Right: corresponding corruption masks.

the ROC space as much as possible. The curves corresponding to the best performance are then plotted from the respective clouds of points. The ROC averaged on the sequence is plotted for the three detectors in figure 5.15.

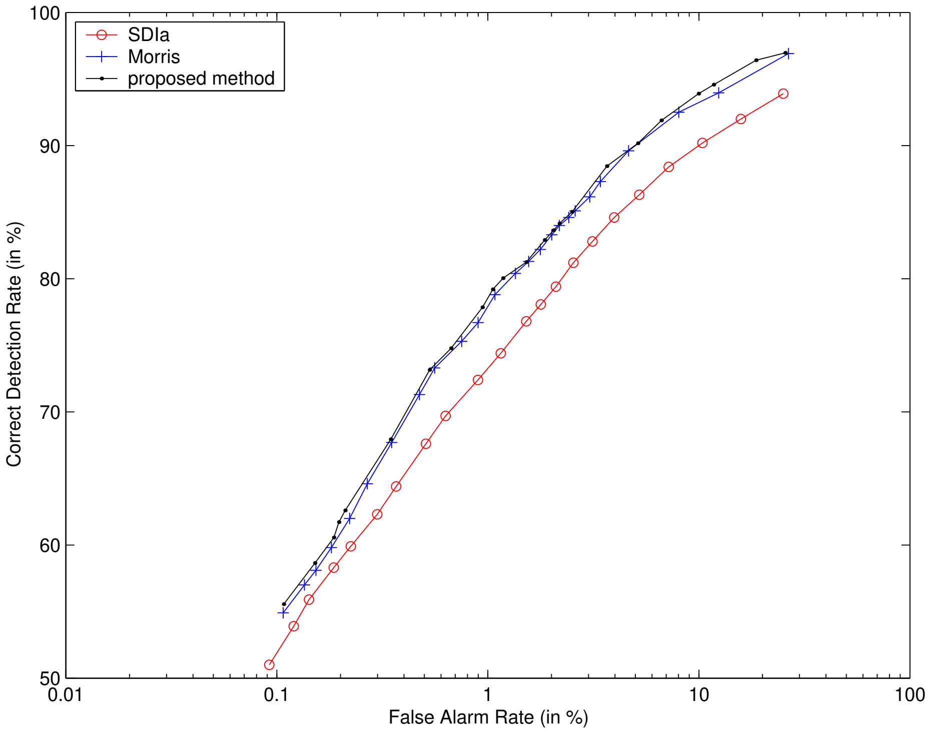


Figure 5.15: ROC averaged on the "Mobile and Calendar" sequence for several detectors

With in mind the limitations of this type of measurement on the counting of false alarms, SDIa seems to perform significantly worse than the two other detectors. On the other hand, the improvement of our method over Morris' detector is nearly imperceptible on these curves for mainly two reasons. Firstly, as the contents of the sequence is poor in terms of pathological motion, this leaves little room for the reduction of false alarms due to it. Secondly, as the level of corruption is very high compared to what typically occurs in real sequences, it is not uncommon that blotches appear at the same or close locations in successive images. For these reasons, the benefit of removing the few false alarms due to pathological motion is cancelled by the decrease in the correct detection rate. Therefore, the testing sequence chosen here does not allow our algorithm to fully express its potential.

5.7.1.2 Visual comparison

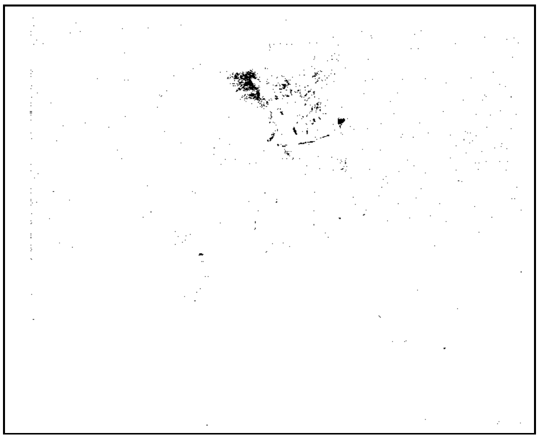
A much better way to compare the three algorithms is to show the detection masks on real examples which contain pathological motion. Among others, this can be illustrated on the "pigeon" sequence (figure 5.16): while it is not corrupted by noticeable artifacts, it contains very difficult motion with the fast displacement of the bird combined with the flapping of its wings. For our Markovian model, the parameters have been set to the following values: $\alpha = 0.005$, $\beta_1 = 0.65$, $\beta_2 = 0.4$, $\beta_3 = -0.07$. The number of levels for the multiscale optimization is set to $K = 7$. For the interpretation, we use a radius $r = 38$ and a threshold on the number of discontinuities $M = 1$. For Morris' algorithm, the parameters α, β_1 and β_2, which exist and play a similar role, are set to the same values. We also include the result for SDIa with a threshold $T_h = 15$. Figure 5.17 clearly shows the benefit of taking into account pathological motion: false alarms are dramatically reduced with our method.

(a) Frame $n - 1$

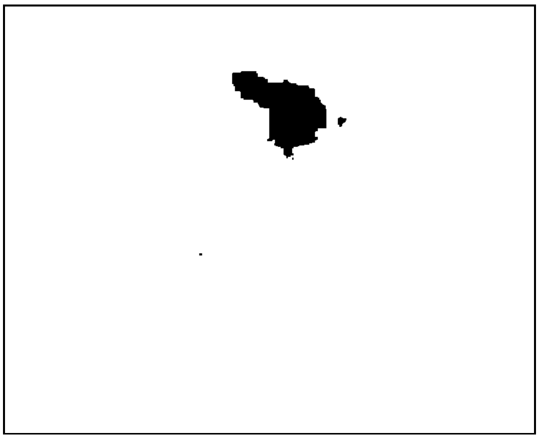
(b) Frame n

(c) Frame $n + 1$

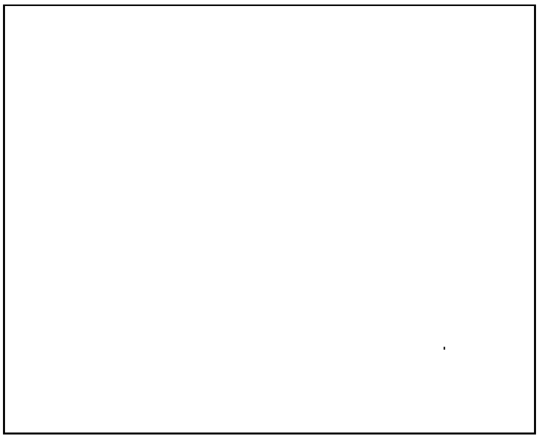
Figure 5.16: the “pigeon” sequence (images by courtesy of the BBC)



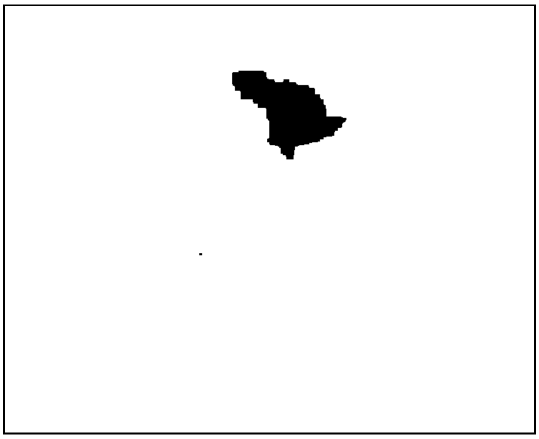
(a) Blotch mask with SDIa



(b) Blotch mask with Morris' method



(c) Blotch mask with our method



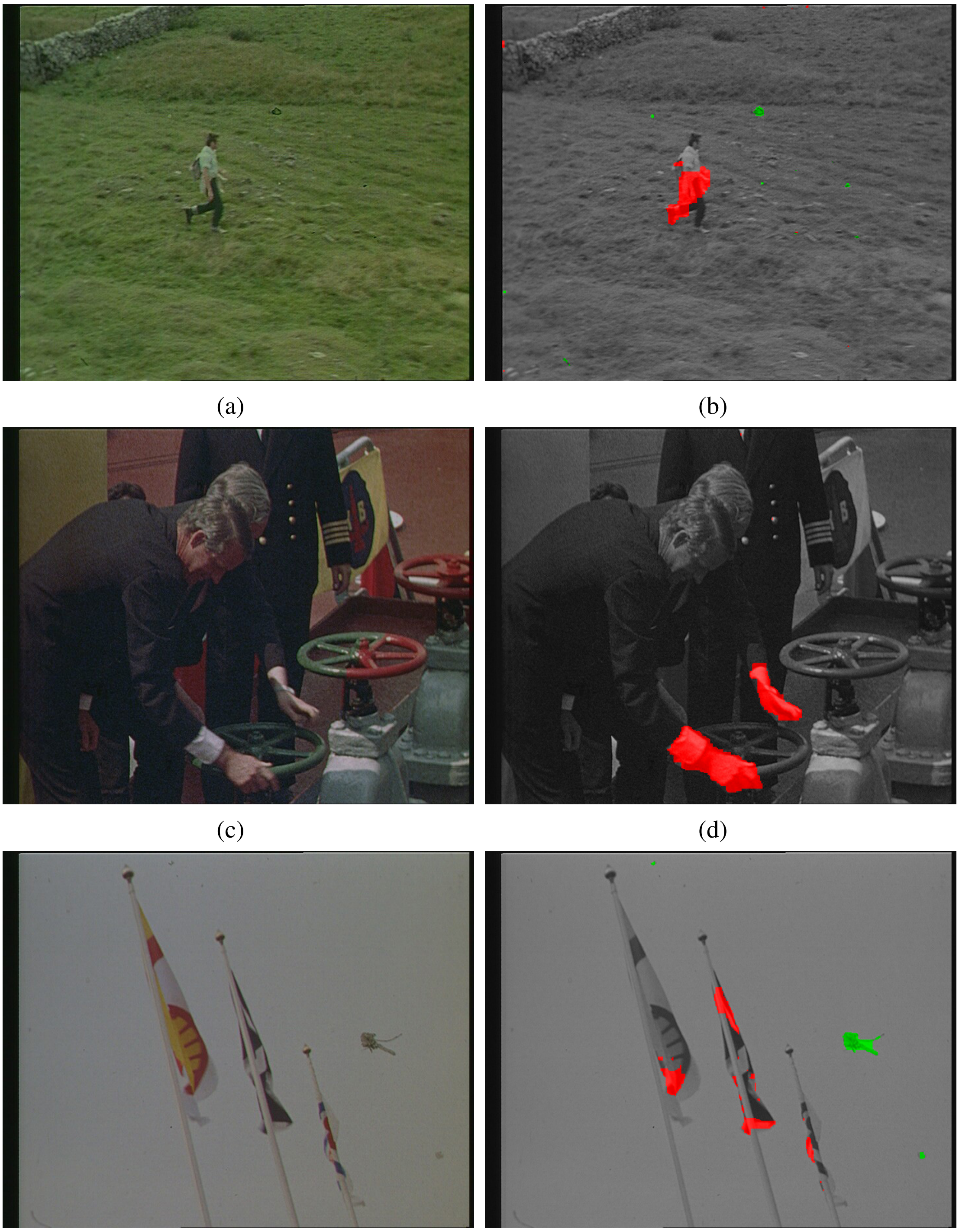
(d) Pathological motion mask with our method

Figure 5.17: blotch masks on the "pigeon" sequence using different methods

5.7.2 Large scale validation on real sequences

On a larger scale, our detection prototype has been validated on several minutes of real video sequences with many blotches. These sequences were gathered from various origins and were all chosen for the complexity of their contents in terms of motion. The prototype flags in green what it detects as blotches and in red the pathological motion.

