Archived Film Analysis and Restoration

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.dr.ir. J.T. Fokkema,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 27 September 2004 om 13:00 uur

door

Andrei RAREŞ

master, Universitatea POLITEHNICA Bucureşti
geboren te Boekarest, Roemenië

Dit proefschrift is goedgekeurd door de promotor:

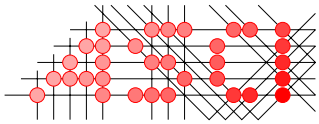
Prof.dr.ir. J. Biemond

Toegevoegd promotor:

Dr.ir. M.J.T. Reinders

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof.dr.ir. J. Biemond	Technische Universiteit Delft, promotor
Dr.ir. M.J.T. Reinders	Technische Universiteit Delft, toegevoegd promotor
Prof.dr.ir. R.L. Lagendijk	Technische Universiteit Delft
Prof.dr.ir. L.J. van Vliet	Technische Universiteit Delft
Prof.dr.ir. J.H.C. Reiber	Leids Universitair Medisch Centrum
Prof.dr.ir. A.C. Kokaram	Trinity College Dublin, Ierland
Prof.dr.ir. C.H. Slump	Universiteit Twente



This work was carried out in the ASCI graduate school.
ASCI dissertation series number 104

ISBN 90-9018588-7

Contents

Părinţilor mei

To my parents

Chapter 1

Introduction

An important part of the scientific and cultural heritage of modern times has been stored in the form of photo, film and video archives. Unfortunately, the classic storage media for these information sources are bound to gradually decay over time, risking the total loss of the valuable information they are carrying. Chemical and physical restoration techniques exist for these media [34, 14, 17, 33], but they are rather expensive, labour intensive, and have many limitations. Alternatively, one can prolong the lifetime of the content stored on such media by regularly copying it onto new film rolls or video tapes. However, the copying process is not perfect. Each copy is slightly degraded with respect to its source, thereby continuing the process of content decay.

Fortunately, with the arrival of the digital era, photographs and films can now be copied easily. The copying process takes place virtually without any information loss, after the initial digitisation of the contents. An equally important aspect of the digital era is the opportunity of applying restoration in superior ways, never possible in the past. As such, information that disappeared partly or completely (at a local scale) from its physical support can now be restored thanks to new, advanced algorithms developed in the restoration community, which take advantage of the local spatial and/or temporal redundancy of the information. Modern technologies have brought along economical benefits, too. The digitised content is cheaper and easier to store, search, reuse, and distribute. Digital broadcasting will take full advantage of these benefits. But today's quality expectations will only be fulfilled when degraded archived content is first digitally restored.

1.1 Degradation Types for Old Visual Material

From the moment they are created, every film roll or video tape undergoes a continuous alteration of its initial physical state. The degradation takes place at various speeds, depending on several factors, such as:

- *The quality of the film roll or video tape.* Each type of film or video tape from each manufacturer has its own chemical and physical characteristics, which influence the lifetime of the product.
- *The storage conditions*, in terms of room temperature, humidity, light, tightness and material composition of the container box, etc. These play an essential role in the conservation of any recorded material.
- *The handling process.* Careless manipulation takes its toll, too: touching of the material with fingers; direct exposure to sun, humid weather or electromagnetic radiation; physical stress; improper transportation etc. may damage the material.
- *The quality of playback devices* on which the material gets played. A lot of damage is caused by improper maintenance of the playback devices. Most notably, vertical film scratches, film breaks, scratches on the video tape etc. are quite often introduced during playback.
- *The number of times a roll or a tape was played.* All types of archived audio-visual material degrade each time they get played.
- *The age* of the material. Degradations tend to accelerate over time.

The original image sequence itself may also have been recorded in improper conditions. For example, dirty or improperly focused lenses, an unsteady camera, worn or dirty recording heads (in case of video cameras) introduce artefacts right from the start. If the original is damaged, the copies will be at least as damaged, unless restoration procedures are employed during the transfer process.

All the aforementioned factors affect the audio-visual material in various ways, resulting in many types of artefacts. Currently, more than 170 audio-visual artefact types have been identified [22, 23]. Among the most frequently encountered ones are those presented in Table 1.1.

With the arrival of digital media, new types of artefacts came into existence. For example, errors in the storage or broadcast medium of digitally compressed image sequences may result in block losses. The appearance of digital media artefacts is specific to the combination of storage medium and encoding format used. These artefacts and their restoration are not explicitly discussed in this thesis, although some of the algorithms presented here can be applied to them, too.

Artefact Name	**Description**
Intensity flicker	Change of overall intensity from one frame to another.
Film unsteadiness	Each frame is displaced with a different offset compared to the neighbouring frames.
Colour fading	Change or total loss of one or more colour components.
Vertical scratches	Bright or dark vertical lines across the image, usually in consecutive frames and at the same location
Blotches (Dirt)	Bright or dark compact, non-repetitive spots in the image. The loss of visual information inside the blotches is usually total.
Vinegar syndrome	Chemical breakdown of cellulose acetate film base, with release of acetic acid. Both film base and emulsion layers get damaged.
Emulsion melting	Melting of the light-sensitive layers of the film, causing irregular spread of the frame contents.
Noise	A (usually) random modification of the pixel intensities in each frame.
Two Inch Scratches	A grid of (semi-)regular dots or small lines due to a scratch on a video tape
Dropouts	Loss of several consecutive lines in a video sequence. The last valid line gets repeated instead of the missing lines, giving rise to a pattern of vertical lines.
Jitter	Loss of line synchronization resulting in horizontal shifts of the start- and endpoints of each line.
Moiré	A spatial aliasing phenomenon occuring when film is transferred to video tape using a telecine device.

Table 1.1: A representative set of artefact types.

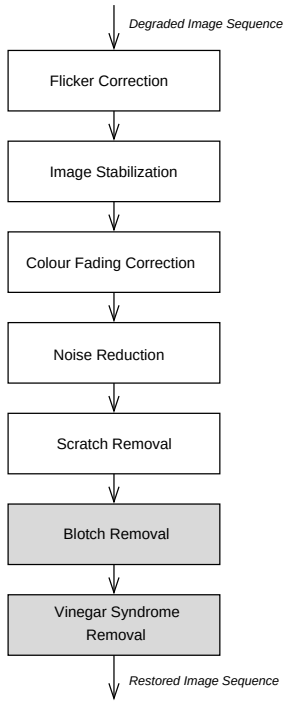


Figure 1.1: A film restoration system. Gray boxes indicate subjects approached in this thesis.

It is possible for artefacts specific to different storage media to be present in the same image sequence. When copying corrupt sequences from one medium to another (e.g. from film to video tapes), the degradations from each medium may combine with each other, giving rise to supplementary problems for the automatic detection and correction algorithms. In this case it is advised to first correct the artefacts specific to the current medium, and then those specific to the previous one.

1.2 Artefact Detection and Restoration

Artefacts have many appearances. For this reason, it is much easier to devise algorithms which know how to detect and restore only a certain type of artefact, rather than a single algorithm which applies to all artefacts. Therefore, one needs to develop an algorithm for every artefact, or at least for every group of artefacts with similar appearance.

Visual artefacts can be categorised according to several criteria, based on their appearances. As such, these artefacts may be only *spatial* (blotches etc.), only *temporal*

(unsteadiness), or *spatiotemporal* (vertical scratches). They may affect only *a local area* (blotches), or *the entire frame* (noise). In terms of spatial regularity, they may be *periodic* (interlacing), *semi-periodic* (moiré), or *non-periodic* (blotches). In terms of temporal regularity, some artefacts may be *periodic* over time (some forms of unsteadiness), may have a *semi-periodic* behaviour (flicker), or may be *non-periodic* (blotches, dirty splices). The information loss may be *total* (blotches, missing frames) or *partial* (colour fading and certain sub-types of vinegar syndrome and noise). They may be shaped as *compact areas* (blotches), as *straight lines* (scratches), or as *branched structures* (a vinegar syndrome sub-type). Finally, the artefact may affect the *pixel values* (blotches, noise, scratches, flicker, moiré, dropouts, most sub-types of vinegar syndrome), the *pixel location* (unsteadiness, jitter, cushion/barrel distortions), or both *pixel values and location* (emulsion melting).

The various artefact appearances translate into a diversity of artefact detection and restoration techniques. The restoration scheme presented in Fig. 1.1 was built to handle the most important artefact types. It is especially tailored for the restoration of films, since the algorithms presented in this thesis concentrate mainly on film-related problems.

The order in which the artefacts are removed is important, because the restoration of a certain type of artefact may affect the detection or the restoration of another type of artefact. For example, by restoring blotches which overlap with a vertical scratch, one may ruin the scratch detection process (and implicitly its restoration). The restoration scheme presented in Fig. 1.1 reflects such an order. In the following, a short account is given for the detection and restoration approaches corresponding to each artefact present in the scheme.

Intensity flicker may be the result of uneven chemical degradation of consecutive frames, or uneven exposure times. It is usually detected as an oscillation in the mean and variance of the frame intensity, either globally for the entire frame, or locally in a region of the frame [36, 31, 15].

Flicker correction involves filtering out the intensity variations from frame to frame. While global flicker is somewhat easier to deal with, local flicker needs more complex models for its correction, since it has to take into account occlusions, scene entrances, or other legitimate intensity changes [36, 31, 15].

Film unsteadiness represents a continuous shaking of the image (usually in the vertical direction), caused by the deterioration of the film perforations, improper film transportation mechanisms, or unstable cameras (e.g. in case of hand-held cameras). It can be estimated based on the variation of the global motion in consecutive frames [15].

Unsteadiness can be corrected by compensating the frame offset with the estimated global motion vector [15]. The focus here lies on assessing the global scene movement (e.g. the camera pan), while avoiding the influence of the local motion of objects.

Colour fading is caused by the attenuation over time or even the complete disappearance of information in one or more colour layers of the film. It can be identified by studying the global properties of each colour layer [38].

Fading can be corrected by boosting the information from the faded colour layers [38]. The nonlinearity of the fading phenomenon in each colour layer must be taken into account.

Vertical scratches are usually caused by dirt and emulsion particles which gather on the components of the projection devices and scratch the films that get rolled over them. They can be detected by studying the shape of the horizontal lines in a frame and by trying to identify the typical cross-section of a scratch [26, 32, 15, 24]. The Hough Transform for detecting straight lines can also be used for scratch detection [26].

Scratches are relatively stationary in consecutive frames. The temporal information is therefore not of much use for the restoration process. This is the reason why most scratch removal algorithms make use only of spatial information [26, 32, 24, 15]. The information to the left and right of the scratch gets propagated inside the scratch.

Blotches are bright or dark spots which are usually caused by the presence of dirt on the film. Together with the noise and scratches, blotches are among the most encountered artefacts, which is why a large part of the research efforts in the restoration field is dedicated to their removal. Blotches are usually detected as impulsive events in the temporal direction, i.e., a sudden change of intensity lasting only one frame [26, 31]. Blotches can also be detected by studying the motion vectors in a local neighbourhood, rather than raw pixel values [26, 36, 31]. If there is complicated object motion around an artefact, then none of the aforementioned methods can be used. In this case, one has to rely only on the extreme brightness or darkness of the blotch with respect to the neighbourhood.

Blotches can be successfully restored by means of temporal or spatiotemporal algorithms, such as motion-compensated median filtering, 3D autoregressive models, Markov Random Field-based methods etc. [26, 31, 4]. When temporal information is unavailable or unreliable, spatial restorations can be applied by means of inpainting schemes [3, 5, 6, 7, 11, 12, 13], texture synthesis [16, 9, 10, 1, 27, 28, 29], projection onto convex sets methods [19], or other interpolation techniques.

The *vinegar syndrome* is caused by a chemical degradation of the film base, which releases acetic acid, thereby producing a vinegar smell [32, 17, 14]. It is one of the artefacts that has more than one appearance. It may show up as bright or dark tree-like branches, a partial loss of colour, non-uniformly blurred images, etc. For such types of artefacts with multiple appearances, several algorithms may be required to detect and restore every sub-type of the artefact separately. For example, the sub-type of *vinegar syndrome* which manifests itself as a local loss of some colour layers can be detected by studying each of these layers separately and finding intensity spikes in the affected layers.

Currently, the treatment of films affected by vinegar syndrome is usually done by chemical means, and it is mostly directed towards their preservation, rather than restoration [17, 34]. In the digital domain, few vinegar syndrome restoration algorithms exist. These are specific to the sub-type of the artefact at hand. As such, the vinegar syndrome sub-types which imply a local loss of information could be restored by the aforementioned blotch removal algorithms. The subtype which shows up as a local, partial loss of colour can also take into account the remaining information inside the artefact. The subtype which shows up as a non-uniform blurring of the image can be restored by means of blur removal algorithms [18, 2, 8].

Noise can be caused by film grain, dust, or charge fluctuations in the CCD elements of a video camera. Virtually all signals measured or recorded with any device contain some degree of noise. The estimation of noise is a well-studied subject in the signal processing community [2, 25, 26, 36, 8, 30, 31]. In general the noise is modelled according to its expected statistical properties in terms of intensity variance, Fourier spectrum, gabor wavelet responses and so on.

Some noise removal algorithms work in the spatiotemporal domain (be it motion-compensated or not), while others take into account only the information in a single frame. The latter group comprises algorithms working in the Fourier domain, the wavelet domain, or the spatial domain. Kalman, Wiener and wavelet-based filters are among the most popular approaches for noise reduction. More detailed overviews can be found in [30, 25, 26, 8, 36, 31].

It is by now obvious that there are many different ways to detect and correct artefacts. The restoration process is sometimes independent from the detection process, but these processes may be related to each other as well. Some algorithms even merge the two processes into a single joint detection and correction algorithm (e.g. the JOMBADI algorithm of Kokaram [26]).

The term “restoration” itself is sometimes used in a broader sense to describe the ensemble of modules which contribute to the final restored result. Accordingly, in Fig. 1.1, each restoration block also comprises an artefact detection step, which is not represented for simplicity reasons.

Artefacts are not always detected in a stand-alone fashion. Instead, they may be incorporated in a general image model which describes both healthy and damaged content, e.g. in a Bayesian fashion [26]. Then, both kinds of content are separated by means of complementary Boolean flags.

1.3 Thesis Background

The algorithms described in this thesis were developed at the Information and Communication Theory Group, Mediamatics Department, Faculty of Electrical Engineering, Mathematics and Computer Science of the Delft University of Technology.

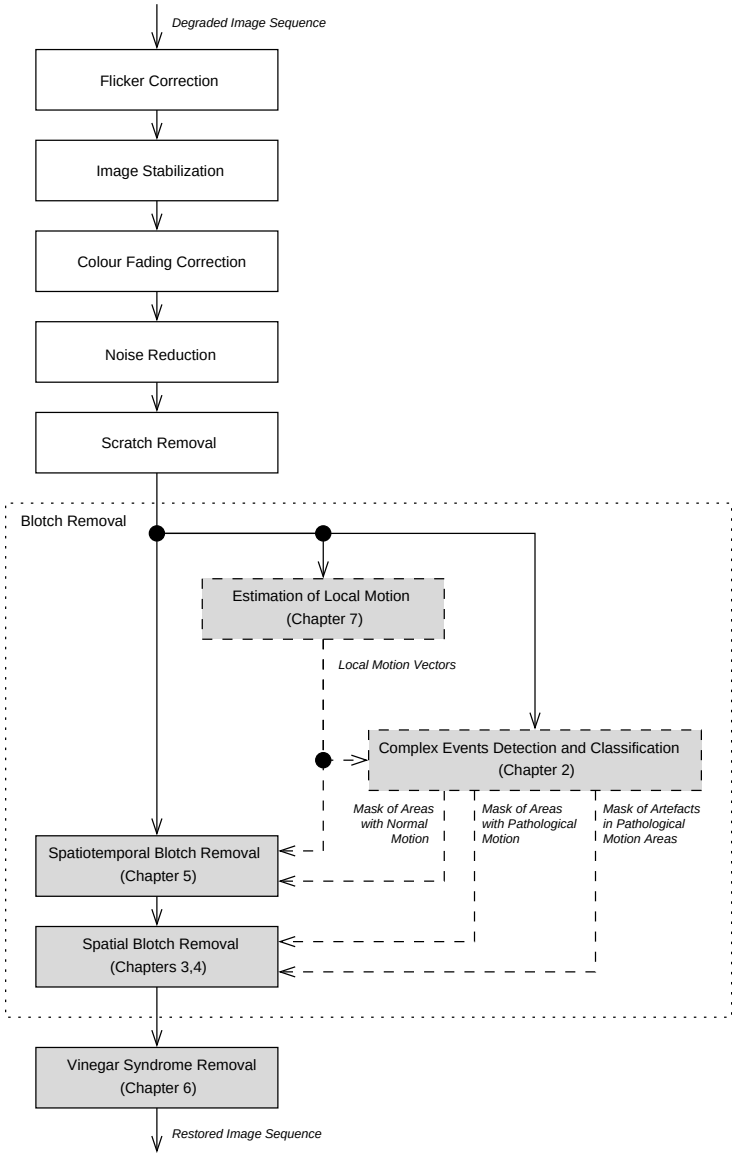


Figure 1.2: The same restoration system, with a detailed view on the "Blotch Removal" block. Gray boxes indicate subjects approached in this thesis. Continuous-line boxes represent modules directly involved in restoration, with continuous lines indicating the path of the original image or the restored image. Dashed boxes represent auxiliary processes required by the restoration modules, and dashed lines represent their outputs.

Most of this research work was carried out within the BRAVA project ("BRoadcast Archives restoration through Video Analysis") of the European Union, between 2000 and 2002. The BRAVA project [21] was part of the EU's IST Research and Technological Development Programme, and was dedicated to the analysis and restoration of degraded audio-visual material from old films and video. The BRAVA project was the successor of another EU project, AURORA ("AUtomated Restoration of ORiginal video and film Archives"), which took place between 1995 and 1998 [20].

The task assigned to TU Delft as a partner in the BRAVA project was to study the possibility of restoring those areas of the film frames where the motion of objects is difficult to estimate. This is reflected by the contents of this thesis. Virtually every chapter is devoted to an algorithm which can be used for restoration under difficult motion circumstances, or to a process which underlies such restoration.

1.4 Scope

A detailed view of the restoration system from Fig. 1.1 is presented in Fig. 1.2. Here, the topics addressed in this thesis are indicated as grey boxes. As the figure shows, most of the work presented in this thesis is related to blotch removal. In particular, the problem of removing blotches under difficult motion circumstances is studied. These circumstances are caused by the limitations of the motion models at hand, or by severe artefacts. They are generically called "complex events". Currently, in the presence of complex events, the usual approach is to protect the affected image areas from any restoration at all, since the restored frames would look even worse than those before the restoration [35].

In this thesis, an attempt is made to restore blotches in complex event areas. The proposed approaches use only spatial information, because of the unreliability of the temporal information. In addition to these spatial restoration algorithms, a spatiotemporal algorithm is proposed for complex events caused by the presence of severe artefacts. In order to assess which parts of the complex event areas represent artefacts, and which ones represent objects with difficult motion, an analysis scheme was devised.

The problems posed by the complex event areas can also be approached from another direction. An appropriate motion model may be able to diminish the amount of complex event areas. In this thesis, an object tracking method is proposed that can be used to estimate motion.

A growing threat to the acetate-based films stored in the present archives is represented by the vinegar syndrome. This is an artefact with multiple appearances, and algorithms for its removal are scarce [18]. In this thesis, a method is especially developed for one of the appearances that was never tackled before. It was developed and tested on degraded samples from the archives of our project partners. Together

with most of the other methods presented in this thesis, the method addresses a part of the current needs in the restoration domain, and can readily be used in real-world applications.

Most algorithms proposed in this thesis are equally applicable to colour and black-and-white images. The algorithm for the analysis of complex events (Chapter 2) is an exception, because it makes use of the information from all layers of a colour film. Another exception is represented by the spatial algorithm dedicated to the restoration of a special type of artefact (*vinegar syndrome* combined with *emulsion melting*) in colour films only (see Chapter 6).

Besides visual content, films usually hold audio information, too. Like the visual quality, the audio quality of old movies degrades over time, resulting in a number of specific artefacts [22]. The subject of audio restoration is, however, not covered in this thesis, which is focused only on aspects related to visual information.

1.5 Thesis Outline

This thesis consists of six chapters covering various subjects related to the restoration of visual material. Each of these chapters represents a paper that has been published in (or has been submitted for publication to) an international journal or conference proceedings. Therefore, the chapters are self-explanatory texts which can be read independently. For this reason a small overlap between some chapters was allowed, as it does not affect the essence of their contents. The chapters were arranged in an order which allows the gradual introduction of new concepts and notations, making for easier reading.

Chapter 2 is devoted to the analysis of difficult motion in image sequences (the "Complex Events Detection and Classification" block in Fig. 1.2). Motion vectors are used for determining object trajectories in consecutive frames, thereby indicating the paths along which localised visual information has relatively constant values. The analysis of motion vectors can also indicate the presence of certain types of artefacts (e.g. blotches), since these represent impulsive temporal events. Current motion estimation methods have their own limitations, which are due to too simple motion model assumptions. Object displacement other than linear motion, motion speeds above certain limits, occlusions, or the presence of artefacts tend to result in wrong motion vectors, which consequently affect the quality of the temporal restoration (see Appendix A for a taxonomy of difficult motion). Moreover, false artefact flagging may occur as a result of difficult (usually impulsive) motion. The result of the restoration of these false artefacts looks worse than the original content, due to the limitations of the current restoration methods and because the wrong motion vectors are extracted. In this chapter a look is taken at one of the types of difficult motion, namely blurred motion caused by fast moving objects. Areas in the image with complex events (difficult motion, or artefacts) are first identified and subsequently

analysed, based on the segmented image. This analysis identifies blurred objects, as well as artefacts which might have caused the failure of the motion estimation.

Because of the unreliability of the extracted motion vectors, it is almost impossible to restore the artefacts lying in areas of difficult motion with the current temporal restoration algorithms. In Chapter 3, a spatial restoration method is developed based on object shape reconstruction and smooth inpainting. Only information from the current frame is employed. The algorithm uses edge shapes, object colours and their spatial order in the image in order to estimate the original edges: those that existed before the degradation took place. The edges are reconstructed by linking pairs of edges with similar features. The reconstructed edges are to be used as "barriers" in the subsequent inpainting operation, which smoothly propagates object colour information into the artefact from the surrounding areas.

The results of the directed smooth inpainting used in Chapter 3 when applied to certain textured areas are not as good as in the case of flat areas. Therefore, in chapter 4 a method is developed for texture restoration constrained by image structure. The features used here for estimating edge connections include texture statistics, and the inpainting operation is replaced by a texture synthesis algorithm.

In Chapter 5, an extension to the temporal domain is introduced for the edge-based approach presented in Chapter 3. The motion is estimated based on the spatially restored frames. The extracted motion vectors are then used in a temporal restoration setup.

Chapter 6 presents a particular type of artefact which affects acetate-based films, together with an algorithm for removing it (in colour films). This type of artefact, vinegar syndrome in combination with emulsion melting, shows up sometimes as colour loss in two of the three colour layers of the film. The algorithm proposed here makes use of the information which still exists inside the artefact in the third colour layer to recover the image structure, and of the information in the spatial surroundings of the artefact to recover the colour information.

In chapter 7, a method is introduced for improving the process of motion estimation by means of object tracking. The objects are statistically represented by N-dimensional mixtures of Gaussians centered around feature vectors consisting of colour, position, and local responses to a set of gabor wavelets. These vectors undergo an update with each new frame through the online expectation-maximisation (EM) algorithm. This general approach to motion estimation at object level is able to cope with incomplete data and thus allows for a flexible representation and tracking of objects.

The final chapter gives general conclusions regarding the algorithms proposed in this thesis, proposals for their potential extensions, and discusses the perspectives of the restoration field.

Bibliography

[1] S.T. Acton, D.P. Mukherjee, J.P. Havlicek, A.C. Bovik, "Oriented texture completion by AM-FM reaction-diffusion", IEEE Transactions on Image Processing, vol. 10, no. 6, pp. 885-896, June 2001.

[2] H.C. Andrews, B.R. Hunt, "Digital Image Restoration", Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1977.

[3] L. Atzori, F.G.B. De Natale, "Error concealment in video transmission over packet networks by a sketch-based approach", Signal Processing: Image Communication, vol. 15, no. 1-2, pp. 57-76, September 1999.

[4] L. Atzori, F.G.B. De Natale, C. Perra, "A spatio-temporal concealment technique using boundary matching algorithm and mesh-based warping (BMA-MBW)", IEEE Transactions on Multimedia, vol. 3, no. 3, pp. 326-338, September 2001.

[5] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Transactions on Image Processing, vol. 10, no. 8, pp. 1200-1211, August 2001.

[6] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, G. Sapiro, "A variational model for filling-in gray and color images", Proceedings of ICCV 2001, vol. 1, pp. 10-16, Vancouver, BC, Canada, July 9-12, 2001.

[7] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image inpainting", Proceedings of ACM SIGGRAPH 2000, pp. 417-424, New Orleans, La, USA, July 2000.

[8] J. Biemond, "Image Restoration - A Linear Stochastic Filtering Approach", PhD Thesis, Delft University of Technology, The Netherlands, 1982.

[9] R. Bornard, E. Lecan, L. Laborelli, J.-H. Chenot, "Missing data correction in still images and image sequences", Proceedings of ACM Multimedia 2002, pp. 355-361, Juan Les Pins, France, December 1-6, 2002.

[10] R. Bornard, "Probabilistic Approaches for the Digital Restoration of Television Archives", PhD Thesis, École Centrale Paris, France, 2002.

[11] T. Chan, J. Shen, "Mathematical models for local non-texture inpainting", SIAM Journal on Applied Mathematics, vol. 62, no. 3, pp. 1019-1043, 2001.

[12] T. Chan, J. Shen, "Non-texture inpainting by curvature-driven diffusions (CDD)", Journal of Visual Communication and Image Representation, vol. 12, no. 4, pp. 436-449, 2001.

[13] T. Chan, S. H. Kang, J. Shen, "Euler's elastica and curvature based inpainting", SIAM Journal on Applied Mathematics, vol. 63, no. 2, pp. 564-592, 2002.

[14] L. Comencini, M. Pavesi, editors, "Restauro, Conservazione e Distruzione dei Film", Il Castoro, 2001, ISBN 88-8033-204-X.

[15] E. Decencière Ferrandière, "Restauration Automatique de Films Anciens", PhD thesis, École Nationale Supérieure des Mines de Paris, France, 1997.

[16] A.A. Efros, T.K. Leung, "Texture synthesis by non-parametric sampling", Proceedings of ICCV 1999, vol. 2, pp. 1033-1038, Kerkyra, Corfu, Greece, September 1999.

[17] Gamma Group, "The Vinegar Syndrome. Prevention, remedies and the use of new technologies. An handbook", 2000.

[18] F. Helt, V. La Torre, "Advances in digital restoration for addressing the vinegar syndrome effects", Proceedings of the 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 4/1-4/7, London, UK, January 16, 2001.

[19] A.N. Hirani, T. Totsuka, "Combining frequency and spatial domain information for fast interactive image noise removal", Proceedings of ACM SIGGRAPH 1996, pp. 269–276, New Orleans, La, USA, August 1996.

[20] http://www.ina.fr/recherche/projets/finis/aurora/index.en.html

[21] http://brava.ina.fr/

[22] http://brava.ina.fr/brava_public_impairments_list.en.html

[23] http://diamant.joanneum.ac.at/film_restoration/index.html

[24] L. Joyeux, "Reconstruction de séquences d'images haute résolution. Application a la restauration de films cinématographiques", PhD Thesis, La Rochelle University, France, 2000.

[25] A.K. Katsaggelos, editor, "Digital Image Restoration", Springer-Verlag, 1991, ISBN 3-540-53292-7 // ISBN 0-387-53292-7.

[26] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[27] A.C. Kokaram, "Parametric texture synthesis for filling holes in pictures", Proceedings of IEEE ICIP 2002, vol. 1, pp. 325-328, Rochester, NY, USA, September 22-25, 2002.

[28] A.C. Kokaram, "Practical MCMC for missing data treatment in degraded video", Proceedings of ECCV 2002 - Workshop on Statistical Methods in Video Processing, pp. 85-90, Copenhagen, Denmark, June 2002.

[29] A.C. Kokaram, S.J. Godsill, "MCMC for joint noise reduction and missing data treatment in degraded video", IEEE Transactions on Signal Processing, Special Issue on MCMC Methods, vol. 50, no. 2, pp. 189-205, February 2002.

[30] R.L. Lagendijk, J. Biemond, "Iterative Identification and Restoration of Images", Kluwer Academic Publishers, 1991.

[31] R.L. Lagendijk, P.M.B. van Roosmalen, J. Biemond, A. Rareş, M.J.T. Reinders, "Video enhancement and restoration", in Al. Bovik, editor, "The Handbook of Image and Video Processing", pp. 275-295, Elsevier Academic Press, 2005, ISBN 0-12-119792-1.

[32] R. D. Morris, "Image Sequence Restoration Using Gibbs Distributions", PhD Thesis, University of Cambridge, UK, 1995.

[33] G. Radu, R. Radu, "Conservarea şi regenerarea dia-foto-filmelor", Editura Tehnică, Bucureşti, 1981, Colecţia Foto-Film.

[34] P. Read, M.-P. Meyer, "Restoration of Motion Picture Film", Butterworth-Heinemann, 2000, ISBN 0-7506-2793-X.

[35] P.M.B. van Roosmalen, "High-level analysis of image sequences", technical report for INA - Paris (Institut National de l'Audiovisuel), France, the EU Aurora Project, July 1999.

[36] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

[37] R.J. Ross, "Color Film for Color Television", Focal Press Limited, 1970, ISBN 0 240 50714 2.

[38] L. Rosenthaler, R. Gschwind, "Restoration of movie films by digital image processing", Proceedings of the 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 6/1-6/7, London, UK, January 16, 2001.

Chapter 2

Complex Event Classification in Degraded Image Sequences

This chapter has been published as "Complex Event Classification in Degraded Image Sequences", by A. Rareş, M.J.T. Reinders and J. Biemond in the Proceedings of the *2001 IEEE International Conference on Image Processing*, vol. 1, pp. 253-256, Thessaloniki, Greece, October 7-10, 2001.

Abstract

In this paper we address the problem of motion estimation failure in degraded image sequences. This failure is caused by some complex events that take place in the image. As a result, the sequence of operations that rely on the motion estimation process, such as motion compensation and motion picture restoration, will fail as well. The statistical analysis of the complex event areas indicates that it is possible to discriminate between the complex events resulted from complicated object motion and the ones resulted from image artefacts. An analysis scheme based on segment matching is proposed for the task of classifying the detected complex event areas.

2.1 Introduction

Recent developments in motion picture restoration show great improvements in noise reduction, flicker correction, blotch detection and removal etc. [1, 6]. However, there are situations where current techniques are still failing. In some of these cases, the quality of parts of the restored sequence is even reduced in the restoration process. For instance, in sequences where objects or persons perform complex motion - called here "pathological motion". The pathological motion represents motion that cannot be easily modelled by current motion estimation techniques. It can be observed in areas with fast or irregular motion, occlusions, scene entrances and/or other specific circumstances[1] [4]. When pathological motion occurs, the errors of the motion estimation process propagate into the motion compensation and image restoration processes. Here, restoration refers to the process of repairing the degraded areas of the image, rather than, for example, removing the motion blur (the latter can be even helpful by enhancing the impression of fast movement).

The usual procedure in case of pathological motion is to prevent the affected areas from being restored [5]. However, a more sophisticated method could be pursued, in order to try to restore these areas as well. To improve the detection scheme for pathological motion in degraded image sequences, a thorough analysis of this particular type of motion is essential. Based on this analysis, one can devise novel object tracking techniques [3] that are able to deal with such complex motions. These techniques will allow the subsequent restoration process to significantly improve its performance. In this paper, we address the problem of complex event classification. We propose an analysis scheme for identifying both pathological motion and artefacts. Section 2.2 presents statistical properties of the pathological motion and artefact areas. In Section 2.3 the analysis scheme is outlined in detail. Section 2.4 discusses the performance of our scheme. Finally, we draw some conclusions and give indications for future developments.

[1] see Appendix A for a detailed list of pathological motion types.

2.2 Statistical Properties of the Pathological Motion Areas

Estimating complex motion is still an open research subject [2, 7, 9]. For deteriorated image sequences, the problem becomes even harder [1, 6, 5], since motion estimation could then fail for two reasons: the pathological motion itself and/or the artefacts that affect the image content. Therefore, for restoration purposes, we propose that, in case of motion estimation failure, the underlying phenomenon to be called a *complex event* (CE). CE thus either refers to *pathological motion* (PM), or *artefacts*. It is necessary to discriminate between the two categories, so that the optimal restoration method can be chosen.

In a recent paper [4] we presented a taxonomy of pathological motion. For the work described here, we concentrated only on one type of pathological motion, namely the motion blur. The statistical properties of motion blur and artefacts that are presented in the following subsections show that one can distinguish them from each other on the basis of histogram examination.

2.2.1 Motion blur

This class of pathological motion happens when an object moves so fast that it looks smeared over the background. In [4] we have performed a series of tests on areas with such motion blur, in order to quantify its characteristics. The tests consisted of inspecting the histograms in the RGB colour space of the objects involved in the motion blur. We have found that in each of the R, G and B channels, the average colour of the blurred object falls always between the average colours of the clear object and background (inspected in a nearby frame).

Further experiments [4] have shown that the average colour of the blurred object does not always fall exactly in the middle of the distance between the clear object and background. Depending on the degree of motion blur, the statistics are biased towards either the foreground object, or the background. The amount of blur, however, is the same for all channels, R, G, and B. Hence, the following ratios between the average colours on the three channels have been found:

$$\frac{\mu_c^r - \mu_{p1}^r}{\mu_{p2}^r - \mu_{p1}^r} \cong \frac{\mu_c^g - \mu_{p1}^g}{\mu_{p2}^g - \mu_{p1}^g} \cong \frac{\mu_c^b - \mu_{p1}^b}{\mu_{p2}^b - \mu_{p1}^b} \tag{2.1}$$

where $\{\mu_c^r, \mu_c^g, \mu_c^b\}$ represents the average colour values of the blurred object in the current frame, and $\{\mu_{pi}^r, \mu_{pi}^g, \mu_{pi}^b\}$, $i = \{1, 2\}$, represent the average colour values of the clear object (i=1) and background (i=2) in a nearby frame. These findings are in line with the existing work on motion blur [8], which is modelled as the application of a point spread function over an image.

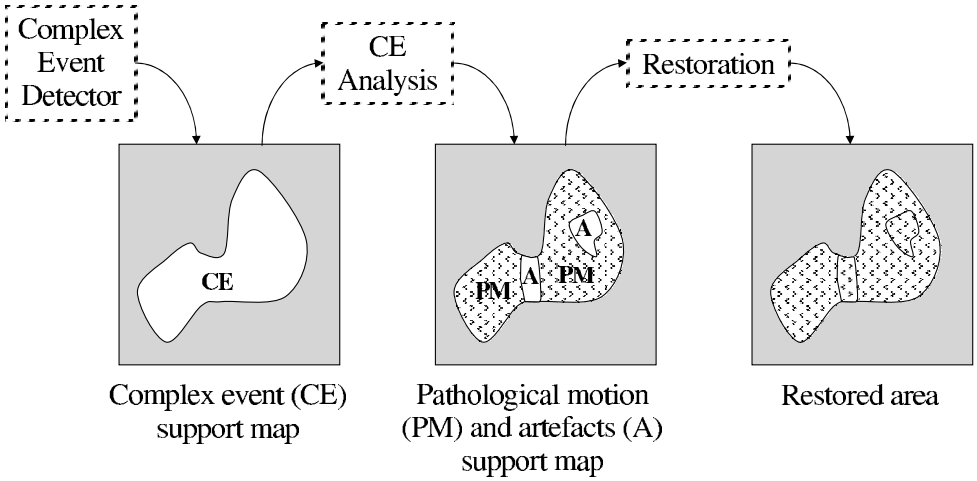


Figure 2.1: General processing flow for the complete restoration chain.

2.2.2 Artefacts

In this paper we consider only artefacts like scratches or blotches. They are usually dark or bright (semi-transparent blotches are rare), so their appearance is placed at the extremes of the colour scale. Our experiments have confirmed this behaviour [4]. Moreover, we have found that for certain artefacts the colour layers of a degraded film may be affected differently, so the artefact "spikes" found at the extremities of the histograms of an object may show up in a subset of the R, G, and B channels only.

2.3 An Analysis Scheme for Complex Event Areas

In order to apply restoration techniques in CE areas, we must separate PM from artefacts so that the most optimal restoration technique can be chosen for each area. We hereby assume that PM and artefacts are being composed of regions with relatively homogeneous colour. Fig. 2.1 displays the general processing flow for the complete restoration chain.

First, CE areas are detected using a CE detector. Then, the analysis scheme separates them into PM and artefact areas. Finally, the image is restored by applying the optimal restoration method for each of the different PM and artefact areas in the image. Next, the first two steps are elaborated. The last one is left out of the scope of this paper.

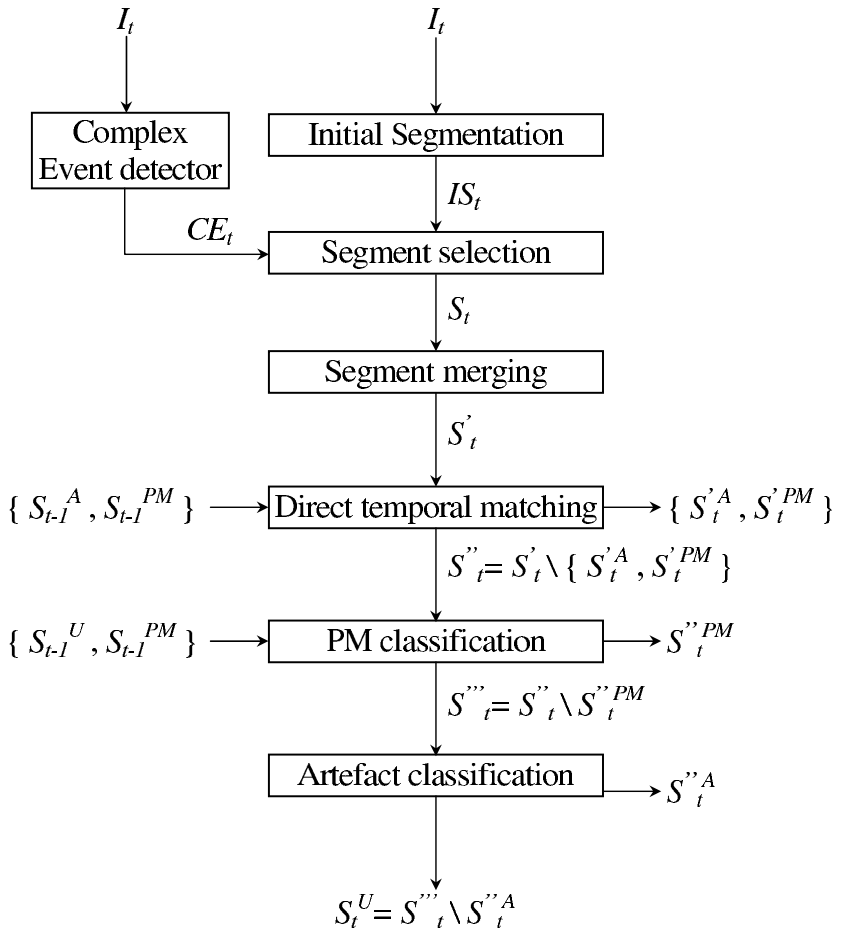


Figure 2.2: The scheme for separating pathological motion segments ($S_t^{PM} = S_t'^{PM} \cup S_t''^{PM}$) from artefact segments ($S_t^A = S_t'^A \cup S_t''^A$). Segments that cannot be labelled as pathological motion, nor as artefacts, are labelled as "unclassified" (S_t^U). CE_t is the set of complex event areas, IS_t is the set of segments provided by the initial segmentation, and S_t, S_t', S_t'', and S_t''' are intermediate sets of segments in the analysis scheme.

2.3.1 The Complex Event Detector

We employ a CE detector [5] that makes use of a large temporal aperture, in order to detect complex events that last for more than one frame (which is the usual behaviour of the pathological motion, as opposed to blotches which usually last for a single frame). Additionally, this method makes use of a sequence of steps including: mean filters, phase correlation, minimum absolute frame difference and other low-level operators.

2.3.2 Analysis of Complex Event Areas

The CE analysis scheme is displayed in Fig. 2.2. It first breaks up the CE areas into segments that exhibit homogeneous colour properties. To avoid overlapping object borders, our segments are constrained to only belong to one object by means of a relative over-segmentation. The CE analysis scheme then proceeds by classifying each segment into motion blur PM or artefact. This is first approached by looking in the previous frame for similar segments (*Direct temporal matching*). If a similar segment is found, then the current segment gets the same label as its counterpart. This ensures temporal consistency of the outcome of the analysis scheme, and speeds up the classification process, as is shown in Section 2.4. If no similar segment is found in the previous frame, then the segment is classified based on the analysis of its colour (*PM and artefact classification*). Next, we elaborate on each of these steps of the analysis scheme.

Initial Segmentation

Segmenting the image consists basically of a sequence of smoothing filters for removing the noise, and a watershed segmentation. A relative over-segmentation is preferred to an under-segmentation, since we do not want adjacent distinct objects (or parts of objects) to be considered as one single entity. That is to say, we try to preserve edges, or else the colour statistics of our segments become meaningless.

Segment Selection

Only segments within CE areas get actually processed. Thus, the actual amount of segments that we must take into account is reduced considerably. Selecting segments within CE areas is, however, not a trivial task, since the CE detector, on one hand, and the segmentation procedure used in our scheme, on the other hand, are created based on different criteria. The first one uses phase correlation, while the second one is based on watershed segmentation. Consequently, the boundaries of the CE masks are in general different from the boundaries of the watershed segments. We chose to select the segments that have at least T_s percent of their area inside the CE masks.

A value of 80% for T_s proved to be a good threshold for all experiments. The set of selected segments, S_t, is thus defined as:

$$S_t = \left\{ s_i \,\middle|\, s_i \in IS_t,\ \frac{A(s_i \cap CE_t)}{A(s_i)} \geq T_S \right\} \tag{2.2}$$

where s_i's represent the segments, IS_t the initial set of segments, $A(s_i)$ the area of segment s_i, and T_s the selection threshold of the relative segment area.

Segment Merging

The resulting set of segments is even further reduced by merging neighbouring sets of segments with similar average colours.

First, a crude but fast merging is employed. Here, all dissimilarities between segments and their neighbours are calculated. The dissimilarity between two segments is given by the mean of the sum of squared differences between the R, G and B values of the two segments. The segments that are neighbours and have a dissimilarity below a certain threshold T_M, are selected in a preliminary list. The crude merging procedure starts by making a set consisting of one segment from the preliminary list. The set is iteratively enlarged by adding all segments from the same list which are neighbours to any of the set's segments, until no more segments can be added. The same set construction procedure is applied to the remaining segments of the preliminary list until the latter gets empty. At this point, all segments of each resulted set are merged into single segments.

After the first merge, another merging procedure is employed, which iteratively merges the two segments that have the lowest dissimilarity. An additional difference to the previous step is that the dissimilarities from all segments to the new segment are updated before going to the next iteration. This procedure is applied until there are no more dissimilarities below $2 * T_M$. This merging procedure is slower than the first one, but more reliable (for which reason we restrict the first merging with a threshold that is lower than in the second one). The resulted set of segments is forwarded to the classification steps that follow.

Direct Temporal Matching

This phase is triggered if we have already labelled segments of CE areas in the previous frame. Here, we look for temporal consistency between frames at the segment level. Each segment belonging to a CE area is matched with segments of the past CE areas. The unmatched CE segments are forwarded to the next phases.

If we note the average colour of the current and previous segments with $\{\mu_c^r, \mu_c^g, \mu_c^b\}$ and $\{\mu_p^r, \mu_p^g, \mu_p^b\}$, respectively, then a match of the two is flagged if

$$\frac{1}{3}\sum_{i=\{r,g,b\}}\left|\mu_c^i-\mu_p^i\right|<T_{DTM} \tag{2.3}$$

where T_{DTM} is the threshold for the matching process (a value between 0 and 1), and the R, G, and B channels have been normalised to range $[0\ldots1]$.

PM Classification

In the PM classification process, we actually look for motion blur PM, in which case an object is smeared over some background. Thus, for the unmatched CE segments that are handed to this phase, we try to find combinations of segments from the CE areas of the previous frame which fulfil the condition of ratio proportionality given in Eq. 2.1. In practice, due to sampling/roundoff errors, the ratios are sometimes slightly different. Therefore, a small variance is allowed:

$$\sigma\left(\frac{\mu_c^r-\mu_{p1}^r}{\mu_{p2}^r-\mu_{p1}^r},\frac{\mu_c^g-\mu_{p1}^g}{\mu_{p2}^g-\mu_{p1}^g},\frac{\mu_c^b-\mu_{p1}^b}{\mu_{p2}^b-\mu_{p1}^b}\right)<T_{PM} \tag{2.4}$$

where σ represents the standard deviation, and T_{PM} a small threshold.

Artefact Classification

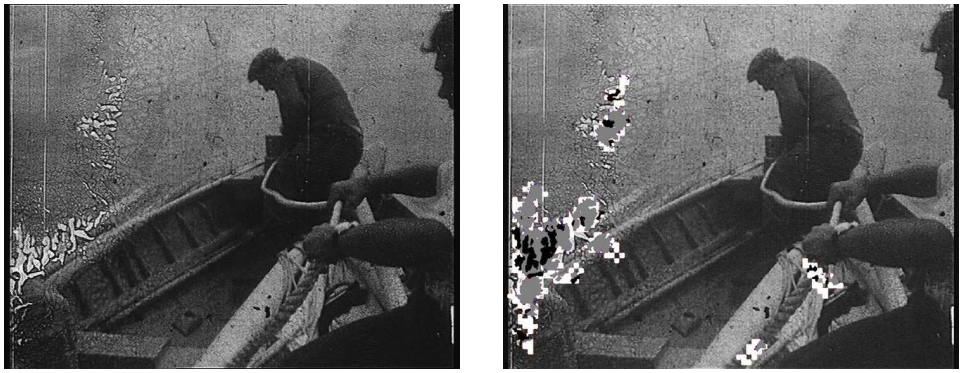
Segments that are not classified as PM are fed to the artefact classification process, which essentially tests for extreme colours. A yet unclassified segment is labelled as "artefact" if

$$\left(\min\left(\mu_c^r,\,\mu_c^g,\,\mu_c^b\right)<T_A\right)\vee\left(\max\left(\mu_c^r,\,\mu_c^g,\,\mu_c^b\right)>1-T_A\right) \tag{2.5}$$

where T_A is a threshold for artefacts. We take the minimum/maximum from the R, G, and B channels and not another combination of the three values because some artefacts show up only in some of the channels (see Section 2.2.2).

Artefacts have slightly different colour characteristics in different sequences, so for the best results T_A has to be tuned for every image sequence.

At the end, all segments that are not labelled as "pathological motion", nor as "artefact", are labelled as "unknown". Since there are many types of pathological motion [4] and artefacts [1], it is still possible that they may represent either one, or the other of the two categories. However, for these ones there is no knowledge incorporated in the scheme yet, and they are subject to further research.



(a) (b)

Figure 2.3: Classification example. a) Original frame. b) Original frame, with classification overlay: complex event areas are displayed in white, pathological motion in grey, and artefacts in black.

2.4 Results and Discussion

Fig. 2.3 shows some results of our analysis scheme. We experimented with three different sequences, each of which was 50-70 frames long. The selected segments were classified either as "pathological motion", "artefact" or "unknown".

The results look quite promising, and a visual inspection throughout the sequence of results indicates that roughly 80% of the classification is correct. This represents a percentage relative to the amount of segments that were classified either as "pathological motion" or "artefact" (around 75% from the total number of selected segments; the other 25% were classified as "unknown"). We rely on visual inspection because there is no ground truth to compare with. The incorrectly classified segments are due to various reasons such as: the lack of previous flicker correction of the input sequence, incorrect input CE masks, etc.

To estimate the complexity of every phase, we assume that the number of segments in a frame is N, on the average. The complexity of the direct temporal matching is then of the order of $O(N^2)$. The complexity of PM classification is $O(N^3)$. Finally, the artefact classification is of the order of $O(N^2)$. Obviously, direct temporal matching speeds up the classification process. It should be mentioned, however, that direct temporal matching may accumulate errors, whereas the other phases do not.

2.5 Conclusions

We have presented here an analysis scheme for the classification of complex event areas, where common motion estimators fail. The scheme is able to discriminate between complex events caused by pathological motion and complex events caused by artefacts. In particular, it performs a segmentation of the frames and tries to find similarities between segments of the current frame and segments of the previous frame. This inter-frame segment matching is done only for the complex event areas, thus avoiding heavy computation. The classification results are quite encouraging, and they give us hope for further developments.

Future work will include the enrichment of our analysis scheme with algorithms for merging segments into objects (to properly steer the restoration process), the use of shot change detection techniques (to avoid segment matching between frames belonging to different shots), and the development of restoration methods for areas degraded by artefacts, as final part of the restoration chain. Further improvements of motion picture restoration are expected to be achieved with object modelling and tracking methods [3], as a qualitative step forward towards a higher level analysis of image sequences.

2.6 Acknowledgements

This work is funded by EU's IST research and technological development programme. It is carried out within the Brava project ("Broadcast Archives Restoration Through Video Analysis").

The authors would like to thank to their colleague Jesper Jensen for the useful discussions on the segment merging procedure.

Bibliography

[1] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[2] V.K. Madisetti, D.B. Williams, editors, "The Digital Signal Processing Handbook", Section XI "Image and video processing", CRC Press and IEEE Press, 1998, ISBN 0-8493-8572-5.

[3] A. Rareş, M.J.T. Reinders, "Object tracking by adaptive modeling", Proceedings of IEEE ICIP 2000, vol. 3, pp. 74-77, Vancouver, BC, Canada, September 10-13, 2000.

[4] A. Rareş, M.J.T. Reinders, J. Biemond, "Statistical analysis of pathological motion areas", The 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 8/1-8/28, London, UK, January 16, 2001.

[5] P.M.B. van Roosmalen, "High-level analysis of image sequences", technical report for INA - Paris (Institut National de l'Audiovisuel), France, the EU Aurora Project, July 1999.

[6] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

[7] M.I. Sezan, R.L. Lagendijk, editors, "Motion Analysis and Image Sequence Processing", Kluwer Academic Publishers, 1993, ISBN 0-7923-9329-5.

[8] K.-C. Tan, H. Lim, B. T. G. Tan, "Restoration of real-world motion-blurred images", CVGIP - Graphical Models and Image Processing, vol. 53, no. 3, pp. 291-299, May, 1991.

[9] J. Watkinson, "The Engineer's Guide to Motion Compensation", Snell & Wilcox Handbook Series, 1994.

Chapter 3

Edge-Based Image Restoration

This chapter has been published as "Edge-Based Image Restoration", by A. Rareş, M.J.T. Reinders and J. Biemond in *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1454-1468, October 2005.

In this paper we propose a new image inpainting algorithm that relies on explicit edge information. The edge information is used both for the reconstruction of a skeleton image structure in the missing areas, as well as for guiding the interpolation that follows. The structure reconstruction part exploits different properties of the edges, such as the colours of the objects they separate, an estimate of how well one edge continues into another one, and the spatial order of the edges with respect to each other. In order to preserve both sharp and smooth edges, the areas delimited by the recovered structure are interpolated independently, and the process is guided by the direction of the nearby edges. The novelty of our approach lies primarily in exploiting explicitly the constraint enforced by the numerical interpretation of the sequential order of edges, as well as in the pixel filling method which takes into account the proximity and direction of edges. Extensive experiments are carried out in order to validate and compare the algorithm both quantitatively and qualitatively. They show the advantages of our algorithm and its readily application to real world cases.

3.1 Introduction

An important part of the scientific and cultural heritage of the modern times has been stored in the form of film and photo archives. Unfortunately, the classic storage media for these information sources are bound to gradually decay in time, risking the total loss of the valuable information they are carrying. Fortunately, with the arrival of the digital era, the digitised films and photographs can now be copied easily and virtually without information loss. An equally important aspect is the opportunity of doing restoration in superior ways, never possible in the past. As such, information that disappeared completely from its physical support can now be restored thanks to advanced algorithms developed in the restoration community. Modern technologies have brought along economical benefits, too. The digitised content is now cheaper and easier to store, search, reuse and distribute. The digitally broadcasted content will take full advantage of these benefits. But today's quality requirements will only be fulfilled when digital restoration is applied to the degraded archived content.

The current restoration algorithms for image sequences exploit both spatial and temporal information, and perform quite well in general [23, 35]. However, they fail when there is difficult motion in the sequence [31, 34], in particular for the detection and correction of blotches. Blotches are artefacts typically related to film that are caused by the loss of gelatin, or the presence of dirt particles on the film [19]. Due to the difficult object movements, wrong motion vectors are extracted from the sequence. As a result, the spatiotemporal restoration process that follows may introduce unnecessary errors that are visually more disturbing than the blotches themselves. The extracted temporal information becomes unreliable, and a source of errors itself.

Instead of protecting the blotches from being restored [34], in our view, the detected artefacts should be restored based on spatial information alone [32, 33], discarding the temporal information. In the BRAVA project [18] of the European Union, we have devised a novel restoration algorithm that takes advantage of the available spatial information in order to restore the degraded film frames. This algorithm is not intended to replace the spatiotemporal algorithms, rather, to complement them in places where they fail. Because of its spatial nature, the algorithm can also be applied for the restoration of missing areas in damaged (usually old) photographs, for the automatic interpolation of damaged pixels from the CCD sensors in new digital cameras, or for the concealment of errors in compressed data due to transmission errors. Another area of application is the reconstruction of occluded objects when they are partly covered by other objects. This can be useful for assessing the correctness of a segmentation procedure, as well as for determining the relative depths of objects [29]. The proposed algorithm only deals with the problem of filling in the missing information.

The task of artefact detection represents a separate problem. With some exceptions, the artefact detection and the restoration are usually treated in different algorithms. In this paper, we explicitly assume that the artefact mask is detected by another algorithm and contains no holes.

3.1.1 Related Work

Several spatial restoration approaches for missing data have been proposed already in the literature. They address the problem of filling in missing data from different points of view. In the following, a short categorised overview is given that presents the most popular approaches.

Restoration based on partial differential equations and variational methods. A recent category of algorithms centered around the idea of "image inpainting" has shown promising results on restoring image structure for large, piecewise smooth patches. Masnou and Morel, for example, present in [27] a simple but effective approach for filling in missing areas based on the connection of the level lines (i.e. isophotes) that have the same values on the artefact contour. The method was further developed in [28]. In [5, 6, 7, 8, 9], Ballester, Bertalmio et al. propose more complex variational approaches for joint interpolation of grey levels and gradient/isophotes directions. In [12, 13, 14], Chan et al. present several inpainting methods based on total variation models, curvature driven diffusions, and Euler's elastica. In [10, 30], Bertalmio, Sapiro, Rane et al. further refine the aforementioned methods by trying to combine in one algorithm different approaches for structure and texture interpolation.

Structure-Based Restoration. Atzori and De Natale propose in [2] a spatial restoration method for recovering missing blocks (corresponding to data packets) in video transmission over packet networks. They use only the information existing in

the same frame, by making a "sketch" of the edges around the missing blocks. These edges are connected in a pairwise fashion, if possible, and a smooth interpolation takes place subsequently in the areas delimited by these sketched edges. While this paper uses a spline interpolation to recover the shapes of the edge connections, in [3] they present an alternative based on Bezier curves. In [4], Atzori et al. present a spatiotemporal algorithm which first uses a temporal interpolation and then applies a spatial, mesh-based warping to reduce the temporal restoration errors mainly caused by complicated motion. In [33], we present a spatial algorithm for the reconstruction of artefacts based on explicit information about the surrounding edges. The main assumption here is that edges are (locally) straight. Simple edge information is extracted from the image and used to recover the edges inside the artefact. The straight edges reconstructed inside the artefact are then used to guide a smooth interpolation between the edges.

Convolution- and Filter-Based Restoration. With their normalised and differential convolution, Knutsson and Westin [22] defined a general method for interpolating N-dimensional data through convolutions based only on valid data. Their approach is more general and flexible than some restricted convolution, by allowing the association of certainty values to each data point and an applicability operator to the filters to be applied. In [21], Khriji et. al. presented a restoration technique based on spatial rational filters.

Texture-Based Restoration. In [16], Efros and Leung present a non-parametric texture synthesis algorithm based on Markov Random Fields. Their approach restores pixels based on the similarity between their local neighbourhood and the surrounding neighbourhoods. From the candidate neighbourhoods, one is randomly selected and the value of its central pixel is pasted at the current location, a process which is able to intelligently imitate the natural randomness of textures. Bornard et al. [11] have further developed the aforementioned texture synthesis for image sequences by incorporating temporal information and imposing some local consistency constraints which allow the algorithm to also synthesise structured objects that do not have random appearances. In [15] a method is presented by Criminisi et al. that also extends the approach of Efros and Leung by imposing higher priorities in the restoration order for pixels lying in the neighbourhood of edges, thereby preserving better edge sharpness and continuity. In [24], Kokaram presents a parametric texture synthesis algorithm which employs *2D* autoregressive models (combined with the Gibbs sampler) in a Bayesian approach. In [25, 26], he introduces a more general framework for restoring image sequences, based on the Markov Chain Monte Carlo methodology. A solution is proposed for jointly detecting and restoring missing data and motion vectors, while also handling occlusion and uncovering situations. In [20], Jia and Tang describe a novel technique based on tensors. Here, edge structure is first reconstructed, followed by texture synthesis. Both steps use adaptive tensor voting. Another way of synthesising texture is presented in [1] by Acton et al. Their approach is based on a diffusion generated by partial differential equations, and a simultaneous reaction based on Gabor filters and AM-FM dominant component

analysis. In [17], Hirani and Totsuka combine spatial and frequency information to reconstruct the missing image structure/texture, in a framework of projection onto convex sets.

Connections with proposed method. Our approach relates most to the sketch-based method of Atzori and De Natale [2]. It generalises the algorithm presented in [33], and employs higher level features extracted from the image. Our approach also bears some similarity with the algorithm of Jia and Tang [20], in what concerns the main steps of the algorithm. Each of these steps is, however, differently approached. The novelty of our method consists of the approximation of the incoming edges with circle arcs, the use of the spatial order of edges, and the directional interpolation scheme that restores missing areas parallel to the recovered edges. As opposed to the classic texture-based restoration algorithms, which do not preserve object shapes, we prefer (together with Atzori and De Natale, and Jia and Tang) to use explicit edge information to capture the image structure. Our main motivation comes from two observations. On the one hand, edges generally separate areas with different content. Therefore, the interpolation should take place independently on both sides of an edge. On the other hand, edges are more robust against intensity changes such as local shading, thereby being more robust than isophote-based algorithms, for example.

Throughout the paper we compare our proposed algorithm with the related restoration scheme of Atzori and De Natale, both qualitatively as well as quantitatively. We also present a qualitative comparison with the algorithm of Masnou [28], which uses a variational approach applied to the image isophotes.

3.1.2 Outline

In Sec. 3.2, we present the main steps of the algorithm. Sec. 3.3 concentrates on how the structure of the missing areas is recovered. Sec. 3.4 describes our interpolation method, which takes into account the structure recovered in the previous section. Sec. 3.5 is devoted to presenting and discussing experimental restoration results, as well as comparisons with other algorithms. Finally, Sec. 3.6 draws conclusions and outlines future work.

3.2 Algorithm Overview

The spatial restoration algorithm that we propose consists of three main steps, depicted in Fig. 3.1: 1) Edge detection and edge feature extraction; 2) Image structure reconstruction; and 3) Edge-based inpainting.

The input to our algorithm is an image and an artefact mask. Here, we assume that the artefact mask is detected by another algorithm. For the sake of simplicity, but

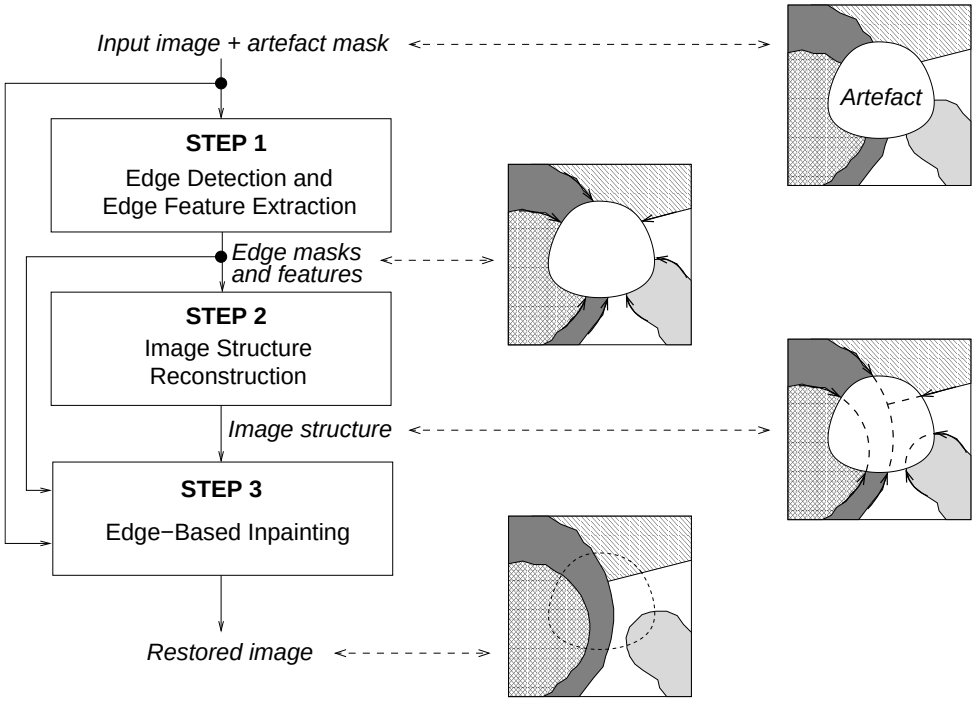


Figure 3.1: General algorithm outline (left) and an illustration of the inputs/outputs for each stage (right).

without loss of generality, in the remainder of this paper we consider that the mask consists of only one artefact, and that the image is grey-valued.

Assuming that the artefact location, size and shape are independent from the image content, then the structure of the original image inside the artefact area is a continuation of the structure outside it. More specifically, the edges inside the artefact are continuations of the outside edges. We therefore use the edge information explicitly to guide the restoration process.

In the first step, edges are detected around the artefact, based on the contours of the segments that result from a watershed segmentation. Ideally, these edges separate two objects (or at least two different homogeneous regions), both of which are partially occluded by the artefact. The object edges are extracted in clockwise order, from a point of view lying inside the artefact. Simple edge features are extracted for each edge, such as: the luminance values on both sides of the edge, and the local gradient magnitude along the edge. Only relevant edges are then kept for the next steps (e.g. those that have at least a certain gradient magnitude).

In the second step, we try to recover the structure of the image within the artefact area. This problem is ill-posed: virtually anything could have existed in the area covered by the artefact, before the degradation took place. We have to "invent" content in places where it was lost, based on some assumptions about the usual image properties. In our case, we have modelled the edges as locally circular shapes (equivalent to a second order polynomial). This modelling was subject to several constraints, such as colour matching, and non-crossing of the object edges. Our model tries to couple edges that are strongly related to each other, thereby reconnecting the pairs of edges that were part of the same object contour. The matching of the edges is on the one hand based on the similarity of the aforementioned edge features, and on the other hand on *continuity* and *sequentiality* criteria. For an edge couple (e.g. $A_1 - A_2$, $B_1 - B_2$ or $C_1 - C_2$ in Fig. 3.2) the continuity is measured by fitting a circle to the pair of edges and measuring the goodness of the fit (e.g. the spatial deviation of both edges from the fitted circle). Unlikely edge couples are ignored and the remaining ones are iteratively joined into edge groups (see Fig. 3.4). An edge group is a (as large as possible) set of consecutive edge couples such that no two couples cross each other. Based on the set of possible groupings, specific configurations that represent potential image structures within the artefact are created. Each configuration is then rated by its "sequentiality", which is a measure indicating the likeliness of a particular configuration (essentially trying to minimize the number of crossing couples). The score of a configuration is based upon the sequentiality together with the other features that estimate the continuity and similarity of edges. The best configuration is then found by selecting the configuration that minimizes this score. After finding the best configuration, spare edges (e.g. $E_1 - E_2$ in Fig. 3.6(a,b), or $E_2 - E_B$ in Fig. 3.7(c)), i.e. edges that were not included in any edge couples, are traced one by one into the selected configuration. They are traced up to the point where they meet another edge (or edge couple), or alternatively they gradually van-

ish towards the opposite side of the artefact. In this way, the structure of the image is recovered inside the artefact area.

Finally, in the third step, the artefact is restored by inpainting, taking into account the recovered image structure. Essentially, the inpainting procedure restores a pixel based on the surrounding recovered edges. The surrounding edges indicate which pixels on the artefact border are used for the interpolation. Then, based on the distance to these border pixels, the pixel inside the artefact is interpolated.

The sketch-based interpolation of Atzori and De Natale [2] follows the three main steps presented in Fig. 3.1. Since the first step does not concern the restoration directly, we address the differences between our scheme and the one of Atzori and De Natale for the other two steps. In step two, the differences concern the set of features used, and, additionally, the way we combine them in order to characterise the overall acceptability of the reconstructed structure. In the last step, our interpolation method tries to draw strips "parallel" to the nearby edges, resulting in smooth patches. Atzori and De Natale have used a "patch repetition" approach, in which the areas around the artefacts are mirrored across the artefact edge. Many other smaller differences between the two methods exist as well in the above steps (e.g. in step two, the way we normalise the values of different features in order to bring them in the same range).

3.3 Image Structure Reconstruction

The structure reconstruction step is crucial to our proposed restoration scheme, since the explicit image structure that is recovered represents the "skeleton" of the restoration process. The input to this step represents a list of *edges* coming into the artefact, in clockwise order. The output of this step is a list of *edge couples* arranged in *groups* of edges, and a list of *spare edges*.

To build accurate pairwise connections between edges, we make use of *local* features, as well as *global* features. Local features describe how well two edges match each other if they were part of the same edge couple. Global features express the goodness of a complete configuration of edge couples. The local features are 1) the two luminance values on both sides of each edge in the edge couple, 2) the local gradient magnitudes of both edges, and 3) the degree to which the edge couple fits a common circle. The global feature expresses the degree to which edge couples do not cross each other within a configuration. The overall cost c^{cfg} of a particular configuration is given by:

$$c^{cfg}(\boldsymbol{Z}) = \frac{4 \times c^{loc}(\boldsymbol{Z}) + c^{glb}(\boldsymbol{Z})}{5} \tag{3.1}$$

where $\boldsymbol{Z}$ represents the configuration of groups of edge couples, c^{loc} is the cost related to the four local features, and c^{glb} is the cost associated with the single

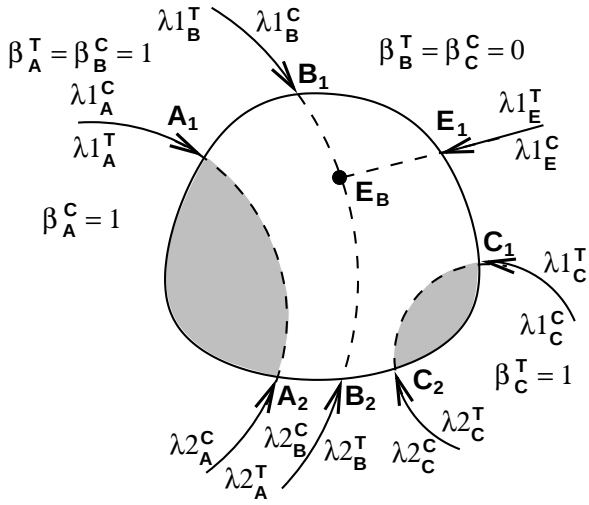


Figure 3.2: The intensity feature for a group formed by the edges couples $A_1 - A_2$, $B_1 - B_2$ and $C_1 - C_2$.

global feature. All costs have values between 0 and 1, with 0 indicating a perfect match and 1 indicating a complete mismatch. The process of building up the final configuration $\boldsymbol{Z}$ is presented later in this section.

3.3.1 Local features

Before specifying the couple-related costs, we first give intensity and magnitude representation of an edge. The set of intensities (λ_i) on the clockwise side of the edges is given by: $\boldsymbol{\Lambda} = \left\{\lambda_i = \widehat{m}\left(\boldsymbol{I}\left(\boldsymbol{b}_i\right)\right) \mid i = 1, \ldots, N^E\right\}$, where $\boldsymbol{I}$ represents the intensity image, $\boldsymbol{b}_i$ is the vector of pixels on the artefact border between edge i and the (clockwise) next edge, $\widehat{m}$ is the median operation, and N^E is the number of edges. The set of edge gradient magnitudes (γ_i) is given by: $\boldsymbol{\Gamma} = \{\gamma_i = \widetilde{m}\left(\boldsymbol{J}\left(\boldsymbol{e}_i\right),\right.$ $\left.\left[N^{MAX}, N^{MAX} - 1, \ldots, 2, 1\right]\right) \mid i = 1, \ldots, N^E\}$, where $\boldsymbol{J}$ is the gradient magnitude of $\boldsymbol{I}$ (obtained after some smoothing, in order to remove noise), $\boldsymbol{e}_i$ is the ordered vector of edge pixels, with its *head* lying on the artefact border and its *tail* stretching outwards, $\widetilde{m}$ is a weighted median operation, $\left[N^{MAX}, N^{MAX} - 1, \ldots, 2, 1\right]$ represents the vector of weights used by $\widetilde{m}$, giving more weight to the edge pixels near the artefact border, and N^{MAX} is the maximum number of pixels in an edge (we used $N^{MAX} = 12$ in our experiments). $\boldsymbol{\Gamma}$ and $\boldsymbol{\Lambda}$, together, are not redundant, since γ_i is not always directly related to $|\lambda_i - \lambda_{i-1}|$. In fact, γ_i indicates the smoothness of edge i.

The weighted median is used to calculate $\boldsymbol{\Gamma}$ because, as we get farther from the artefact, it is obvious that the local properties of the edge tend to become less and

less related to the missing edge inside the artefact. It may also happen that a third object present in the image lies close to the artefact, without touching it. In this case, the actual edge is partly occluded, and the detected edge bends to follow the border of the third object. As a result, the edge tail is not related to the structure to be recovered in the artefact. Weighing the tail less than the head tries to overcome this situation.

The cost related to the local features of a configuration $\boldsymbol{Z}$, c^{loc}, is computed by averaging the costs c_{ij} of every edge couple within that configuration:

$$c^{loc}(\boldsymbol{Z}) = \frac{\sum_{i=1}^{N^G} \sum_{j=1}^{\|\boldsymbol{G}_i\|} c_{ij}}{\sum_{i=1}^{N^G} \|\boldsymbol{G}_i\|} \tag{3.2}$$

where $\boldsymbol{G}_i$, $i = 1, \ldots, N^G$ are the groups of edge couples in configuration $\boldsymbol{Z}$, $\|\boldsymbol{G}_i\|$ is the number of edge couples in group $\boldsymbol{G}_i$, and c_{ij} is the individual cost of edge couple $\boldsymbol{q}_{ij}$ (the j^{th} couple of group $\boldsymbol{G}_i$).

$$c_{ij} = \sqrt{\frac{\left(\beta_{ij}^C \left(\lambda 1_{ij}^C - \lambda 2_{ij}^T\right)^2 + \beta_{ij}^T \left(\lambda 1_{ij}^T - \lambda 2_{ij}^C\right)^2 + \beta_{ij}^C \beta_{ij}^T \left(\gamma 1_{ij} - \gamma 2_{ij}\right)^2 + \omega_{ij}^2\right)}{\beta_{ij}^C + \beta_{ij}^T + \beta_{ij}^C \beta_{ij}^T + 1}} \tag{3.3}$$

The cost c_{ij} of a specific couple $\boldsymbol{q}_{ij}$ indicates how well the two edges within the couple match each other, i.e. whether they describe the border of the same object. Since they belong to the same object, it seems natural to require that the intensities on both sides of the edges have similar values (first two terms in Eq. 3.3) and that the strength of the edges match as well (third term in in Eq. 3.3). Further, we assume that the object edges continue each other smoothly, without abrupt changes of direction (fourth term in the same equation). The cost c_{ij} is then defined by the formula in Eq. 3.3, with λk_{ij}^s representing the intensity on side s of edge k ($k \in \{1, 2\}$) in the couple, as shown in Fig. 3.2. The *side* index s indicates whether the intensity belongs to the side lying in *clockwise* ($s = C$), or in *trigonometrical* ($s = T$) direction. γk_{ij} represents the gradient magnitude along edge k of couple $\boldsymbol{q}_{ij}$. The intensity and gradient subscript notations are different here in order to reflect the affiliation of the edge to couple $\boldsymbol{q}_{ij}$ from group $\boldsymbol{G}_i$. $\beta_{ij}^s \in \{0, 1\}$ are flags indicating whether the next edge on side s of edge "1" from edge couple $\boldsymbol{q}_{ij}$ belongs to a couple in the same group ($\beta_{ij}^s = 1$) or not ($\beta_{ij}^s = 0$). These binary flags effectively switch off cost contributions of the respective luminances and gradients in places where they are rendered irrelevant by spare edges (e.g. spare edge E_1 between edge couples $B_1 - B_2$ and $C_1 - C_2$ in Fig. 3.2 prevents the comparison of $\lambda 1_B^C$ with $\lambda 2_B^T$), or by edge couples from other groups. ω_{ij} is the cost of fitting a circle to couple $\boldsymbol{q}_{ij}$.