Our testing architecture is described in appendix C. The evaluation of the results is purely visual. These results are very encouraging: computed masks are subjectively relevant and correspond to what is intuitively expected. Examples of results with the parameter settings given previously are shown on figures 5.18 and 5.19.



(a) (b) (c) (d) (e) (f)

Figure 5.18: results with our detection prototype. Left: original. Right: detection masks. Original images by courtesy of the BBC.

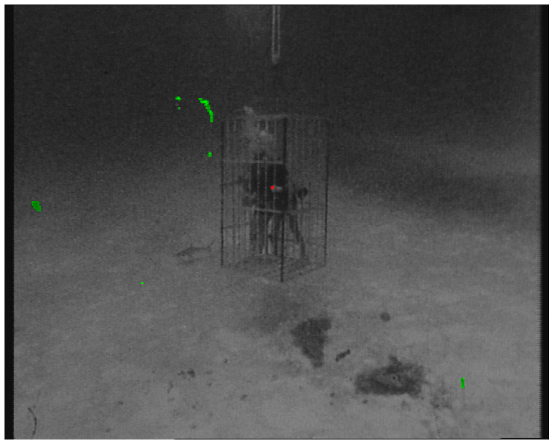
(a)

(b)

(c)

(d)

(e)



(f)

Figure 5.19: results with our detection prototype (continued). Left: original. Right: detection masks. Original images by courtesy of the BBC.

The carried out experiments entitle to hope for a very high degree of automation: good results have been obtained on very different sequences without having to change the set of parameters. This hopefully allows us to avoid the very tricky issue of parameter estimation in our Markovian context: this is usually performed with schemes based on the Expectation-Maximization (EM) [Dem 77] or Iterative Conditional Estimation (ICE) [Pie 92] algorithms (see [Cha 00, Per 98]), at the cost however of a very significant computational overhead. Our parameters can be understood and tuned intuitively once and for all: α is left to a reference value, β_1 is changed to increase or decrease spatial coherence, β_2 acts as the equivalent of the threshold T_h for SDIa, β_3 allows slight modulations of the previous parameter as it affects more pathological motion than blotches; as for M and the radius r for interpretation, they can roughly be seen as changing the ratio among flagged pixels between blotches and pathological motion. Because computed masks are usually well delineated, the incorporation of spatial edges as considered in the original model was not thought as absolutely necessary; it would be simply expected to bring a slight improvement in the accuracy of the masks at the expense of a computational overhead.

5.8 Final comments

In this chapter, we presented a new method for blotch detection which dramatically reduces the number of false alarms. The use of appropriate multiscale schemes has here a significant importance to maintain the computational load to a reasonable level. An additional step could be taken to go even further: although this has much less critical consequences, some blotches are now incorrectly flagged as pathological motion, either because they are *close to pathological motion* or because they are *within pathological motion areas*. This is discussed in the concluding chapter 8.

Chapter 6

Correction in missing data areas

The ability to replace data in damaged regions is of key importance for the efficient concealment of missing data artifacts. A reliable correction scheme, performing well even in the presence of pathological motion, would allow to significantly alleviate the shortcomings of many detection techniques: the better the correction scheme, the less critical the impact of false alarms resulting from the detection step.

In this chapter, we develop an algorithm which automatically interpolates missing information in the corrupted regions from the surrounding. The locations of the pixels to be corrected are here assumed to be known. Although the proposed technique has been primarily developed for blotch correction (and its obvious extension to dropout correction), it can also be applied to the correction of scratches, which belong to the same family of missing data artifacts. Beyond the scope of archives restoration, other applications have the same requirement of being able to fill-in whole regions in a "natural" way and can therefore greatly benefit from our algorithm. Superimposed logos, dates, names, subtitles or others are sometimes intentionally added to a document at a given moment and can be undesirable for use in a different context whereas the original document is no longer available. Another typical application is the concealment of selected objects in digital photograph retouching or special effects (wires, unwanted characters, ungainly details).

Unlike what was the case for detection, we now deal with unknowns which are not binary any more but belong to a much larger state space, e.g. $\{0, ..., 255\}$ for greyscale images or $\{0, ..., 255\}^3$ for colour images. In addition, the complexity of what we attempt to model is greater as this is no less than the underlying original sequence: larger neighbourhood systems would certainly be necessary to account for this complexity. For these reasons, Markovian models such as the one developed in the previous chapter cannot be considered any more: they would undoubtedly have a prohibitive cost. Our algorithm therefore relies on non-parametric Markovian models: it is inspired by texture synthesis techniques and especially by the algorithm presented in [Efr 99]. Our general-purpose algorithm is suited to any complex natural scene and not restricted to stationary patterns. It has the property to be adapted to both still images and image sequences. The resulting computational cost is relatively low and corrections are usually produced within seconds.

6.1 Algorithm

Efros and Leung's original algorithm is based on a non-parametric Markovian model. This statistical model is based on the assumption of spatial locality: the probability distribution for one pixel given the values of its neighbourhood is independent of the rest of the image. The neighbourhood $\mathcal{N}(p)$ of a pixel p is here chosen to be a square window around this pixel (illustrated in figure 6.1 for a 5×5 neighbourhood). The model is non-parametric in the sense that the probability function is not imposed or constructed explicitly. Instead, it is approximated from a reference sample image which must be large enough to capture the stationarity of the texture.

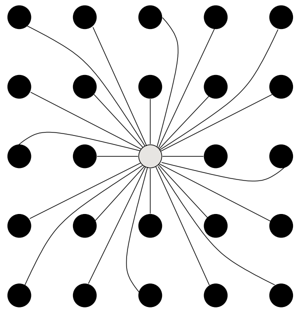


Figure 6.1: square neighbourhood

During the synthesis process, the approximation of the probability distribution $P(I(p) = i_p | I(q) = i_q, q \in \mathcal{N}(p))$ is made as follows: the sample image is first searched in order to find all pixels that have a "similar" neighbourhood to the one of the pixel being synthesized. In addition to the neighbourhood giving the best similarity d_{best}, all neighbourhoods that give a similarity d such as $d < (1+\epsilon)d_{\text{best}}$ are considered as candidates for replacement. Then one of these candidates is randomly drawn and the centre value of this neighbourhood is assigned to the pixel being processed.

The similarity of two neighbourhoods is measured according to the normalized sum of square differences (L^2 distance). Since it is desirable to give more importance to the pixels that are near the centre of the window than to those at the edge, this measure is weighted by a two-dimensional Gaussian. The pixels within the neighbourhood window that have not been synthesized yet are not taken into account in the sum. The distance between the partially filled neighbourhood $\mathcal{N}_1$ of the pixel being synthesized and neighbourhood $\mathcal{N}_2$ from the sample image (figure 6.2) can thus be expressed as

$$d(\mathcal{N}_1, \mathcal{N}_2) = \frac{\sum_{p \in \mathcal{N}} b_1(p) G(p - p_{\text{centre}}) (I_1(p) - I_2(p))^2}{\sum_{p \in \mathcal{N}} b_1(p) G(p - p_{\text{centre}})} \tag{6.1}$$

where the index p specifies both a pixel in $\mathcal{N}_1$ and its corresponding pixel in $\mathcal{N}_2$, b_1 is the binary mask set to zero for the pixels to be replaced, G is the square window of Gaussian weights and I denotes a greyscale value or a three-dimensional colour vector.

It can be noted that by drawing from the conditional probabilities after their approximation, this algorithm can be seen as a heuristic equivalent of a single pass of a Gibbs

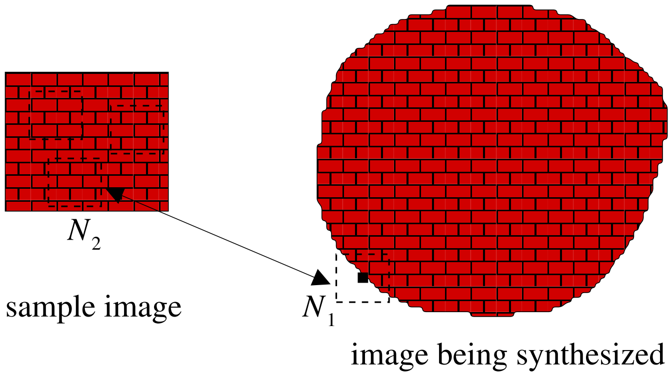


Figure 6.2: Efros and Leung's algorithm overview

sampler. Another point to which attention must be drawn is the choice of the similarity measure. Ideally, the chosen metric should be a good measure of perceptual similarity. Although this is certainly not the case for a metric based on the L^2 distance, simplicity and low computational cost are the main arguments in favour of this metric. Readers are invited to refer to [Efr 99] for a more detailed description of the original algorithm.

6.1.1 Constrained synthesis and pixel ordering

This algorithm is of interest to us not to generate isolated patches, but in the framework of synthesis with boundary constraints: missing regions are contained within parts of the image that must not be changed. We choose the synthesis ordering as follows: from the binary mask b which defines the pixels to be replaced, we can count for each missing pixel p the number of its valid neighbours. This number $M(p)$ is the number of unflagged pixels within the square neighbourhood window, weighted as previously by a Gaussian kernel:

$$M(p) = \sum_{q \in \mathcal{N}(p)} b(q) G(q-p) \tag{6.2}$$

Pixels are then replaced starting from the ones having the most valid neighbours. All missing regions are thus simultaneously and progressively filled from the edges to the centre (see figure 6.3). From our own experience, the choice of the pixel ordering has a significant influence on the result of the synthesis; this non-linear ordering ensures a good inward propagation of the surrounding information.

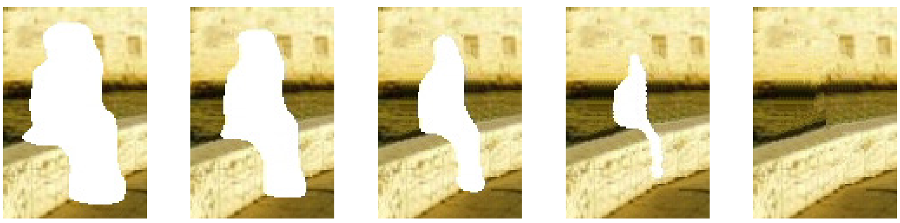


Figure 6.3: filling-in process

6.1.2 Adaptive sample image

The key difference between texture and an ordinary natural image is that the latter does not have the property of stationarity. It would therefore be much better modelled as a *non-stationary MRF*: the conditional probability distribution $P(I(p)|I(q), q \in \mathcal{N}(p))$ should then be different for each pixel p (see section 4.1.4.5). This can be achieved by approximating it from a different sample subimage for each pixel to be synthesized, rather than querying the same sample image for all pixels.

For each pixel, the corresponding adaptive sample image is constructed as follows: we start from a sample image with an initial size centred on the pixel under consideration. This sample image is then grown as long as it does not contain at least a minimum number of pixels outside the degraded area. Once this condition is met, the sample image is frozen and is ready to be subsequently searched. As a consequence, the sample image will be smaller for pixels near the edges of the missing regions than for those at the centre (figure 6.4).