Let us discuss in more detail why the fourth term in Eq. 3.3, ω_{ij}, is essential. First, when an edge couple has spare edges on both sides, none of the first three features

is of any help. Therefore, we need a supplementary feature in order to be able to do the matching. Second, when there are more objects with similar appearance that are occluded by the artefact (e.g. the fingers of a hand), the three intensity-based features alone are not sufficient to discriminate between them. Third, exploiting the continuity of the edges within a couple can help in selecting the right couples. For example, the shape of the potential edge couple $e_1 - e_8$ in Fig. 3.4(b) is less *natural* than the shape of the couple $e_1 - e_{12}$. Obviously, the reconstruction of object shapes is an ill-posed problem which we need to avoid. We do it by putting constraints on the edge reconstruction. Namely, we use smoothness and convexity constraints that we implement by means of a model fitting (to ensure reliable parameters).

The *naturalness* of a couple is a psychological term, rather than a physical measurement. It describes the way humans perceive the edge continuation, and not the deviation from a theoretically objective ground truth (which does not exist in practice). Naturalness is discussed in the Gestalt theory on perceptual grouping, the grounds of which were laid as early as 1923 by psychologist Max Wertheimer [36]. This theory has shown that some visual cues, such as proximity, similarity, good continuity, closure, etc., allow us to group parts of an image into objects (or groups of related objects). For example, in Fig. 3.4(b), the naturalness of edge couple $e_1 - e_{12}$ is expressed by a combination of properties such as similar local direction (i.e. tangent) and constant curvature.

These observations led us to define the naturalness of a couple by how well they fit a circle[1]. The cost ω_{ij} of fitting a circle to the edge couple is defined by:

$$\omega_{ij} = 1 - \delta_{ij} \times \varphi_{ij} \times \theta_{ij} \tag{3.4}$$

where δ_{ij} is the *spatial deviation* of the couple from the fitted circle, φ_{ij} is the *angular consistency* factor, and θ_{ij} is the *aperture quality* factor. ω_{ij} returns values between 0 (ideal case) and 1 (worst case). For the δ_{ij}, φ_{ij} and θ_{ij} parameters, the significance of these values is reversed (0 represents the worst case, while 1 represents the ideal case). This enables us to propagate a "worst case" value identified with either δ_{ij}, φ_{ij} or θ_{ij}, to the circle fitness measure ω_{ij}.

The *spatial deviation*, δ_{ij}, indicates how far on the average the edge pixels lie with respect to the fitted circle. First, a distance δ^*_{ij} that represents the median of the distances from the edge pixels in couple $\boldsymbol{q}_{ij}$ to their closest points on the fitting circle is defined:

$$\delta^*_{ij} = \widehat{m}\left(\left|d\left(\left[\boldsymbol{e}^1_{ij}, \boldsymbol{e}^2_{ij}\right], \boldsymbol{x}^{\odot}_{ij}\right) - r^{\odot}_{ij}\right|\right) \tag{3.5}$$

where $\boldsymbol{e}^1_{ij}$ and $\boldsymbol{e}^2_{ij}$ are the two edges of couple $\boldsymbol{q}_{ij}$, concatenated here in a single vector for the median operation, $r^{\odot}_{ij}$ and $\boldsymbol{x}^{\odot}_{ij}$ are the radius and the center of the

[1] To avoid numerical problems for straight edge couples (i.e. a large radius of the fitted circle), all radii above a certain threshold (10^4) were limited to that threshold.

fitting circle, respectively, and $d()$ represents the euclidean distance. In order to bring the value of δ_{ij}^* between 0 and 1, we use the following normalisation:

$$\delta_{ij} = \frac{K^\delta}{\delta_{ij}^* + K^\delta} \tag{3.6}$$

where $K^\delta = 2.16$ is a constant chosen to calibrate δ_{ij} in such a way that if it is above a predefined threshold, it indicates a valid edge couple.

The spatial deviation determines how well the couple fits a circle, but does not take into account the "direction" of the edges. From Fig. 3.3(a) one can observe that the normal edge couple lies on the fitted circle in the following clockwise order: *tail* 1 − *head* 1 − *head* 2 − *tail* 2, while an erroneous edge couple lies in the order *tail* 1 − *head* 1 − *tail* 2 − *head* 2 (see Fig. 3.3(b)). In both cases, the spatial deviation is small. To penalise these incorrect continuations, we introduce the *angular consistency*:

$$\varphi_{ij} = \frac{|\Delta^*\alpha_i - \Delta^*\alpha_j|}{|\Delta^*\alpha_i| + |\Delta^*\alpha_j| + \varepsilon^\varphi} \tag{3.7}$$

where $\Delta^*\alpha = \alpha^t \setminus \alpha^h$, with α^h and α^t being the angles measured from the center of the fitted circle to the *head* and *tail* of the edge, respectively, and $\varepsilon^\varphi = 10^{-6}$ a small value to avoid a potential division by zero. The operator "\" defines the smallest angle (in absolute value) between two angles α_1 and α_2 (both between 0 and 2π) as follows:

$$\alpha_1 \setminus \alpha_2 = \begin{cases} \Delta\alpha_{1,2} = \alpha_1 - \alpha_2 & , if\ |\Delta\alpha_{1,2}| \leq \pi \\ -\operatorname{sign}(\Delta\alpha_{1,2}) \times (2\pi - |\Delta\alpha_{1,2}|) & , otherwise \end{cases} \tag{3.8}$$

Finally, we also want to penalise very wide angles between the heads of the two edges in a couple, since such configurations are very unlikely. For example, the edge couple in Fig. 3.3(d) is much less common than edge couple in Fig. 3.3(c). This is measured by the *aperture quality*:

$$\theta_{ij} = \begin{cases} \sqrt{1 - \frac{\left|\widetilde{\Delta}\alpha_{ij}^h\right|}{2\pi}} & , if\ \left|\widetilde{\Delta}\alpha_{ij}^h\right| < \left|\widetilde{\Delta}\alpha_{ij}^t\right| \\ \sqrt{\frac{\left|\widetilde{\Delta}\alpha_{ij}^h\right|}{2\pi}} & , otherwise \end{cases} \tag{3.9}$$

where $\widetilde{\Delta}\alpha_{ij}^h = \alpha_i^h \setminus \alpha_j^h$, $\widetilde{\Delta}\alpha_{ij}^t = \alpha_i^t \setminus \alpha_j^t$, and $\theta_{ij} \in [0, 1]$.

The square roots in Eq. 3.9 are meant to approximately calibrate the values returned by θ_{ij}. Note that the *aperture quality* measure is (more or less) equivalent to the *proximity* property stated by Wertheimer [36]. Moreover, the *aperture quality* is scale independent, which is a desirable property of any extracted feature.

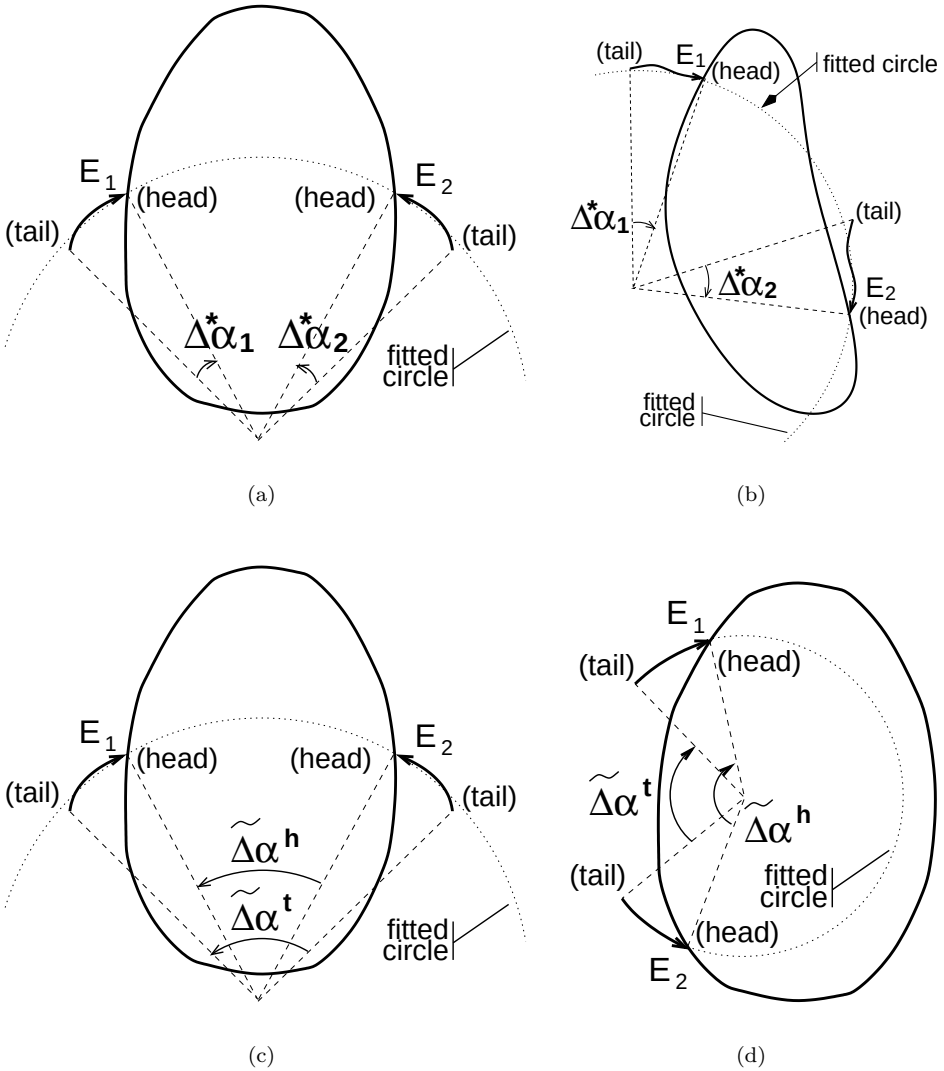


Figure 3.3: Behaviour of the circle-fitting related measures. a) Both the spatial deviation and the angular consistency indicate a good edge couple. b) The spatial deviation indicates a good edge couple, whereas the angular consistency indicates a bad one. c) Both the angular consistency and the aperture quality indicate a good edge couple. d) The angular consistency indicates a good couple, whereas the aperture quality indicates a bad one.

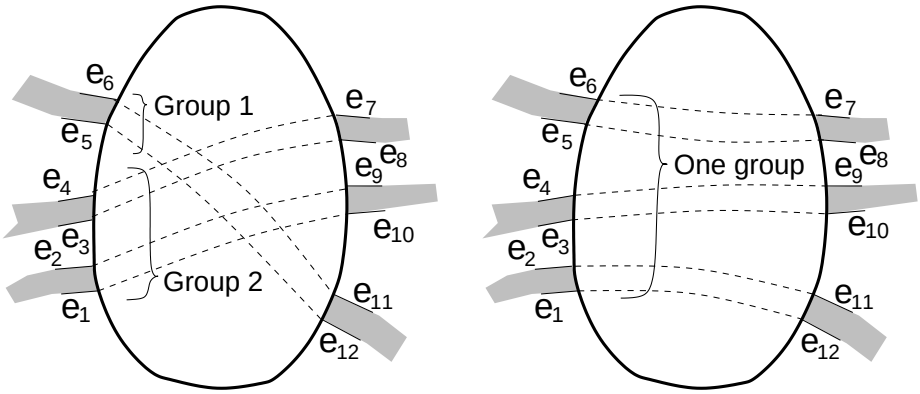


Figure 3.4: Contribution of the *sequentiality* parameter. a) Configuration penalised by the sequentiality parameter. b) Configuration given preference by the sequentiality parameter.

3.3.2 Global Feature and Prediction of the Final Configuration

Besides looking at how well edges match within a couple, we also take into account the global configuration that is created, in order to exclude false edge couples. Hereto we measure the edge order, or *sequentiality* of the edges, which validates the configuration. Suppose we are dealing with an artefact that splits a number of horizontal objects in two, i.e. they appear (once) on both the left and right sides of the artefact. As an example see Fig. 3.4. If we inspect the artefact border in clockwise order, the object edges on one side (such as edges $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_6\}$ in Fig. 3.4(b)) appear in exactly opposite order compared to the ones on the other side (edges $\{\boldsymbol{e}_7, \ldots, \boldsymbol{e}_{12}\}$). This is a very useful property of the edges around artefacts, since it is extremely robust against noisy data. For example, Fig. 3.4(a) shows how edges can be connected in a wrong way when only the local features are accounted for. Here, the presence of noise resulted in slightly tilted edges (which affected the circle fitting cost), as well as distorted grey levels and gradient magnitudes (which affected the other costs). When the sequentiality of the edge couples is also taken into account, the right configuration can be better predicted (Fig. 3.4(b)). Edge displacement and changed grey levels do not change edge order so they do not influence the sequentiality feature. The only way in which noise can affect this feature is by hampering the edge detection process, introducing false edges, or missing existing ones. However, the other edges still lie in the same consecutive order, which contributes to the stability

of the cost. Most probably, an erroneously introduced edge, or the remaining pair of a missed edge will be treated as spare edges and thus the impact on the sequentiality cost is reduced (since this cost is computed over pairs of edges only).

The sequentiality represents a natural property of most object edges. If edges are not sequential, then they should change their order in the image very often, i.e. they should cross each other, as in interwoven patterns. While interwoven patterns are not unusual, they are certainly not encountered very often.

It is worth pointing out that the sequentiality parameter does not forbid a configuration containing crossing groups - rather, it penalises it. If the evidence coming from the local features strongly indicates a crossing, separate groups are formed accordingly (resulting in a configuration such as the one in Fig. 3.4(a)).

Sequential configurations usually have smooth edge couples. This does not mean that the features based on sequentiality and circle fitting are the same: smooth edge couples are not necessarily sequential. Besides, in practice, the detected edges are sometimes displaced or tilted, which affects the smoothness feature. The sequentiality comes to correct for such cases.

Three problems arise when determining the sequentiality of a configuration. First, we must find a way to express it as a number. Second, despite the fact that it is used to calculate the configuration cost, we can measure it only *after* the configuration of edge couples has been formed, based on some cost that does not depend on sequentiality. And third, the sequentiality does not represent a measurement of each edge couple alone. Rather, it is a measurement of the complete configuration, which is an ensemble of edge couples. The latter problem gave rise actually to the formula in Eq. 3.1.

For the moment, let us assume that the configuration of edge couples has already been found. The groups of edge couples in the current configuration are denoted by $\boldsymbol{G}_i$, $i = 1, \ldots, N^G$. Eq. 3.10 then defines the cost related to the global property of sequentiality:

$$c^{glb}(\boldsymbol{Z}) = \begin{cases} 1 - \frac{\sum_{i=1}^{N^G}(\|\boldsymbol{G}_i\|-1)}{\left\lfloor \frac{N^E}{2} \right\rfloor - 1} & , \left\lfloor \frac{N^E}{2} \right\rfloor \geq 2 \, and \, \sum_{i=1}^{N^G} \|\boldsymbol{G}_i\| \geq 1 \\ 0 & , \left\lfloor \frac{N^E}{2} \right\rfloor = 1 \, and \, \sum_{i=1}^{N^G} \|\boldsymbol{G}_i\| = 1 \\ 1 & , otherwise \end{cases} \tag{3.10}$$

Operator $\lfloor \, \rfloor$ rounds to the nearest smaller integer. Thus, the term $\left\lfloor \frac{N^E}{2} \right\rfloor$ represents the maximum number of edge couples that can be achieved out of the N^E edges. To exemplify this measure, the six couples in Fig. 3.4(a), have a sequentiality cost of $1 - ((4-1) + (2-1))/(6-1) = 0.2$, while the six couples in Fig. 3.4(b), have a sequentiality cost of $1 - (6-1)/(6-1) = 0$. It is clear now that the global feature favours fewer but larger groups of edge couples (e.g. Fig. 3.4(b)), and penalises

more, but smaller groups (e.g. Fig. 3.4(a)). As a result, it imposes a (desired) natural constraint on the configurations (in most of the images, edges do not cross each other locally).

The main steps for building up the final configuration are summarized into pseudocode in Fig. 3.5.

3.3.3 Spare Edges Reconstruction

Before one can use the selected configuration $\boldsymbol{Z}$ to restore the artefact, the spare edges must be integrated with the edge couples. Ideally, we should be able to fit circles to spare edges, similarly to what we did with the edge couples, and then calculate where they intersect with the couples. Unfortunately, experiments have shown that fitting circles to spare edges is unreliable and frequently gives unnatural results. This happens mostly because 1) the edges are usually small (remember that when fitting a circle to a couple, the two edges are relatively far apart, making the fit reliable), and 2) they could be quite noisy (spatially). This motivated us to approximate the spare edges with straight lines (a choice which was validated by experimental results).

To reconstruct the structure of the spare edge inside a strip of the artefact, we iteratively pick the spare edge with the biggest difference between the luminances on its two sides $\left|\lambda_i^C - \lambda_i^T\right|$, approximate it with a straight line, and recover it. This is repeated until all spare edges have been traced.

When recovering a spare edge, two situations may occur: 1) the recovered spare edge does not intersect with any other reconstructed edge within the artefact area (e.g. edges $E_1 - E_2$ and $E_3 - E_4$ in Fig. 3.6(a,b)); and 2) the recovered spare edge intersects with another edge that was already recovered inside the artefact (e.g. edge $E_2 - E_B$ in Fig. 3.7(c)). In situation 1 we are dealing with a *fading* edge, while in situation 2 the edge is part of a *T junction*. In Fig. 3.6(b), the reconstructed spare edge increments the number of *middle strips*[2] existing inside the artefact. In all other cases, it only adds a new *side strip*[3] - even in Fig. 3.6(a), where the strip $A_1 - A_2 - E_2 - E_1 - A_1$ will be considered a side strip with *fragmented* contour. Fragmented contours occur in places where a reconstructed fading edge intersects the same contour a second time, cutting out a side strip and fragmenting the old contour (e.g. in Fig. 3.6(a), the *continuous* contours $A_1 - A_2$ and $A_1 - B_1$ become $A_1 - E_1; E_2 - A_2$ and $A_1 - E_3; E_4 - B_1$, respectively).

[2] A middle strip is an area that spans from one side to the other of the artefact and is usually delimited by two consecutive edge couples from the same group (e.g. strips $A_1 - E_1 - E_2 - A_2$ in Fig. 3.6(b), or $A_1 - B_1 - B_2 - A_2$ and $A_1 - E_2 - E_B - B_2 - A_2$ in Fig. 3.7(b) and (c), respectively).

[3] A side strip is an area delimited usually by a single edge couple, or by one or two spare edges (e.g. $E_1 - E_2 - E_1$, $E_3 - E_4 - E_3$ or $B_1 - B_2 - B_1$ in Fig. 3.6(a), $E_2 - E_B - B_1$ in Fig. 3.7(c), or $A_1 - E_{A_1} - E_1$ in Fig. 3.7(d)).

1. Calculate the feature distance d_{kl} between all possible edge pairs $(\boldsymbol{e}_k, \boldsymbol{e}_l)$ with Eq.3.3 by setting all β flags to 1 (at this point we do not know which edges are spare ones). d_{kk} is set to 1 for all k in order to avoid edges to be connected to themselves.

2. Calculate the potential of two edges to make a couple ($K^E = \sqrt{0.1}$ is introduced here in order to avoid that two spare edges are fitted to each other):

$$\pi_{kl} = \pi_{lk} = \begin{cases} TRUE & , if\ d_{kl} < \min\left(2 \times \min_k d_{kl},\ 2 \times \min_l d_{kl},\ K^E\right) \\ FALSE & , otherwise \end{cases}$$

3. FOR every possible (not necessarily maximal) set of couples $\boldsymbol{Q}_i = \{\boldsymbol{q}_j = (\boldsymbol{e}_k, \boldsymbol{e}_l)\,|\pi_{kl} = TRUE\}$

   (a) All edges which do not belong to a couple from $\boldsymbol{Q}_i$ are considered spare edges and are taken into account only in step [b.vi].

   (b) FOR every couple $\boldsymbol{q}_j$ from $\boldsymbol{Q}_i$

      i. Initialize group $\boldsymbol{G}_1$ with couple $\boldsymbol{q}_j = (\boldsymbol{e}_k, \boldsymbol{e}_l)$ and remove the couple from $\boldsymbol{Q}_i$.

      ii. Find next edges: $\boldsymbol{e}_k^{next}$ clockwise from $\boldsymbol{e}_k$ and $\boldsymbol{e}_l^{next}$ anti-clockwise from $\boldsymbol{e}_l$, provided that: $\exists\, u, v$ such that $\boldsymbol{e}_k^{next} \in \boldsymbol{q}_u$, $\boldsymbol{q}_u \in \boldsymbol{Q}_i$, and $\boldsymbol{e}_l^{next} \in \boldsymbol{q}_v$, $\boldsymbol{q}_v \in \boldsymbol{Q}_i$.

      iii. IF $u = v$ AND $\boldsymbol{q}_u$ does NOT intersect any couple form the current group THEN
      add couple $\boldsymbol{q}_u$ to the current group, remove it from $\boldsymbol{Q}_i$ and GO TO step ii.

      iv. Repeat steps ii-iii, this time searching on the other side of the first couple from the current group: anti-clockwise from $\boldsymbol{e}_k$ and clockwise from $\boldsymbol{e}_l$.

      v. If $\boldsymbol{Q}_i \neq \emptyset$ and no more couples can be added to the current group, initialize another group with a couple from $\boldsymbol{Q}_i$ and repeat steps i-iv for the new group.

      vi. Calculate the cost c^{cfg} of the current configuration of groups, $\boldsymbol{Z}_h$ according to Eq.3.1 (the β flags are now enabled according to the existing spare edges).

4. Choose the configuration with the minimal cost:

$$\boldsymbol{Z} = \boldsymbol{Z}_m,\ \ m = \arg\min_h c^{cfg}(\boldsymbol{Z}_h)$$

Figure 3.5: Pseudocode for the grouping procedure.

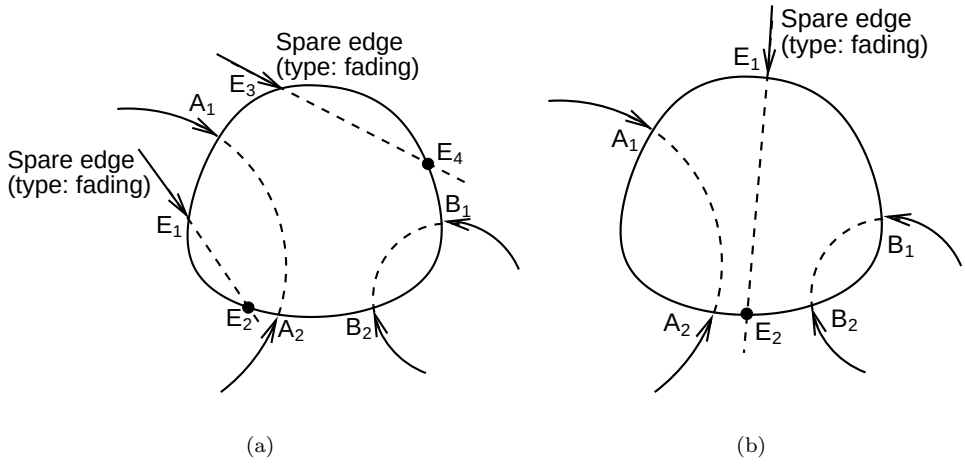


Figure 3.6: Reconstruction examples for fading spare edges.

3.4 Edge-Based Inpainting

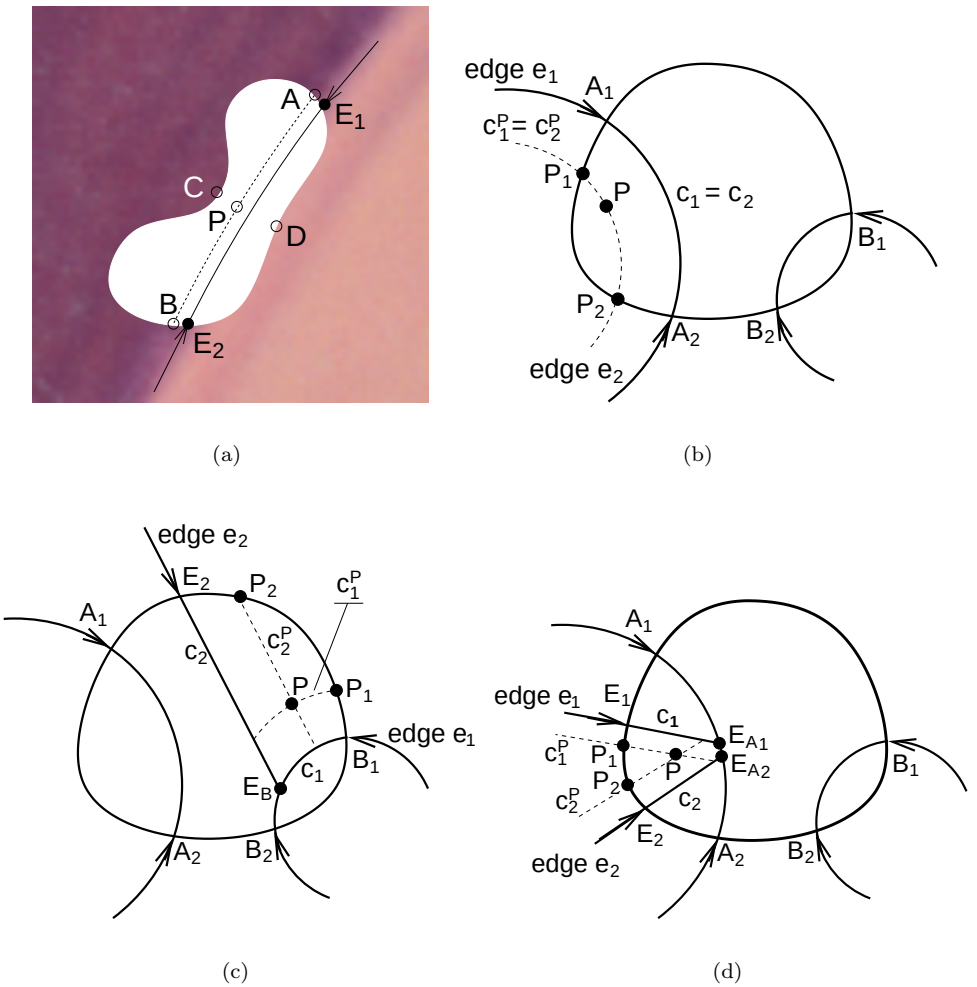
If the structure reconstruction step builds the "skeleton" of the missing areas, then we could say that the inpainting step adds the "flesh". During the inpainting process, the middle strips and the side strips will undergo different types of interpolation. In all cases, however, we rely on the finding that the image structure around an edge is usually "parallel" to that edge.

In the case where we have more edge groups (i.e. crossing edge couples as in Fig. 3.4(a)), we have to assume that one group lies in the front of the others. Since the information extracted so far provides no guidelines as to which one is in the front and which one in the background, the choice is made arbitrarily. Only groups consisting of a single edge couple (e.g. a horizon line) are "pushed" to the background, since their reconstruction in the foreground may cover entirely all other groups.

The following subsections describe our interpolation method, starting with the simplest case.

3.4.1 Inpainting of a Side Strip with Continuous Contour, Bounded Only by an Edge Couple

This is the simplest case of inpainting. We have a continuous contour and we know that there is an edge at each of the two ends of the contour (e.g. contour $A_1 - A_2$ in Fig. 3.7(b)). When the two edges form an edge couple, then a restoration "parallel"



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

Figure 3.7: a) Pixel similarity along edges: the value of P is closer to the values of A and B, rather than C or D. b-d) Inpainting of side strips with continuous contour. The side strips are bounded by: b) An edge couple; c) An edge couple and a spare edge; d) Two spare edges.

to the edges is (broadly speaking) equivalent to drawing circle arcs on both sides of the couple. These arcs are concentric with the couple's fitted circle, and span from one side of the artefact to the other (e.g. the $P_1 - P - P_2$ arc in Fig. 3.7(b)). A pixel along such an arc (e.g. P) is interpolated from the ending pixels of the arc (P_1 and P_2, in our case), which lie on the artefact border.

To understand the reason why we restore in this way, consider the example in Fig. 3.7(a). The missing area in region P is likely to be more similar to areas A and B, rather than C or D, although the last two are closer spatially. In fact, D and P are probably very different from each other, since they lie across edge couple $E_1 - E_2$, which means that they belong to two different objects.

Let us denote the circle fitted to edge $\boldsymbol{e}_k$ by c_k, $k \in \{1, 2\}$ (in Fig. 3.7(b), $c_1 = c_2$ because $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ belong to the same couple). The circle that passes through P, and is "parallel" to (i.e concentric with) c_k is denoted by c_k^P. It intersects with the artefact border at two points, $P_1^k = P_1$ and $P_2^k = P_2$. These two pixels are called the *source* pixels from which the intensity λ^P of pixel P is calculated as follows:

$$\lambda^P = \frac{w_1 \lambda^{P_1} + w_2 \lambda^{P_2}}{w_1 + w_2} \tag{3.11}$$

where w_i, $i \in \{1, 2\}$, is inversely proportional to the distance from P to P_i: $w_i = 1/d\left(\boldsymbol{x}^{P_i}, \boldsymbol{x}^P\right)$. λ^{P_i} represents the intensity of pixel P_i.

3.4.2 Inpainting of a Side Strip with Continuous Contour Bounded by an Edge Couple and a Spare Edge

This is a slightly more complicated case. As an example see Fig. 3.7(c). Now the side strip is not bounded by a single edge couple, but an edge that belongs to a couple, and one spare edge. To ease the discussion, the straight line fitted to the spare edge is also denoted as a circle. Like in the previous situation, we strive towards restoring the artefact as much as possible parallel to the edges. This means that, if we are close to one of the edges, for example, then we should interpolate from points on the artefact border close to the circle fitted to that edge. Again this is accomplished by defining two source pixels on the artefact border (one for each bounding edge) from which pixel P is interpolated, but now one of them is related to the spare edge and the other one to the edge couple.

In Fig. 3.7(c), the first source pixel P_1 is created from the edge $\boldsymbol{e}_1$ by intersecting c_1^P with the artefact border. The other source pixel is then found by intersecting c_2^P (based on the spare edge $\boldsymbol{e}_2$) with the artefact border. The intensity of pixel P is now calculated from the source pixels P_1 and P_2 according to:

$$\lambda^P = \frac{w_1^a \lambda^{P_1} + w_2^a \lambda^{P_2}}{w_1^a + w_2^a} \tag{3.12}$$

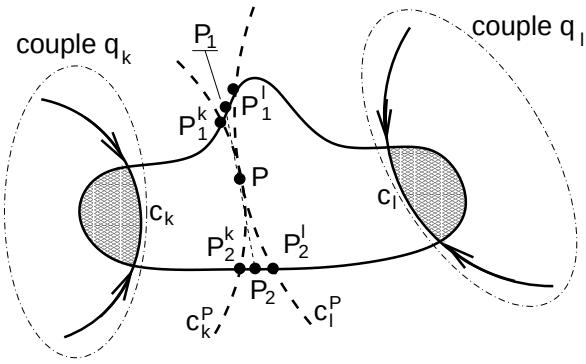


Figure 3.8: Inpainting of a middle strip with continuous contours

where w_i^a, $i \in \{1, 2\}$, is inversely proportional to *both* the distance from P to P_i as well as the distance from P to c_i:

$$w_i^a = \frac{1}{\max\left(d\left(\boldsymbol{x}^P, \boldsymbol{x}^{P_i}\right), \varepsilon^D\right) \times \max\left(d^c\left(\boldsymbol{x}^P, c_i\right), \varepsilon^D\right)} \tag{3.13}$$

where $\varepsilon^D = 0.01$ is used for protecting against potential divisions by zero, as well as for avoiding unusually large weights due to the proximity of pixel P to either of the circles c_1 and c_2. $d^c(\boldsymbol{x}, c)$ represents the euclidean distance between a pixel with coordinates $\boldsymbol{x}$ and the closest point on circle c. The weights w_i^a place more emphasis on source pixels close to pixel P. Also, in the immediate neighbourhood of a reconstructed edge, the source pixel that is close to that edge will dominate, thereby preserving the edge sharpness.

Notice that a side strip with continuous contour can also be formed by two spare edges, for example $E_1 - E_{A_1} - E_{A_2} - E_2$ in Fig. 3.7(d). Here, P_1 and P_2 are created in a similar way, by intersecting circles c_1^P and c_2^P (corresponding to spare edges $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, respectively) with the artefact border. The intensity of point P is then again estimated with the formulas in Eq. 3.12 and Eq. 3.13.

3.4.3 Inpainting of a Middle Strip with Continuous Contours

The next case of inpainting is a middle strip. In its simplest form, it is only bounded by two edge couples from the same group (see Fig. 3.8). Again, the interpolation is driven by the structure defined by the bounding edge couples.

Similarly to the side strip case, source pixels from the artefact border are calculated, upon which the interpolation is based. Since we now have two bounding edge couples, two sets of source pixels are created, $\left\{P_1^k, P_2^k\right\}$ and $\left\{P_1^l, P_2^l\right\}$, each based on one of

the two edge couples, $\boldsymbol{q}_k$ and $\boldsymbol{q}_l$ respectively. From the two source pixels that belong to the same part of the contour, $\left\{P_1^k, P_1^l\right\}$ and $\left\{P_2^k, P_2^l\right\}$, two virtual source pixels are created: P_1 and P_2. The position of such a virtual source pixel (see also Fig. 3.8) is defined by:

$$\boldsymbol{x}^{P_i} = \frac{w_1^a \boldsymbol{x}^{P_i^k} + w_2^a \boldsymbol{x}^{P_i^l}}{w_1^a + w_2^a}, \; i = 1, 2 \tag{3.14}$$

where w_i^a is defined as in Eq. 3.13.

The intensity of these virtual source pixels P_i are defined as:

$$\lambda^{P_i} = \frac{w_1^a \lambda^{P_i^k} + w_2^a \lambda^{P_i^l}}{w_1^a + w_2^a} \tag{3.15}$$

Based on the coordinates and intensities of the virtual source pixels P_1 and P_2, the intensity of pixel P can now be determined as follows:

$$\lambda^P = \frac{w_1^b \lambda^{P_1} + w_2^b \lambda^{P_2}}{w_1^b + w_2^b} \tag{3.16}$$

where w_i^b is inversely proportional to the distance from point P to the virtual source pixel P_i.

3.4.4 Other cases

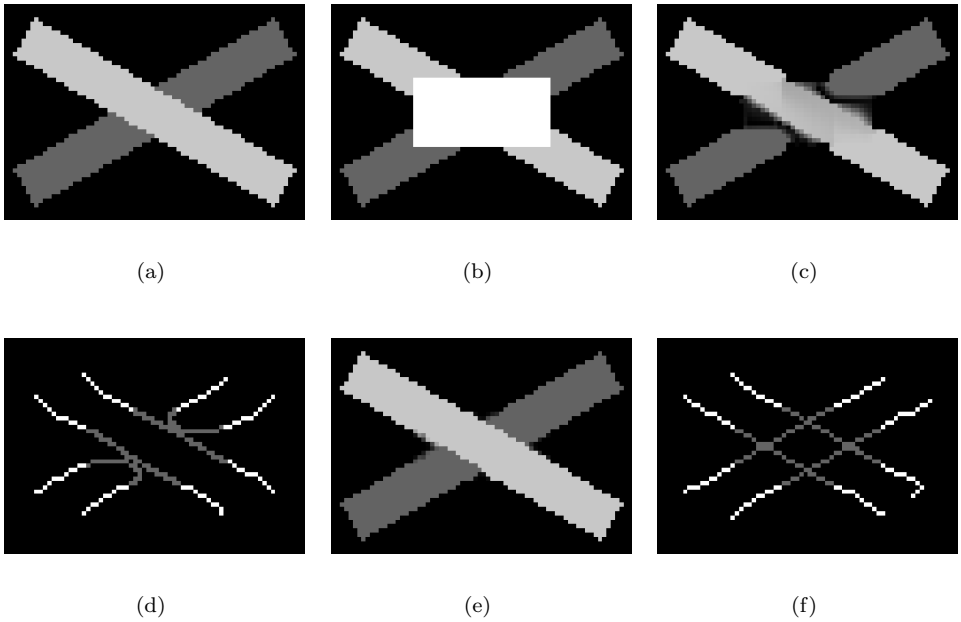
When a side strip or a middle strip has fragmented contours (e.g. Fig. 3.6(a)), they are interpolated similarly to the strips with continuous contours. However, in this case, a virtual source pixel is calculated for each fragment independently (e.g. one for fragment $A_1 - E_3$ and one for fragment $E_4 - B_1$ in Fig. 3.6(a)) and then the virtual source pixel of the entire fragmented contour $(A_1 - E_3; E_4 - B_1)$ is computed as a weighted average of its fragments' source pixels. The rest of the procedure is similar to the previous subsections.

If no edges are detected around the artefact, then the artefact lies probably in a smooth area. In such a case, the intensity of an artefact pixel P is simply the weighted average of the pixels on the artefact border. The weights are inversely proportional to the distance from P to the border pixels.

(a) (b) (c)

(d) (e) (f)

Figure 3.9: Restoration results. a) “Lena” image, degraded with artificial artefacts; b) Restored “Lena”; c) Zoom-in on the restored image; d) “Lena” image, artificially degraded over a T junction; e) Zoom-in on the original image; f) Zoom-in on the restored image.



(a) (b) (c)

(d) (e) (f)

Figure 3.10: Comparison on an artificial example with crossing structures. a) Original image; b) Degraded image; c) Restoration by the algorithm of Atzori and De Natale [2]; d) Structure recovered by the algorithm of Atzori and De Natale; e) Restoration by our proposed algorithm; f) Structure recovered by our algorithm;.

3.5 Results

3.5.1 Qualitative Evaluation

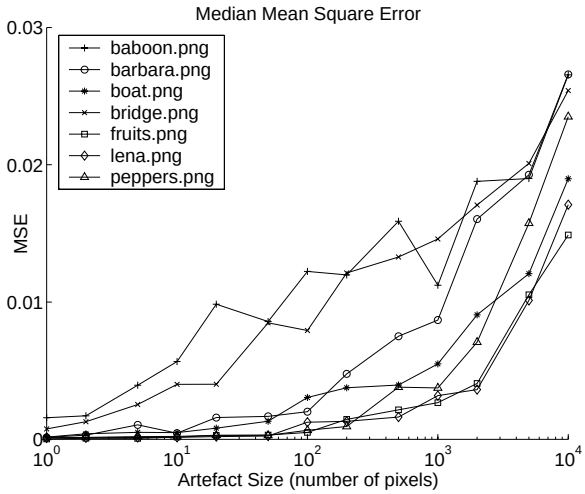
In this subsection the performance of our proposed algorithm is demonstrated by some visual examples. Fig. 3.9(a-c) shows an artificially degraded version of the "Lena" image, the restored version and a zoom-in on one of the artefact areas in the restored image (for every artefact, the restored structure consisted of a single group of coupled edges). Fig. 3.9(d-f) shows an example of interpolated spare edge (a T junction). Visual inspection of these results shows a good restoration quality. Both sharp and smooth edges are well recovered.

One of the strengths of our restoration scheme comes from its capability of finding and interpolating crossing structures. Fig. 3.10(e) shows such an example. Here, a group of two edge couples (the margins of the dark grey bar) is crossed by another group of two edge couples (the margins of the light grey bar). The restoration shows that the proposed algorithm is capable of reconstructing the correct configuration (Fig. 3.10(f)).

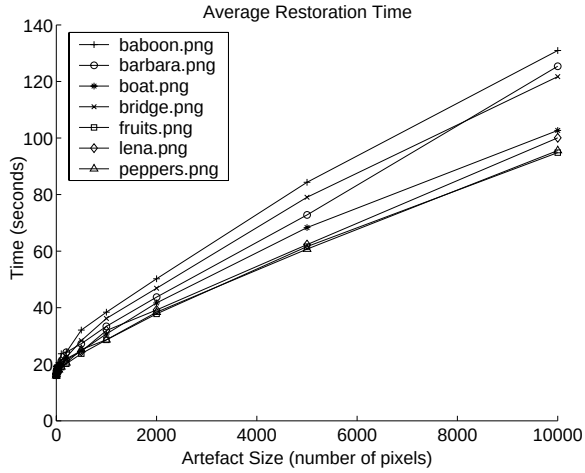
Obviously, our algorithm works well for objects which fit our assumptions. When edges are neither straight, nor circular (e.g. wiggly edges), the structure reconstruction will not be able to reproduce the initial image content. Also, when the structure becomes complex (e.g. in textured areas), the structure reconstruction step will fail, unless there is a dominant structure orientation (e.g. an image of straws). In these complex cases, the abundance of edges will make the algorithm more prone to errors than in usual cases. Similarly, if many of the edges detected around an artefact are spare edges, the structure reconstruction becomes a very difficult task. In such a case, the luminosity-related costs of most edge couples are cancelled by spare edges, so the final costs may become dependent on circle fitting and sequentiality costs only. Since less features are taken into account, the edge matching gets less reliable than in a normal case, so the probability of mismatches grows. Some edge couples may get treated as two spare edges, or they become coupled with wrong edges, while some spare edges may get erroneously assigned to couples. A thorough analysis of the reconstructed structure can only be done if a large database with manually segmented images would exist.

3.5.2 Quantitative Evaluation

Besides using visual inspection, we have also assessed the performance of the algorithm in a *quantitative* manner. A set of experiments was performed on a set of seven 512×512 images (see the name list in the legends of Fig. 3.11). These images were chosen because they exhibit some local structure. We have conducted the following series of experiments for each image. Artefacts with random shapes and locations were generated, having sizes of 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000,



(a)



(b)

Figure 3.11: Plots for the experiments done on the image test set. Each point represents the median result of 100 experiments done on the same image, with random artefacts of the same size. a) Median MSE, calculated on the grey-value images (grey range: $0 \ldots 1$). b) Average restoration time, under Matlab (interpreted code).

and 10000 pixels. For each size and each image, a single artefact was generated and restored in 100 consecutive experiments (each time with a different, random shape and location). For each restoration, the mean square error (MSE) was measured between the original and the reconstructed image. The MSE plots are shown in Fig. 3.11(a), with artefact sizes on a logarithmic scale. For each size and each image, the median MSE for the 100 experiments was plotted (this was chosen in order to avoid the influence of a small percentage of outliers). The MSE values stay within acceptable ranges, in general. A growing trend for bigger artefacts is present, as expected (the trend seems to accelerate at larger sizes because of the logarithmic scale used).

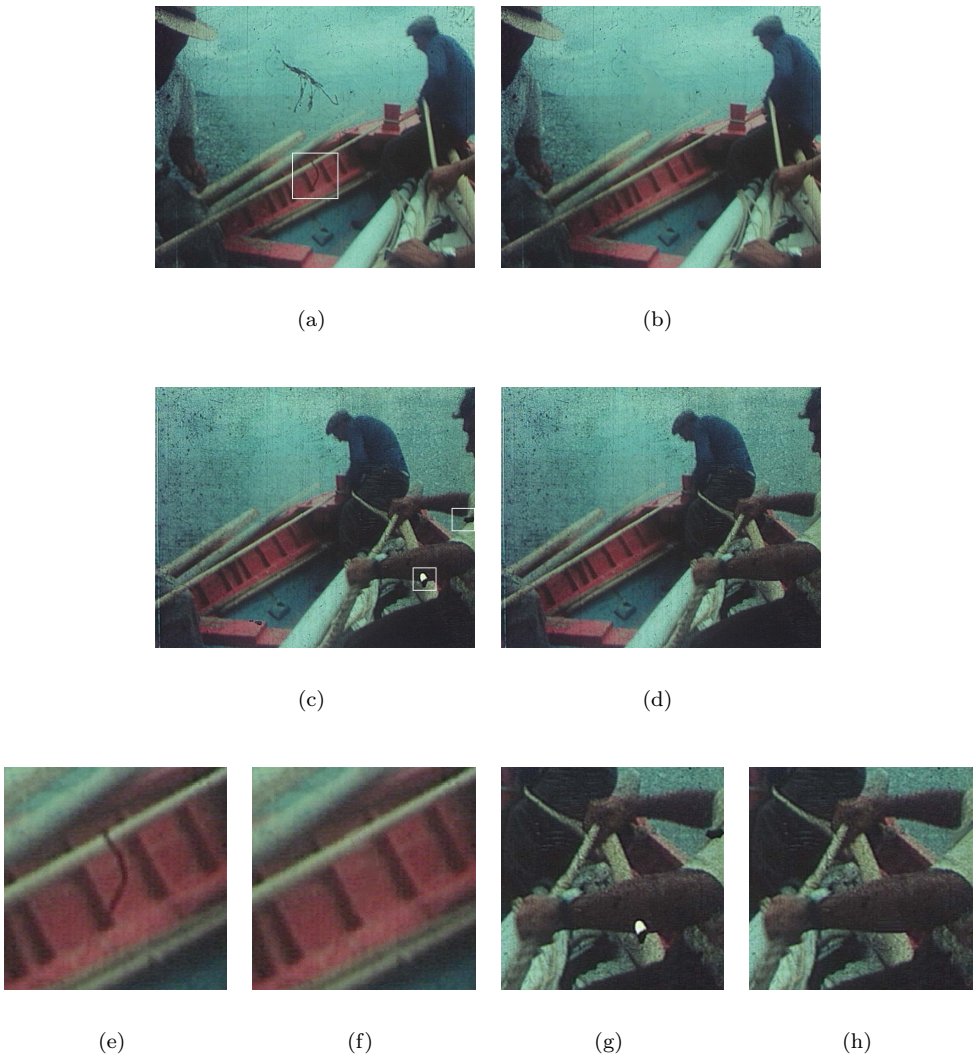
Additionally, the associated restoration times are displayed in Fig. 3.11(b). The artefact sizes are presented here on a linear scale, in order to show the almost linear dependency between the restoration time and the artefact size. The plot also shows a constant overhead, regardless of the artefact size. This overhead is related to the first part of the algorithm, in which object edges are detected, pixels on the artefact borders (together with the list of edges) are arranged in clockwise order, and edge features are computed.

From a perceptual point of view, our algorithm performed satisfactorily for MSE values up to about 0.005. Above this value, the quality of the restoration degraded in a more visible manner. This value is only a rough estimate and should not to be taken as an absolute reference, since the MSE is not strictly correlated with the visual quality. Depending on the textural content and the structural complexity of each image, the restoration errors may start becoming visible at smaller or larger MSE values and/or artefact sizes.

All experiments have been done with the same parameter setting. This showed that the parameter setting was not really sensitive to different images (i.e. different structure configurations), nor to different artefact shapes. Also, adding together costs with different variances did not seem to have a significantly negative impact on the quality of the restoration.

3.5.3 A Real Case Experiment

We also demonstrate the algorithm performance on a real case of degraded old film. Each row in Fig. 3.12 contains, from left to right: an original frame from a degraded film, and the same frame in which the main artefacts were subject to restoration using our algorithm (we concentrate only on those artefacts which cover areas containing structure and moving objects). White boxes are used in the original frames to mark artefacts of interest for our algorithm. These areas of interest are enlarged and displayed next to the full-size frames. The examples from Fig. 3.12 show that the algorithm performs equally well in real cases of degraded films.

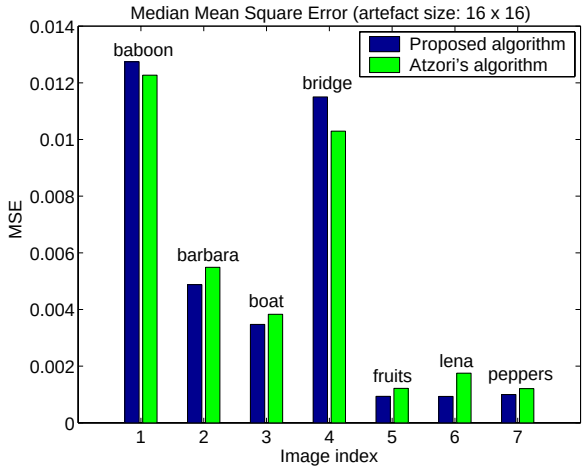


(a) (b)

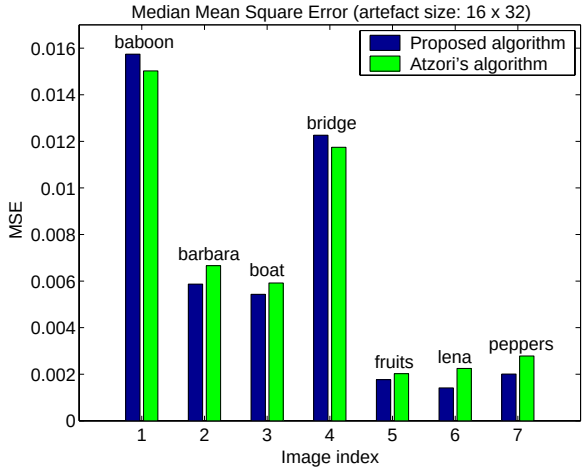
(c) (d)

(e) (f) (g) (h)

Figure 3.12: Real case example of film restoration. a,c) Original frames, with artefacts of interest surrounded by a white box; b,d) Same frames, with main artefacts restored; e,g) Zoom-in on the areas of interest in the original frames; f,h) Zoom-in on the areas of interest in the restored frames.

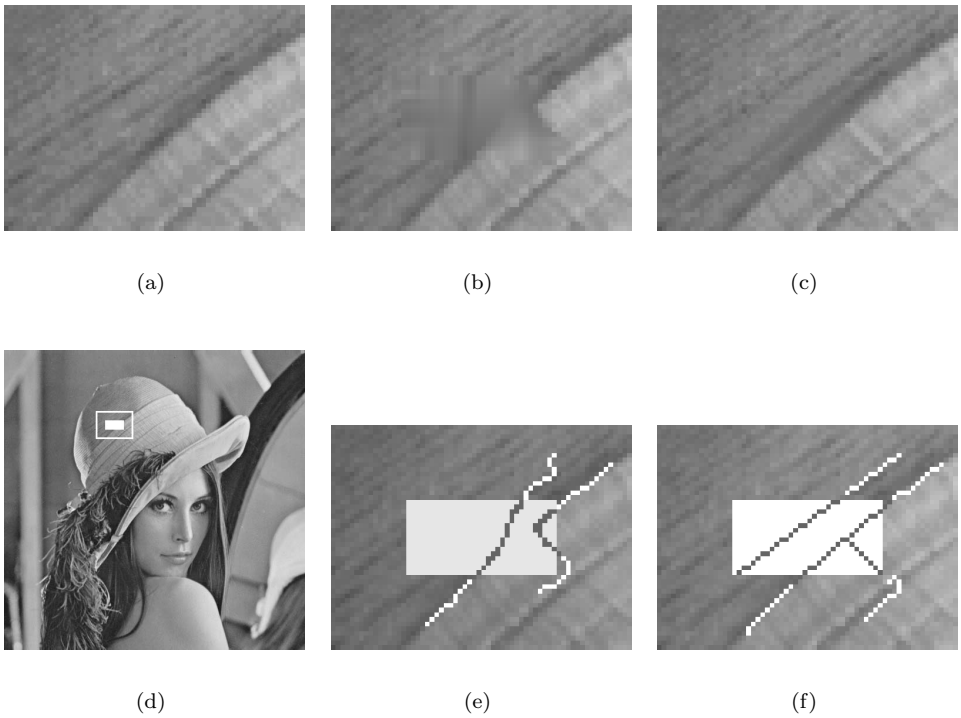


(a)



(b)

Figure 3.13: Comparison of the median MSE for the proposed algorithm (dark bars) and the algorithm of Atzori and De Natale [2] (light bars), for artefact sizes of a) 16×16 pixels, and b) 16×32 pixels.



(a) (b) (c)

(d) (e) (f)

Figure 3.14: Comparison with the algorithm of Atzori and De Natale [2]. a) Original image (zoomed in); b) Restoration by the algorithm of Atzori and De Natale; c) Restoration by our proposed algorithm; d) Full degraded image; e) Structure recovered by the algorithm of Atzori and De Natale; f) Structure recovered by our algorithm.

3.5.4 Comparisons with Other Algorithms

We have performed a comparison of our algorithm and the sketch-based interpolation of Atzori and De Natale [2]. For each of the seven images from our test set, we have generated artefacts with different sizes and random locations (1000 iterations for each size). For reasons of compatibility with the code we received from Atzori and De Natale, the artefacts were chosen to be only rectangular, having 16×16 or 16×32 pixels. In order to allow a proper comparison of both algorithms, the code of Atzori has been modified such that the input edges for both algorithms are the same, namely, the edges extracted in the first step of our algorithm.

The median MSE of all experiments for each image was measured for both algorithms. The comparison graph is displayed in Fig. 3.13. For both artefact sizes, our algorithm scored better in 5 out of the 7 images. The fact that both algorithms show larger MSE values for the highly textured images is an indication that the edge detection step had lower performances.

Visually, the restoration quality was not strikingly different for the two algorithms. This is not surprising, given the fact that the algorithms share some similarities. There are, however, more situations in which our algorithm outperforms the other one. Fig. 3.14 shows an example taken from our quantitative experiments. The circle fitting used in our algorithm enforced a more natural continuation of the edges, by connecting the upper-right edge with the lower-left one. Fig. 3.10 shows an artificial example of two bars crossing each other and an artefact covering their intersection. Our algorithm was able to detect and reconstruct the right image structure, while the algorithm of Atzori and De Natale failed. The fact that the input edge mask is not 100% the same comes from the fact that Atzori's algorithm considers the edges to start right from the artefact border, while we look at edges starting one pixel away from the artefact (thus the two algorithms never have *exactly* the same edge input). At times, our algorithm benefited from the use of the sequentiality. This global feature has contributed decisively in cases where several edge connections were equally possible.

Due to the type of interpolation used in the last step, our algorithm may sometimes produce smoother than normal areas. However, the patch repetition used by Atzori and De Natale (in this case a mirroring across the artefact border) may introduce its own type of defects, for example when another object lies close to the artefact. In this case, the patches that would be pasted would repeat the object (or parts of it) inside the artefact, although that object does not even touch the artefact. Patch repetition may go wrong in other cases, too. If a strip that presents a constant change of intensity is "interrupted" by an artefact, the mirroring process reverses the gradient direction in the artefact area, introducing a sudden change of intensity in the middle of the artefact. From the bar graphs presented in Fig. 3.13 it becomes clear that our algorithm performs better for piecewise smooth images, or moderately textured ones. For highly textured images the algorithm of Atzori and De Natale performs better, mainly due to their interpolation scheme based on patch repetition.

We have also performed a comparison with the algorithm of Masnou [28], shown in Fig. 3.15. The comparison was performed on the example presented in [28]. Both algorithms give good results, as expected. While the two methods may perform similarly in many cases, for overlapped structures or T junctions (as defined in our paper) our algorithm would outperform Masnou's algorithm, which cannot handle them properly.

3.6 Conclusions and Future Work

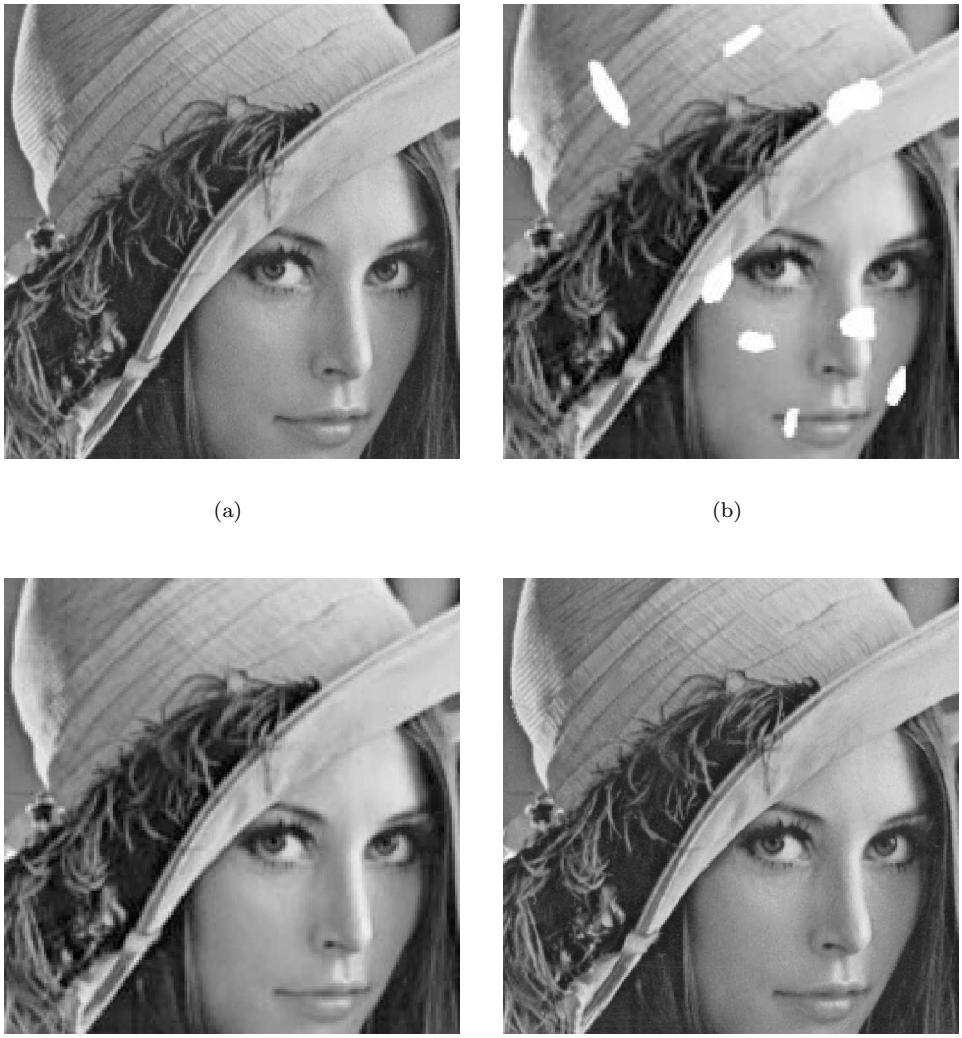
We have presented here an algorithm for the spatial restoration of images. Our goal is to restore frames from image sequences that exhibit "difficult" object motion, making the temporal restoration ineffective. The algorithm uses edge information extracted from the area surrounding the artefact. Based on this information, the missing structure inside the artefact (in the form of object borders) is reconstructed and then the areas between the reconstructed borders are filled by a smooth continuation of the surrounding image data.

The algorithm performs best with piecewise smooth images. In these cases, the restoration results are very good (both visually and numerically), as long as there is enough information around the artefact that is strongly related to the missing data. For highly textured images, the restoration is less effective because the image does not possess a certain "structure" - rather, it is a pattern with some degree of irregularity. In these cases, a texture interpolation method should be employed. This, however, would guarantee only a visually pleasing result, and not a lower error.

One of the main advantages of our method, is that it makes use of both local and global features of the edges in the image. The use of a global feature that validates the edge couples with respect to each other within the recovered structure is a new approach to image restoration. To our knowledge, this is the first algorithm which explicitly takes into account such a global feature, i.e. the sequentiality. The way the interpolation is done, along the reconstructed structures, is also new.

The validity of our structural model was demonstrated by evaluating the algorithm both visually and numerically on various images and across several artefact sizes. Moreover, the same set of parameters was used for all experiments, which demonstrates the robustness of our approach.

By reconstructing overlapped structures, our algorithm actually steps into the three-dimensional area, bringing one structure in front and pushing the others in the background. At this stage, these abilities are rather rudimentary. A superior analysis may certainly be added in the future to ensure the correct depth order of the structures. In any case, since the edge groups that cross each other may give us some depth information, applying the proposed grouping scheme could reveal object occlusions in undegraded images, provided that one can achieve a satisfactory segmentation of the image.



(a) (b)

(c) (d)

Figure 3.15: Comparison with the algorithm of Masnou [28]. a) Original image; b) Degraded image; c) Restoration with the algorithm of Masnou; d) Restoration with the proposed algorithm.

One of the implicit assumptions made in this paper is that the artefact masks do not have holes. Indeed, the overwhelming majority of artefacts from old films does not have holes. When they do have them, a few solutions could be applied. The simplest one is to simply consider that the artefact does not have holes, restore it in the way presented in our paper, and then paste the original content of the artefact holes back into the image (thus overriding a part of the restoration result). This, of course, neglects the structure that may be present inside the artefact holes, which might help guiding the structure reconstruction process. In some cases, this information may even be used to decide which group gets painted in in the foreground. Another solution would be to split the artefact mask conveniently such that no resulting submask contains any holes, and then proceed with the normal restoration algorithm.

There are several ways to improve the performances of our algorithm. First, it should be noted that the algorithm presented here uses only a 1 pixel-wide layer of pixels around the artefact. By increasing the amount of pixels taken into account, we expect to get more reliable edge features and useful neighbourhood information, which will improve the results in situations where the present algorithm has limited effectiveness.

Since the proposed algorithm works well with piecewise smooth images, rather than textured ones, whereas texture-based restoration shows opposite behaviour in general, we expect that the combination of the two approaches would improve the spatial restoration of images [10, 30]. Clearly, one needs to be able to decide which scheme to use depending on the surrounding area of the artefact. A special analysis module should be employed for this purpose.

Finally, a more sofisticated approach can be developed for the treatment of the available temporal information, along with the spatial information. Useful information can be extracted about the type of motion that causes the failure of motion estimation [31, 32], and then used to further enhance the results of the current algorithm. These subjects will constitute the focus of our future research.

3.7 Acknowledgements

This work was funded by the EU's IST research and technological development programme. It was carried out within the *BRAVA* project ("Broadcast Archives Restoration Through Video Analysis") [18].

The sequence used in the real case restoration (Fig. 3.12) is courtesy of RTP (Radiotelevisão Portuguesa).

We are grateful to Mr. Luigi Atzori for making available to us the code of his restoration algorithm.

Bibliography

[1] S.T. Acton, D.P. Mukherjee, J.P. Havlicek, A.C. Bovik, "Oriented texture completion by AM-FM reaction-diffusion", IEEE Transactions on Image Processing, vol. 10, no. 6, pp. 885-896, June 2001.

[2] L. Atzori, F.G.B. De Natale, "Error concealment in video transmission over packet networks by a sketch-based approach", Signal Processing: Image Communication, vol. 15, no. 1-2, pp. 57-76, September 1999.

[3] L. Atzori, F.G.B. De Natale, "Reconstruction of missing or occluded contour segments using Bezier interpolations", Signal Processing, vol. 80, no. 8, pp. 1691-1694, 2000.

[4] L. Atzori, F.G.B. De Natale, C. Perra, "A spatio-temporal concealment technique using boundary matching algorithm and mesh-based warping (BMA-MBW)", IEEE Transactions on Multimedia, vol. 3, no. 3, pp. 326-338, September 2001.

[5] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Transactions on Image Processing, vol. 10, no. 8, pp. 1200-1211, August 2001.

[6] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, G. Sapiro, "A variational model for filling-in gray and color images", Proceedings of ICCV 2001, vol. 1, pp. 10-16, Vancouver, BC, Canada, July 9-12, 2001.

[7] C. Ballester, V. Caselles, J. Verdera, "A variational model for disocclusion", Proceedings of IEEE ICIP 2003, vol. 3, pp. 677-680, Barcelona, Spain, September 2003.

[8] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image inpainting", Proceedings of ACM SIGGRAPH 2000, pp. 417-424, New Orleans, La, USA, July 2000.

[9] M. Bertalmio, A. Bertozzi, G. Sapiro, "Navier-Stokes, fluid-dynamics and image and video inpainting", Proceedings of IEEE CVPR 2001, vol. 1, pp. 355-362, Kauai, Hawaii, USA, December 2001.

[10] M. Bertalmio, L. Vese, G. Sapiro, S. Osher, "Simultaneous structure and texture image inpainting", IEEE Transactions on Image Processing, vol. 12, no. 8, pp. 882-889, August 2003.

[11] R. Bornard, E. Lecan, L. Laborelli, J.-H. Chenot, "Missing data correction in still images and image sequences", Proceedings of ACM Multimedia 2002, pp. 355-361, Juan Les Pins, France, December 1-6, 2002.

[12] T. Chan, J. Shen, "Mathematical models for local non-texture inpainting", SIAM Journal on Applied Mathematics, vol. 62, no. 3, pp. 1019-1043, 2001.

[13] T. Chan, J. Shen, "Non-texture inpainting by curvature-driven diffusions (CDD)", Journal of Visual Communication and Image Representation, vol. 12, no. 4, pp. 436-449, 2001.

[14] T. Chan, S. H. Kang, J. Shen, "Euler's elastica and curvature based inpainting", SIAM Journal on Applied Mathematics, vol. 63, no. 2, pp. 564-592, 2002.

[15] A. Criminisi, P. Pérez, K. Toyama, "Object removal by exemplar-based inpainting", Proceedings of IEEE CVPR 2003, vol. 2, pp. 721-728, Madison, Wis, USA, June 2003.

[16] A.A. Efros, T.K. Leung, "Texture synthesis by non-parametric sampling", Proceedings of ICCV 1999, vol. 2, pp. 1033-1038, Kerkyra, Corfu, Greece, September 1999.

[17] A.N. Hirani, T. Totsuka, "Combining frequency and spatial domain information for fast interactive image noise removal", Proceedings of ACM SIGGRAPH 1996, pp. 269–276, New Orleans, La, USA, August 1996.

[18] http://brava.ina.fr

[19] http://brava.ina.fr/brava_public_impairments_list.en.html

[20] J. Jia, C.-K. Tang, "Image repairing: robust image synthesis by adaptive ND tensor voting", Proceedings of IEEE CVPR 2003, vol. 1, pp. 643-650, Madison, Wis, USA, June 2003.

[21] L. Khriji, M. Gabbouj, G. Ramponi, E. Decencière Ferrandiere, "Old Movie Restoration Using Rational Spatial Interpolators", Proceedings of IEEE ICECS 1999, Paphos, Cyprus, September 5-8, 1999.

[22] H. Knutsson, C.-F. Westin, "Normalized and differential convolution: methods for interpolation and filtering of incomplete and uncertain data", Proceedings of IEEE CVPR 1993, pp. 515-523, New York, NY, USA, June 1993.

[23] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[24] A.C. Kokaram, "Parametric texture synthesis for filling holes in pictures", Proceedings of IEEE ICIP 2002, vol. 1, pp. 325-328, Rochester, NY, USA, September 22-25, 2002.

[25] A.C. Kokaram, "Practical MCMC for missing data treatment in degraded video", Proceedings of ECCV 2002 - Workshop on Statistical Methods in Video Processing, pp. 85-90, Copenhagen, Denmark, June 2002.

[26] A.C. Kokaram, S.J. Godsill, "MCMC for joint noise reduction and missing data treatment in degraded video", IEEE Transactions on Signal Processing, Special Issue on MCMC Methods, vol. 50, no. 2, pp. 189-205, February 2002.

[27] S. Masnou, J.-M. Morel, "Level-lines based disocclusion", Proceedings of IEEE ICIP 1998, vol. 3, pp. 259-263, Chicago, Ill, USA, October 1998.

[28] S. Masnou, "Disocclusion: a variational approach using level lines", IEEE Transactions on Image Processing, vol. 11, no. 2, pp. 68-76, February 2002.

[29] N. Nitzberg, D. Mumford, T. Shiota, "Filtering, Segmentation and Depth", Lecture Notes in Computer Science 662, Springer-Verlag, 1993.

[30] S. D. Rane, G. Sapiro, M. Bertalmio, "Structure and texture filling-in of missing image blocks in wireless transmission and compression applications", IEEE Transactions on Image Processing, vol. 12, no. 3, pp. 296-303, 2003.

[31] A. Rareş, M.J.T. Reinders, J. Biemond, "Statistical analysis of pathological motion areas", The 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 8/1-8/28, London, UK, January 16, 2001.

[32] A. Rareş, M.J.T. Reinders, J. Biemond, "Complex event classification in degraded image sequences", Proceedings of the IEEE ICIP 2001, vol. 1, pp. 253-256, Thessaloniki, Greece, October 8-10, 2001.

[33] A. Rareş, M.J.T. Reinders, J. Biemond, "Image sequence restoration in the presence of pathological motion and severe artifacts", Proceedings of IEEE ICASSP 2002, vol. 4, pp. 3365-3368, Orlando, Fla, USA, May 13-17, 2002.

[34] P.M.B. van Roosmalen, "High-level analysis of image sequences", technical report for INA - Paris (Institut National de l'Audiovisuel), France, the EU Aurora Project, July 1999.

[35] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

[36] M. Wertheimer, "Laws of Organization in Perceptual Forms", in W. Ellis, editor, "A Source Book on Gestalt Psychology", pp. 71-88, Routledge & Kegan Paul, London, UK, 1938. Translated from German, "Untersuchungen zur Lehre von der Gestalt II", in "Psycologische Forschung", 4, pp. 301-350, 1923.

Chapter 4

Constrained Texture Restoration

This chapter has been published as "Constrained Texture Restoration", by A. Rareş, M.J.T. Reinders and J. Biemond in *Eurasip Journal on Applied Signal Processing*, vol. 2005, no. 17, pp. 2758-2771, 2005.

A method is proposed for filling in missing areas of degraded images through explicit structure reconstruction, followed by texture synthesis. The structure being reconstructed represents meaningful edges from the image, which are traced inside the artefact. The structure reconstruction step relies on different properties of the edges touching the artefact and of the areas between them, in order to sketch the missing edges within the artefact area. The texture synthesis step is based on Markov Random Fields and is constrained by the traced edges in order to preserve both the shape and the appearance of the various regions in the image. The novelty of our contribution concerns the constraining of the texture synthesis, which proves to give results superior to the original texture synthesis alone, or to the smoothness-preserving structure-based restoration.

4.1 Introduction

This paper addresses the problem of image restoration, in situations where areas of the image are completely missing. This type of information loss takes place in old photographs, films, drawings, paintings, etc. [26, 33, 21]. The problem of information loss does not only affect old media. Even the most recent types of media are affected by similar problems. Packet losses during the transmission of streaming digital video (especially live broadcasts, or other broadcasts where no retransmission is possible) result in corrupted image areas, or even the complete loss of one or more consecutive frames [2]. The visual appearance of these errors can be quite disturbing.

The work presented here was carried out in the context of archived film restoration. Film restoration is usually carried out by means of temporal or spatiotemporal algorithms. However, when pathological motion occurs (i.e. when the objects in the image are difficult to track), the temporal restoration algorithms fail due to inaccurate motion vectors. Several solutions have been proposed [12, 31, 32], which discard the temporal information when pathological motion is detected, since this type of information is unreliable. This paper continues the aforementioned restoration efforts and represents a solution that also uses only spatial information during the restoration.

In spatial restoration, the missing image information has to be recovered using the remaining valid portions of the image. The approaches taken so far for solving the problem at hand can basically be classified in two categories: smoothness-preserving algorithms, and texture synthesis ones.

The algorithms belonging to the first category range from simple isophote connection [28] to PDE-based or variational inpainting models using isophotes, gradients, curvatures, etc. [6, 7, 8, 9, 10, 13, 14, 15, 29], normalized convolution [25], or inpainting guided by explicitly sketched edges [3, 4, 5, 32]. Although the term “inpainting” was mostly used in association with the aforementioned PDE-based or variational

methods, in the following we will use the terms “inpainting”, “restoration” and “interpolation” interchangeably.

The second category comprises a number of different approaches for texture synthesis. The algorithms presented in [18, 12, 16] are non-parametric, and are based on Markov Random Fields (MRF), while the algorithm presented in [27] is parametric, and is based on a Bayesian, *2D* autoregressive model. The approach presented in [1] uses partial differential equations and Gabor filters. And in [19], spatial and frequency information is combined in a framework of projection onto convex sets.

Smoothness-preserving algorithms are good at reconstructing piecewise flat (or relatively flat) areas, but they do not reconstruct texture satisfactorily. On the other hand, the texture synthesis methods tend to neglect the image structure. While the appearance of the restored texture may be pleasant, the object edges suffer sometimes deformations that may be visually disturbing. To overcome these drawbacks, some first algorithms for hybrid structure and texture interpolation are reported in [30, 11, 23, 24], in which various approaches for combined structure and texture interpolation are used. The approach presented in [30] by Rane et al. classifies complete 8×8 artefact blocks either as flat or texture areas, based on their surrounding uncorrupted blocks. The blocks classified as flat areas are restored using the smoothness-preserving inpainting method described in [9], while the textured blocks are restored using the texture synthesis algorithm described in [18]. In [11], Bertalmio et al. approach the problem differently, by decomposing the image into two completely overlapping images of the same size, one with no texture, and one containing only texture, whose sum represents the original image. These are then restored in parallel, using the same aforementioned algorithms for inpainting and texture synthesis, and then combined back into one image by simply adding them up. In [23, 24], Jia and Tang describe a novel technique based on tensors. Here, edge structure is first reconstructed, followed by texture synthesis. Both steps use adaptive tensor voting.