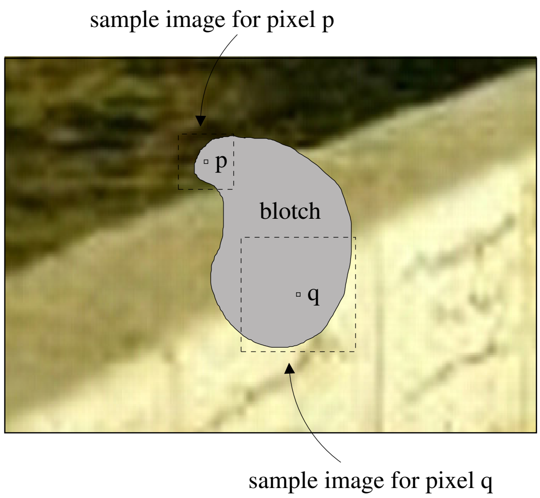
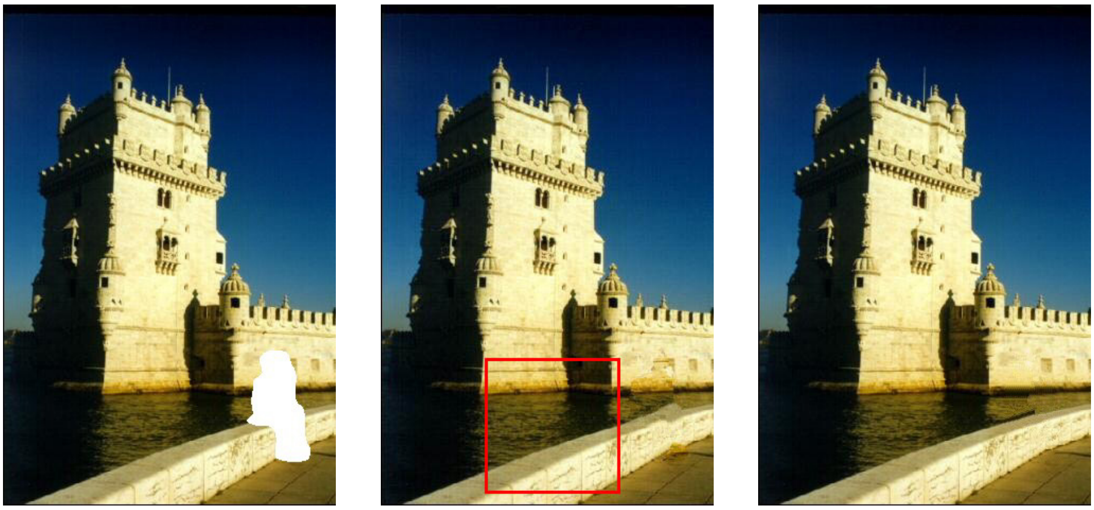


Figure 6.4: adaptive sample image. A different sample image is assigned to each pixel to be synthesized.

The improvement introduced here plays a crucial role in obtaining good quality results on general natural images. This is illustrated in figure 6.5: in this example, the use of adaptive sample images leads to a much better preservation of the curvature of the wall boundaries. In addition to making the sample subimage contain less irrelevant information, this also allows it to be much smaller. It has therefore the additional benefit of significantly reducing the computation time.

6.1.3 Coherence search and partial similarity computation

Most of the computation time is due to the similarity measure between neighbourhoods during the exhaustive search of the sample images. In order to reduce this cost, we



(a) Degraded image

(b) Correction with a single manual sample image (red box)

(c) Correction with adaptive sample images

Figure 6.5: correction with and without adaptive sample images

adopt the principle of *"coherence" search* introduced in [Ash 01] and also exploited in other applications [Her 01]. The idea is to rely, for each new pixel p, on the pixels used to replace the adjacent already synthesized pixels instead of starting the search from scratch. Each adjacent synthesized pixel q generates a "shifted" candidate t: the relative displacement between t and the pixel s used to replace q is the same as between p and q (figure 6.6). The most similar neighbourhood is found among these "shifted" candidates only. If the similarity is good enough, p is replaced accordingly; otherwise, the adaptive sample is constructed and exhaustively searched as before. Practically, we consider the similarity as good enough if $d(\mathcal{N}(p), \mathcal{N}(t)) \leq d(\mathcal{N}(q), \mathcal{N}(s))$.

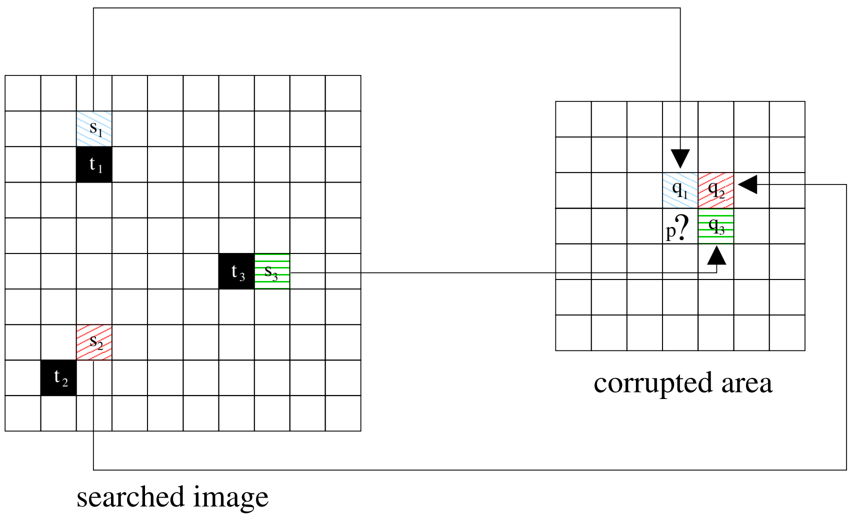


Figure 6.6: coherence search. The black pixels in the searched image are the "shifted" candidates.

With this approach, the vast majority of pixels are actually replaced without exhaustive search. A step further can be taken when the exhaustive search is triggered. For many neighbourhoods in the sample images, the similarity measure is undoubtedly too high. In many cases, the computation of the similarity can be stopped as soon as the partial measure is higher than a constantly updated reference value as detailed hereafter. During the exhaustive search of a particular sample image, let us denote d_{ref} the best similarity for the already tested neighbourhoods. When a new neighbourhood $\mathcal{N}_2$ is queried in its turn, if the computation of the similarity is frozen when only a portion $\mathcal{N}_{\text{part}}$ of the complete square window has been taken into account, we have the partial similarity:

$$d_{\text{part}}(\mathcal{N}_1,\mathcal{N}_2) = \frac{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}})(I_1(p)-I_2(p))^2}{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}})} \tag{6.3}$$

This similarity would decrease the most if all pixels in the remaining of the square window $\mathcal{N}\setminus\mathcal{N}_{\text{part}}$ were identical for the two neighbourhoods. The final similarity can thus be bounded by:

$$d(\mathcal{N}_1,\mathcal{N}_2) \geq \frac{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}})(I_1(p)-I_2(p))^2}{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}}) + \sum_{p\in\mathcal{N}\setminus\mathcal{N}_{\text{part}}} G(p-p_{\text{centre}})} \tag{6.4}$$

The denominator of the right-hand side can be bounded by a constant equal to the sum of all the Gaussian weights over the square window, $G_{\text{total}} = \sum_{p\in\mathcal{N}} G(p-p_{\text{centre}})$. We therefore have the following lower bound for the final similarity:

$$d(\mathcal{N}_1,\mathcal{N}_2) \geq \frac{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}})(I_1(p)-I_2(p))^2}{G_{\text{total}}} \tag{6.5}$$

For these reasons, we can simply test the following condition at each step of the similarity measure, each time a new pixel is taken into account in the partial sum:

$$\frac{\sum_{p\in\mathcal{N}_{\text{part}}} b_1(p)G(p-p_{\text{centre}})(I_1(p)-I_2(p))^2}{G_{\text{total}}} > (1+\epsilon)d_{\text{ref}} \tag{6.6}$$

As $d_{\text{ref}} \geq d_{\text{best}}$, if condition (6.6) is met, this necessarily implies $d(\mathcal{N}_1,\mathcal{N}_2) > (1+\epsilon)d_{\text{best}}$. Therefore, the neighbourhood already has no chance to be among the candidates for replacement and continuing to compute its similarity becomes pointless. On the opposite side, if the computation proceeds to its end and if the resulting similarity is lower than d_{ref}, then d_{ref} is updated accordingly.

6.1.4 Spatio-temporal synthesis

When considering an image within a sequence, the correction must take into account information from the previous and next frames; otherwise, even if the correction is unnoticeable when the image is frozen, its lack of consistency with the rest of the sequence will be grossly visible when animated. This can be simply achieved by searching three sample subimages instead of one: one is taken from the current image as we would do for still image correction, and the two others are taken from the previous and next frame respectively. For the previous and the next frame, the problem is that we must know where to extract the relevant information from, i.e. we need motion vectors. However, as the sample subimages cover quite a large area, there is no need for accurate motion vectors for each pixel and an approximate global motion vector is sufficient (figure 6.7).

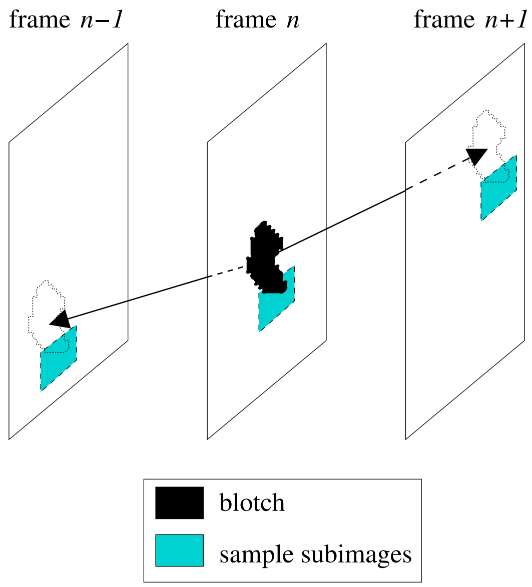


Figure 6.7: three sample subimages taken in the previous, current and next frame for each pixel to be synthesized

With this strategy to incorporate spatio-temporal information, the algorithm turns out to behave differently depending on the situation. When there is no or little motion, almost every pixel used for replacement belongs to the previous or the next frame. In this case, coherence search is remarkably efficient: whole regions are copied and there is nearly no computation at all. When the motion is more complex, such as in the case of pathological motion or when the same information is missing across several frames, much more similarities will be found with neighbourhoods belonging to the current frame. The algorithm thus implicitly falls back on a spatial correction. Experiments clearly confirm this relation between the perceived complexity of the motion and the ratio between temporal replacement (pixels taken from the previous or next frames) and spatial replacement (pixels taken from the current frame). This smooth adaptation of the algorithm without any user intervention is a major advantage and makes it suited to a wide range of motion conditions.

6.1.5 Summary

The main steps of the algorithm are summarized in figure 6.8.

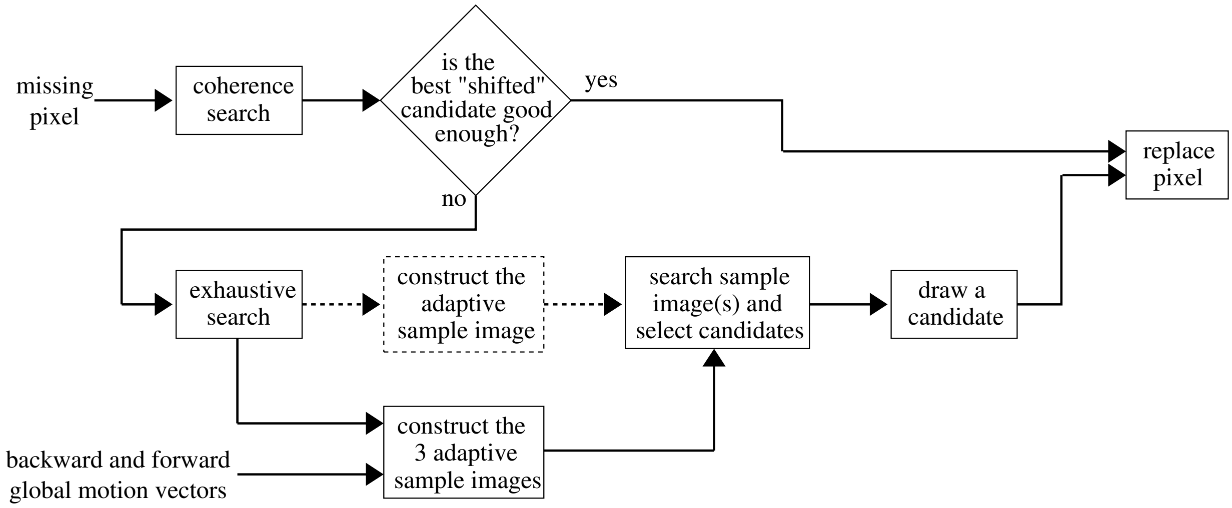


Figure 6.8: algorithm overview. The dashed part represents the modifications for still image correction.