All aforementioned combining approaches have advantages and disadvantages. While the first one benefits from the distinction between different parts of the image, it is hampered by the fact that the structure/texture separation is done on a block basis. The second one benefits from the power of handling smooth texture changes (especially shadings), but it may fail when the structure (i.e. the prominent edges) resulting from the two parallel interpolations are different. The third approach benefits from the power of tensor-based texture reconstruction, but the edge pairing approach is rather basic.

We have opted in our approach for a hybrid, structure and texture restoration, in which different areas of the artefact are restored independently by means of texture synthesis. The structure reconstruction part of the algorithm is based on the one we proposed in [32], while for the texture restoration part we use a modified version of the algorithm described in [12]. The advantage of the new approach over these two methods lies in the fact that the algorithm described in [32] cannot handle textural

contents, while the algorithm described in [12] does not preserve edge smoothness generally and cannot handle overlapped structures. The proposed method is able to reproduce texture, while preserving the smoothness of object edges, and has the ability to deal with crossing structures, if needed.

In the algorithm that we propose, we separate different areas of the artefact and interpolate them independently, so we do not run the risk of "double edges", as in [11]. Moreover, our approach is superior to the one presented in [30] because it explicitly constructs the image skeleton (i.e. the object edges), and then uses it to *constrain* the texture restoration in a precise manner during the reconstruction step. The constrained texture synthesis approach is preferable to a simple one, since the latter does not use superior knowledge for preserving the object boundaries, thus being prone to "spilling texture" over the edges. Our method also has the advantage that it is not confined to square artefacts, and it is actually able to adaptively split them into smaller pieces, according to the recovered structure. The algorithm we propose also benefits from an edge connection scheme which is more elaborate than the one used in [23, 24], being thus capable of recovering more complex structures.

4.1.1 Algorithm Overview

The spatial restoration algorithm that we propose consists of three main steps, depicted in Fig. 4.1:

1. Segmentation and Feature Extraction;
2. Structure Reconstruction;
3. Texture Synthesis.

The input to our algorithm is an image and an artefact mask[1]. Without loss of generality, in the remainder of this paper we consider that the mask consists of only one artefact. The general assumption that we make is that there is enough redundant information in the input image, outside the artefact, which allows us to restore what is missing inside it.

In the first step (Section 4.2), a simple segmentation takes place which separates areas with distinct properties. Ideally, this results in a separate mask for each object. The second step of the algorithm (Section 4.3) builds the structure of the image inside the artefact. Hereto, we try to reconstruct the missing object boundaries inside the artefact (see the dashed lines in Fig. 4.1(d)) as continuations of the object boundaries from outside the artefact. The outer boundaries separate the areas segmented in the first step. Finally, in the third step (Section 4.4), the reconstructed structure is used to guide a texture synthesis procedure which fills in the artefact

[1] We assume that the artifact mask is detected by another algorithm.

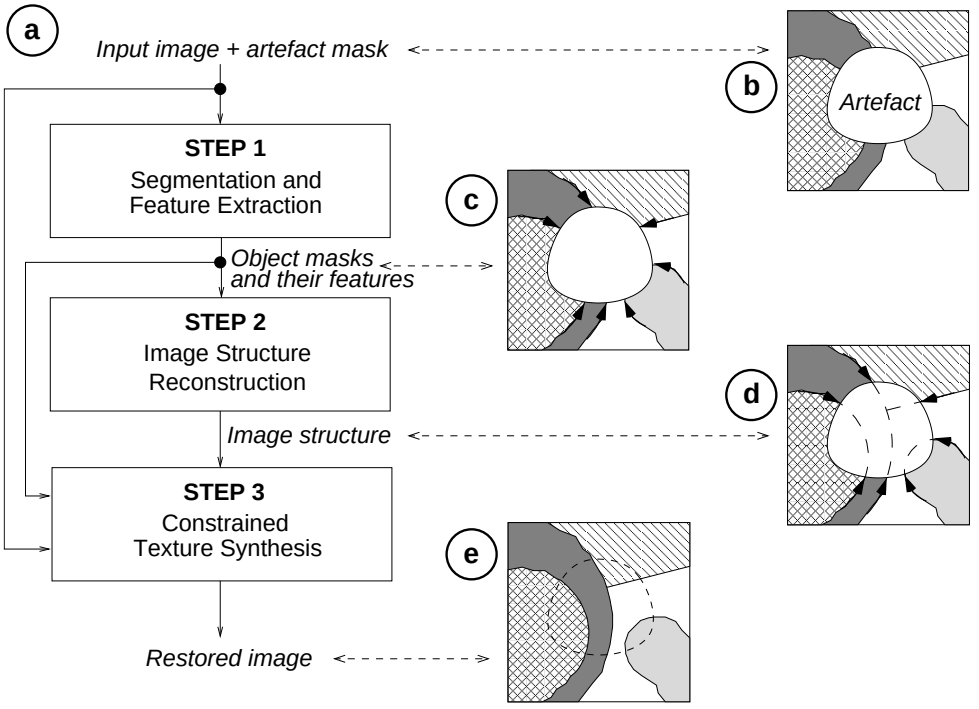


Figure 4.1: a) General algorithm outline (left) and an illustration of the inputs/outputs for each stage (right); b) Artefact mask; c) Detected object mask and extracted boundaries (thick arrows); d) Reconstructed structure inside the artefact (dashed lines); e) Restored artefact using constrained texture synthesis.

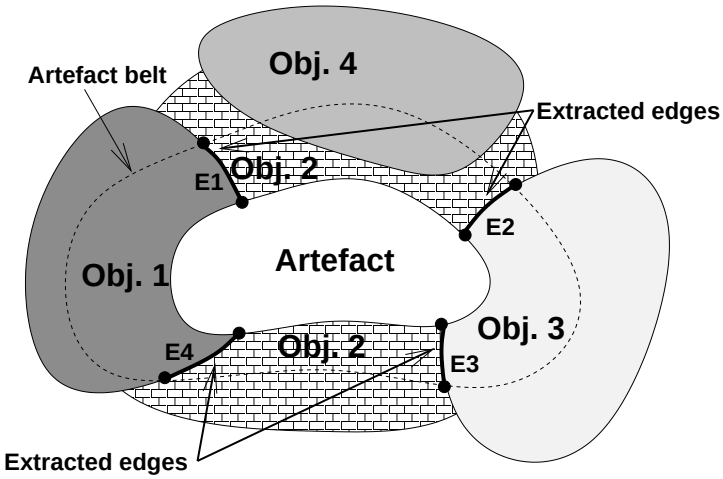


Figure 4.2: Objects lying close to the artefact may influence the feature values.

area. Here, a key aspect of our method contributes to its strength. Namely, the sampling neighbourhood for a certain (reconstructed) area inside the artefact is taken *only* from areas around the artefact that are belonging to the same object mask.

4.2 Segmentation and Extraction of Object Edge Features

The segmentation procedure is an essential step which allows us to identify distinct areas around the artefact. We have used the segmentation algorithm described in [17]. Ideally, we should be able to distinguish between all objects within an image. This is, however, not a trivial task, and no current algorithm is able to output a segmentation mask which is very similar to the ground truth (i.e. a manually segmented image), except in some restricted contexts. This is a limitation we have assumed for our current experiments. In the following, we consider that the segmentation mask we get is accurate enough.

Once the segmentation is performed, the edges which separate different objects are extracted. In addition, for every area between consecutive edges, a number of characteristic features are computed. These features will help us later to identify pairs of edges with similar properties (e.g. pairs $E_1 - E_4$ and $E_2 - E_3$ in Fig. 4.2). These pairs of edges originate from the same object border, after a part of it was occluded by the artefact. The segments are considered up to a certain distance from the artefact (20 pixels in our case), forming a *belt* around the artefact.

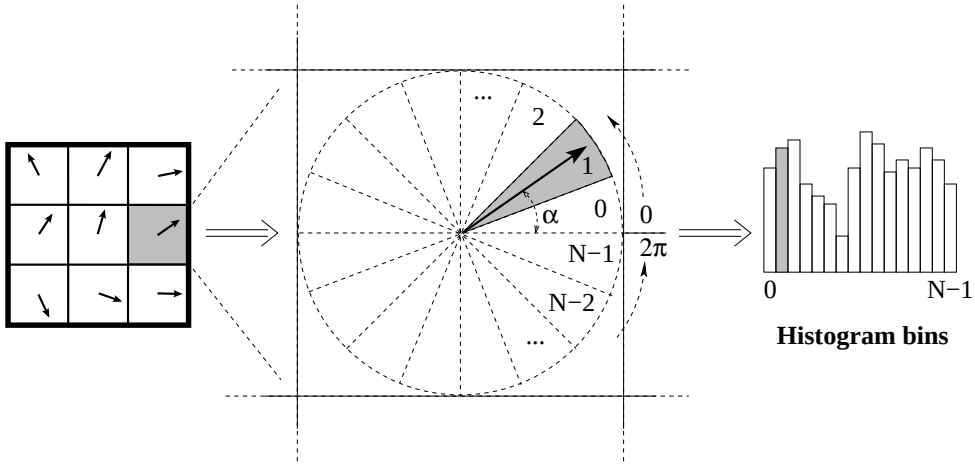


Figure 4.3: The gradient angle histogram is calculated from the gradient vectors. The angle interval $[0 \dots 2\pi]$ is divided into N "pie slices", and each gradient vector contributes with one unit to the bin corresponding to its angle.

The features that describe the segments are the intensity histogram, measured on the original image intensity, as well as the gradient angle histogram (in radians) within these segments, measured on the image gradient. The gradient angle histogram shows the dominant gradients and is calculated from the angle of each gradient vector, as shown in Fig. 4.3. The angle interval $[0 \dots 2\pi]$ is divided into N "pie slices", corresponding to the N bins of the histogram. Each gradient vector contributes with *one* unit to the bin that corresponds to its angle. Since each gradient vector contributes with only one unit to the histogram, the gradient magnitude information is discarded, making the histogram insensitive to contrast changes. One needs to pay attention to the value of N. If N is too small, the bins are too coarse, covering a wide angle range and giving rise later to false histogram matches. If N is too big, the histograms may become sparse, giving rise to false histogram mismatches. In our experiments, N was chosen to be 100. However, one should adapt this number to the situation at hand, since histogram sparsity may also arise if N is too big with respect to the numbers of pixels belonging to the segments.

The feature that describes the likelihood that two object edges belong to the same object boundary is calculated by measuring how well they fit the same circle. We rely here on the observation that most object edges have a rather constant curvature, locally.

The extracted histograms may get disturbed by various factors. For example, if an object was missed during the segmentation process and lies close to the artefact (such as object 4 in Fig. 4.2), then the histogram of the corresponding segment (e.g. the upper part of object 2) may display significant perturbations, resulting later

on in histogram mismatches. Other examples of feature perturbations include: the presence of shading along an object, the bending of textured objects (which affects the gradient angle histograms), etc. In all these cases, those parts of the segments (from the same objects) which lie close to the artefact will have histograms more similar to each other than those parts which lie further away from the artefact. For this reason, we calculated weighted histograms, instead of normal ones. As such, a pixel contributes to a histogram bin with a quantity given by a gaussian weight proportional with the distance to the closest artefact point (the standard deviation used is the belt width). In the end, the weighted histograms are normalized by the sum of the histogram bins. All these operations are applied in order to reduce histogram discrepancies between similar segments with different numbers of pixels, and to avoid the influence of non-informative areas.

At this point it is important to mention that we do not work with segments in our structure reconstruction step - rather, with the edges which separate them. Accordingly, each edge will be described by five features: four individual features and a shared one. The individual features are: the histograms of the intensity and gradient angles, both on the right and left side of the edge. The fifth, shared feature is the shape fitting cost.

Additionally, there is a sixth feature, *sequentiality*, which is not calculated only for edges (or edge pairs). Rather, it is calculated for the entire configuration of edge connections. For every potential configuration that we tentatively construct, this feature will measure the degree to which the edge connections in the configuration at hand lie in consecutive order. Alternatively, it may be seen as the degree to which the edge connections do not cross each other. Since this feature is only computed after pairs of edges have been connected to each other, it will be described later in this paper.

4.3 Image Structure Reconstruction

The structure reconstruction step is crucial to our proposed restoration scheme, since the explicit image structure that is recovered represents the "skeleton" of the restoration process. The input to this step represents a list of *edges* coming into the artefact, in clockwise order. The output of this step will be a list of *edge couples* arranged in *groups* of couples, and a list of *spare edges*. An *edge couple* is composed of two edges that are considered to represent the same object border. These two edges will be connected, in order to recover the object border inside the artefact. A *group* of couples is a collection of edge couples which lie in strict consecutive order, i.e. they do not cross each other, and no edge from a couple that does not belong to the current group lies between two couples of this group. The groups will be formed after the edge couples are constructed. The *spare edges* are edges for which there was no other edge that could be matched. Hence, it is assumed that a spare edge

represents a *T-junction* (e.g. the upper right edge in Fig. 4.1(d)), or a *fading edge* (an edge that gradually dissolves).

The procedure for reconstructing the image structure inside the artefact is based upon our approach presented in [32]. However, in the current approach, the feature description of the edges (in particular the histogram features) is different. We briefly describe here the procedure and the cost functions involved, but more details can be found in [32].

The basic idea is that we build several particular configurations (edge couples, groups, and spare edges), test how well they match the measured features, and then select the one which scores the best. The overall cost c^{cfg} of a particular configuration is computed from the five features shared by the two edges within each edge couple (c^{cpl}), and from the additional feature characterizing a global property of the configuration (c^{seq}):

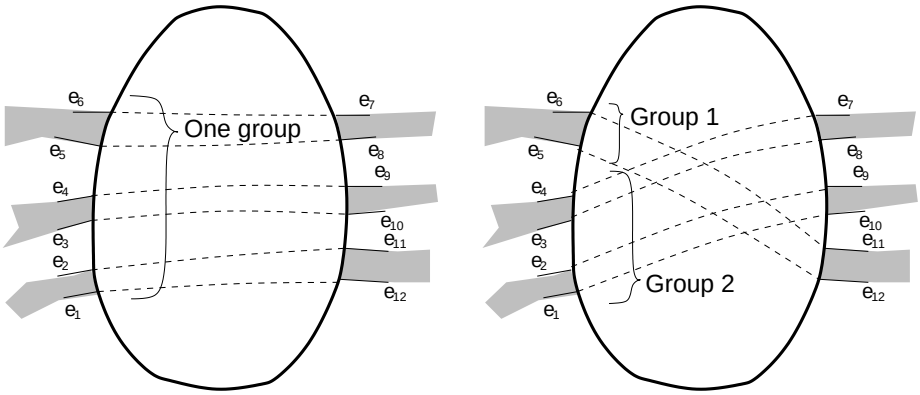
$$c^{cfg}(\boldsymbol{Z}) = \frac{5 \times c^{cpl}(\boldsymbol{Z}) + c^{seq}(\boldsymbol{Z})}{6} \tag{4.1}$$

where $\boldsymbol{Z}$ represents the configuration of (groups of) edge couples. c^{seq} is the cost associated with the *sequentiality* feature, defined later in this section. The cost c^{cpl} is defined in Eq. (4.2) and expresses how well the two edges in every couple match each other, i.e. how smoothly they can be continued into each other, and how similar are the regions they separate. All aforementioned costs (c^{cfg}, c^{cpl} and c^{seq}) have values between 0 and 1, with 0 indicating a perfect match and 1 indicating a complete mismatch. c^{cpl} contributes with 5/6 of the final configuration score c^{cfg} because it is computed from 5 features. For the same reason, c^{seq} contributes with 1/6 of c^{cfg}.

The couple-related cost is calculated as follows:

$$c^{cpl}(\boldsymbol{Z}) = \sum_i \sqrt{\frac{\beta_i^C \left(\chi_i^{\lambda,C} + \chi_i^{\gamma,C}\right) + \beta_i^A \left(\chi_i^{\lambda,A} + \chi_i^{\gamma,A}\right) + \omega_i^2}{2 * \beta_i^C + 2 * \beta_i^A + 1}} \tag{4.2}$$

where χ_i is the mean square error (*MSE*) operation applied on the difference of the histograms belonging to the two edges in edge couple $\boldsymbol{q}_i$. This histogram difference is a simple one-to-one subtraction of the arrays holding the histograms. Since all histograms are normalized to sum up to 1, the χ_i's always return values between 0 and 1. Each χ_i is calculated for a certain feature (λ for the histogram of intensities, and γ for the histogram of gradient angles) and on a certain side of the edge (C for the clockwise side, and A for the anticlockwise side, these sides being considered with respect to only one of the two edges in the couple). If the next edge on a certain side of a couple is a *spare edge* (see Fig. 4.1), or if it belongs to an edge couple which intersects the current couple (see Fig. 4.4), then the shared features on that side of the current couple become irrelevant. For this reason, we have used the binary flags $\beta_i \in \{0, 1\}$ to switch off cost contributions of the respective features, when needed.



(a) (b)

Figure 4.4: Contribution of the *sequentiality* parameter. a) Configuration given preference by the sequentiality parameter; b) Configuration penalised by the sequentiality parameter.

ω_i is the cost of fitting a certain shape to both edges of couple $\boldsymbol{q}_i$, and returns values between 0 and 1. We have chosen to fit circles because we assume that object boundaries have constant curvature, locally. ω_i is computed based on three measurements: the spatial deviation of the edge pixels from the fitted circle; the angular proximity of the points where the two edges touch the artefact (computed with respect to the center of the fitted circle); and a formula that checks the spatial order of edge pixels, making sure that the two edges stretch in opposite directions with respect to the artefact (see [32] for more details).

In both formulas from Eq. (4.1) and (4.2) we combined the feature costs through addition rather than multiplication for several reasons. First of all, in case of multiplication, a feature cost that is close to zero would cancel the contribution of all other feature costs. This is not a desirable behaviour, since these feature costs cannot describe any possible edge configuration. Rather, they are chosen to describe configurations that are encountered most often. In case of a valid configuration that is not properly modelled by one of our feature costs, the final cost should not be influenced too much by the faulty feature cost. Even when the edge configuration is properly modelled by the chosen features, the cost calculations may have imperfections (e.g. due to faulty segmentation masks). As such, some costs may again get lower than normal, which could strongly influence the final configuration score if multiplication was used instead of addition.

The sequentiality cost c^{seq} tries to measure a special property of object edges around the artefacts. Namely, for each edge that touches an artefact on one side, there is

usually a second edge on the other side, belonging to the same object. Moreover, consecutive objects (i.e. object margins) which are occluded by the same artefact give rise to consecutive, non-overlapping edge couples (see Fig. 4.4(a)). c^{seq} returns a cost of 0 when a maximum number of couples is formed with a minimum number of groups (i.e. one group), and a gradually increasing cost for less "sequential" configurations. The sequentiality feature is a very useful property of the edges around artefacts, since it is extremely robust against noisy data.

It is worth pointing out that the sequentiality parameter does not forbid a configuration containing crossing edges - rather, it penalises it. If the evidence coming from the other features strongly indicates a crossing, the edge couples are formed accordingly (resulting in a configuration such as the one in Fig. 4.4(b)).

Three problems arise when determining the sequentiality of a configuration. First, we must find a way to express it as a number. Secondly, despite the fact that it is used to calculate the configuration cost, we can measure it only *after* the configuration of edge couples has been formed. Thirdly, the sequentiality does not represent a measurement of each edge couple alone - rather, it is a measurement of the complete configuration, which is an ensemble of edge couples.

The sequentiality of a configuration $\boldsymbol{Z}$ was defined as follows:

$$c^{seq}(\boldsymbol{Z}) = \begin{cases} 1 - \frac{\sum_{i=1}^{N^G}(\|\boldsymbol{G}_i\|-1)}{\left\lfloor \frac{N^E}{2} \right\rfloor - 1} & , \left\lfloor \frac{N^E}{2} \right\rfloor \geq 2 \, and \, \sum_{i=1}^{N^G} \|\boldsymbol{G}_i\| \geq 1 \\ 0 & , \left\lfloor \frac{N^E}{2} \right\rfloor = 1 \, and \, \sum_{i=1}^{N^G} \|\boldsymbol{G}_i\| = 1 \\ 1 & , otherwise \end{cases} \tag{4.3}$$

where $\boldsymbol{G}_i$, $i = 1, \ldots, N^G$ are the groups of edge couples in configuration $\boldsymbol{Z}$, $\|\boldsymbol{G}_i\|$ is the number of edge couples in group $\boldsymbol{G}_i$, and $\left\lfloor \frac{N^E}{2} \right\rfloor$ represents the maximum number of edge couples that can be achieved out of the N^E edges detected around the artefact.

The algorithm for building the final configuration of groups of edge couples is described in pseudocode in Fig. 4.5. The computational complexity of this structure reconstruction step is (in a worst-case scenario) $O\left(2^n \times n^3\right)$, since, in step 3 from the pseudocode, trying every subset from the selected edges is $O\left(2^n\right)$, then building a different configuration starting from every edge is $O\left(n\right)$, and building every single configuration is $O\left(n^2\right)$ (adding couples to the current group is $O\left(n\right)$, and the step of checking couple crossing for each newly added couple is also $O\left(n\right)$, assuming that the amount of couples in the current subset is $O\left(n\right)$). Although the complexity of this step may seem high at a first glance, in reality, however, the computational demands can be reduced drasticly. We severely eliminate unlikely edge couples right from the beginning. As a result, only a fraction of the possible couples are left for each edge, in general. This reduces the search space for making subsets of couples, and for checking couple crossing. Additionally, it is not necessary to check all crossings. We

1. Calculate c^{cpl} for all possible edge pairs with Eq.(4.2) by setting all β flags to 1 (at this point we do not know which edges are spare ones).
2. Eliminate from further consideration those couples whose costs c^{cpl} are too big.
3. FOR each subset of couples from the remaining ones, and FOR each couple from the current subset, build potential configurations $\boldsymbol{Z}_j$ in a greedy fashion:
   (a) Start a new group with the current couple.
   (b) Incrementally add neighbouring couples to the group, on both sides of the initial couple, until one reaches a couple that intersects any couple from the current group.
   (c) Start another group with the intersecting couple and add again neighbouring couples from the remaining ones.
   (d) Repeat the above steps until no more couples are left.
   (e) All remaining edges are considered spare edges
4. Choose the configuration with the minimal cost: $\boldsymbol{Z} = \boldsymbol{Z}_k, \quad k = \arg\min_j c^{cfg}(\boldsymbol{Z}_j)$ (the β flags are now enabled according to the existing spare edges).

Figure 4.5: Pseudocode for the structure reconstruction procedure.

stop at the first crossing that is found. Finally, one should limit the maximal number of initial edges (to about 15) because generally only the main structure within the artefact area needs to be reconstructed.

To reconstruct the missing structure, all edge couples are traced inside the artefact according to the circle fitted to the couple. Finally, the spare edges are continued as straight lines within the artefact area (possibly stopping into an already reconstructed object boundary inside the artefact). Spare edges are traced as straight lines because circle fitting proved to be much less reliable for single edges, as opposed to edge couples. All contiguous areas lying between consecutive sketched edges, inside and outside the artefact, are distinctly labelled, resulting into disjoint masks of objects.

In the case where we have more groups of edge couples (i.e. crossing edge couples as in Fig. 4.4(b)), each group is reconstructed separately. We have to assume, however, that one group lies in front of the others. Since the information extracted so far provides no guidelines as to which one is in the front and which one in the background, this choice is made arbitrarily. Only groups consisting of a single edge couple (e.g. a horizon line) are "pushed" to the background, since their reconstruction in the foreground may obliterate all other groups.

4.4 Textural Inpainting

4.4.1 Texture Synthesis

The structure reconstruction step builds only a virtual sketch of the missing areas. The process of filling these areas must take care that object appearance inside the artefact area is the same as outside it. This will be achieved by means of texture synthesis, constrained by the current edge configuration.

We have chosen to apply the texture reconstruction algorithm of Bornard et al. presented in [12], modified as described in Section 4.4.2. This algorithm is able to reconstruct both regular and irregular textures. It stems from Efros and Leung's algorithm for texture reconstruction [18]. In contrast to Efros and Leung's algorithm, the algorithm of Bornard et al. is based on non-stationary Markov Random Fields (*MRF*), and imposes coherence constraints on the texture interpolation.

The underlying statistical model of the algorithm assumes that each pixel's probability distribution function (*PDF*) is independent of the rest of the image. The neighbourhood of the pixel is assumed - for the time being - to be a square window centered around the pixel which must be synthesized. The algorithm is non-parametric in the sense that the PDF is not imposed, nor constructed explicitly. Rather, it is approximated from a sample region of the image which has the same

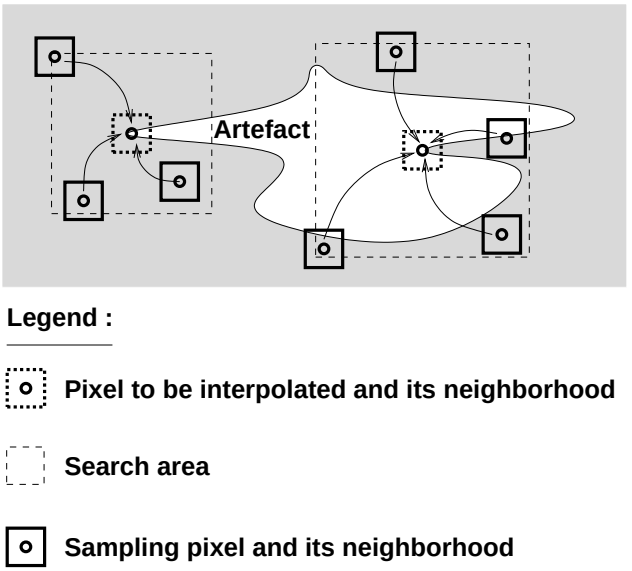


Figure 4.6: Search for candidates during texture synthesis.

size as the aforementioned pixel neighbourhood. It should be large enough to capture the texture characteristics, but not too large, or else we run the danger of falling out of the textured area.

The algorithm goes as follows. For each artefact pixel, a search is started in an area around that pixel (we used a search area size of 41×41 pixels in our experiments) in order to find valid pixels (non-artefact pixels, or pixels which were already interpolated) whose neighbourhoods are similar enough with the neighbourhood of the current artefact pixel (our choice for neighbourhood size was again 41×41 pixels). Out of the set of candidates that were found, one is randomly drawn and its value is pasted into the current artefact pixel (see Fig. 4.6). When computing the neighbourhood similarity, the L^2 norm is used. Since pixels in the closer vicinity are more relevant than pixels lying further away, gaussian weights are imposed upon calculating the norm. The artefact pixels that have not been synthesized yet and happen to lie in the neighbourhood windows are not taken into account for the norm calculation.

The aforementioned non-stationarity of the MRF is modelled by using an adaptive window size [12], rather than a fixed size (as used in Efros and Leung's approach [18]). Whenever there are less than a given number of valid pixels in the neighbourhood of the artefact pixel (1500 pixels in our case), the neighbourhood is enlarged until we have the requested number of pixels. As such, each pixel can have different conditional PDFs.

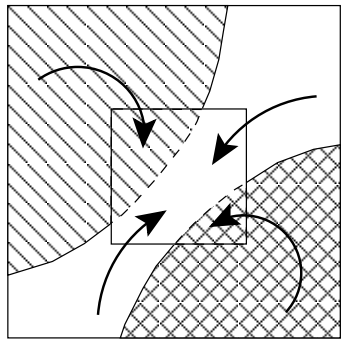


Figure 4.7: Key idea for the constrained texture synthesis: the artefact pixels are interpolated based only on pixels from the same object mask.

A second improvement introduced by Bornard et al. [12] speeds up the algorithm considerably without sacrificing the quality of the restoration. For this purpose, it imposes coherence constraints on the candidate search. If an artefact pixel to be synthesized has one or more neighbours that have already been synthesized, then the candidates used for restoring these neighbours are also used to generate candidates for the current pixel. Namely, the shifts between the current pixel and the previously synthesized neighbours are used to shift their respective candidates in order to generate the replacement candidates for the current pixel. From the generated candidates whose scores stay above a certain threshold, the best scoring one is selected for restoring the current pixel. If no shifted candidate has a score above the chosen threshold, a complete search is initiated as previously described. It should be noted that the coherence constraint can only be imposed after pixels from the outermost layer of the artefact have been already synthesized.

4.4.2 Integration of Structure and Texture Reconstruction

The mask created in the Structure Reconstruction step consists of distinctly labelled areas, corresponding to different objects (see Fig. 4.1(e)). Each object's mask consists of a part lying inside the artefact, and one lying outside it, namely in the *belt* area (see Fig. 4.2; for simplicity reasons we will not represent the belt in the following figures). We disregard those parts of the masks lying farther away than the belt. Since the texture of an area inside the artefact should resemble the texture of the *same* object outside the artefact, we can use the computed masks to constrain the texture synthesis scheme of Bornard et al. such that the sampling step takes place only in the belt area covered by the same object mask (see Fig. 4.7).

Nevertheless, when computing the difference between the neighbourhood of a pixel inside the artefact and the neighbourhood of a pixel from the corresponding belt

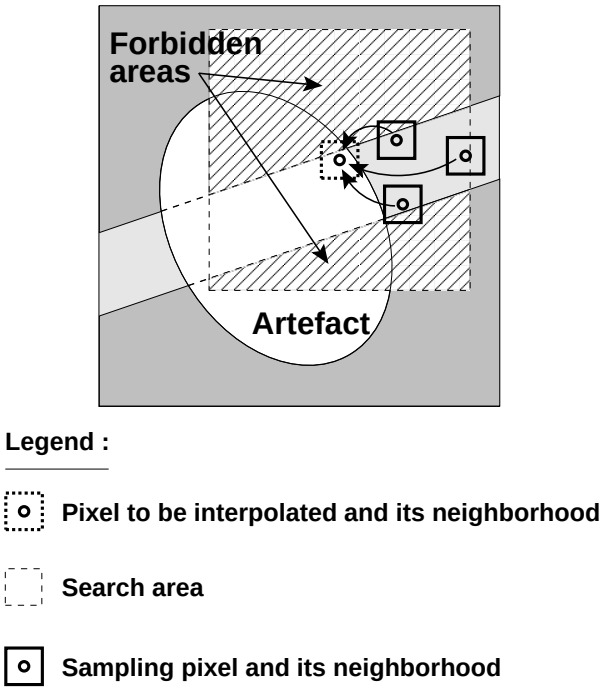


Figure 4.8: Texture interpolation near the object edges. Preference will be implicitly given for the sampling pixels lying next to the same edge (e.g. the top candidate).

subset, the neighbourhood window is allowed to fall outside the current object mask. In practice, for artefact pixels lying along object edges (especially straight edges), this gives priority to candidate pixels lying along the same edge, effectively reconstructing the effect of transition between two objects in an image (see Fig. 4.8).

4.5 Experiments and Results

This section describes the performance of our proposed restoration algorithm. In Figures 4.9-4.15 we demonstrate it by some visual examples of restoration based on the current algorithm, compared to the algorithms from [12] and [32]. All images have a size of 512×512 pixels, except the image from Fig. 4.15, which has a size of 200×200 pixels.

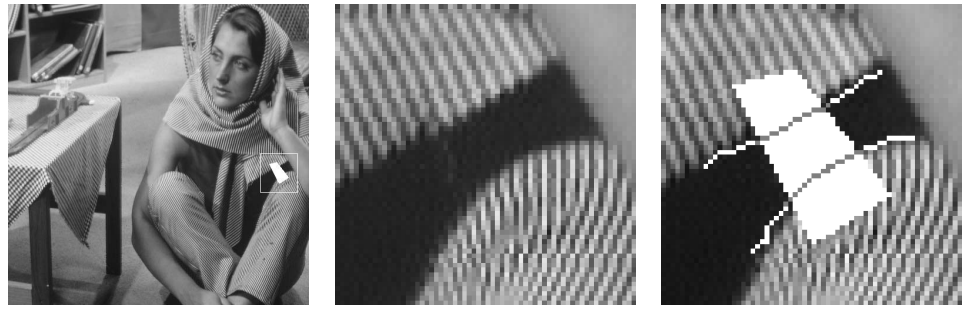
In general, the structure-based algorithm from [32] was affected by the failure of the edge detection in the textured areas, and by the smooth interpolation approach used therein. At times, the edge detection failure manifested itself not only as missed edges, but also as supplementary, fake edges, as was the case in Fig. 4.9(e). On its turn, the texture-based algorithm from [12] was not able to accurately render the object edges. Only the combined approach managed to preserve both object borders and their texture (see Fig. 4.9-4.14).

It is interesting to note that the two main reconstruction steps, the structure reconstruction and the texture synthesis, generally help each other, resulting in better restoration results than each of the steps separately. As such, a wrong texture interpolation (which tends to give irregular object borders) is "straightened" by the structure within which it must fit. Conversely, errors which take place sometimes in the structure reconstruction step may get corrected by the texture restoration algorithm, as shown in Fig. 4.11 and Fig. 4.12.

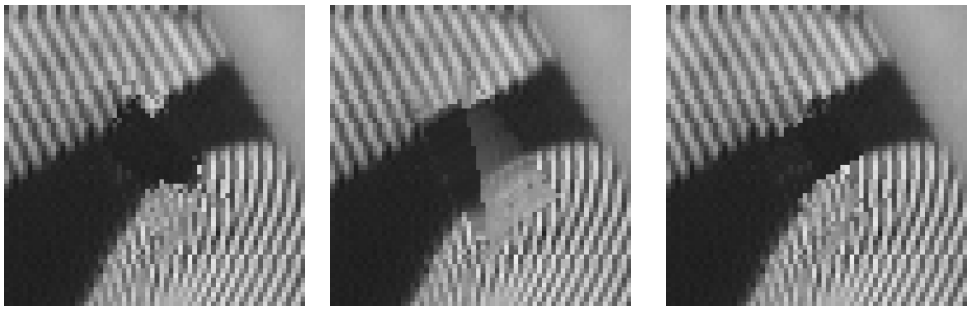
As opposed to the texture synthesis approach, the proposed method is able to handle situations where overlapping structures are involved. This is the case in Fig. 4.15, where two grey bars cross each other. The texture synthesis algorithm could only reconstruct one of the bars, while the other one looks "interrupted", since the reconstruction of the first one brought along some neighbourhood colour (the black background). The proposed method correctly reconstructed the overlapped bars. One can clearly see in Fig. 4.15(e) that two groups of edge couples were formed that cross each other.

4.6 Discussion and Conclusions

We have presented here a restoration algorithm which combines structure reconstruction and texture synthesis. Both algorithms make use of spatial information

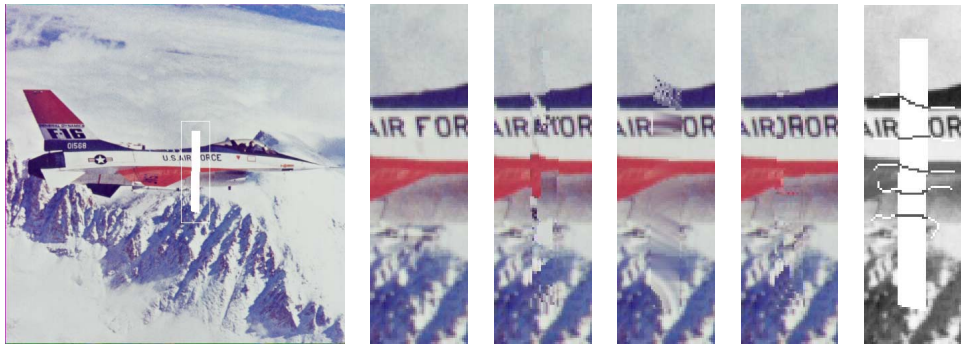


(a) (b) (c)



(d) (e) (f)

Figure 4.9: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Structure reconstructed in the proposed method; d) Texture-only restoration; e) Smooth restoration based on edges; f) Proposed method.



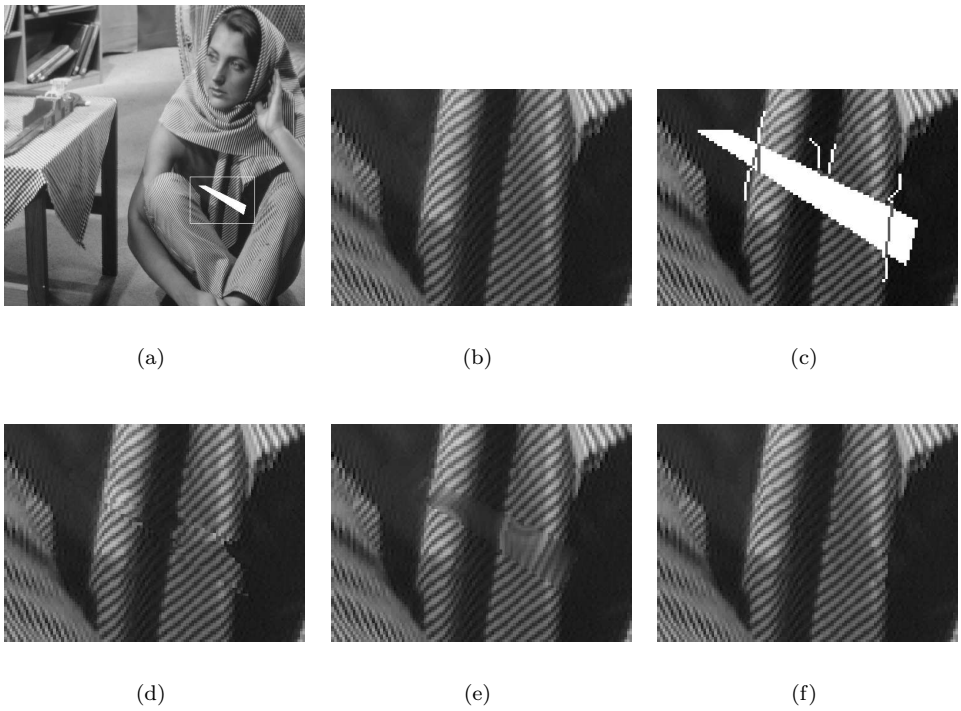
(a) (b) (c) (d) (e) (f)

Figure 4.10: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Texture-only restoration; d) Smooth restoration based on edges; e) Proposed method; f) Structure reconstructed in the proposed method.

only. Based on edge information coming from an object segmentation process, the structure reconstruction step recovers the structure of the image (the object borders) inside the artefact. The texture reconstruction step is then used to paint in the missing areas of the objects with their respective texture.

One of the main advantages of our method (over the texture synthesis method alone), is that the non-stationarity of the MRF is modelled not only by using an adaptive window size. It is actually modelled by the various object masks the texture restoration process is confined to. In this way, the actual sampling windows are shaped by the object masks, thus getting more accurate conditional PDFs for each pixel. Of course, questions may raise about the dependence on the object segmentation results. Indeed, at times, this may influence negatively the structure reconstruction step. In our opinion, however, an object segmentation is certainly preferable to a grid-like segmentation into square blocks that would be independent from the image content, followed by a "blind" texture synthesis that would sample indiscriminately from the entire neighbourhood, as is the case in [30].

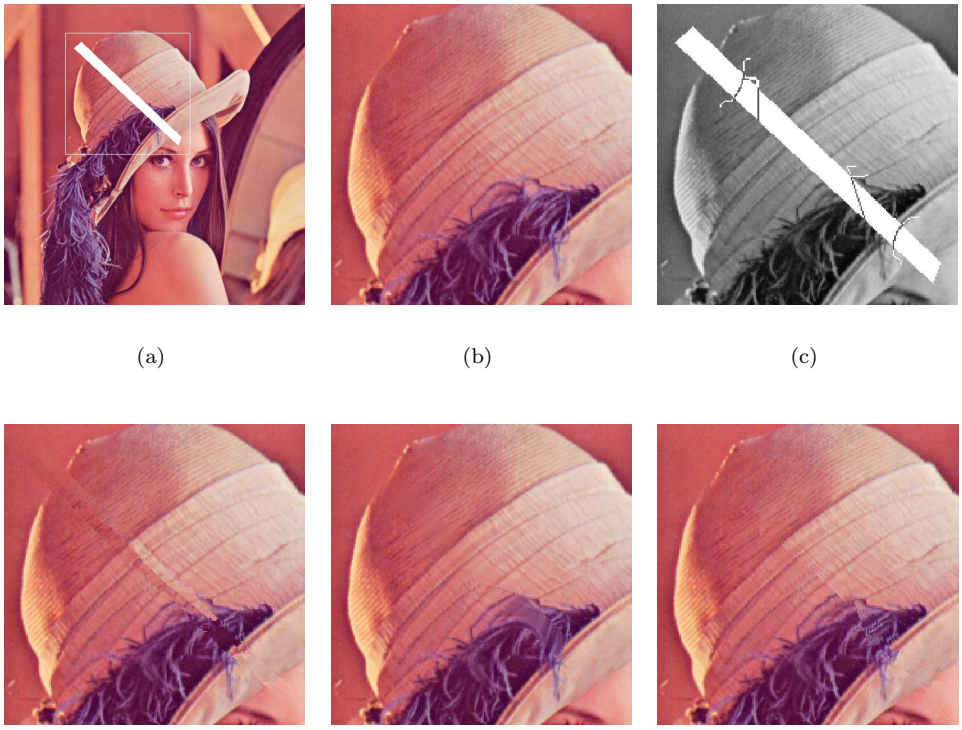
Despite the fact that the segmentation used is among the best available ones, the current state of the art in object segmentation is still far from perfect. Many relevant edges are still missed, and non-existent ones get erroneously detected. Although the texture synthesis process helps recovering from some of these cases, both types of errors may still affect the results negatively. Nevertheless, our proposed restoration algorithm showed its superiority even in such far from ideal conditions.



(a) (b) (c)

(d) (e) (f)

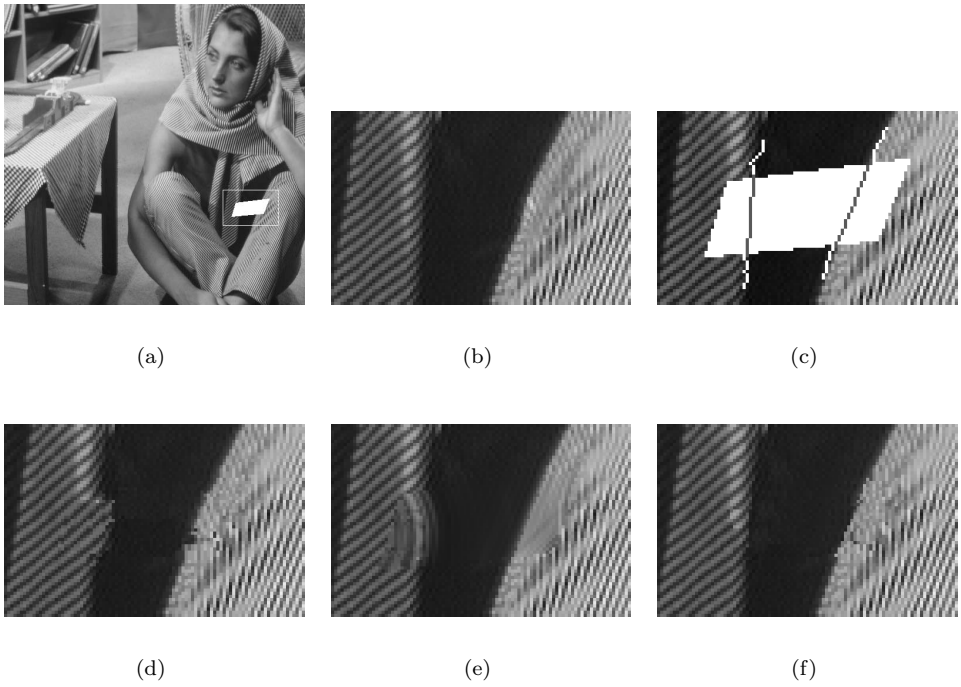
Figure 4.11: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Structure reconstructed in the proposed method; d) Texture-only restoration; e) Smooth restoration based on edges; f) Proposed method.



(a) (b) (c)

(d) (e) (f)

Figure 4.12: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Structure reconstructed in the proposed method; d) Texture-only restoration; e) Smooth restoration based on edges; f) Proposed method.



(a) (b) (c)

(d) (e) (f)

Figure 4.13: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Structure reconstructed in the proposed method; d) Texture-only restoration; e) Smooth restoration based on edges; f) Proposed method.

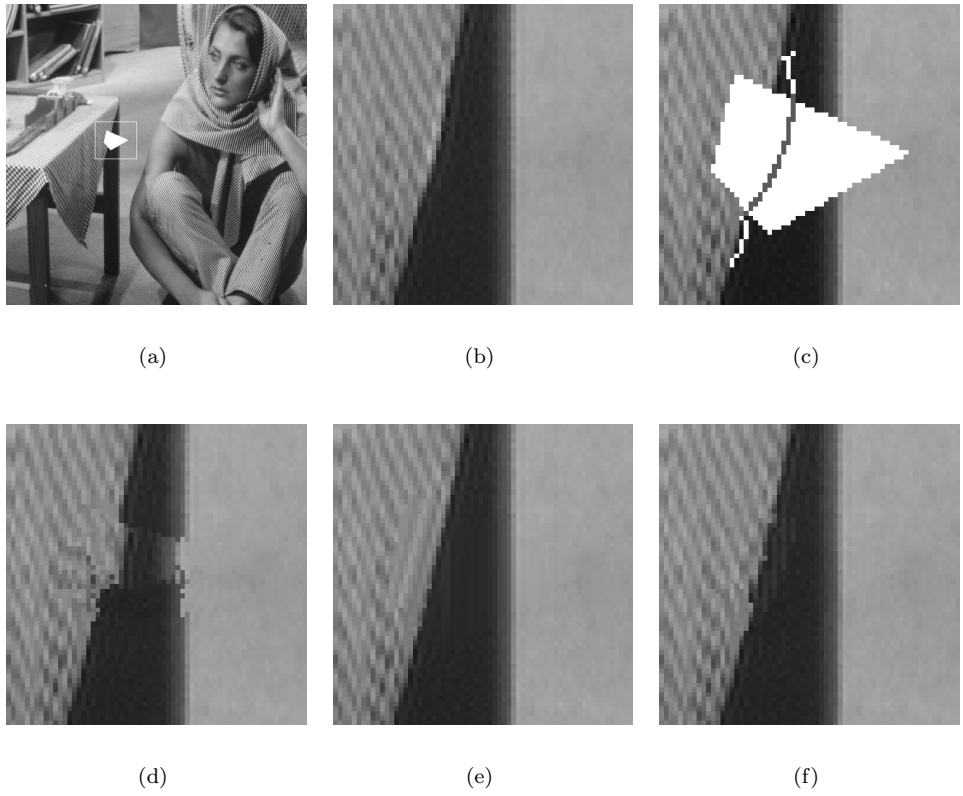
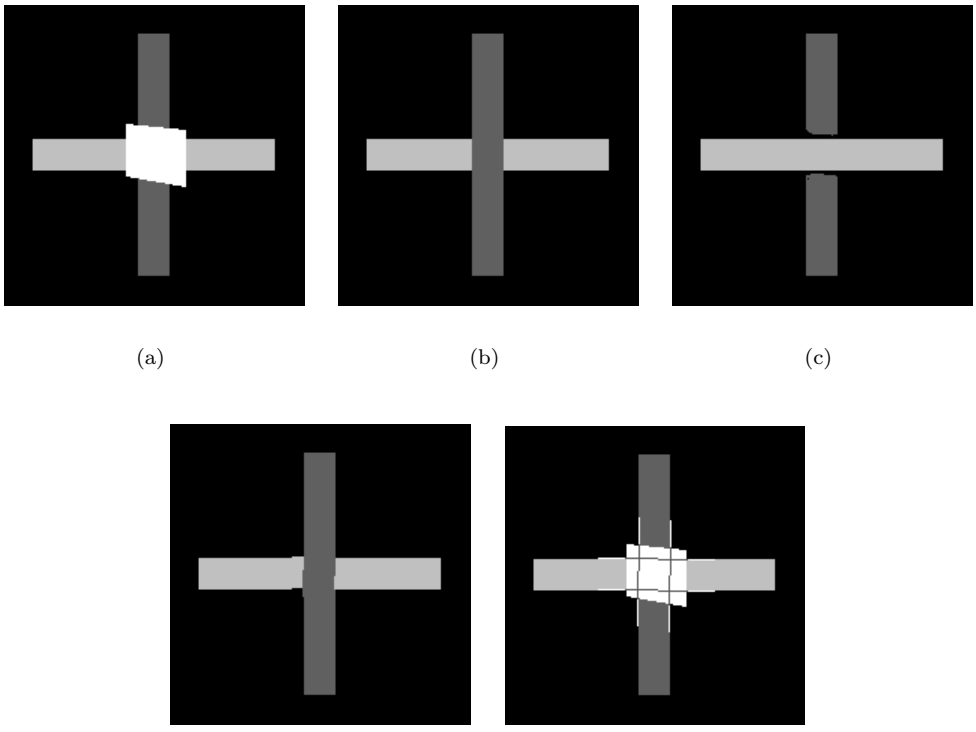


Figure 4.14: Comparison between different types of restoration (all images are zoom-ins on the artefact area, except the first one). a) Original image with artefact overlay; b) Original artefact contents; c) Structure reconstructed in the proposed method; d) Texture-only restoration; e) Smooth restoration based on edges; f) Proposed method.



(a) (b) (c)

(d) (e)

Figure 4.15: Example of crossing structures. a) Original image with artefact overlay; b) Original image contents; c) Texture-only restoration; d) Proposed method; e) Structure reconstructed in the proposed method.

The same holds for the texture synthesis algorithm. While the one we used is among the top ones, it may go (partially) wrong at times. For example, it does not handle shadings smoothly. When the opposite sides of an artefact belong to the same object and differ in intensity, a sudden luminosity change can be seen in the middle of the restored artefact area, where the iteratively synthesised sides meet each other. Imposing continuity constraints on the low frequency contents within each object mask will help overcome these problems.

The work presented here can be improved in several ways. One of these is by assigning local certainties to the object segmentation masks (and consequently to the locally extracted features). These certainties could then be used to avoid building structure in areas with low-certainty features, or to trigger a process for improving the segmentation masks.

The structure reconstruction step can be enhanced by a thorougher analysis of the shapes of the extracted edges and their coupling. For example, having a measure of the smoothness of transition between adjacent objects can contribute more information to the process of making edge couples.

One of the implicit assumptions made in this paper is that the artefact masks do not have holes. Indeed, the overwhelming majority of artefacts from old films does not have holes. When they do have them, a few solutions could be applied. The simplest one is to consider that the artefact does not have holes, restore it in the way presented in our paper, and then paste the original content of the artefact holes back into the image (thus overriding a part of the restoration result). This, of course, neglects the structure that may be present inside the artefact holes, which might help guiding the structure reconstruction process. In some cases, the information present in the artefact holes may even be used to decide which group gets painted in in the foreground. Another solution would be to split the artefact mask conveniently such that no resulting sub-mask contains any holes, and then proceed with the normal restoration algorithm.

Currently, in case that several groups have been formed, the order of inpainting is arbitrary, except for the groups consisting of one edge couple, which are pushed to the background. More clues may actually be used to decide the inpainting order. If depth information is available (e.g. from range sensors, or from the analysis of the object motion in neighbouring frames of an image sequence), this would help establishing the right inpainting order.

When tracing the object boundaries inside the artefacts, we made simple assumptions about their shapes, considering them to have constant curvature locally (i.e. being either circular or straight). These assumptions do not hold in all cases. However, it would be very hard and even hazardous to try to detect more complex shapes. The largest radical improvement of object boundary reconstruction can take place in the context of temporal restoration of image sequences. Here, tracking an object along several frames can give us precious information about the actual object shape.

The results of the algorithm we present here are very promising. They indicate that the current algorithm inherits the strengths of the two algorithms it combines, while avoiding (some of) their weaknesses. The algorithm's main strength comes from the fact that the structure reconstruction and the texture synthesis algorithms help each other to achieve better results. As such, the *constrained* texture synthesis that we propose shows robustness against errors of the modules which it is built upon.

4.7 Acknowledgements

This work was partly funded by the EU's IST research and technological development programme. Portions of it were carried out within the *Brava* project ("Broadcast Archives Restoration Through Video Analysis") [20].

We would like to thank the project partners from INA (Institut National de L'Audiovisuel) Paris, France, namely Raphaël Bornard, Emanuelle Lecan, Louis Laborelli, and Jean-Hugues Chenot, for providing the source code of their texture synthesis algorithm.

We would also like to thank the authors of the segmentation algorithm we used [17], Y. Deng and B.S. Manjunath, who made their code publicly available on the Internet [22].

Bibliography

[1] S.T. Acton, D.P. Mukherjee, J.P. Havlicek, A.C. Bovik, "Oriented texture completion by AM-FM reaction-diffusion", IEEE Transactions on Image Processing, vol. 10, no. 6, pp. 885-896, June 2001.

[2] M.E. Al-Mualla, C.N. Canagarajah, D.R. Bull, "Video Coding for Mobile Communications", Academic Press, Boston, Mass, USA, 2002.

[3] L. Atzori, F.G.B. De Natale, "Error concealment in video transmission over packet networks by a sketch-based approach", Signal Processing: Image Communication, vol. 15, no. 1-2, pp. 57-76, September 1999.

[4] L. Atzori, F.G.B. De Natale, "Reconstruction of missing or occludded contour segments using Bezier interpolations", Signal Processing, vol. 80, no. 8, pp. 1691-1694, 2000.

[5] L. Atzori, F.G.B. De Natale, C. Perra, "A spatio-temporal concealment technique using boundary matching algorithm and mesh-based warping (BMA-MBW)", IEEE Transactions on Multimedia, vol. 3, no. 3, pp. 326-338, September 2001.

[6] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Transactions on Image Processing, vol. 10, no. 8, pp. 1200-1211, August 2001.

[7] C. Ballester, V. Caselles, J. Verdera, M. Bertalmio, G. Sapiro, "A variational model for filling-in gray and color images", Proceedings of ICCV 2001, vol. 1, pp. 10-16, Vancouver, BC, Canada, July 9-12, 2001.

[8] C. Ballester, V. Caselles, J. Verdera, "A variational model for disocclusion", Proceedings of IEEE ICIP 2003, vol. 3, pp. 677-680, Barcelona, Spain, September 2003.

[9] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image inpainting", Proceedings of ACM SIGGRAPH 2000, pp. 417-424, New Orleans, La, USA, July 2000.

[10] M. Bertalmio, A. Bertozzi, G. Sapiro, "Navier-Stokes, fluid-dynamics and image and video inpainting", Proceedings of IEEE CVPR 2001, vol. 1, pp. 355-362, Kauai, Hawaii, USA, December 2001.

[11] M. Bertalmio, L. Vese, G. Sapiro, S. Osher, "Simultaneous structure and texture image inpainting", IEEE Transactions on Image Processing, vol. 12, no. 8, pp. 882-889, August 2003.

[12] R. Bornard, E. Lecan, L. Laborelli, J.-H. Chenot, "Missing data correction in still images and image sequences", Proceedings of ACM Multimedia 2002, pp. 355-361, Juan Les Pins, France, December 1-6, 2002.

[13] T. Chan, J. Shen, "Mathematical models for local non-texture inpainting", SIAM Journal on Applied Mathematics, vol. 62, no. 3, pp. 1019-1043, 2001.

[14] T. Chan, J. Shen, "Non-texture inpainting by curvature-driven diffusions (CDD)", Journal of Visual Communication and Image Representation, vol. 12, no. 4, pp. 436-449, 2001.

[15] T. Chan, S. H. Kang, J. Shen, "Euler's elastica and curvature based inpainting", SIAM Journal on Applied Mathematics, vol. 63, no. 2, pp. 564-592, 2002.

[16] A. Criminisi, P. Pérez, K. Toyama, "Object removal by exemplar-based inpainting", Proceedings of IEEE CVPR 2003, vol. 2, pp. 721-728, Madison, Wis, USA, June 2003.

[17] Y. Deng, B.S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 8, pp. 800-810, August 2001.

[18] A.A. Efros, T.K. Leung, "Texture synthesis by non-parametric sampling", Proceedings of ICCV 1999, vol. 2, pp. 1033-1038, Kerkyra, Corfu, Greece, September 1999.

[19] A.N. Hirani, T. Totsuka, "Combining frequency and spatial domain information for fast interactive image noise removal", Proceedings of ACM SIGGRAPH 1996, pp. 269–276, New Orleans, La, USA, August 1996.

[20] http://brava.ina.fr/

[21] http://brava.ina.fr/brava_public_impairments_list.en.html

[22] http://vision.ece.ucsb.edu/segmentation/jseg

[23] J. Jia, C.-K. Tang, "Image repairing: robust image synthesis by adaptive ND tensor voting", Proceedings of IEEE CVPR 2003, vol. 1, pp. 643-650, Madison, Wis, USA, June 2003.

[24] J. Jia, C.-K. Tang, "Inference of segmented color and texture description by tensor voting", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 6, pp. 771-786, June 2004.

[25] H. Knutsson, C.-F. Westin, "Normalized and differential convolution: methods for interpolation and filtering of incomplete and uncertain data", Proceedings of IEEE CVPR 1993, pp. 515-523, New York, NY, USA, June 1993.

[26] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[27] A.C. Kokaram, "Parametric texture synthesis for filling holes in pictures", Proceedings of IEEE ICIP 2002, vol. 1, pp. 325-328, Rochester, NY, USA, September 22-25, 2002.

[28] S. Masnou, J.-M. Morel, "Level-lines based disocclusion", Proceedings of IEEE ICIP 1998, vol. 3, pp. 259-263, Chicago, Ill, USA, October 1998.

[29] S. Masnou, "Disocclusion: a variational approach using level lines", IEEE Transactions on Image Processing, vol. 11, no. 2, pp. 68-76, February 2002.

[30] S. D. Rane, G. Sapiro, M. Bertalmio, "Structure and texture filling-in of missing image blocks in wireless transmission and compression applications", IEEE Transactions on Image Processing, vol. 12, no. 3, pp. 296-303, 2003.

[31] A. Rareş, M.J.T. Reinders, J. Biemond, "Image sequence restoration in the presence of pathological motion and severe artifacts", Proceedings of IEEE ICASSP 2002, vol. 4, pp. 3365-3368, Orlando, Fla, USA, May 13-17, 2002.

[32] A. Rareş, M.J.T. Reinders, J. Biemond, "Edge-based image restoration", IEEE Transactions on Image Processing, vol. 14, no. 10, pp. 1454-1468, October 2005.

[33] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

Chapter 5

A Spatiotemporal Image Sequence Restoration Algorithm

This chapter has been published as "A Spatiotemporal Image Sequence Restoration Algorithm", by A. Rareş, M.J.T. Reinders, J. Biemond and R.L. Lagendijk in the Proceedings of the *2002 IEEE International Conference on Image Processing*, vol. 2, pp. 857-860, Rochester, NY, USA, September 22-25, 2002.

Temporal restoration of image sequences may fail because of erroneously estimated motion. Motion estimation failures may occur for several reasons: complicated motion, severe artefacts, or a combination of both. We present here an algorithm that tries to overcome at least the problems related to the severe artefacts. It consists of a spatial restoration, which is supposed to recover the general image structure within the missing areas, followed by a temporal restoration. First performing spatial restoration has the advantage that the the motion estimation during temporal restoration can rely on the spatially restored frames. The efficiency of the spatial restoration scheme is demonstrated for both artificial and real life examples. Restoration results when using only temporal restoration are presented for the real life sequences. Finally, we compare both restoration methods with results of the proposed spatiotemporal algorithm which shows its superior behaviour.

5.1 Introduction

Film and video archives nowadays go through an accelerated process of degradation. Since the preservation of the cultural heritage plays an important role in our society, motion picture restoration has drawn a lot of attention lately. As a result, quite a number of digital restoration algorithms are now available [4, 9]. The success of current methods is, however, limited when there is complicated motion, which we call "*pathological motion*"(PM), or when there are severe artefacts, or both [6]. In such cases, the restored version may even look worse than the original. The main reason is that the temporal restoration (generally accepted as the most successful category of restoration algorithms) relies on motion estimation/compensation, which fails in the aforementioned situations. A trivial solution that prevents the restored frames from becoming worse than the originals is to detect areas where motion estimation is likely to fail and protect them against any restoration. Unfortunately, this also protects artefacts within these areas from being restored. Clearly this is not a desirable solution.

In this paper, we present a restoration scheme that is based on applying a spatial restoration, followed by a temporal restoration. The spatial restoration recovers the missing areas using only information from the same frame. Although this spatial restoration may not be perfect, it makes it possible to increase the accuracy of a subsequent motion estimation. Thus, when applying temporal restoration on the spatially restored frame, the quality of the restored areas can increase significantly. We should remark that in the case of PM the temporal restoration is not able improve the results since then the motion estimation still fails. In that case, we can only rely on the spatial restoration. Therefore, we need to make a distinction between model failure areas due to PM and those due to severe artefacts. In [7], we have introduced a scheme that is able to perform such a classification.

In Sections 5.2 and 5.3, we present the proposed spatial and temporal restoration schemes, respectively. Section 5.4 provides results achieved by combining the two

restoration schemes. Section 5.5 presents our conclusions and gives indications on future work.

5.2 Spatial Restoration

Several spatial restoration techniques have been proposed already, but many of them do not preserve high frequency components that define the image structure, so they are suitable only for artefacts inside relatively flat areas. Some recent research centred around the idea of "image inpainting" has shown promising results on restoring image structure for large patches [2, 3, 5]. All these methods interpolate the missing areas by connecting the isophotes that arrive at the artefact's border. This interconnection is achieved by minimizing a cost functional that takes into account parameters like gradient, isophote curvatures, etc. In [1], however, Atzori and De Natale, proceed along a different line. They recover missing blocks by first making a "sketch" of the edges around those blocks. These are continued inside the blocks, and used to interpolate the missing areas.

Our approach, first presented in [8], relates most to the sketch-based approach of Atzori and De Natale. We, however, detect straight incoming edges and use a directed interpolation scheme that restores missing areas parallel to the recovered edges. We prefer (together with Atzori and De Natale) to use gradient information to capture the image structure, instead of isophotes. Our main motivation comes from the observation that gradient information is more robust against intensity changes, such as local shading, noise, etc. The restoration scheme we propose consists of the following three steps: 1) straight edge detection, 2) edge matching, and 3) image reconstruction. These are highlighted in the following subsections.

5.2.1 Straight Edge Detection

To eliminate the influence of noise and other small artefacts, the image is first smoothed with a relatively large Gaussian kernel. The gradient magnitude is then calculated over the smoothed image. To avoid the influence of the artefact area on the gradient, the artefact masks are dilated with a factor that is proportional to the size of the smoothing kernel. Incoming edges are then detected at the border of the dilated artefact area. These detected edges are subsequently tracked outwards, following a path of local gradient magnitude maxima. Only those edges that are straight are kept for interpolation inside the artefact area. The selected incoming edges are described by: the edge angle θ_i, the gradient magnitude m_i, and the gradient direction φ_i.

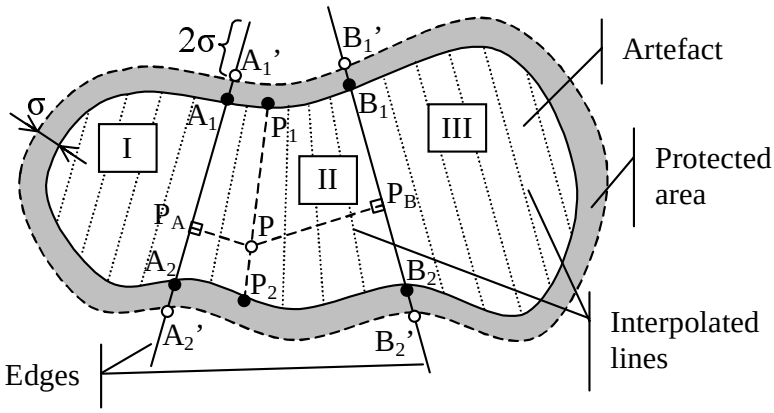


Figure 5.1: Spatial restoration. Dotted lines inside the artefact are linearly interpolated between their nearest ends.

5.2.2 Edge Matching

When there are several incoming edges, we need to match them pairwise in order to restore the missing connections between them. This is done by calculating the distances between all edges. Parallel, non-collinear lines will have their distance set by default to ∞. The inter-edge distance is calculated as an euclidean distance in the parameter space, where each term is normalized to an interval of [0,1]:

$$D\left(e_1, e_2\right) = \sqrt{\left(\frac{\theta_1 \setminus (\theta_2 + \pi)}{\pi}\right)^2 + \left(\frac{\varphi_1 \setminus \varphi_2}{\pi}\right)^2 + \left(\frac{m_1 - m_2}{grayrange}\right)^2} \tag{5.1}$$

where e_i denotes an edge, "\" is a modified "minus" operator that would return 0 as the difference between 0 and 2π and $grayrange$ is proportional to the gray scale.

5.2.3 Edge-Driven Inpainting

Considering all incoming edges being matched, Fig. 5.1 visualizes the actual restoration of the artefact area. Straight lines between each pair of matched edges (e.g. between $A_1^{'}$ and $A_2^{'}$) are drawn. The main assumption here is that edges can be approximated locally by straight lines. This results in several areas within the artefact (regions I to III). For regions bounded only by one pair of matched edges (such as regions I and III), we restore by interpolating lines parallel to it. The restored pixel intensities on these lines are then a linear interpolation from the intensities of

(a) (b)

(c) (d)

Figure 5.2: a) Original “Lena” image; b) Artefact mask, detected outside edges and reconstructed inside edges; c) Restored image; d) Zoom-in of a restored area.

the two pixels on the contour of the artefact area (e.g. A_1 and A_2). These values are sampled from the original images. Special care is taken for regions that are bounded by two restored edges (region II, in Fig. 5.1). Here, the angle of each interpolated line is also linearly interpolated between the angles of the neighbouring restored edges (see line $P_1 - P_2$). In the case where no incoming edges are detected, we simply use a smooth interpolation scheme based on a weighted sum of the pixels on the artefact border.

5.2.4 Spatial Restoration Results

First we show results of our proposed spatial restoration scheme for artificial artefacts introduced in the "Lena" image (see Fig. 5.2). The results are promising. Only a close look at the pixel level allows the identification of some traces of restoration. Close inspection of Fig. 5.2(b) shows that the right edge of the nose was estimated a few pixels off from the actual edge. However, Fig. 5.2(d) shows that this did not really influence the restoration, which provides evidence that our method is relatively robust to errors in the placement of the estimated edges. This robustness comes from the fact that in most of the cases, the image structure around the edges is parallel to them.

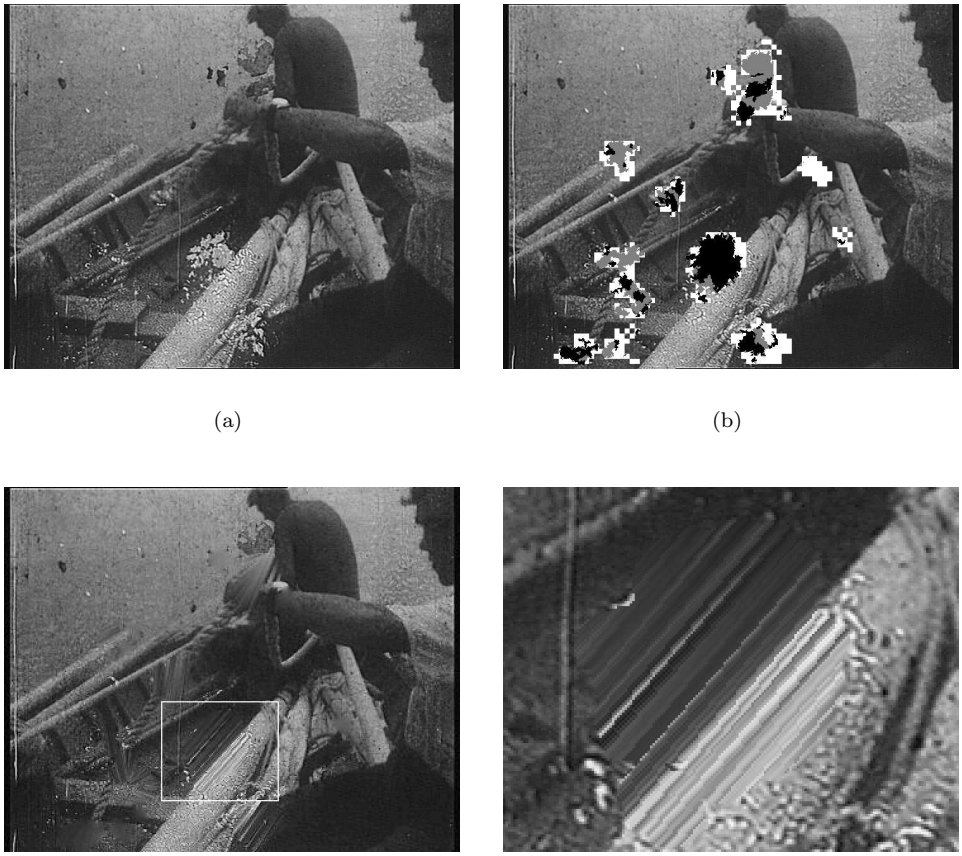
Fig. 5.3 shows the results achieved on a real life image that is affected by severe artefacts. The masks in Fig. 5.3(b) were computed according to the algorithms presented earlier in [7]. Most of the image structure has been recovered. The reconstruction is not as accurate as in the "Lena" case, mainly because virtually all image pixels are degraded. This makes the restoration rely on noisy pixels, resulting in the lining effect seen in Fig. 5.3(d).

From the examples presented above, we may conclude that the spatial restoration described here is most suitable for piecewise smooth images. However, severely degraded films are generally not smooth. In such cases, the spatial restoration itself may still be affected by degradations, and further processing is certainly needed.

5.3 Temporal Restoration

This section describes the temporal restoration technique used in our experiments. The results presented in this section are obtained by applying the algorithm on the original frames (i.e. without any previous spatial restoration).

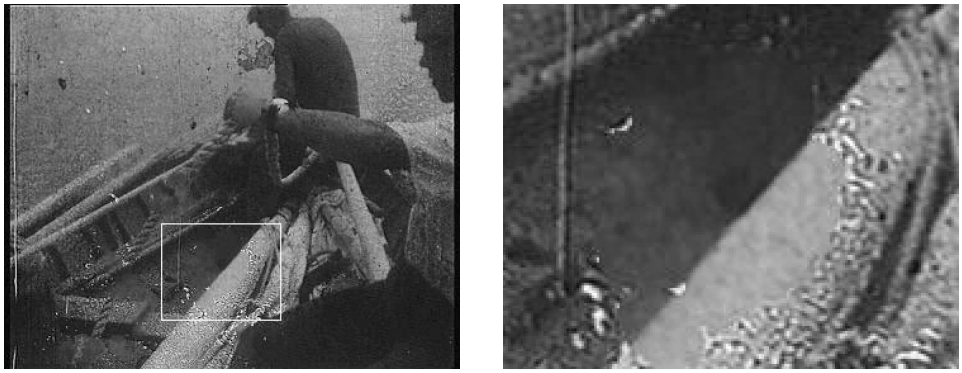
For a successful temporal restoration, one should compensate for the motion in the considered frames, in order to (relatively) align the objects in each frame before interpolation takes place [4]. Since the current image is corrupted, we must estimate the motion between the *previous* frame and the *next* frame. Then, assuming a linear uniform movement, the objects in the *current* frame should lie halfway along the



(a) (b)

(c) (d)

Figure 5.3: Spatial restoration of a real life example. a) Original image; b) Original image with a layout of complex event areas (in white), pathological motion areas (in gray) and artefacts (in black); c) Restored version; d) Zoom-in of a restored artefact. Sequence courtesy of RTP (Radiotelevisão Portuguesa).



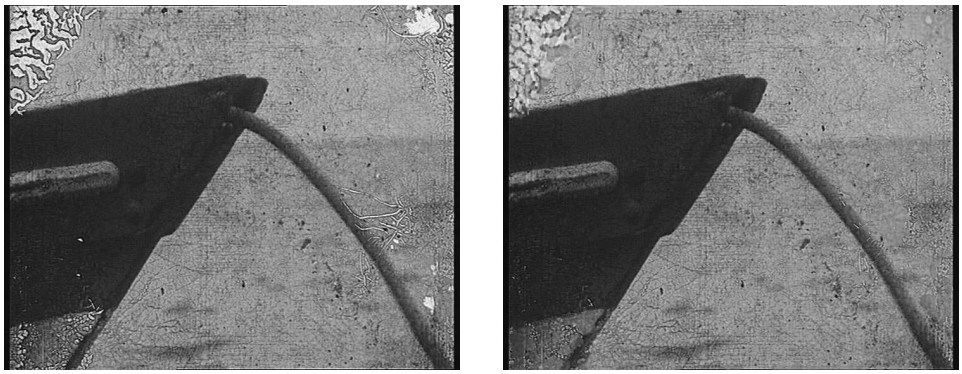
(a) (b)

Figure 5.4: Temporal restoration of the same real life example. a) Restored image; b) Zoom-in of a restored artefact.

computed motion vectors. We have used a motion estimator based on phase correlation. For restoration, we have performed a median filtering of a vector obtained by concatenating the pixels in a 3x3 neighbourhood in the (motion-compensated) *previous* and *next* frames. Fig. 5.4 displays results of the described temporal restoration scheme for the same example as in Fig. 5.3.