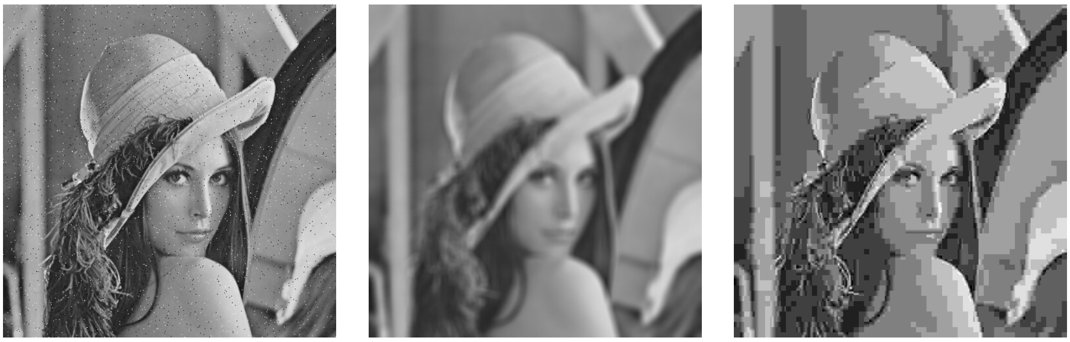
6.2 Discussion on evaluation

Methods for the measurement of perceived quality, which is at the core of the assessment of a correction, have been presented in section 3.1.4. Unfortunately, these current means of quantitative evaluation are far from satisfactory. In particular, PSNR and MSE are notorious for being very poor measures of perceived visual quality [Gir 93, Dal 93, Wan 02]. They are in widespread use because they are easy to compute and not completely deprived of interest for very specific distortions, such as noise, but they are definitely not adapted for the general-purpose measurement of image quality. This is, among others, illustrated in [Wan 02], where different distorted images with the same MSE but with a dramatically different perceived quality are shown side-by-side (see figure 6.9).

Surprisingly, more sophisticated measures do not necessarily perform better. A study group of the ITU, the Video Quality Experts Group (VQEG) has recently compared many of these measures and tested them against subjective evaluation [VQEG 00]. Although most of the tests were related to MPEG compression artifacts, some of the tests were also concerned with analog degradations. Among the conclusions of the study are the facts that most of the measures were not statistically distinguishable from PSNR and that “no objective measurement system in the test is able to replace subjective testing” ([VQEG 00], p. 50). For these reasons, it can be said that there is a clear lack of a well-accepted metric for the evaluation of image quality.

Given the fact that the current state of the art is still quite far from an ideal measurement, simple visual evaluation is believed to remain the best solution. We have therefore chosen to leave results to the subjective evaluation of the reader.

(a) Original Lena image (b) Contrast stretched, MSE = 225 (c) Gaussian noise contaminated, MSE = 225



(d) Impulsive noise contaminated, MSE = 225 (e) Blurred, MSE = 225 (f) JPEG compressed, MSE = 215

Figure 6.9: evaluation of “Lena” images distorted by different means (reproduced from [Wan 02])

6.3 Experimental results

The algorithm has been tested and validated on many different images and image sequences. The photograph dimensions are 700×466 in portrait or landscape orientation. All the sequences are excerpts from PAL video (720×576 pixels). Colour correction is performed in the YCbCr colour space, with one luminance and two chrominance components. The main parameters of the algorithm were chosen as follows:

- $\epsilon = 0.1$ for the selection of candidate neighbourhoods during exhaustive searches This means that all neighbourhoods that give a similarity within a 10% range of the best similarity are candidates for drawing and replacement.
- the standard deviation of the Gaussian kernel used in the similarity measure is chosen to be one third of the neighbourhood window size.
- the initial size of the adaptive sample subimage is taken to be 41×41. If necessary, it is extended until the sample contains at least 1500 valid pixels.

All these parameters are considered as internal settings and are never modified. As for the size of the neighbourhood window, unless stated otherwise, it was chosen to be 11×11: this gave the best results on all our testing dataset including the examples shown here. However, this size would have to be set to a different value for images with significantly different resolutions, e.g. high-definition (HD) material.

6.3.1 Results on still images

6.3.1.1 Comments on results

The algorithm gives very good results in various situations without any boundary artifact. It is able to reconstruct not only textured areas (figures 6.11(c)-(d) and 6.10(c)-(d)) or mixed regions (figure 6.10(a)-(b)), but also structured details or even sharp isolated elements such as the cable in the upper-right part of figure 6.11(a)-(b) or the edges of the pyramid in figure 6.11(e)-(f). The computation time significantly depends on the difficulty of the correction. It can range roughly from 0.2 to 1 ms per synthesized pixel using C++ code on a PC with a 1.5 GHz Pentium 4 processor. For a complete correction, the order of magnitude is usually from a few tenths of seconds to a few seconds per image. Typically, between 60 and 95% of the corrupted pixels are replaced by coherence search.

Figure 6.10: automated retouching

(a) (b)

(c) (d)

(e) (f)

Figure 6.11: removal of blotches, scratches, superimposed text or logo (original image (c) by courtesy of Emmanuelle Lecan)

6.3.1.2 Comparison with "image inpainting"

Our method is also compared to the "image inpainting" algorithm (section 3.3.1.2) based on the use of PDEs and which gives to our knowledge the most impressive results for still images. Images in this thesis generated by this algorithm were taken from the web page of the project[1]. As shown in figure 6.12, both algorithms give very good results of comparable quality though our algorithm requires a much smaller computational cost: a few minutes are required for image inpainting. For the correction of figure 6.12(c), our neighbourhood window was chosen to be 9×9 given the low resolution of the original image. While having a comparable quality, the results have a different "look and feel" for the two algorithms: "image inpainting" results have a tendency to look "rubbed off" and the results of our algorithm usually exhibit what could be described as a "high frequency" aspect. The difference is more visible for textured regions (figure 6.13), especially when these regions are large: in these cases information tends to be over-blurred with the inpainting algorithm.

(a) Original image

(b) Correction with image inpainting

(c) Correction with our algorithm

Figure 6.12: comparison of correction results

[1] http://www.ece.umn.edu/users/marcelo/restoration.html

(a) Detail of the original image

(b) Correction with image inpainting

(c) Correction with our algorithm

(d) Detail of the original image

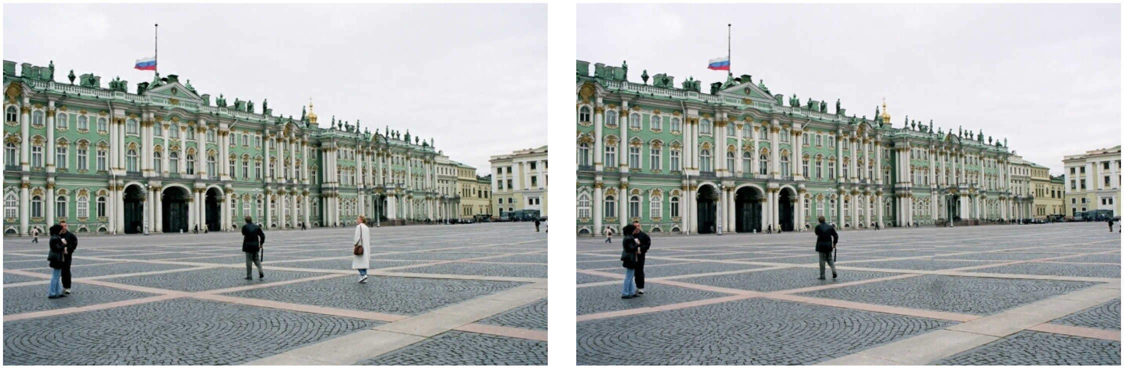
(e) Correction with image inpainting

(f) Correction with our algorithm

Figure 6.13: correction details for textured regions. Here, (b) and (e) tend to be over-blurred.

6.3.1.3 Limitations

A clear limitation of our algorithm as of most other techniques is their lack of high-level understanding: artifacts will appear when the surrounding does not contain the necessary information. For example, it will not able to reconstruct the head of a character if nothing in the surrounding gives any "hint" of what a head should be. Apart from that, very few examples of visible imperfections have been reported. Figure 6.14 is one very interesting illustration of noticeable correction: it contains in the same time textured regions and geometric elements with great variations in size because of the perspective.



(a) Original image and correction

(b) Zoom in on the region of interest

Figure 6.14: example of visible correction

6.3.2 Results on image sequences

The algorithm has been tested on whole sequences with impulsive defects or on areas missing across several frames. The global motion vectors taken as an input by the algorithm as explained in section 6.1.4 are computed using phase correlation (see appendix B). The algorithm is applied recursively, i.e. frame $n-1$ used in the correction of frame n is the already corrected frame $n-1$. This has proved to be much more efficient than non-recursive correction. The image sequence of figure 6.15 illustrates how the algorithm adapts to the spatio-temporal context: the blotches and the background behind the scratch are corrected using temporal information whereas the man which is spanned by the scratch over several frames is mainly reconstructed from the spatial surrounding. The algorithm has also proved to be efficient on sequences with pathological motion, as illustrated on figure 6.16 with motion blur and figure 6.17 with erratic motion. In this case, the algorithm implicitly falls back on a spatial correction and relatively few pixels are taken from the previous or the next frame. The computation time and the ratio of pixels replaced by coherence search are within the same range as for still images.

(a) Degraded frame $n-1$

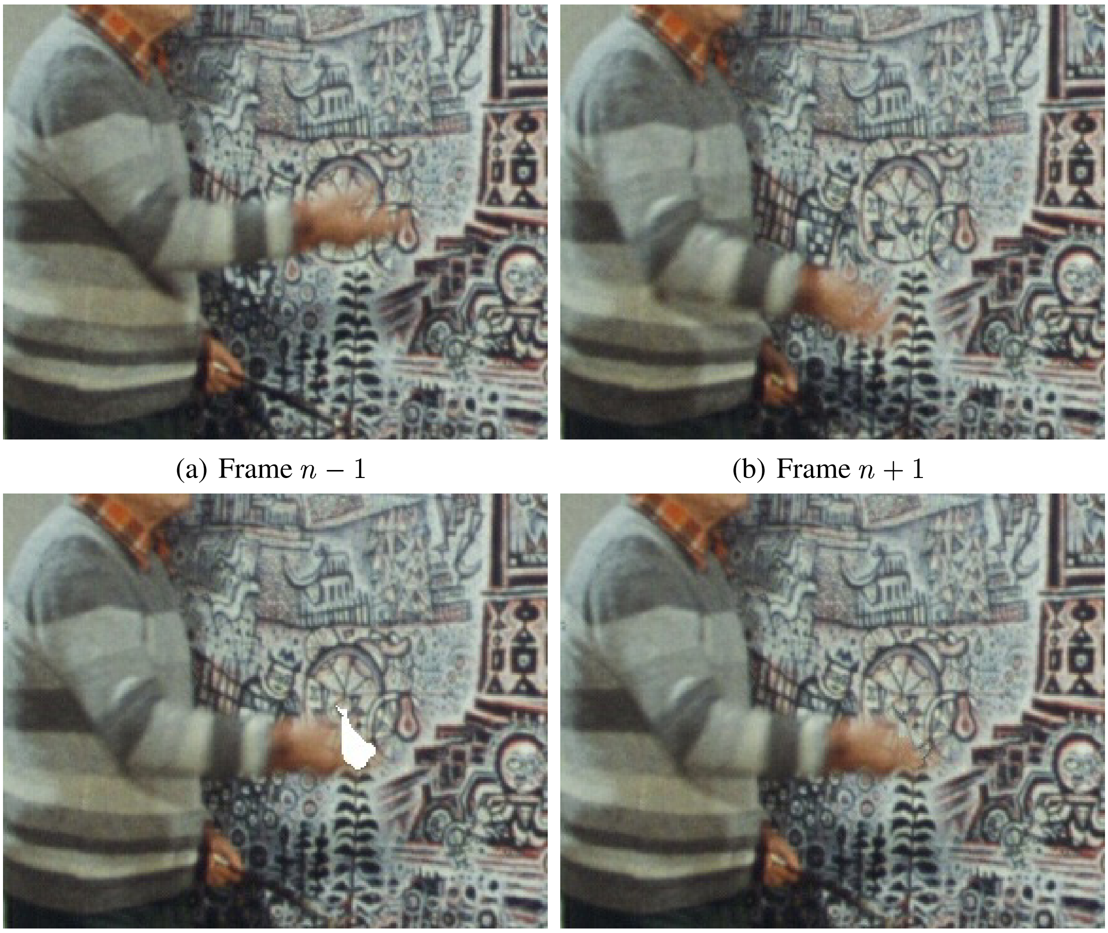
(b) Degraded frame $n+1$

(c) Degraded frame n

(d) Frame n after correction

Figure 6.15: restoration of an image sequence with artificial blotches and scratch

However, a distinction in the results should be made between impulsive blotches and missing areas spanning several frames. For impulsive defects, all the results obtained are of great quality and virtually invisible. For defects remaining across several frames, some imperfections may still occasionally be noticeable. They appear as unnatural activity at precise moments in the reconstructed area and are usually not noticeable when the image is frozen. In other words, at these moments, the correction is not coherent enough temporally with the rest of the sequence despite the use of the previous and next frame in the correction.



(a) Frame $n-1$ (b) Frame $n+1$

(c) Degraded frame n (d) Frame n after correction

Figure 6.16: restoration within a pathological motion area (motion blur)

6.4 Final comments

The algorithm presented in this chapter for general-purpose missing data correction produces very good visual results and runs reasonably fast without any user intervention. When processing video, which is of prime importance to us, it has the additional benefit to avoid the tricky step of motion vector repair as it does not involve motion compensation. Even more important is the fact that it smoothly adapts to the motion conditions: much more information is automatically fetched from the spatial vicinity of the missing area in the case of pathological motion.

(a) Frame $n-1$

(b) Frame $n+1$

(c) Degraded frame n

(d) Frame n after correction

Figure 6.17: restoration within a pathological motion area (erratic motion)

Two directions could be investigated to improve even further the results of our algorithm. Firstly, as just mentioned, there is a need for more temporal coherence when the damaged region spans several frames. Secondly, as for most other correction methods, it is assumed here that there is no more underlying information in the damaged regions, which is generally not true. These two point are developed in the concluding chapter 8.

Chapter 7

Experimentation of the complete prototype

Our detection scheme described in chapter 5 is now combined with our correction technique detailed in the previous chapter to build a complete prototype for impulsive defect concealment. A delay of one frame must be kept between the detection and correction processes: correction in frame $n-1$ can only proceed after detection in frame n (figure 7.1). This is required to ensure that when correcting frame $n-1$, blotched pixels are known in frame n and will consequently not be used for replacement.

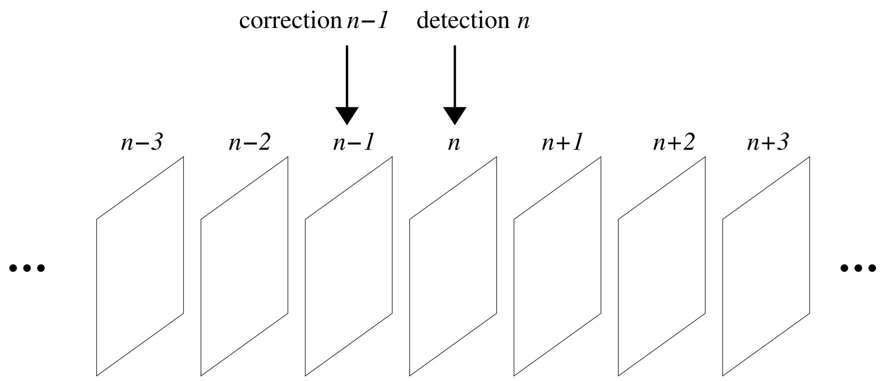
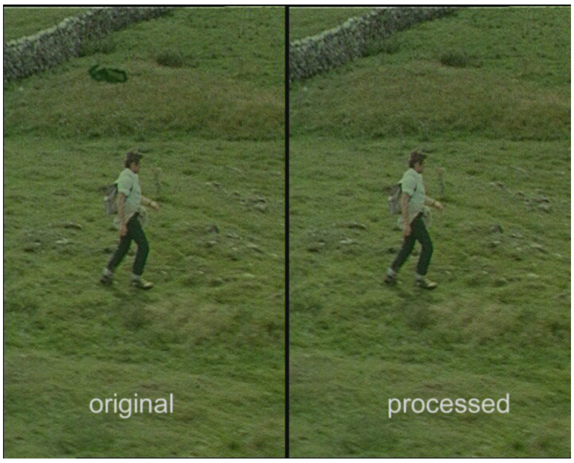


Figure 7.1: combination of detection and correction

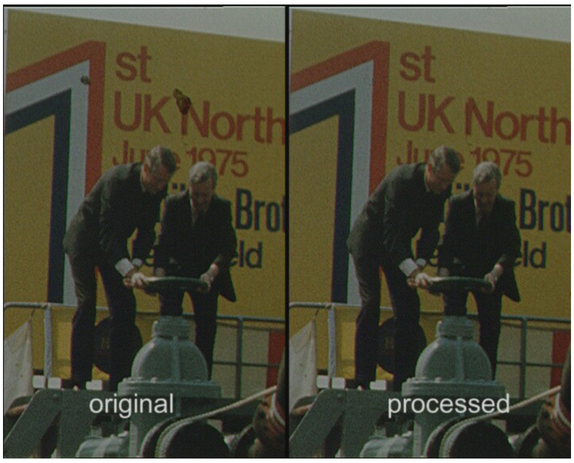
When the two steps are combined, the settings of the detection step can be "lowered" to detect as many blotches as possible, even to the point where false alarms start to appear: slight false alarms are generally well handled by our correction scheme. Typical detection settings that have been used for the complete concealment system are $\alpha = 0.005$, $\beta_1 = 0.3$, $\beta_2 = 0.26$, $\beta_3 = 0.01$ for the detection of temporal discontinuities, $r = 25$ and $M = 15$ for the interpretation. The size of the square neighbourhood window for correction is kept to 11×11. These settings have been chosen after a few trials and errors, by tuning the parameters according to their intuitive meaning and to the visual characteristics of the outcome. The same video sequences stretching over several minutes used for the validation of detection have been processed successfully without modifying these parameters.

(a)

(b)

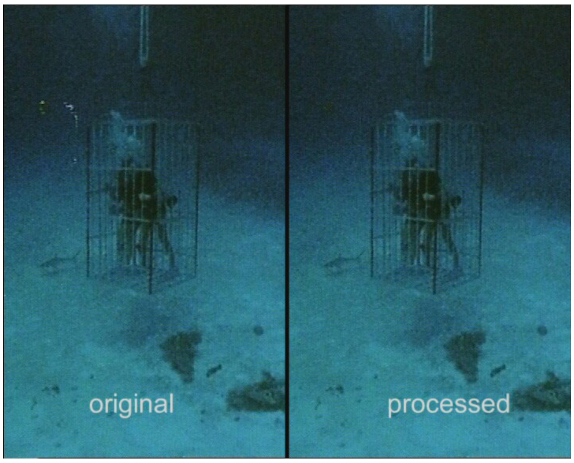


(c)



(d)

(e)



(f)

Figure 7.2: processing with the complete prototype

The total computation time per frame for the complete prototype can be roughly decomposed as follows (for a 1.5 GHz Pentium 4 processor):

- 7s for motion estimation and compensation
- 5s for blotch detection (localization of discontinuities and interpretation)
- 0.5s for blotch correction

Chapter 8

Conclusions and further research

This thesis has addressed the problem of impulsive defects by introducing the important notion of pathological motion. Based on this notion, we presented a new detection method which dramatically reduces the number of false alarms. We also developed in this thesis a general-purpose correction algorithm which is able to handle successfully these cases as well. These algorithms have proven to be efficient in a wide range of conditions. They achieve a high degree of automation as their parameters do not need to be changed to maintain efficiency in different contexts. The resulting computational cost is relatively low and frames are usually processed within seconds. As a continuation of our work, four promising points can be highlighted as interesting directions for future research.

8.1 Go further in and around PM

A marginal drawback of our detection scheme is the fact that some blotches are now incorrectly flagged as pathological motion, either because they are *close to pathological motion* or because they are *within pathological motion areas*. To be able to correctly handle these cases, detection must there undoubtedly rely on criteria other than temporal impulsivity. The approach undertaken in [Rar 01] is to classify and characterise further each subtype of pathological motion, e.g. motion blur. While this approach could be interesting in very specific situations, we believe that it may lack generality. An alternative point that could be exploited and has never been yet is the fact that blotches often look similar within the same sequence. In other words, we could make the assumption of stationarity of well-chosen blotch characteristics along the sequence. This could be exploited by gathering statistics on blotches identified by our detection method and by using these statistics to take more sophisticated decisions on what our method flags as pathological motion. On each detected blotch, global characteristics can thus be computed and the corresponding statistics can be updated on-the-fly. In a first step, these statistics could be applied to whole connected sets flagged as PM: if the set complies with the gathered statistics, it is likely to be a real blotch after all; otherwise, the set is likely to be mainly pathological motion and is left for closer analysis. This would allow to correctly re-classify blotches *around pathological motion* and it is believed that this could be achieved without major difficulties. In a second step, if we want to be able to identify blotches *within pathological motion areas*, a segmentation of these areas would be necessary and each segment would have to be individually tested against the gathered statistics. This step would probably require much more effort to become really efficient. The exact nature of the characteristics to be chosen (size, mean luminance, degree of uniformity, compacity,...) remains an open question.

8.2 More temporal coherence for missing areas across several frames

For our correction algorithm, when the damaged region spans several frames such as in the case of line scratches or the concealment of a character tracked over a sequence, there is a need for more temporal coherence. This could for example be achieved by incorporating more frames in our correction, e.g. two frames before and two frames after. It could also be interesting to incorporate frames that are not immediately before and after, e.g. frames $n - 20$ and $n + 20$: it is not uncommon that with the evolution of the scene, information which is missing in frame n and in the surrounding frames becomes uncovered and therefore available in more distant frames. As our algorithm provides enough flexibility to incorporate these changes while preserving its simplicity, it is believed that this could be tested and improved quite easily.

8.3 Incorporate remaining underlying information

Our correction method as well as most others makes the implicit assumption that there is no more underlying information in the damaged region. While this is true for dropouts, this is generally not true for other missing data artifacts: dirt is often not completely opaque and gelatine sparkles or scratches can be more or less superficial. Ideally, we should attempt to employ this remaining information to some extent. This implies in particular being able to estimate the degree of underlying information in the corrupted area. Incorporating this information within the correction algorithm presented in chapter 6 is believed to be a very promising point for further investigation.