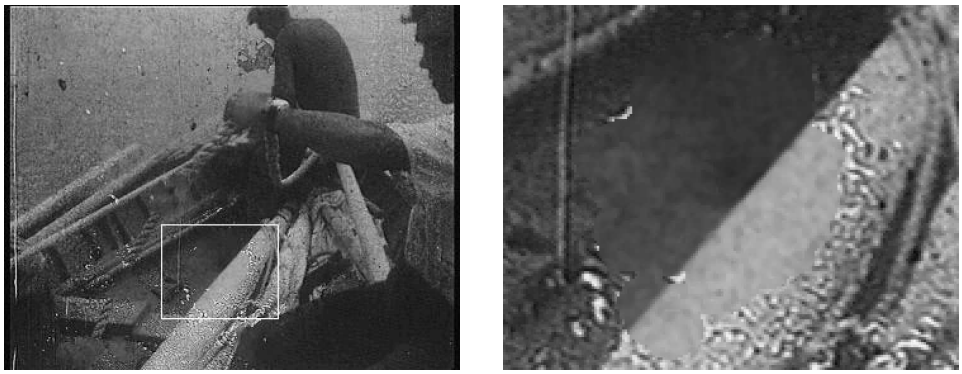
In this example, the temporal restoration outperforms the spatial one. Some mistakes are, however, still present. For example, the watch on the hand of the right fisherman is spread around, due to erroneous motion vectors (not visible). A more obvious case of temporal restoration failure is displayed in Fig. 5.5, where a succession of artefacts around the same location (top-left corner) in consecutive frames rendered the temporal restoration ineffective.

Besides series of similar artefacts in consecutive frames, other causes for temporal restoration failure are possible. Generally speaking, the further apart the two frames are and/or the bigger the artefacts they contain, the higher the chances that the motion estimator fails, together with the restoration that follows. A better motion can be estimated if we could rely on the current frame as well, and if our frames have already had their artefacts (at least) partly removed.



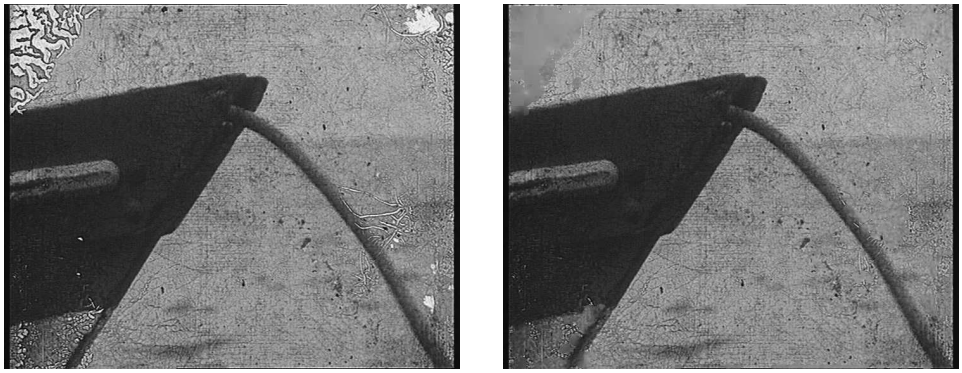
(a) (b)

Figure 5.5: Temporal restoration failure (top-left corner). a) Original image; b) Restored image.



(a) (b)

Figure 5.6: Spatiotemporal restoration of the first real life example. a) Restored image; b) Zoom-in.



(a) (b)

Figure 5.7: Spatiotemporal restoration of the second real life example. a) Original image; b) Restored image.

5.4 Spatiotemporal Restoration

The combined spatial and temporal restoration scheme that we propose here tries to exploit the current frame when estimating the motion. By restoring the rough structure in the artefact area of the current frame, a more accurate motion can be estimated between the current frame and its neighbouring frames. The spatial restoration of the previous/next frames allows us to interpolate from areas relatively free of artefacts. The spatially interpolated frames are then used in the temporal restoration as described in Section 5.3. Fig. 5.6 and 5.7 show results obtained with our combined spatiotemporal restoration scheme.

Careful inspection shows that the spatiotemporal restoration looks better than either the spatial-only or temporal-only restorations. The spatial- and temporal restoration artefacts are gone. Of course, some artefacts can still be seen, but they are usually outside the flagged artefact area that we were confined to. These, however, are subject to other noise reduction and/or further artefact detection and removal.

One may observe that in the first real life example, the rope that goes from the fisherman's hand to the bottom-left corner is not properly restored by any of the restoration methods described so far. This, however, is due to the pathological nature of the motion.

5.5 Conclusions and Future Work

We have presented here an algorithm for image sequence restoration, consisting of a succession of spatial and temporal restorations. This combination proved to behave better than each of the two restorations separately. The spatial restoration part of our algorithm can be successfully applied to static images as well, as demonstrated in this paper.

Currently, during the spatial restoration only straight edges are interpolated. As a next step, we will take into account also curved edges, T-junctions, and texture information. We expect that these enhancements will enrich our results in a consistent manner.

An analysis of the image structure and motion vectors may further indicate whether it is suitable to apply a spatial- and/or a temporal restoration, effectively switching on and off these modules.

Bibliography

[1] L. Atzori, F.G.B. De Natale, "Error concealment in video transmission over packet networks by a sketch-based approach", Signal Processing: Image Communication, vol. 15, no. 1-2, pp. 57-76, September 1999.

[2] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", IEEE Transactions on Image Processing, vol. 10, no. 8, pp. 1200-1211, August 2001.

[3] T. Chan, J. Shen, "Mathematical models for local non-texture inpainting", SIAM Journal on Applied Mathematics, vol. 62, no. 3, pp. 1019-1043, 2001.

[4] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[5] S. Masnou, J.-M. Morel, "Level-lines based disocclusion", Proceedings of IEEE ICIP 1998, vol. 3, pp. 259-263, Chicago, Ill, USA, October 1998.

[6] A. Rareş, M.J.T. Reinders, J. Biemond, "Statistical analysis of pathological motion areas", The 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 8/1-8/28, London, UK, January 16, 2001.

[7] A. Rareş, M.J.T. Reinders, J. Biemond, "Complex event classification in degraded image sequences", Proceedings of the IEEE ICIP 2001, vol. 1, pp. 253-256, Thessaloniki, Greece, October 8-10, 2001.

[8] A. Rareş, M.J.T. Reinders, J. Biemond, "Image sequence restoration in the presence of pathological motion and severe artifacts", Proceedings of IEEE ICASSP 2002, vol. 4, pp. 3365-3368, Orlando, Fla, USA, May 13-17, 2002.

[9] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

Chapter 6

Restoration of Films Affected by Partial Colour Artefacts

This chapter has been published as "Restoration of Films Affected by Partial Color Artefacts", by A. Rareş, M.J.T. Reinders and J. Biemond in the Proceedings of the *EUSIPCO 2002*, vol. 1, pp. 609-612, Toulouse, France, September 3-6, 2002.

Abstract

In this paper we present a novel restoration algorithm for a special type of film artefacts resulting mainly from film emulsion melting and the vinegar syndrome. We have devised the algorithm based on the observation that the affected areas did not lose their contents entirely. Rather, one of the film layers still preserves some of the original image structure. This artefact behaviour is exploited for both identifying and restoring these artefacts. The algorithm is validated by showing some results on real case examples.

6.1 Introduction

Film archives with material from the last century face deterioration at a progressive pace. Even if films are stored in protective conditions, the degradation process is not stopped, but rather slowed down. While there are several chemical and physical treatments that are able to clean, or at least slow down some types of degradations, in many cases it is impossible to do so, or the costs involved get too high.

The advent of powerful computers has permitted lately the digital processing of image sequences. Accordingly, the restoration of (digitized) films is now possible by means of intelligently devised algorithms [4, 8]. Some of these algorithms use spatial information to recover a damaged area, while others use temporal information from consecutive frames, or a combination of them. These algorithms proved to be quite successful in the restoration of digitized film and video.

Nevertheless, a problem occurs in the course of digital restoration of motion picture, when the so-called *pathological motion* (PM) occurs. This is a complicated motion that may take place in the image sequence, and it includes fast or highly irregular motion, occlusions, scene entrances, and/or other specific circumstances [5]. In such cases, as well as in the presence of severe artefacts, the motion estimation and the subsequent motion compensation, fail altogether. Since these two processes are essential ingredients for the temporal restoration algorithms, the restoration outcome may at times look worse than the original. A solution in this case is to detect PM areas and protect them against any restoration [7]. Unfortunately, this also protects the artefacts inside the PM areas against restoration. In order to avoid this side effect, algorithms have been developed that identify artefacts lying within PM areas, which would allow their further restoration [6].

In this context, we have devised an algorithm that takes advantage of the available spatial/colour information in order to restore the degraded colour films. The algorithm we propose here is demonstrated on a real case of heavily degraded film that also exhibits pathological motion, making temporal-based restoration an impossible task.

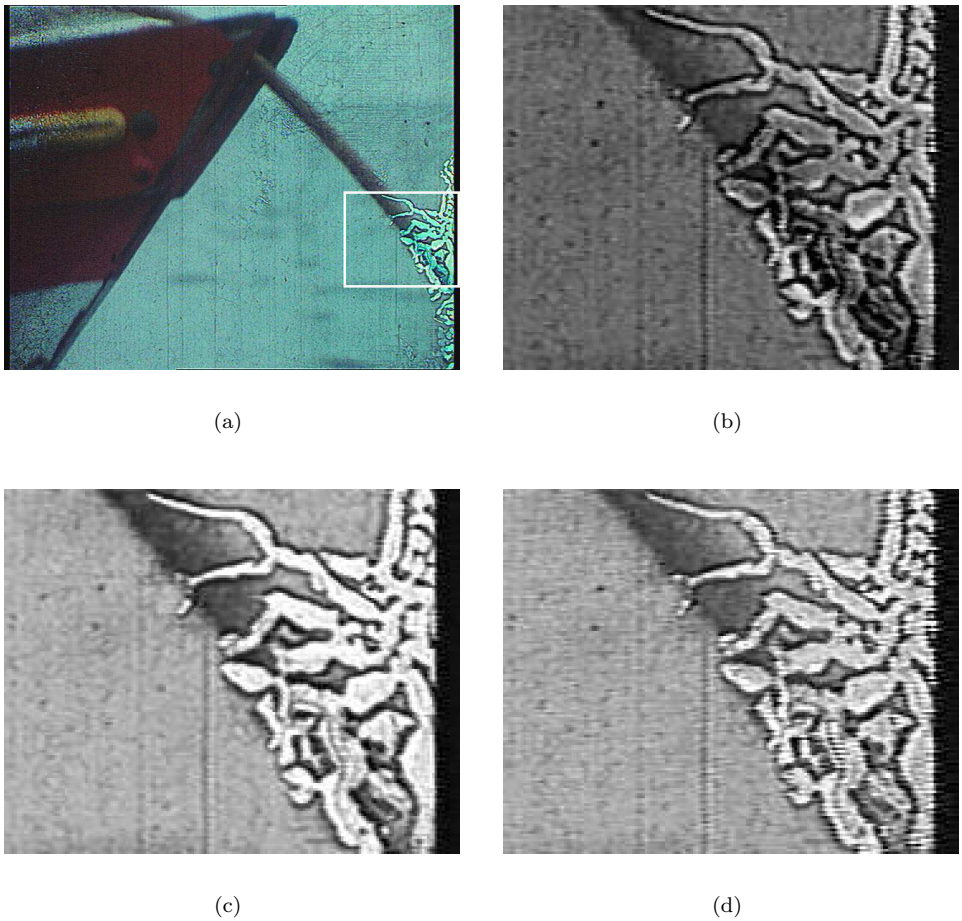
In Section 6.2, we present the general film structure and the properties of the artefacts that we concentrate on. Section 6.3 lays out our new restoration algorithm. Section 6.4 presents and discusses the results we have achieved so far, and outlines future work.

6.2 Film Structure and Properties of the Artefacts

The structure of a positive colour film usually consists of an arrangement of layers as follows: an anti-halation layer, the safety film base (the actual film support), a blue layer, a green layer, a red layer (these three layers consist of some special, colour-sensitive emulsion), some gelatin layers in between these layers and a protective gelatin layer on the surface [9]. Although the order may change in some cases, the aforementioned structure can be considered as general.

Films can be affected by dozens of types of film artefacts [2, 3], making artefact detection and correction a non-trivial task. Each artefact has its own properties and may show up in particular circumstances. In this paper, we try to identify and correct a special type of artefact for which the original information seemed to be completely lost at a first glance. However, an intense study of it revealed that only the *green* and *blue* film layers (i.e. the outermost ones) are totally lost, while the inner *red* layer still contains some original information. We will call this type of degradation a *partial colour artefact*. Film experts have established that the partial colour artefacts in the examples at hand are mainly due to emulsion melting and vinegar syndrome. In the shown images, other less noticeable artefacts are present as well, such as small dirt, hair, vertical scratches, glue traces, colour fading, etc. These, however, will not be considered here.

The emulsion melting happens when the dye layers dissolve, producing irregular patterns in each frame. It is an irreversible process. The vinegar syndrome appears due to poor ambient storage conditions and it got its name from the vinegar odour it produces (in the course of their chemical breakdown, the acetate-based film bases start to release acetic-acid; a 5% acetic-acid solution in water actually represents the common vinegar as we know it). This type of degradation represents one of the biggest problems of film archives, and, like the emulsion melting, it is an irreversible process. Moreover, from a certain moment on, the vinegar syndrome is an auto-catalytic process, progressively fuelling itself in the course of time. Both emulsion melting and vinegar syndrome can have various appearances [2, 3], but it is not the purpose of this paper to describe them. Fig. 6.1 shows a frame from our sequence, with emulsion melting and vinegar syndrome symptoms.



(a) (b)

(c) (d)

Figure 6.1: Frame corrupted by partial colour artefacts. a) Original image, with the main artefact indicated by the white rectangle; b-d) Zoom-in on the artefact in the red, green and blue layers, respectively.

6.3 Restoration Scheme

The restoration flow is depicted in Fig. 6.2. First, the partial colour artefacts are detected as regions exhibiting extreme colours only in the green and blue layers. Then, the (less affected) red layer is used in order to guide the restoration. Based on the image structure that is still present in the red layer, we find for each pixel inside the partial colour artefacts, a "sibling" pixel belonging to the same object, but lying outside the artefact. The RGB value of the sibling will then be pasted into the current pixel.

6.3.1 Detection of Partial Colour Artefacts

In order to detect the partial colour artefacts, ideally, we should be able to set some thresholds for selecting the normal/abnormal levels of red, green and blue. However, setting fixed thresholds is not desirable, because artefact intensities differ from film to film, or from frame to frame. A conservative threshold, while less prone to errors, would miss a good deal of the artefacts. On the other hand, a relaxed threshold would result in too many false artefacts detected. To avoid these problems, the detection of the partial artefacts uses a *hysteresis thresholding* [1], that proves to work quite efficiently in our case. The general idea of the hysteresis thresholding is that the image is first thresholded with a conservative threshold. The regions resulted from this step are then dilated in a connected neighbourhood that satisfies the second, relaxed threshold. In our case, each of the three layers - red, green and blue - has its own thresholds, since the red and the blue layers are somewhat darker, in general.

Furthermore, as shown in Fig. 6.1, the partial colour artefacts may be surrounded by some dark borders. We also want to add them to the partial artefact mask. This is realized by selecting the pixels in a strip around the partial artefacts detected so far, that satisfy $\sqrt{\frac{p_R^2+p_G^2+p_B^2}{3}} < T$, where p_R, p_G, and p_B represent the pixel values in the red, green and blue layers, respectively, and T is a predefined threshold.

Finally, to eliminate the influence of noise, and to add the smooth edges of the artefacts, the mask is twice closed and then dilated once.

Fig. 6.3 shows the original frame with an overlay of masks for the partial colour artefacts (a), as well as a zoom-in on the artefact area (b). Visual inspection of the masks confirms that the detection algorithm performs as desired.

6.3.2 Restoration of the Partial Colour Artefacts

As previously mentioned, the red layer is still intact in some cases (or at least partly intact, as is the case with our samples). The underlying assumption of the restoration

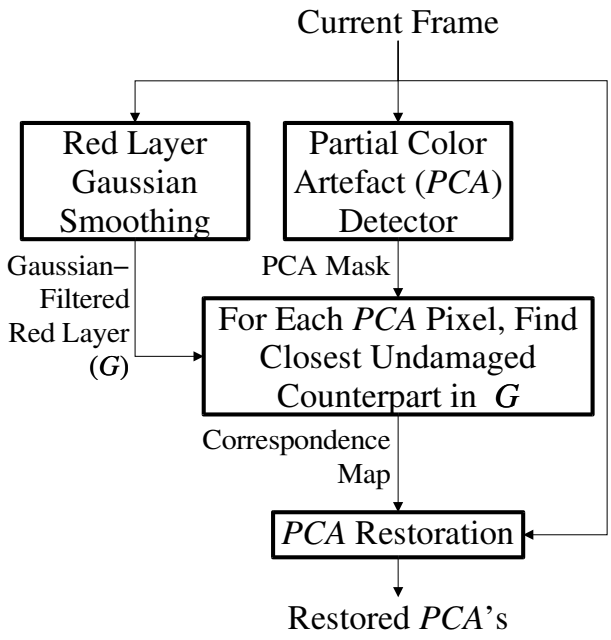
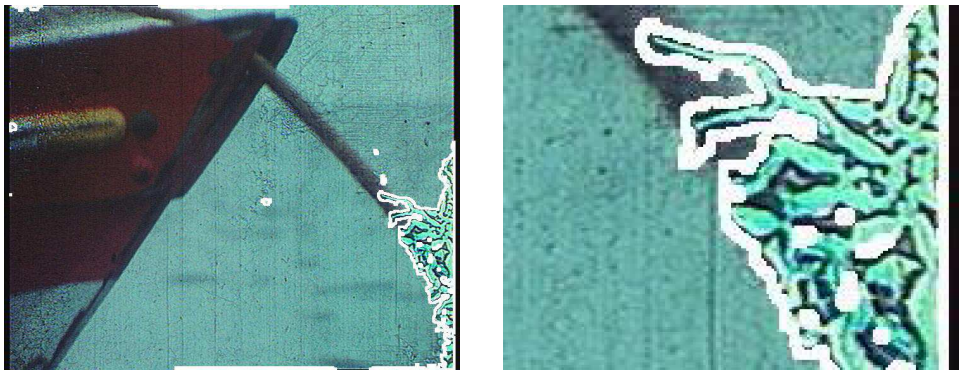
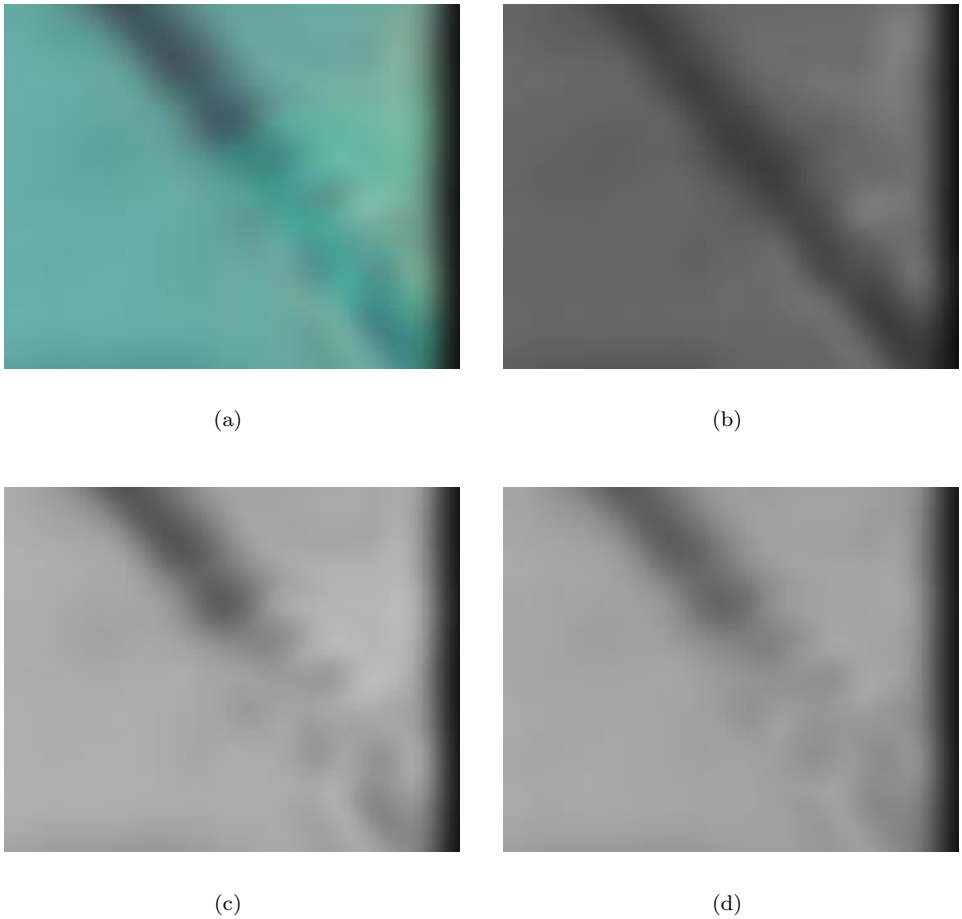


Figure 6.2: Restoration scheme.



(a) (b)

Figure 6.3: a) Original image overlayed with artefact masks (white border); b) Zoom-in on the main artefact.



(a) (b)

(c) (d)

Figure 6.4: Zoom-in on the artefact in the gaussian-filtered image. a) RGB image; b) Red layer (the only one to be further used); c) Green layer; d) Blue layer.

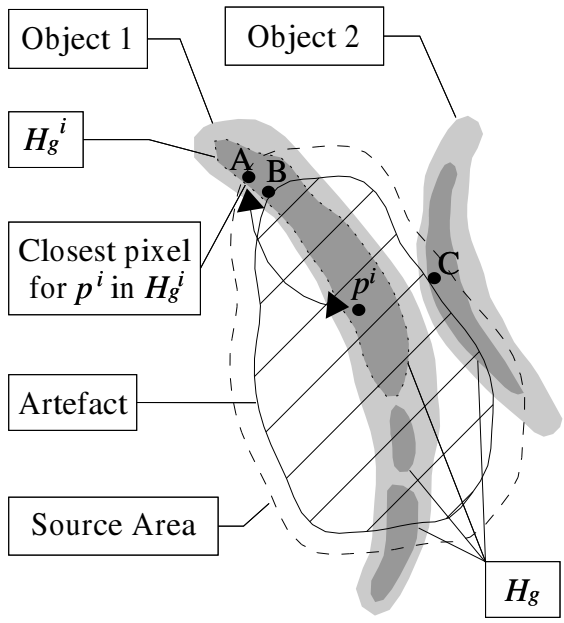


Figure 6.5: Scheme of the smoothed red layer appearance, with artefact overlay.

algorithm is that the general structure of the original artefact-free image, is still present in the red layer from our input image, and can be recovered by smoothing it. The isophotes of the smoothed red layer are then (almost) similar to the isophotes of the original artefact-free image. Although we do not draw samples from the smoothed red layer itself in the restoration process, it is used to indicate the areas where we may draw samples from. Further in this subsection we elaborate on these aspects. The restoration method will be outlined for one *partial colour artefact* (PCA), but in practice it is repeated separately for each PCA of the current frame.

We have already mentioned that the artefact influence is partly present in the red layer as well. In order to clean it, together with the noise, we smoothed the red layer with a large gaussian filter (a standard deviation $\sigma = 7$ was used in our experiments). We shall note the gaussian-filtered red layer of the image with G. Fig. 6.4 shows a zoom-in on the smoothed RGB image, as well as each smoothed layer, separately.

For each PCA, a *source area* (SA) was defined as a strip of pixels around the artefact (5 pixels, in our experiments). SA is depicted with a dashed border in Fig. 6.5. The source area will be used to draw samples from, in the final steps of the restoration. Of course, care was taken to eliminate from SA those pixels that already belonged to other artefacts. We have iterated through each grey level g in $G \cap PCA$, and computed a hysteresis thresholding, resulting in a mask of pixels having a value of $g \pm \Delta$

(e.g. $\Delta \cong 4\%$ from the overall greyvalue range). We shall name this the *hysteresis mask* (H_g). Normally, H_g follows the structure of the object(s). However, different objects with the same grey level g may exist within the same PCA (see Fig. 6.5), so H_g will consist of several disjoint areas. For this reason, each pixel p_g^i in $G \cap PCA$, having a grey level of g will possess its own hysteresis mask, H_g^i. H_g^i is given by that part of H_g which is connected with p_g^i. Further, pixel p_g^i will draw its restored value from that part of H_g^i that lies outside the artefact, $Q_g^i = H_g^i \cap SA$. Namely, the values of p_g^i will be replaced by the RGB tuple (from the initial, unprocessed image) of $p_{Q_g^i}^k$, with:

$$k = \arg\min_j \sqrt{d_r\left(G\left(p_g^i\right), G\left(p_{Q_g^i}^j\right)\right) + \left(\frac{d_p\left(p_g^i, p_{Q_g^i}^j\right)}{d_p\left(p_g^i, p_{Q_g^i}^j\right) + 1}\right)^2} \tag{6.1}$$

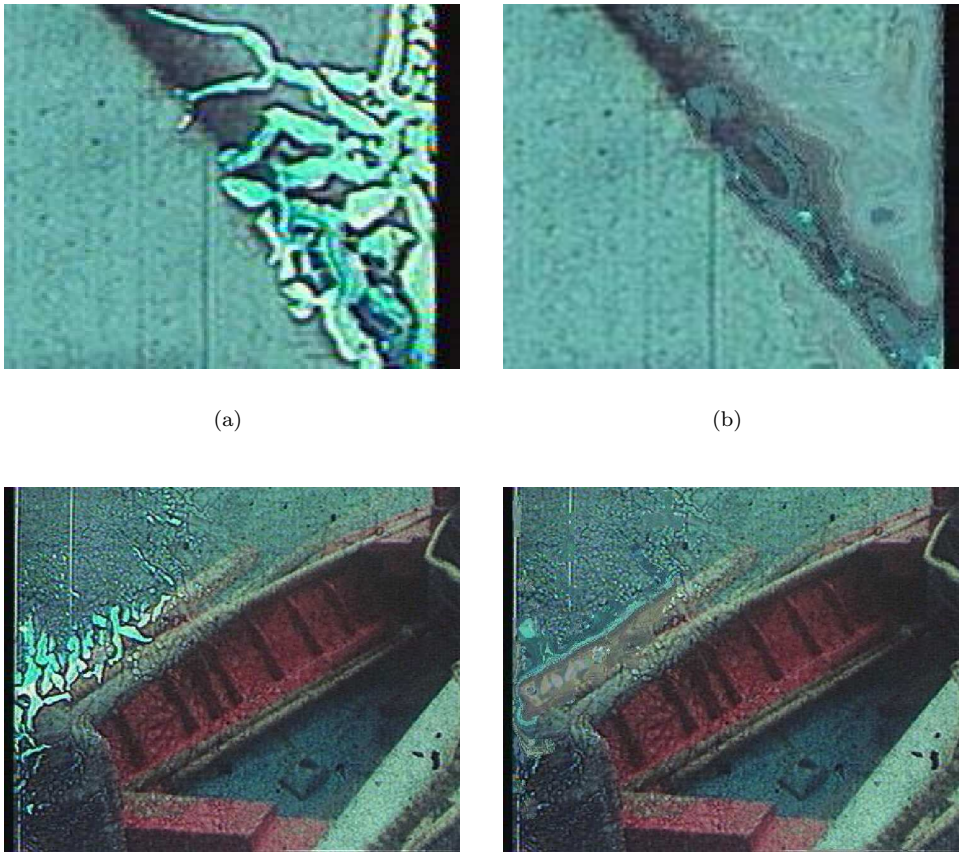
where $d_r\left(G\left(p_g^i\right), G\left(p_{Q_g^i}^j\right)\right)$ is the red value difference between the two pixels, $d_p\left(p_g^i, p_{Q_g^i}^j\right)$ is the physical (euclidean) distance, and $G(p)$ is the value of pixel p in the smoothed red layer G. The physical distance can take values in $[0 \ldots MAX]$, with MAX an unknown value. In order to both eliminate this ambiguity and enable us to combine it with the colour distance having values in $[0 \ldots 1]$, the physical distance was normalized to fit in the same range. The resulting distance tries to combine the two measurements with different, orthogonal natures in an euclidean manner. For pixels whose $Q_g^i = H_g^i \cap SA = \emptyset$, we use $Q_g^i = SA$ instead. To ensure the colour consistency of the interpolation, the RGB values from $p_{Q_g^i}^k$ are pasted altogether into p^i.

6.4 Results and Conclusions

Fig. 6.6 shows results of the method presented in the previous section. The general structure of the image has been recovered out of heavily degraded images. Although the restored images are not perfect (they still bear some artefact traces), we should stress again that they were recovered out of almost nothing. From this point of view, the results are promising.

One could also use general restoration algorithms here. However, those ones would not take advantage of the image structure that persists in the red layer. In our case, we do not have to "reinvent" that image structure - rather, we rely on it, since it already exists there, and enhance it. From this point of this view, we use artefact's properties to both detect it and eliminate it.

We conclude that the restoration technique presented here successfully revived portions of the sequence affected by partial colour artefacts, which were presumed to



(a) (b)

(c) (d)

Figure 6.6: Restoration results.a,c) Original frames (zoomed in); b,d) Restored frames (zoomed in). Sequence courtesy of RTP (Radiotelevisão Portuguesa).

have been lost. The method can be used (with some alterations) for artefacts caused by processes other than emulsion melting or vinegar syndrome, if they exhibit similar colour properties. Moreover, the present algorithm can be used as an intermediate step for a temporal restoration. The motion estimation in this case (a basic ingredient for temporal restoration) would no longer fail because of the severe artefacts. This, together with an investigation of other types of artefacts, are subject to further research.

Bibliography

[1] J. Canny, "A computational approach to edge detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679-698, 1986.

[2] http://www.ina.fr/Recherche/Brava/
Brava_public_impairments_list_3.light.html

[3] http://diamant.joanneum.ac.at/defects/index.html

[4] A.C. Kokaram, "Motion Picture Restoration: Digital Algorithms for Artifact Suppression in Degraded Motion Picture Film and Video", Springer Verlag, Berlin, Germany, 1998, ISBN 3-540-76040-7.

[5] A. Rareş, M.J.T. Reinders, J. Biemond, "Statistical analysis of pathological motion areas", The 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 8/1-8/28, London, UK, January 16, 2001.

[6] A. Rareş, M.J.T. Reinders, J. Biemond, "Complex event classification in degraded image sequences", Proceedings of the IEEE ICIP 2001, vol. 1, pp. 253-256, Thessaloniki, Greece, October 8-10, 2001.

[7] P.M.B. van Roosmalen, "High-level analysis of image sequences", technical report for INA - Paris (Institut National de l'Audiovisuel), France, the EU Aurora Project, July 1999.

[8] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

[9] R.J. Ross, "Color Film for Color Television", Focal Press Limited, 1970, ISBN 0 240 50714 2.

Chapter 7

Object Tracking by Adaptive Modelling

This chapter has been published as "Object Tracking by Adaptive Modeling", by A. Rareş and M.J.T. Reinders in the Proceedings of the *2000 IEEE International Conference on Image Processing*, vol. 3, pp. 74-77, Vancouver, BC, Canada, September 10-13, 2000.

Abstract

This paper addresses the problem of object tracking in image sequences. The approach taken is based upon adaptive statistical models. An object selected in a frame by a user is tracked throughout the sequence by using a blob-like description of its features. The object features are continuously updated by using the on-line version of the Expectation-Maximization algorithm. The proposed object description results in a flexible representation.

7.1 Introduction

To restore old movie material with high quality one has to deal with numerous artefacts [6]. Restoration techniques for these artefacts can, however, be complementary in their behaviour (e.g. one sharpens and the other smoothens) so that it becomes important to select the appropriate restoration technique locally and automatically. In such a restoration framework, a restorer selects regions or objects that must be restored. The restoration technique can then be chosen such that it is most effective for the selected region. For a successful restoration these selected items have to be tracked throughout the sequence. This paper addresses the tracking of such regions.

In this paper, we present a system that tracks objects in an image sequence. The proposed system tracks both the position and the shape of the object. In the first frame, the user marks the object by drawing a polygon around it. Calculating the statistics of the respective areas then initializes the models of the object and of the background (remaining part of the scene). For the subsequent frames, these object models are then analysed and updated according to the tracked data. The tracked items (object or the background) are represented statistically and approximated with a mixture of Gaussians. By including position into the feature set a coarse blob-like description of the shape of the region is readily achieved. The analysis consists of two phases: *segmentation* and *learning*. In the first phase, we calculate support maps for the tracked objects. The second phase uses the support maps for updating the statistics of the object models. Because not much data of the objects is available, one needs statistical techniques that can deal with incomplete data sets. Further, we need flexible learning techniques that can deal with rapidly changing object appearances.

A number of algorithms developed in the frame of statistical mathematics are suitable for such computer vision based object tracking. For achieving the real-time performances that are required only a part of these algorithms can be employed, such as the iterative Expectation Maximization (EM) algorithm [2], Icondensation [4], or Ransac [1]. In this paper, we use the on-line version of the EM algorithm as proposed in [5].

7.2 The Feature Space

When tracking an object in an image sequence, we look for features that can distinguish it from other objects. We have chosen a feature set that describes position, normalized colour, and Gabor-based textures [3].

The position feature consists of the two dimensions of the input image. The colour feature includes normalized red and green (calculated in the RGB colour space). We used normalized colours for their relative invariance to lighting changes

The texture feature is represented by a multi-dimensional set consisting of the responses of the Gabor wavelets, called "jets". For a pixel $\boldsymbol{x} = (x, y)$ and a pixel value $I(\boldsymbol{x})$, the jets $\boldsymbol{J}_j(\boldsymbol{x})$ are calculated using a convolution, as described in [3]:

$$\boldsymbol{J}_j(\boldsymbol{x}) = \int \boldsymbol{I}(\boldsymbol{x'})\boldsymbol{\Psi}_j(\boldsymbol{x} - \boldsymbol{x'})d^2\boldsymbol{x'} \tag{7.1}$$

with a family of Gabor kernels $\boldsymbol{\Psi}_j(\boldsymbol{x})$:

$$\begin{aligned} \boldsymbol{\Psi}_j(\boldsymbol{x}) \;=\; & \exp\left[-\tfrac{1}{2}\left\{\tfrac{(x\cos\varphi_\mu + y\sin\varphi_\mu)^2}{\sigma_x^2} + \tfrac{(-x\sin\varphi_\mu + y\cos\varphi_\mu)^2}{\sigma_y^2}\right\}\right] \\ & \bullet \exp\left[\tfrac{2\pi(x\cos\varphi_\mu + y\sin\varphi_\mu)}{k_\nu} i\right] \end{aligned} \tag{7.2}$$

where we used 3 different frequencies (ν={0, 1, 2}) and 4 orientations (μ={0, 1, 2, 3}):

$$\begin{array}{lll} k_\nu = 2^{-\frac{\nu+2}{2}}\pi, & \varphi_\mu = \mu\frac{\pi}{4} & j = \mu + 4\nu \\ \sigma_x = 0.5\lambda & \sigma_y = 0.5\lambda & \end{array} \tag{7.3}$$

Thus, in our implementation, a jet $\boldsymbol{J}$ is defined by a set of $3 \times 4 = 12$ complex values for each pixel. Although we used a number of 12 values, this number can vary, depending on the chosen number of directions and frequencies. Each calculated jet value can be written as:

$$\boldsymbol{J}_j = a_j \exp(i\varphi_j) \tag{7.4}$$

where a_j represents the magnitude, and φ_j represents the phase. The magnitude does not change quickly with translation, distortion, rotation, and scaling, whereas the phase changes rapidly with translation. Therefore, we have chosen to include in our feature set only the magnitudes. The magnitudes were normalized with respect to their energy:

$$a'_j = \frac{a_j}{\sqrt{\sum_i a_i^2}} \tag{7.5}$$

In Fig. 7.1, one can see how the responses of each Gabor kernel make up the feature set for the texture.