8.4 Video quality metrics

This thesis raised the remark that current means for the quantitative evaluation of detection and correction are far from satisfactory. We also stressed that the detection step cannot be properly evaluated without keeping in mind the fact that it will be followed by a correction step with shortcomings of its own. A proper evaluation of both steps therefore revolves around a good objective video quality metric. While this is a longer-term prospect, it is believed that the temporal dimension should play a very significant role in this metric. Such a metric would have to be fully validated by tedious but necessary psychovisual experiments involving human viewers.

It should be finally stressed that all these measurements will at best be good hints about the efficiency of one technique or another. They can prove a very valuable tool and can ideally complement but not replace extensive experiments on real degraded sequences with all their variety and unpredictability. International efforts on this issue such as the second phase currently under way of the work conducted by the Video Quality Experts Group should be followed closely.

Appendix A

The YCbCr colour space

The YCbCr colour space used in digital video can be defined from the RGB colour space by the following formulas, with values between 0 and 255:

$$
\begin{array}{rcrlrlrlrl}
Y & = & & 0.2989\,R & + & 0.5866\,G & + & 0.1145\,B & & \\
Cb & = & - & 0.1688\,R & - & 0.3312\,G & + & 0.5000\,B & + & 128 \\
Cr & = & & 0.5000\,R & - & 0.4184\,G & - & 0.0816\,B & + & 128
\end{array}
\tag{A.1}
$$

Conversely, the RGB values are given from the YCbCr values by:

$$
\begin{array}{rcrlll}
R & = & Y & + & 1.4022\,(Cr - 128) & \\
G & = & Y & - & 0.3456\,(Cb - 128) & - \; 0.7145\,(Cr - 128) \\
B & = & Y & + & 1.7710\,(Cb - 128) &
\end{array}
\tag{A.2}
$$

These conversion rules are specified with an accuracy of three digits after the first decimal in the 4:2:2 standard (see [ITU 95] and section 2.1.2.2).

In the 8-bit version of the 4:2:2 standard, 220 quantization levels are authorised for the luminance signal, with the black level corresponding to 16 and the peak white level corresponding to 235. Similarly, 225 quantization levels are possible for the Cb and Cr signals, ranging from 16 to 240. For these two signals, the zero value corresponds to level 128. To get values within these ranges, the formulas above have to be scaled accordingly, which yields from RGB to YCbCr:

$$
\begin{array}{rcrlrlrlrl}
Y & = & & 0.2567\,R & + & 0.5038\,G & + & 0.0983\,B & + & 16 \\
Cb & = & - & 0.1483\,R & - & 0.2909\,G & + & 0.4392\,B & + & 128 \\
Cr & = & & 0.4392\,R & - & 0.3675\,G & - & 0.0717\,B & + & 128
\end{array}
\tag{A.3}
$$

and from YCbCr to RGB:

$$
\begin{array}{rclll}
R & = & 1.1644\,(Y - 16) & + \; 1.5963\,(Cr - 128) & \\
G & = & 1.1644\,(Y - 16) & - \; 0.3934\,(Cb - 128) & - \; 0.8134\,(Cr - 128) \\
B & = & 1.1644\,(Y - 16) & + \; 2.0161\,(Cb - 128) &
\end{array}
\tag{A.4}
$$

with Y $\in [16, 235]$, Cb and Cr $\in [16, 240]$, R, G and B $\in [0, 255]$.

Appendix B

Phase-correlation motion estimation

B.1 Quick overview of motion estimation

Motion estimation deals with estimating the “apparent” motion (or optical flow) between two images based on intensity variations. Dense flow fields which associate a motion vector to each pixel are usually estimated using only the luminance information. The fundamental assumption on which most estimators rely is the conservation of optical flow: the intensity of a given point is assumed to remain constant along its motion trajectory. The main families of motion estimation techniques are:

- *differential methods*, which are based on the differential equation expressing the conservation of optical flow. This equation is coupled with a regularization equation usually imposing a smoothness constraint on the motion field. Expressed as an energy minimization problem, these methods often lead to a large linear system of equations. Well-known differential methods include [Hor 81, Nag 86, Luc 81].

- *block-based methods*, which consider in a first step the motion of entire blocks of pixels. They can take place in the spatial domain, such as block-matching algorithm [Bie 88] or in the Fourier domain, such as phase-correlation methods. These methods are very widespread in hardware encoders for MPEG-1 or MPEG-2 compression because of their relatively low complexity.

- *pel-recursive methods*, which can be seen as predictor-corrector estimators. The initial prediction for a given pixel is taken as the final estimation for the previously processed pixel. This prediction is updated by minimizing a quantity related to the optical flow equation. [Rob 83, Bie 87] are typical pel-recursive estimators.

- *Bayesian methods*, such as [Kon 92] based on the tools described in chapter 4 with motion fields as the unknowns.

A more detailed overview of the wealth of existing motion estimation techniques can be found in [Sti 99], [Bar 94], [Bea 95] and [Tek 95] chap. 5 to 8.

We implemented for our purpose a phase-correlation motion estimator. This type of estimator was mainly chosen for its ease of implementation and its computational speed. It is definitely not the most advanced motion estimator, but it is sufficient to illustrate the validity of our approach as this latter is meant to be efficient with any type of motion estimation.

B.2 Principle of phase-correlation motion estimation

Let us consider the case of a translation d between images I_n and I_{n+1}:

$$I_{n+1}(x + d) = I_n(x) \tag{B.1}$$

In the Fourier domain, this relation becomes:

$$\hat{I}_{n+1}(f) = \hat{I}_n(f) \exp(2i\pi f \cdot d) \tag{B.2}$$

where $\hat{}$ denotes the Fourier transform. In other words, a translation in the spatial domain is equivalent to a phase shift in the Fourier domain. Phase-correlation motion estimation relies on this principle [BBC 87, Wat 94b].

In the phase-correlation method, we compute the cross-correlation function of the two frames:

$$C_{n,n+1}(x) = I_n(x) * I_{n+1}(-x) \tag{B.3}$$

where $*$ is the 2-D convolution operation. As convolutions in the spatial domain are equivalent to multiplications in the Fourier domain, taking the Fourier transform of this equation gives:

$$\hat{C}_{n,n+1}(f) = \hat{I}_n(f)\hat{I}^*_{n+1}(f) \tag{B.4}$$

in which the exponent * denotes the complex conjugate. Normalizing this quantity yields the phase of the cross-power spectrum:

$$\hat{c}_{n,n+1}(f) = \frac{\hat{I}_n(f)\hat{I}^*_{n+1}(f)}{|\hat{I}_n(f)\hat{I}^*_{n+1}(f)|} \tag{B.5}$$

We finally take the inverse Fourier transform of this expression, $c_{n,n+1}(x)$, which is called the phase-correlation function.

If we now consider a translational motion such as the one expressed by relation (B.2), we then have:

$$\begin{aligned}
\hat{c}_{n,n+1}(f) &= \frac{\hat{I}_n(f)\hat{I}^*_n(f)\exp(-2i\pi f \cdot d)}{|\hat{I}_n(f)\hat{I}^*_n(f)\exp(-2i\pi f \cdot d)|} \\
&= \frac{|\hat{I}_n(f)|^2 \exp(-2i\pi f \cdot d)}{|\hat{I}_n(f)|^2} \\
&= \exp(-2i\pi f \cdot d)
\end{aligned} \tag{B.6}$$

From that point, the inverse Fourier transform gives:

$$c_{n,n+1}(x) = \delta(x - d) \tag{B.7}$$

where δ is the Dirac function. In practice, even by replacing the Fourier transform by the discrete Fourier transform (DFT), image functions by discrete arrays and knowing that a periodic extension of the images outside their support is not realistic, this succession of operations works remarkably well and a large peak centred on the displacement d is noticeable. For subpixel displacements, it can be shown that the Dirac function becomes a 2-D Dirichlet kernel, which can be closely approximated by a 2-D sinc kernel [For 02]. When there are multiple displacements in the frame, all the corresponding peaks are present in the phase-correlation function.

An important property of phase-correlation motion estimation is that it is largely insensitive to global intensity fluctuations. In particular, in the case of film and video archives, this type of motion estimation is very robust to intensity flicker (see section 2.3). The main drawback of these motion estimators is that they do not handle very well rotational and zooming motion.

Based on this principle, motion estimation is performed in two steps: motion vector candidates are first extracted on a block basis and one of these candidates is then assigned to each pixel.

B.3 Candidate extraction

The image is first divided into half-overlapping blocks, 128×128 blocks in our case. The half-size of the blocks corresponds to the maximum displacement which can be estimated with this method. For each block in frame n and the corresponding block in image $n+1$, we compute the phase-correlation function. The main dominant peaks are then detected and their coordinates are considered as motion vector candidates. We chose to detect the 4 main peaks and to obtain their subpixel location with a simple curve fitting strategy. The relative heights of the peaks reflect the relative areas of moving objects or background (figure B.1).

B.4 Candidate assignment

During the second step, the candidates are assigned on a pixel basis. As most pixels belong to 4 overlapping blocks, 16 candidates have to be tested for these pixels in our case. The chosen candidate is the one minimizing the mean DFD over a small window centred on the pixel. These local differences can be computed efficiently using separable recursive filters. Figure B.2 shows an example of motion field obtained by phase correlation.

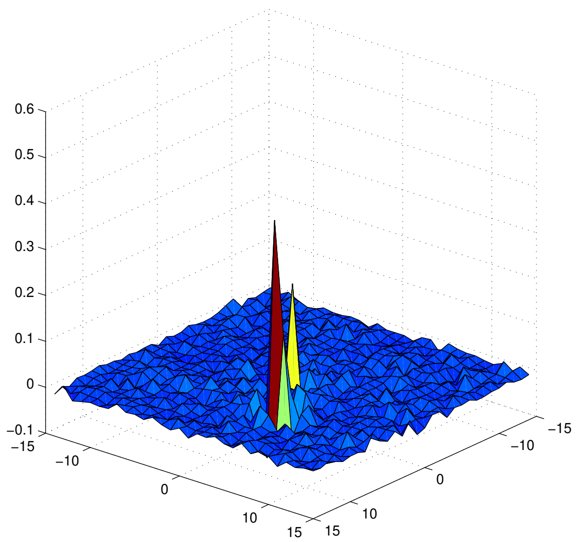


Figure B.1: phase-correlation function. The two peaks correspond here to a large moving object on a fixed background.

Figure B.2: overlaid dense motion field

Appendix C

Testing architecture

In this appendix, we describe the testing architecture that we had to set up to conduct experiments on whole sequences extracted from video tapes. Our testing architecture involves the following elements:

- a *professional VTR* with digital input/output capabilities. In our case, this was a Digital Betacam player/recorder.
- a *video server* which is able to store uncompressed video. We used a MMS ProntoServer with a capacity of approximately 40 minutes of uncompressed video. This video server is connected to the VTR through a digital video link (SDI link, Serial Digital Interface). It is also able to control the VTR using a standard protocol for the remote control of broadcast devices (RS 422 protocol).
- a *remote host* connected the video server through a SCSI bus (Small Computer System Interface). This remote host is a Silicon Graphics O2 in our case.
- a *local computer* which performs all the processing. We used a Linux PC with a 1.5 GHz Pentium 4 processor.

In our C++ code, we implemented the built-in capability to issue system calls over the local network from the local computer to the remote host. Our prototype which is run on the local computer is thus able to ask the remote host to get or put back images from/to the video server and to transmit them over the network.

The processing of whole sequences from a video tape is conducted as follows. The sequences are first entirely transferred from the VTR to the video server. A few frames are then transmitted from the video server to the local computer which can start to process them. Each time a frame has been processed, the local computer sends the result back to the video server via the remote host and asks for a new frame. The frames are thus progressively exchanged between the video server and the local computer, which only needs to keep a very limited number of frames in memory. For our combined detection and correction algorithms, this number amounts to five frames. At the end of the processing, the whole sequences can be put back to tape.

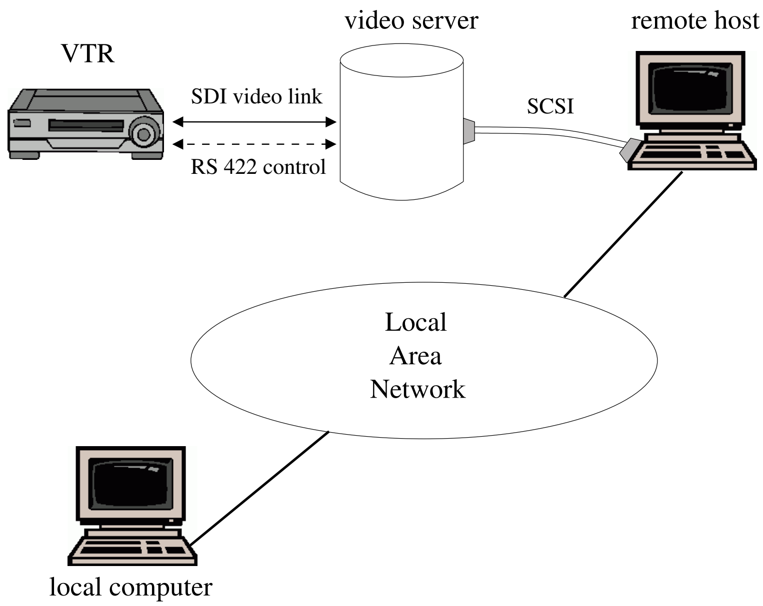


Figure C.1: testing architecture

With this testing architecture, only a very moderate amount of disk space is required on the local computer as the sequences remain on the video server. The other advantage of this architecture is its flexibility: the computational burden can be laid on any machine connected to the network without having to change the cable configuration. The overhead resulting from network queries and transmissions is around 2 seconds per frame.

Bibliography

94 listed references

[All 90] Arnold O. Allen. *Probability, Statistics and Queuing Theory with Computer Science Applications* (2nd edition). Academic Press, San Diego, 1990.

[Arc 91] Gonzalo R. Arce. *Multistage Order Statistic Filters for Image Sequence Processing*. IEEE Transactions on Signal Processing, Vol. 39, No. 5, pp. 1146-1163, May 1991.

[Arm 99] Steven Armstrong. *Film and Video Restoration using Rank-Order Models*. PhD Thesis, Cambridge University, United Kingdom, June 1999.

[Ash 01] Michael Ashikhmin. *Synthesizing Natural Textures*. Proceedings of the ACM Symposium on Interactive 3D Graphics, pp. 217-226, Research Triangle Park, USA, March 2001.

[Bar 94] John L. Barron, David J. Fleet and Steven S. Beauchemin. *Performance of Optical Flow Techniques*. International Journal of Computer Vision, Vol. 12, No. 1, pp. 43-77, 1994.

[BBC 84] Richard Storey. *Electronically detecting the presence of film dirt*. UK Patent Specification No. GB2139039, British Broadcasting Corporation, 31 October 1984.

[BBC 87] Graham A. Thomas. *TV picture motion measurement*. UK Patent Specification No. GB2188510, British Broadcasting Corporation, 30 September 1987.

[Bea 95] Steven S. Beauchemin and John L. Barron. *The Computation of Optical Flow*. ACM Computing Surveys, Vol. 27, No. 3, pp. 433-467, September 1995.

[Ber 91] Claude Berge. *Graphs* (3rd revised edition). North-Holland, Amsterdam, 1991.

[Ber 00] Marcelo Bertalmío, Guillermo Sapiro, Vicent Caselles and Coloma Ballester. *Image Inpainting*. Proceedings of SIGGRAPH 2000, pp. 417-424, New Orleans, USA, July 2000.

[Ber 01] Marcelo Bertalmío. *Processing flat and non-flat image information on arbitrary manifolds using Partial Differential Equations*. PhD Thesis, University of Minnesota, USA, March 2001.

[Bes 74] Julian Besag. *Spatial Interaction and the Statistical Analysis of Lattice Systems*. Journal of the Royal Statistical Society, Series B, Vol. 36, No. 2, pp. 192-236, 1974.

[Bes 86] Julian Besag. *On the Statistical Analysis of Dirty Pictures*. Journal of the Royal Statistical Society, Series B, Vol. 48, No. 3, pp. 259-302, 1986.

[Bie 87] J. Biemond, L. Looijenga, D.E. Boekee and R.H.J.M. Plompen. *A pel-recursive Wiener-based displacement estimation algorithm*. Signal Processing, Vol. 13, pp. 399-412, 1987.

[Bie 88] M. Bierling. *Displacement estimation by hierarchical blockmatching*. Proceedings of SPIE Conference on Visual Communications and Image Processing, Vol. 1001, pp. 942-951, Cambridge, USA, November 1988.

[Bra 95] James C. Brailean, Richard P. Kleihorst, Serafim N. Efstratiadis, Aggelos K. Katsaggelos and Reginald L. Lagendijk. *Noise Reduction Filters for Dynamic Image Sequences: a Review*. Proceedings of the IEEE, Vol. 83, No. 9, pp. 1272-1292, September 1995.

[Bre 99] Pierre Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulations and Queues*. Springer-Verlag, New York, 1999.

[Bui 97] Olivier Buisson. *Analyse de séquences d'images haute résolution, application à la restauration numérique de films cinématographiques*. PhD Thesis, Université de La Rochelle, France, December 1997.

[Can 86] John Canny. *A Computational Approach to Edge Detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 8, No. 6, pp. 679-698, November 1986.

[Cha 00] Annabelle Chardin. *Modèles énergétiques hiérarchiques pour la résolution des problèmes inverses en analyse d'images, application à la télédétection*. PhD Thesis, Université de Rennes I, France, January 2000.

[Che 98] Jean-Hugues Chenot, John O. Drewery and David Lyon. *Restoration of Archived Television Programmes for Digital Broadcasting*. Proceedings of the International Broadcasting Convention (IBC'98), pp. 26-31, Amsterdam, The Netherlands, September 1998.

[Cho 97] M.N. Chong, P. Liu, W.B. Goh and D. Krishnan. *A new spatio-temporal MRF model for the detection of missing data in image sequences*. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'97), Vol. IV, pp. 2977-2980, Munich, Germany, April 1997.

[Dal 93] Scott Daly. The visible differences predictor: an algorithm for the assessment of image fidelity. In *Digital Images and Human Vision* (A.B. Watson, ed.), pp. 179-206, MIT Press, Cambridge, 1993.

[Dec 97] Etienne Decencière Ferrandière. *Restauration automatique de films anciens*. PhD Thesis, Ecole des Mines de Paris, France, December 1997.

[Dek 01] Fabien Dekeyser. *Restauration de séquences d'images par des approches spatio-temporelles : filtrage et super-résolution par le mouvement*. PhD Thesis, Université de Rennes I, France, November 2001.

[Dem 77] A.P. Dempster, N.M. Laird and D.B. Rubin. *Maximum Likelihood from Incomplete Data via the EM Algorithm*. Journal of the Royal Statistical Society, Series B, Vol. 39, No. 1, pp. 1-38, 1977.

[Der 87] Rachid Deriche. *Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector*. International Journal of Computer Vision, Vol. 1, No. 2, pp. 167-187, May 1987.

[Duv 95] Loïc Duval. *Le système de télévision couleur Pal*. INA Formation, Bry-sur-Marne, 1995.

[Duv 86] Loïc Duval. *Le système de télévision couleur Secam*. INA Formation, Bry-sur-Marne, 1986.

[Duv 96] Loïc Duval. *Le système de télévision couleur NTSC*. INA Formation, Bry-sur-Marne, 1996.

[Efr 99] Alexei A. Efros and Thomas K. Leung. *Texture Synthesis by Non-parametric Sampling*. Proceedings of the IEEE International Conference on Computer Vision (ICCV'99), Vol. 2, pp. 1033-1038, Corfu, Greece, September 1999.

[Efr 01] Alexei A. Efros and William T. Freeman. *Image Quilting for Texture Synthesis and Transfer*. Proceedings of SIGGRAPH 2001, pp. 341-346, Los Angeles, USA, August 2001.

[For 02] Hassan Foroosh, Josiane Zerubia and Marc Berthod. *Extension of Phase Correlation to Subpixel Registration*. IEEE Transactions on Image Processing, Vol. 11, No. 3, pp. 188-200, March 2002.

[Gem 84] Stuart Geman and Donald Geman. *Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 6, No. 6, pp. 721-741, November 1984.

[Gem 92] Stuart Geman, Donald E. McClure and Donald Geman. *A Nonlinear Filter for Film Restoration and Other Problems in Image Processing*. CVGIP: Graphical Models and Image Processing, Vol. 54, No. 4, pp. 281-289, July 1992.

[Gir 93] Bernd Girod. What's wrong with mean-squared error? In *Digital Images and Human Vision* (A.B. Watson, ed.), pp. 207-220, MIT Press, Cambridge, 1993.

[Gra 95] Christine Graffigne, Fabrice Heitz, Patrick Pérez, Françoise Prêteux, Marc Sigelle and Josiane Zerubia. *Hierarchical Markov Random Field Models Applied to Image Analysis: a Review*. Proceedings of SPIE Conference on Morphological Image Processing and Random Image Modeling, Vol. 2528, pp. 2-17, San Diego, USA, July 1995.

[Har 97] Neil R. Harvey and Stephen Marshall. *Application of Non-linear Image Processing: Digital Video Archive Restoration*. Proceedings of the IEEE International Conference on Image Processing (ICIP'97), Vol. 1, pp. 731-734, Santa Barbara, USA, October 1997.

[Har 01] Paul Harrison. *A non-hierarchical procedure for re-synthesis of complex textures*. Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG 2001), pp. 190-197, Plzen, Czech Republic, February 2001.

[Hei 94] Fabrice Heitz, Patrick Pérez and Patrick Bouthemy. *Multiscale Minimization of Global Energy Functions in Some Visual Recovery Problems*. CVGIP: Image Understanding, Vol. 59, No. 1, pp. 125-134, January 1994.

[Her 01] Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless and David H. Salesin. *Image Analogies*. Proceedings of SIGGRAPH 2001, pp. 327-340, Los Angeles, USA, August 2001.

[Hir 96] Anil N. Hirani and Takashi Totsuka. *Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal*. Proceedings of SIGGRAPH 1996, pp. 269-276, New Orleans, USA, August 1996.

[Hor 81] Berthold K.P. Horn and Brian G. Schunk. *Determining optical flow*. Artificial Intelligence, Vol. 17, pp. 185-203, 1981.

[Ige 97] Homan Igehy and Lucas Pereira. *Image Replacement through Texture Synthesis*. Proceedings of the IEEE International Conference on Image Processing (ICIP'97), Vol. 3, pp. 186-189, Santa Barbara, USA, October 1997.

[ITU 95] Recommendation ITU-R BT.601-5. *Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios*. International Telecommunication Union, section 11B, 1995.

[ITU 00] Recommendation ITU-R BT.500-10. *Methodology for the Subjective Evaluation of the Quality of Television Pictures*. International Telecommunication Union, 2000.

[Joy 99] Laurent Joyeux, Olivier Buisson, Bernard Besserer and Samia Boukir. *Detection and Removal of Line Scratches in Motion Picture Films*. Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'99), pp. 548-553, Fort Collins, USA, June 1999.

[Joy 00] Laurent Joyeux. *Reconstruction de séquences d'images haute résolution, application à la restauration de films cinématographiques*. PhD Thesis, Université de La Rochelle, France, January 2000.

[Kal 97] S. Kalra, M.N. Chong and D. Krishnan. *A new auto-regressive (AR) model-based algorithm for motion picture restoration*. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'97), Vol. IV, pp. 2557-2560, Munich, Germany, April 1997.

[Kin 90] T. Kinugasa, N. Yamamoto, H. Komatsu, S. Takase and T. Imaide. *Electronic image stabilizer for video camera use*. IEEE Transactions on Consumer Electronics, Vol. 36, No. 3, pp. 520-525, August 1990.

[Kok 93] Anil C. Kokaram. *Motion Picture Restoration*. PhD Thesis, Cambridge University, United Kingdom, May 1993.

[Kok 95a] Anil C. Kokaram, Robin D. Morris, William J. Fitzgerald and Peter J.W. Rayner. *Detection of Missing Data in Image Sequences*. IEEE Transactions on Image Processing, Vol. 4, No. 11, pp. 1496-1508, November 1995.

[Kok 95b] Anil C. Kokaram, Robin D. Morris, William J. Fitzgerald and Peter J.W. Rayner. *Interpolation of Missing Data in Image Sequences*. IEEE Transactions on Image Processing, Vol. 4, No. 11, pp. 1509-1519, November 1995.

[Kok 97] Anil Kokaram, Peter Rayner, Peter van Roosmalen and Jan Biemond. *Line Registration of Jittered Video*. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'97), Vol. IV, pp. 2553-2556, Munich, Germany, April 1997.

[Kok 98] Anil C. Kokaram. *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*. Springer-Verlag, London, 1998.

[Kok 02] Anil Kokaram. *Parametric Texture Synthesis for Filling Holes in Pictures*. Proceedings of the IEEE International Conference on Image Processing (ICIP 2002), Vol. I, pp. 325-328, Rochester, USA, September 2002.

[Kon 92] Janusz Konrad and Eric Dubois. *Bayesian Estimation of Motion Vector Fields*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 9, pp. 910-927, September 1992.

[Lub 95] Jeffrey Lubin. A visual discrimination model for imaging system design and evaluation. In *Vision Models for Target Detection and Recognition* (E. Peli, ed.), pp. 245-283, World Scientific Publishers, 1995.

[Luc 81] Bruce D. Lucas and Takeo Kanade. *An Iterative Image Registration Technique with an Application to Stereo Vision*. Proceedings of the DARPA Image Understanding Workshop, pp. 121-130, 1981.

[Mar 85] Jose Luis Marroquin. *Probabilistic Solution of Inverse Problems*. PhD Thesis, Massachusets Institute of Technology, USA, September 1985.

[Mas 98] Simon Masnou and Jean-Michel Morel. *Level-lines based disocclusion*. Proceedings of the IEEE International Conference on Image Processing (ICIP'98), pp. 259-263, Chicago, USA, October 1998.

[Mor 95] Robin D. Morris. *Image sequence restoration using Gibbs distributions*. PhD Thesis, Cambridge University, United Kingdom, May 1995.

[Mur 01] Kevin Murphy. *An introduction to graphical models*. Technical Report, UC Berkeley, USA, May 2001.

[Nag 86] Hans-Hellmut Nagel and Wilfried Enkelmann. *An Investigation of Smoothness Constraints for the Estimation of Displacement Vector Fields from Image Sequences*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 8, No. 5, pp. 565-593, September 1986.

[Par 88] Stephen K. Park and Keith W. Miller. *Random number generators: good ones are hard to find*. Communications of the ACM, Vol. 31, No. 10, pp. 1192-1201, October 1988.

[Per 93] Patrick Pérez. *Champs markoviens et analyse multirésolution de l'image : application à l'analyse du mouvement*. PhD Thesis, Université de Rennes I, France, July 1993.

[Per 98] Patrick Pérez. *Markov Random Fields and Images*. Technical Report PI-1196, Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), Rennes, France, July 1998.

[Pie 92] Wojciech Pieczynski. *Statistical image segmentation*. Machine Graphics and Vision, Vol. 1, No. 1/2, pp. 261-268, 1992.

[Rar 01] Andrei Rareş, Marcel J.T. Reinders and Jan Biemond. *Statistical Analysis of Pathological Motion Areas*. Proceedings of the IEE Seminar on Digital Restoration of Film and Video Archives, London, UK, January 2001.

[Rar 02] Andrei Rareş, Marcel J.T. Reinders, Jan Biemond and Reginald L. Lagendijk. *A Spatiotemporal Image Sequence Restoration Algorithm*. Proceedings of the IEEE International Conference on Image Processing (ICIP 2002), Vol. II, pp. 857-860, Rochester, USA, September 2002.

[Res 92] Sidney I. Resnick. *Adventures in Stochastic Processes*. Birkhäuser, Boston, 1992.

[Ric 95] Paul Richardson and David Suter. *Restoration of Historic Film for Digital Compression: a Case Study*. Proceedings of the IEEE Conference on Image Processing (ICIP'95), Vol. II, pp. 49-52, Washington DC, USA, October 1995.

[Rob 83] J.D. Robbins and A.N. Netravali. Recursive motion compensation: a review. In *Image Sequence Processing and Dynamic Scene Analysis* (T.S. Huang, ed.), pp. 76-103, Springer-Verlag, Berlin, 1983.

[Roo 99a] Peter M.B. van Roosmalen. *Restoration of Archived Film and Video*. PhD Thesis, Technische Universiteit Delft, The Netherlands, October 1999.

[Roo 99b] Peter M.B. van Roosmalen, Reginald L. Lagendijk and Jan Biemond. *Correction of Intensity Flicker in Old Film Sequences*. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 9, No. 7, pp. 1013-1020, October 1999.

[Rua 96] Joseph J.K. Ó Ruanaidh and William J. Fitzgerald. *Numerical Bayesian Methods Applied to Signal Processing*. Springer-Verlag, New York, 1996.

[Sap 90] Gilbert Saporta. *Probabilités, analyse des données et statistique*. Technip, Paris, 1990.

[Sid 02] Denis N. Sidorov and Anil C. Kokaram. *Suppression of moiré patterns via spectral analysis*. Proceedings of SPIE Conference on Visual Communications and Image Processing, Vol. 4671, pp. 895-906, San Jose, USA, January 2002.

[Sti 99] Christoph Stiller and Janusz Konrad. *Estimating motion in image sequences: a tutorial on modeling and computation of 2D motion*. IEEE Signal Processing Magazine, Vol. 16, No. 4, pp. 70-91, July 1999.

[Sto 85] Richard Storey. *Electronic Detection and Concealment of Film Dirt*. SMPTE Journal, pp. 642-647, June 1985.

[SW 01] Martin Weston. *Video scratch repair*. UK Patent Specification No. GB2361133, Snell & Wilcox, 10 October 2001.

[Tek 95] A. Murat Tekalp. *Digital Video Processing*. Prentice Hall, Upper Saddle River, NJ, 1995.

[Teo 94] Patrick C. Teo and David J. Heeger. *Perceptual image distortion*. Proceedings of SPIE Conference on Human Vision, Visual Processing and Digital Display, Vol. 2179, pp. 127-141, San Jose, USA, February 1994.

[Uom 90] K. Uomori, A. Morimura, H. Ishii, T. Sakaguchi and Y. Kitamura. *Automatic image stabilizing system by full-digital signal processing*. IEEE Transactions on Consumer Electronics, Vol. 36, No. 3, pp. 510-519, August 1990.

[Vla 96] Theodore Vlachos and Graham Thomas. *Motion estimation for the correction of twin-lens telecine flicker*. Proceedings of the IEEE Conference on Image Processing (ICIP'96), Vol. I, pp. 109-112, Lausanne, Switzerland, September 1996.

[Vle 02] Christophe De Vleeschouwer, Jean-François Delaigle and Benoît Macq. *Invisibility and Application Functionalities in Perceptual Watermarking – An Overview*. Proceedings of the IEEE, Vol. 90, No. 1, pp. 64-77, January 2002.

[VQEG 00] Video Quality Experts Group (VQEG). *Final report from the video quality experts group on the validation of objective models of video quality assessment*. Technical Report, March 2000.

[Wan 02] Zhou Wang, Alan C. Bovik and Ligang Lu. *Why is image quality assessment so difficult?* Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2002), Vol. IV, pp. 3313-3316, Orlando, USA, May 2002.

[Wat 94a] John Watkinson. *An introduction to Digital Video*. Focal Press, Oxford, 1994.

[Wat 94b] John Watkinson. *The Engineer's Guide to Motion Compensation*. Snell & Wilcox Handbook Series, Petersfield, 1994.

[Wei 00] Li-Yi Wei and Marc Levoy. *Fast Texture Synthesis using Tree-structured Vector Quantization*. Proceedings of SIGGRAPH 2000, pp. 479-488, New Orleans, USA, July 2000.

[Yeh 98] Edmund M. Yeh, Anil C. Kokaram and Nick G. Kingsbury. *Psychovisual measurement and distortion metrics for image sequences*. Proceedings of the European Signal Processing Conference (EUSIPCO'98), Vol. 2, pp. 1061-1064, Island of Rhodes, Greece, September 1998.

[Yu 02] Zhenghua Yu, Hong Ren Wu, Stefan Winkler and Tao Chen. *Vision-Model-Based Impairment Metric to Evaluate Blocking Artifacts in Digital Video*. Proceedings of the IEEE, Vol. 90, No. 1, pp. 154-169, January 2002.

[Zhu 98] Song Chun Zhu, Yingnian Wu and David Mumford. *Filters, Random Fields and Maximum Entropy (FRAME): Towards a Unified Theory for Texture Modeling*. International Journal of Computer Vision, Vol. 27, No. 2, pp. 107-126, 1998.

Résumé

Dans le contexte de la restauration d'archives, nous abordons dans cette thèse la suppression des défauts impulsifs (taches, "dropouts" vidéo). Les méthodes de détection et correction existantes sont limitées par les défaillances de l'estimation de mouvement dues à la présence de phénomènes naturels complexes. Nous cherchons à prendre en compte ces phénomènes que nous qualifions de mouvement pathologique. Pour les deux étapes de détection et de correction, une approche probabiliste est privilégiée et nos algorithmes sont exprimés à l'aide de champs de Markov paramétriques ou non-paramétriques.

La méthode de détection que nous proposons s'inscrit dans le cadre de la théorie bayesienne de l'estimation. Nous considérons une fenêtre temporelle plus large que les trois images utilisées habituellement afin de mieux distinguer les défauts des mouvements pathologiques et éviter ainsi les fausses alarmes. Nous proposons également une méthode de correction dans les zones d'information manquante inspirée de travaux sur la synthèse de texture. Après généralisation aux images naturelles, nous intégrons ces approches dans un contexte spatio-temporel qui permet un repli implicite sur une correction spatiale lorsque le mouvement est trop complexe. Les méthodes proposées sont validées séparément puis intégrées dans un prototype complet de suppression des défauts impulsifs.

Probabilistic Approaches for the Digital Restoration of Television Archives

Abstract

Within the context of archives restoration, we investigate in this thesis the concealment of impulsive defects (blotches, video dropouts). Existing detection and correction methods reach their limits with the presence of motion estimation failures due to complex natural events. We aim at taking into account these events that we shall call pathological motion. For both detection and correction steps, we investigate probabilistic approaches and our algorithms are expressed by means of parametric or non-parametric Markov random fields.

The proposed detection method relies on the framework of the Bayesian theory of estimation. We consider a larger temporal window than the usual three frames, in order to better distinguish defects from pathological motion and thus dramatically reduce the number of false alarms. We also propose a method to correct missing data areas which is inspired by works on texture synthesis. After generalizing these techniques to natural images, we integrate them in a spatio-temporal context which allows an implicit fallback on a spatial correction when motion is too complex. We first validate the proposed methods separately before combining them to create a complete prototype for the concealment of impulsive defects.