The complete feature set is built by concatenating all features. In the N-dimensional feature space, a tracked object ω_i is represented by a mixture $\boldsymbol{\Theta}_i$ of N-dimensional Gaussians, θ_{ij} (with mean $\boldsymbol{\mu}_{ij}$ and covariance $\boldsymbol{\Sigma}_{ij}$). Increasing the number of Gaussians in such a description allows for a more specific representation (the coarsest description is thus achieved by using only one Gaussian).

7.3 Object Tracking Using the EM Algorithm

Having such statistical representation, it becomes possible to classify each pixel x into one of the objects (say, ω_λ) by using a Maximum A Posteriori (MAP) decision that makes use of the Bayes rule:

$$\boldsymbol{x} \rightarrow \omega_\lambda,\ \lambda = \arg\max_\lambda p(\boldsymbol{\Theta}_\lambda \,|\boldsymbol{x}\,) = \arg\max_\lambda p(\boldsymbol{x}\,|\boldsymbol{\Theta}_\lambda\,) \tag{7.6}$$

with

$$p(\boldsymbol{x}\,|\boldsymbol{\Theta}_i\,) = \sum_{j=1}^{M_i} w_{ij} p(\boldsymbol{x}\,|\boldsymbol{\theta}_\lambda\,),\ i = 1, \ldots, K;\ \sum_{j=1}^{M_i} w_{ij} = 1 \tag{7.7}$$

where $\boldsymbol{\Theta}_i = \{\theta_{ij}\,|j = 1, \ldots, M_i\,\}$ denotes the object model for the i-th object ω_i, M_i is the number of components (Gaussians) of that object, θ_{ij} is the j-th component of object model $\boldsymbol{\Theta}_i$, $p(\boldsymbol{x}\,|\boldsymbol{\theta}_{ij}\,)$ is the probability density function (*pdf*) corresponding to component θ_{ij}, and w_{ij} is the weight of this component with respect to the object model. In Eq. 7.6 we assumed that all objects are equally probable.

The steps of the proposed algorithm are then as follows. The statistics of each object are initialized in the first frame by letting the user select it manually with a polygon. After this initial phase, the algorithm performs two steps for each frame. First, based on the available statistical model (coming from the initialization phase, or from the previous frame), the MAP decision is made for every pixel of the new image to be processed. In this way, support maps for the tracked object(s) are created (indicating that these pixels belong to the specified object). Together with the original image, these support maps offer new information about the statistics of the region to be tracked. In the next step, the statistics of each object model are updated on the basis of these support maps by using the EM learning algorithm. The updated object models are finally used as estimates when processing the next frame.

These steps of the algorithm are illustrated in Fig. 7.2. The scheme is made more efficient by incorporating a focus-of-attention mechanism, namely the motion between

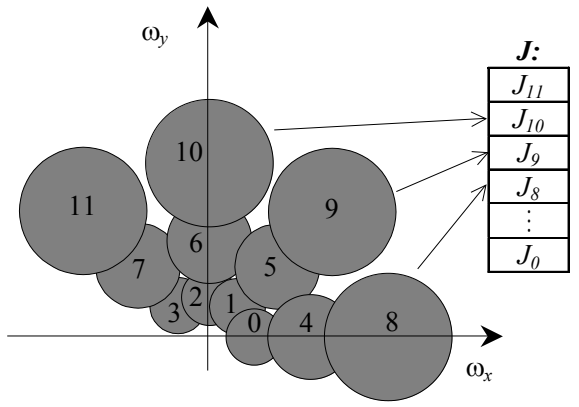


Figure 7.1: The responses of the Gabor kernels constitute the jets.

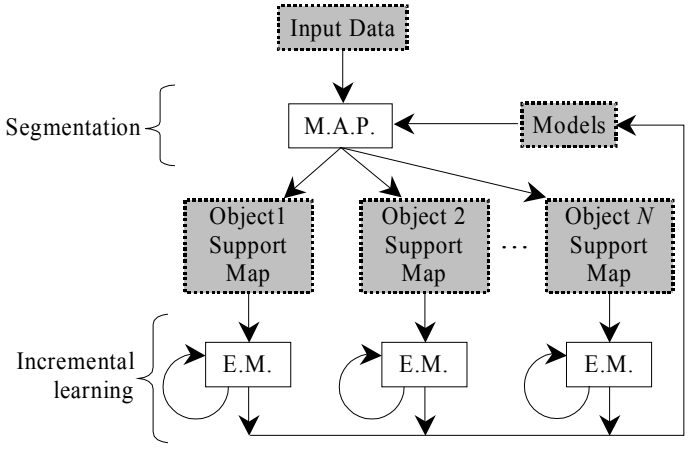


Figure 7.2: The analysis loop of the object tracker.

the frames (a simple thresholded frame difference) which makes the distinction between moving objects and still background.

For the learning step, we used update formulas derived from [5]. If n pixels of the current frame already have been processed, then, at the *n+1*-th pixel, the updating goes as follows. First, we know to which object the *n+1*-th pixel belongs (from the support maps that are initially created), say this is denoted by label λ. Then, for each component k of object model $\mathbf{\Theta}_\lambda$, we calculate the "responsibility" $h_{\lambda k}^{n+1}$:

$$h_{\lambda k}^{n+1} = \frac{w_{\lambda k}^n p(\boldsymbol{x}^{n+1} | \boldsymbol{\theta}_{\lambda k}^n)}{\sum_{j=1}^{N_i} w_{\lambda j}^n p(\boldsymbol{x}^{n+1} \Big| \boldsymbol{\theta}_{\lambda j}^n)}, \; k = 1, \ldots, N_i \tag{7.8}$$

where $\boldsymbol{\theta}_{\lambda k}^n$ or $\boldsymbol{\theta}_{\lambda j}^n$ represent the k-th and j-th component of the object model $\mathbf{\Theta}_\lambda$, respectively. The responsibility $h_{\lambda k}^{n+1}$ can be interpreted as the normalized contribution of the k-th component to object model $\mathbf{\Theta}_\lambda$ based on the measured values of pixel $\boldsymbol{x}_{n+1}$. Given these responsibilities, we can update the model's statistics (for all components $\boldsymbol{\theta}_{\lambda k}$, including their weights $w_{\lambda k}^{n+1}$):

$$w_{\lambda k}^{n+1} = w_{\lambda k}^n + \frac{h_{\lambda k}^{n+1} - w_{\lambda k}^n}{n+1} \tag{7.9}$$

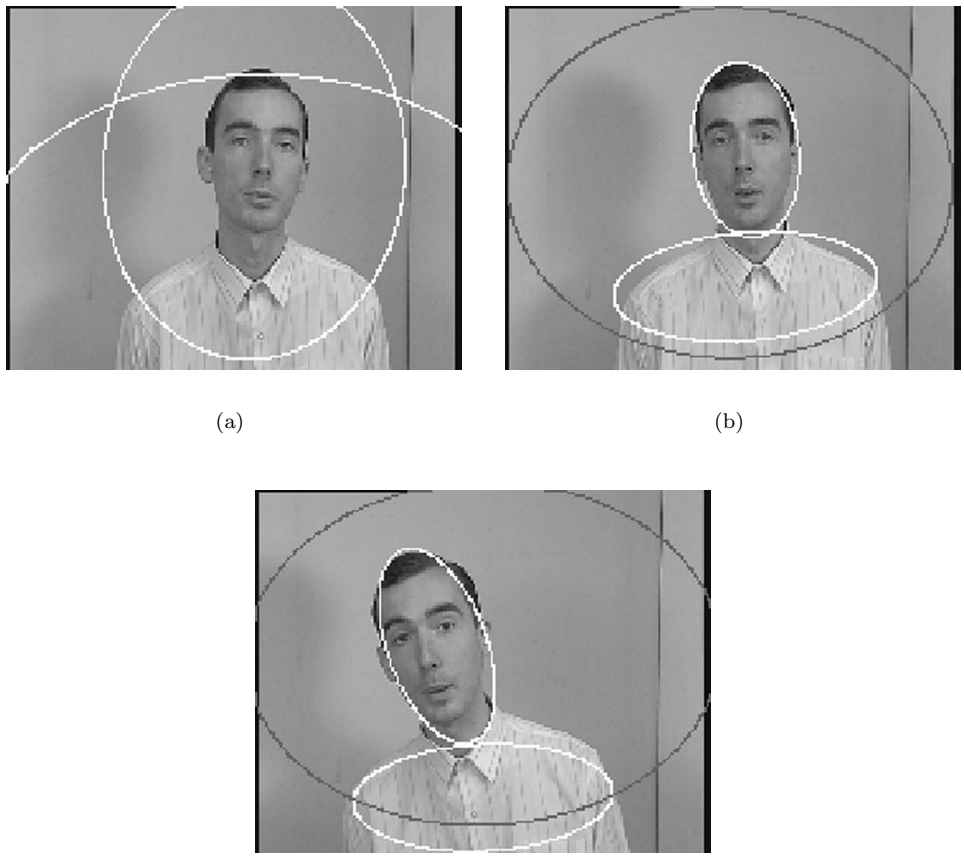
$$\boldsymbol{\mu}_{\lambda k}^{n+1} = \boldsymbol{\mu}_{\lambda k}^n + \frac{h_{\lambda k}^{n+1}}{(n+1) w_{\lambda k}^{n+1}} (\boldsymbol{x}^{n+1} - \boldsymbol{\mu}_{\lambda k}^n) \tag{7.10}$$

$$\mathbf{\Sigma}_{\lambda k}^{n+1} = \mathbf{\Sigma}_{\lambda k}^n + \frac{h_{\lambda k}^{n+1}}{(n+1) w_{\lambda k}^{n+1}} \left((\boldsymbol{x}^{n+1} - \boldsymbol{\mu}_{\lambda k}^n)^T (\boldsymbol{x}^{n+1} - \boldsymbol{\mu}_{\lambda k}^n) - \mathbf{\Sigma}_{\lambda k}^n \right) \tag{7.11}$$

Some special comments have to be made about the initial and maximum number of objects in the system, and the initial and maximum number of components for each object. These values can be set in a configuration file. The current number of objects and/or components varies as follows:

Number of objects. In the MAP step, if a pixel is not well explained by any of the current objects (i.e. the Mahalanobis distance to any of the current object models exceeds a certain threshold), an object *can* be added to the system (note that if one would require a constant number of objects this can be skipped). This object has one component, having as mean values the calculated features for the current pixel. The covariance matrices are set to some predefined value.

Number of components. In the EM step, if a pixel that was assigned to an object is too far from any of the object's components (i.e. the Mahalanobis distance to any of the current object's components exceeds another threshold), then a new component *can* be added to that object, with mean values the calculated features for the current pixel, and some predefined covariances.



(a) (b)

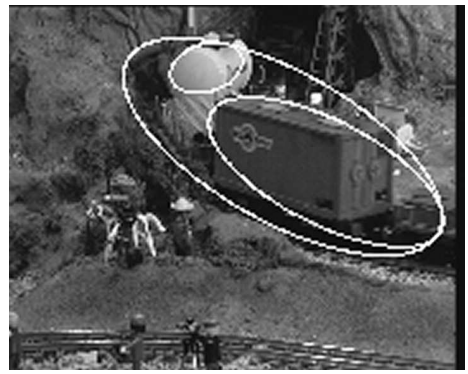
(c)

Figure 7.3: Results for the "person" sequence: a) Frame 1; b) Frame 21; c) Frame 133.

(a)

(b)

(c)



(d)

Figure 7.4: Results for the train sequence. a) Frame 1; b) Frame 17; c) Frame 27; d) Frame 49.

Cleanup. After each frame has been processed, components that cannot explain at least one pixel, or have a very small weight, are purged, in order to avoid computational overload. Objects left with no components are then deleted as well.

7.4 Results and Discussion

We have performed several experiments on different image sequences, and different parameters. Fig. 7.3 shows an example where we used only position and colour features. The ellipse represents a cross-section through the two-dimensional Gaussian representing the positional feature set (x and y position of the object pixels). The cross-section is taken at about 2σ distance from its mean.

The black ellipse corresponds to the background. The ellipses are initialized with large values (see the first frame, where the background's ellipse falls actually outside the image), but they rapidly shrink to adapt to the tracked items. The tracking is quite accurate, and robust, i.e. even partial occlusions are handled. We should, however, emphasize that in this particular case, the sequence is relatively unconstrained, i.e. there is no clutter and the background is not moving.

Fig. 7.4 shows an example of a result when using a more difficult sequence, where the content is harder to distinguish, and there are scene exits, occlusions and appearances in the scene. Again, we used only colour and position as object features.

The toy train sequence consists of a locomotive, and three wagons. Although we initially indicated only two wagons as objects of interest (Fig. 7.4(a)), in the second frame, the tracker has correctly identified that there is a preceding moving object (the locomotive) that is different in colour.

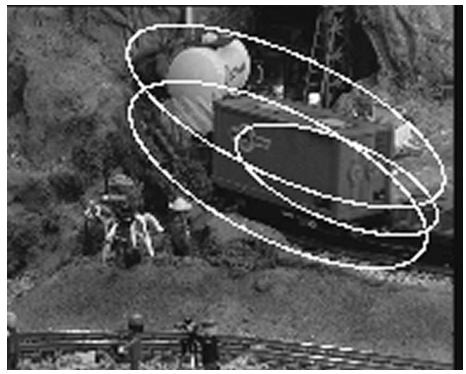
As we can see, the tracking performs satisfactorily as long as the situation is relatively clear. At some point in the image sequence, a third wagon appears (Fig. 7.4(b) and 7.4(c)). At that moment, the last ellipse tends to extend to the third wagon because of its proximity and similarity in colour between parts of the last two wagons (Fig. 7.4(c)). Meanwhile, the locomotive enters the tunnel (Fig. 7.4(d)) and the ellipse which was representing it now extends over the rest of the wagons. This happens, because the lower parts of these wagons have again a similar colour. Having a less confusing task, the middle ellipse follows its wagon quite well. From this experiment, we conclude that the tracking scheme is quite adaptive which enables it to quickly fit the changing appearances. However, in this sequence, this strong feature works somewhat against it since, due to the appearance and disappearance of the objects, the tracker becomes distracted.

We have performed another experiment on the same sequence, including Gabor-based textures this time. The results can be seen in Fig. 7.5.

It seems that using the additional Gabor features does not improve the object representation and consequently the tracking results. Similar results can be observed

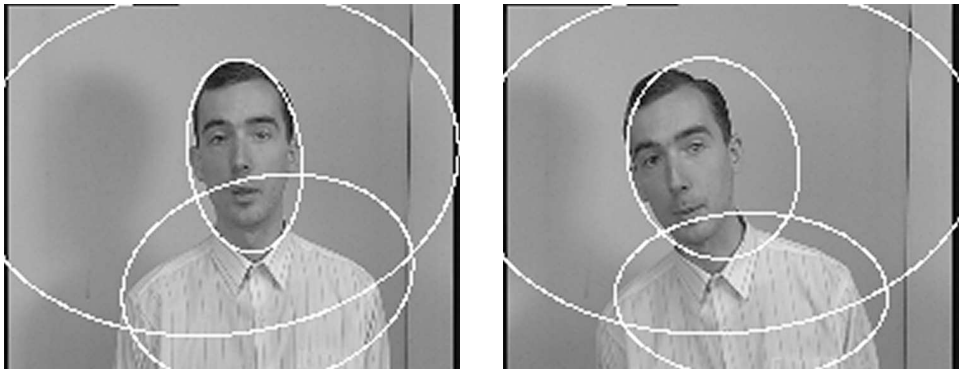
(a)

(b)



(c)

Figure 7.5: Results for the train sequence (when using texture). a) Frame 17; b) Frame 27; c) Frame 49.



(a) (b)

Figure 7.6: Results for the "person" sequence (when using texture). a) Frame 21; b) Frame 133.

for the "person" sequence (Fig. 7.6), although its disturbing influence is less then for the train sequence.

In our opinion, it turns out that Gabor-based textures cannot be used in highly cluttered images, since they cannot make a clear distinction between the objects. In fact they seem to represent the borders of the object instead of regional aspects. This hampers the tracking process, resulting in wrongly placed ellipses.

7.5 Conclusions and Future Work

In the view of our current results, the tracking performs satisfactorily for "clear" situations, i.e. the regions are described relatively accurate and the system rapidly adapts to the tracked items. The flexible object model can cope relatively easy with changing object appearances. Unlike motion estimation techniques, this method can adapt to changes of lighting conditions, to deformation and/or occlusion of objects, and it can overcome the problems posed by noise, up to a certain limit.

However, there are situations where the tracking fails, such as the heavily cluttered images. Here, even the texture recognition does not help in improving the results. The latter has mainly to do with the high adaptivity of the tracker.

Future work should include some more experiments in the area of textures, in order to gain more insight on how we can describe and identify textures. Also, as an addition to the current work, the feature set can be extended to include other features, such

as object motion. As a final remark, by representing the feature set of the objects statistically, these statistics can be used for selecting the appropriate restoration techniques to be applied to these areas, which needs to be explored in future work.

Bibliography

[1] R. C. Bolles, M. A. Fischler, "A RANSAC-based approach to model fitting and its application to finding cylinders in range data", Proceedings of IJCAI 1981, pp. 637-643, Vancouver, BC, Canada, 1981.

[2] A.P. Dempster, N.M. Laird, D.B. Rubin, "Maximum likelyhood from incomplete data via the EM algorithm", Journal of the Royal Statistical Society, Series B (methodological), vol. 39, no. 1, pp. 1-38, 1977.

[3] Y. Hamamoto, "A Gabor filter-based method for fingerprint identification", in L.C. Jain et al., editors, "Intelligent Biometric Techniques in Fingerprint and Face Recognition", CRC Press, NJ, pp. 137-151, 1999.

[4] M. Isard, A. Blake, "ICONDENSATION: unifying low-level and high-level tracking in a stochastic framework", Proceedings of ECCV 1998, vol. 1, pp. 893-908, Freiburg, Germany, 1998.

[5] N. Oliver, A. Pentland, F. Bérard, "LAFTER: lips and face real time tracker", Proceedings of CVPR 1997, pp. 123-129, San Juan, Puerto Rico, June 1997.

[6] P.M.B. van Roosmalen, "Restoration of Archived Film and Video", PhD Thesis, Delft University of Technology, The Netherlands, 1999.

Chapter 8

Discussion

This thesis focused on various aspects of film restoration, with an emphasis on restoration in conditions of difficult motion. Various subjects have been covered such as spatial and spatiotemporal restoration, classification of complex event areas, as well as motion estimation by means of object tracking. Since most of the proposed restoration algorithms circumvent the use of motion information (due to its unreliability), they can also be used for the restoration of still images. The novelty of each algorithm has been explained, and conclusions about merits and pitfalls of each approach have been given.

In the following, we discuss various issues related to the proposed algorithms and to the restoration subject in general. Ideas are offered for extending the proposed algorithms, and for new algorithms as well. Finally, prospects are given about future developments in the restoration domain.

Artefact Detection Issues

Usually, there is not enough knowledge built into the artefact detectors. One of the reasons is that artefacts display a certain amount of randomness in terms of spatial and temporal location, shape, colour etc., which is difficult to model. The presence of noise further complicates the modelling of artefacts. Yet, the artefact detectors have to take "hard" decisions (artefact or not artefact) for each pixel or group of pixels. In ambiguous situations, this leads to two categories of failures: missed artefacts, and false alarms (valid content that is flagged incorrectly as artefact). Since reducing the failures from one category automatically increases those from the other one, a trade-off has to be achieved.

Instead of taking "hard" decisions, it may be useful to output the certainty of the detection. Accordingly, if the artefact detector is very sure that a pixel represents an

artefact, it should mark it in the artefact mask and associate it with a high certainty of the detection (close to 1). If the artefact detector is not very sure whether or not the pixel is an artefact (e.g. in areas with difficult motion), it should mark it with a low detection certainty (close to 0). Restoration algorithms could then be designed such that they accept these certainty values (like the normalised convolution algorithm of Knutsson and Westin[1]). According to these "soft" detection masks, the restoration algorithms may either choose to keep or restore the associated pixels, or to modify them partially. The certainty value of a pixel may also be used as a weighing factor in situations where the pixel is used as an information source for the restoration of other deteriorated pixels.

Artefacts have been flagged until now by means of binary masks. In such a mask, a pixel is either an artefact or original image data. In practice, this sometimes oversimplifies the pixel status. Especially on the artefact borders (e.g. in case of scratches or blotches), there is a smooth transition from artefact to valid content that spans a few pixels. The information from this transition area could still be used to restore the original content (or at least to guide the restoration). For this reason, artefact flagging should output "soft" artefact masks instead of the current binary masks. These "soft" masks could contain values between 0 and 1, with 0 indicating no artefact, and 1 indicating a total information loss. The restoration algorithms may then choose between "boosting" the existing information to the desired levels, or replacing the pixels with interpolated information, or a combination of the two. Care should be taken in colour films, since different scratch depths, for example, may result in different degrees of degradation for each colour layer.

The "soft" certainty and degradation masks represent different types of information. For example, an artefact detection algorithm can be very sure that a pixel represents a transition between artefact and non-artefact areas. In this case, the certainty value is close to 1, while the degradation value is about 0.5. In another situation, the artefact detector may have some doubts whether a very bright pixel represents an artefact or not. The extreme brightness of the pixel indicates that the information loss is total, if the pixel is indeed an artefact. In this case, the certainty value could be about 0.5, with a degradation value close to 1.

Spatial Restoration Algorithms

In the edge-based algorithms presented in this thesis (Chapters 3 and 4), edges are described by their properties on either side. If a T junction is formed on one side of the edge, the intensity-related features extracted from that side become irrelevant and have to be discarded from the process of comparing pairs of edges. An alternative approach which would get rid of this problem would be to use the same features to

[1] H. Knutsson, C.-F. Westin, "Normalized and differential convolution: methods for interpolation and filtering of incomplete and uncertain data", Proceedings of IEEE CVPR 1993, pp. 515-523, New York, NY, USA, June 1993.

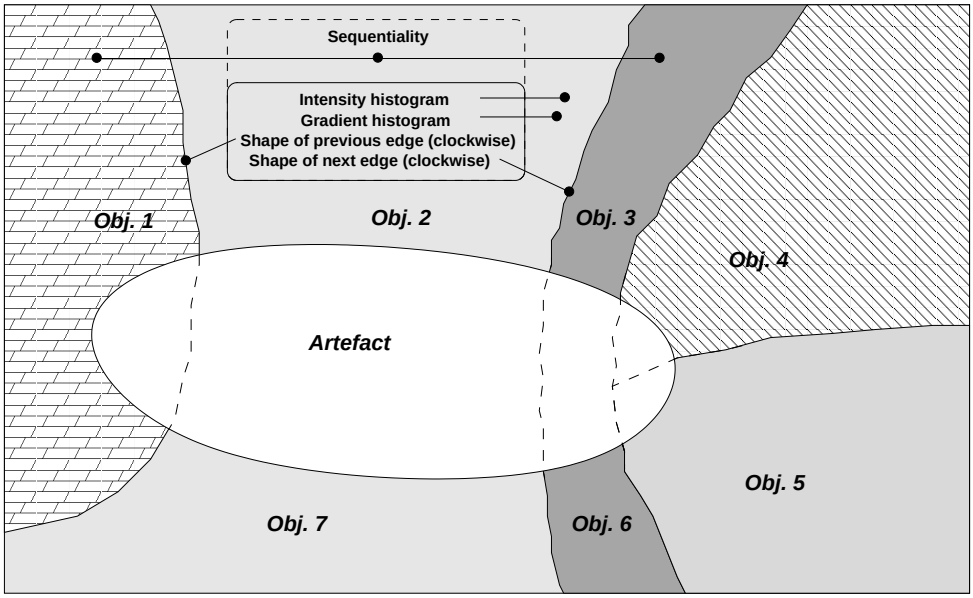


Figure 8.1: Alternative to the edge-based restoration algorithms from Chapters 3 and 4. Here, instead of building edge couples, object couples are created, based on the similarity of their features.

describe the objects themselves, rather than the edges between them, and to try to continue the objects inside the artefact (see Fig. 8.1). Initially, the objects are the regions between two neighbouring edges that touch the artefact. The objects can then be represented by the intensity-related features (intensity and gradient histograms in Fig. 8.1), and the circles fitted to the edges on both sides of the object (the shapes of previous and next edges in the same figure). The intensity-related features need not be flagged. The sequentiality parameter would be measured based on objects, rather than on the edges that separate them.

The edges were modelled as circle arcs. In many cases, this proved to be a sufficient approximation of the edge shape. In some special cases, more complex shapes would be needed. This should be done with caution, however, since fitting a more flexible shape may give better results in those special cases, but decreases the fit for simpler ones. In the latter case, the parameter estimation may become ill-posed: the complex models allow more degrees of freedom, thereby giving rise to unstable systems when used on small input data sets (the edge pixels). Besides rejecting valid couples, complex edge models may accept a larger number of invalid ones. From our experience, this was already the case for elliptical models. Although they fitted better the correct edge couples, they also fitted a lot of incorrect ones. Especially in conditions of noisy data, degenerated solutions may occur. This led us to conclude that spline fitting could not be successful either. A compromise solution to this dilemma would be to use circle fitting for the structure reconstruction step, and ellipse or spline fitting for the inpainting step, in order to achieve smoother edge continuity.

A category of shapes which is not covered by our approaches but is found in many images is represented by corners made by pairs of straight edges (e.g. the corners of a window). When such corners lie inside an artefact, our approach would always reconstruct a round corner between the corresponding couple of edges (see Fig. 8.2). In reality, it is natural to assume that if the incoming edges are both straight, then they probably belong to an object that has a sharp corner. The problem of finding straight edges in conditions of incomplete and/or noisy data needs further investigation.

The cost formulas used in our edge-based spatial restoration approaches involve variables of different natures such as: physical distances, pixel intensities, or spatial order of edges. Merging these variables to provide a single number which characterises the entire edge configuration is by no means a trivial task. Care was taken to approximately calibrate the different contributions of each variable in the final cost. Nevertheless, more could be done in this respect to make sure that the value of one parameter does not consistently dominate the values of the other parameters. For this purpose, the parameters could be normalised against their variances before they are added up. However, for some parameters the variance is not clearly defined (e.g. for the *sequentiality* parameter).

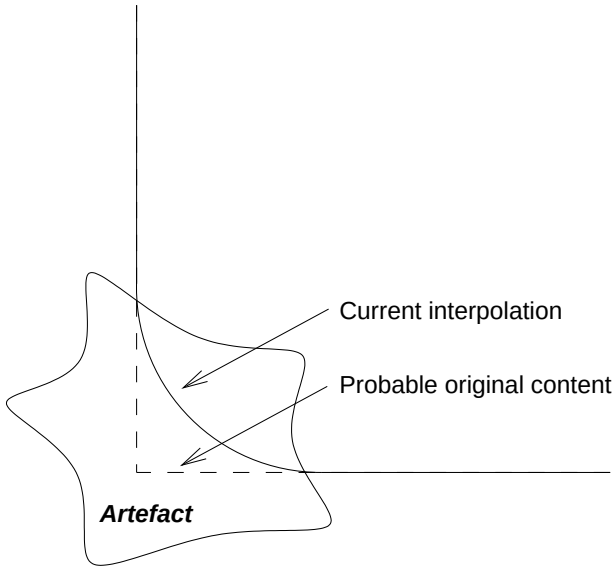


Figure 8.2: The edge-based restoration algorithms presented in this thesis interpolate round corners. Straight edges, however, usually form a sharp corner.

Spatiotemporal Restoration Algorithms

If a blotch-like artefact persists over time at the same place (e.g. because of dirt on the lenses, or some form of vinegar syndrome), the classic (spatio)temporal algorithms are not useful. In this case, however, one can apply our proposed spatial restoration algorithm on a frame-by-frame basis.

When applying the structure-based algorithms from Chapters 3 and 4, it is possible that a blotch which persists in the same place gets interpolated differently in consecutive frames. This would happen if several alternative structures were to have similar costs. In this case, however, it would be more acceptable from a visual point of view to keep the same interpolated structure over consecutive frames (even if it is not the best possible one), rather than having a flickering appearance of the result.

The algorithm presented in Chapters 3 and 4 can be extended in this respect to the temporal domain. The structure cost can be modified to accept a bias towards building the same edge couples as in another frame. This bias could be, for example, the best scoring structure from all frames around the current one. Clearly, this will only be helpful in those cases where objects do not modify significantly over the given time span.

In Chapters 3 and 4, the edge-based algorithms try to "invent" the object shape inside the artefact area by looking at a part of the remaining shape. Instead of this

approach, it may be worthwhile to extract entire objects bordering the artefact, and then to look for the same object in one of the previous or next frames, based on the incomplete information about the extracted object. Upon finding the object, its missing part can be copied into the current frame. This suggests the use of object tracking as part of the restoration process. A practical problem, however, is the fact that current object tracking systems were not built with possible degradations of films and video in mind. The deteriorated content will affect the object tracker's ability to do its task. In this respect, object tracking algorithms should be designed such that they allow for an incomplete description of objects.

3D Spatial Restoration

Nowadays, 3D data processing is becoming commonplace in areas such as magnetic resonance imaging (MRI) or ultrasounds for medical purposes, geological prospection of mineral deposits, weather forecasting etc. The structure-based restoration algorithms proposed in this thesis (Chapters 3 and 4) can be extended to restore missing volumes in such three-dimensional (3D) data. For this purpose, the segmentation step which provides edges should be replaced with a 3D segmentation step. The intensity-related features are then estimated over a volume, rather than a surface. Edges are replaced with surfaces, and the circle fit becomes a sphere fit. The *spatial deviation* from a circle is replaced by the spatial deviation from the sphere. The *angular consistency* and *aperture quality* parameters, however, need to be adapted. The 3D angular consistency might be calculated as an average of 2D angular consistencies measured in a series of consecutive slices through the artefact. The 3D aperture quality could be measured as a 2D aperture quality over the slice which returns the maximum value, or as an average of 2D aperture qualities measured on the same slices as for the angular consistency. In 3D, objects intersect far less than in 2D, which makes the *sequentiality* parameter even more needed in 3D than in 2D. However, the sequentiality parameter is more difficult to calculate, since the clockwise order does not exist in 3D. In 2D, the sequentiality parameter penalises the intersections implicitly, by encouraging sequential edge couples. In 3D, the sequentiality cost should be used as a kind of a penalty score for the number of intersections. In the structure reconstruction step, the edge couples are replaced by surface couples, modelled by spheres. The inpainting step can be adapted to 3D quite simply: it should be carried out "parallel" (i.e. concentric) with the spheres fitted to the surface couples. In the case of texture synthesis, the search neighbourhood and the sampling neighbourhood should be cubic instead of square, and should be limited by the object surfaces instead of object edges.

Supporting Algorithms

Supporting algorithms such as motion estimation and edge detection were all originally devised to work in artefact-free conditions. However, even with perfect,

artefact-free images (or image sequences), their results are far from perfect. The deteriorated content will further affect their ability to do their tasks, resulting eventually in a "chicken-and-egg" problem. For example, when good a motion estimate is available, the restoration improves visibly; however, motion can be estimated best on artefact-free content, that is to say, on content which has already been restored properly. A common solution to such interdependent problems is to iterate through the two processes until a stop condition is met (i.e. a certain number of iterations have been gone through, or the solution has reached a stable point, etc.). This solution is similar in some sense to the Expectation Maximisation approach used in Chapter 7.

Ideally, as long as the supporting processes do not take into account possible image degradations, any restoration method which relies on them should be aware of their limitations. For example, in edge-based restoration, for each edge that was not coupled with another edge, the algorithm should go back and fine-tune the edge detection step in order to try to find a similar edge (around the artefact) that was missed in the initial edge detection step. It should also check whether the uncoupled edge is indeed a valid edge, or whether it was mistakenly detected and introduced in the edge set.

In order to become artefact aware, supporting processes have to implement important changes in the way they function and in their inputs and outputs. Edge detection, for example, should accept as input, besides the image itself, a mask of artefacts. No edge should be detected in the flagged areas, and care should be taken on how the artefacts might influence the detection process in their vicinity.

Motion estimation processes should also adapt their outputs. Currently, all motion vectors in a frame point to the next/previous frame, resulting in forward/backward motion vectors. In case of total occlusion, or scene entrance, there are no valid forward or backward motion vectors, respectively. Therefore, special labels should be assigned to the motion vectors to indicate this.

Motion estimation would also benefit from taking into account frames beyond the next or previous frames. As shown in Chapter 2, a bird flapping its wings, or the propeller of a plane are currently detected as impulsive, single-frame events, and flagged accordingly as artefacts. However, looking a few frames ahead or behind, the right motion vectors can be properly estimated. Taking into account motion vectors which point to distant frames would require another change in the current structure of motion vectors.

Evaluation and Standardisation Issues

A lot of insight could be gained (from many points of view) from using two copies of the same film to restore the contents of that film. First of all, the detection of artefacts would be very simple, since it is very unlikely that two artefacts occur at the

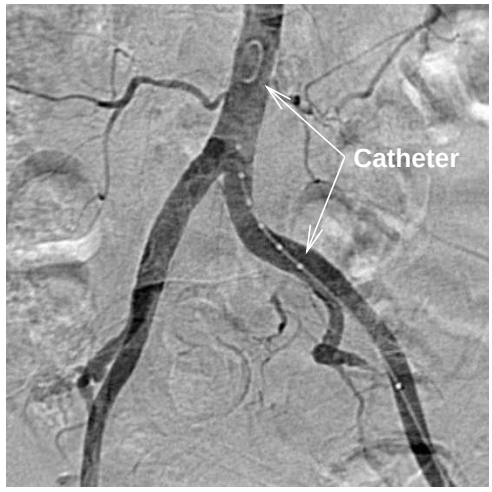


Figure 8.3: X-ray of an aortic bifurcation with a catheter inside.

same place in the same frame of the two film rolls (unless these were developed from the same deteriorated negative). Secondly, the restoration of these artefacts would be very easy - one would only have to copy undegraded content from the other film (after proper intensity and contrast correction). Thirdly, a common problem in estimating the quality of a restoration algorithm is the absence of the undegraded original data. Having access to the undegraded sequence would allow a *ground truth evaluation* of the restoration algorithms, through comparison with the original content. Fourthly, the sequence of extracted ground truth artefact masks can be used to artificially degrade other original sequences. These could then be used for similar ground truth evaluation purposes. Last but not least, the sequence of ground truth artefact masks can give us knowledge about the statistical properties of the artefacts. This, in turn, has a twofold advantage: it helps tuning the artefact detection algorithms, and it facilitates model building for artificial artefact generators that are employed nowadays to validate the restoration results.

A forward step in terms of standardisation would be the gathering of a representative, public set of samples for each type (and sub-type) of artefact. This would benefit the entire restoration field, as it would supply researchers with a common ground for comparing the results of their algorithms, much in the same way as the "Lena", "baboon" and other images represent a *de facto* common ground for result comparison for the image processing community.

Restoration Benefits Outside the Archives World

The restoration benefits reach further than film and video archives. We already mentioned some other applications when we discussed the restoration of 3D data. Some of the algorithms presented in this thesis can be easily used without modifications in the medical image domain. Fig. 8.3 shows an example in which a catheter is introduced in the blood vessel to be studied, and used to inject a contrast agent substance into the blood in order to make the vessels visible in X-ray images. The catheters may also be used to calibrate the images in order to relate the various dimensions measured in pixels to the actual dimensions in millimeters. For this purpose, they are manufactured with a known width and bear calibration markers which are placed at fixed distances from each other. However, the presence of a catheter in the blood vessel hampers the visualisation of the vessel and has a bad influence on the estimation of blood flow or the detection of vessel edges. In the latter case, the contour detector may follow the catheter contour instead of the vessel contour, from the point where the catheter touches the vessel wall. It becomes obvious that if automatic analysis of the image has to be performed, the removal of the catheter from the image (as if it was an artefact) can be beneficial.

An expanding field for applying restoration algorithms is related to compressed images and image sequences. These are commonly used in live TV transmissions, video chat sessions on the Internet, and for general storage of visual information. Each application has its own reason why errors can occur: the broadcast medium may be noisy (and retransmission may not be feasible), packets may be lost due to a network congestion, and the storage medium may suffer from a partial degradation process, respectively. The errors will often show up as one or more consecutive blocks with missing or distorted information. These blocks can be regarded as artefacts and restored with one of the algorithms proposed in this thesis.

The restoration techniques can be great tools for retouching photos, and not only strictly to restore damaged content. Sometimes the presence of an object or some persons in the image affects the composition of a photo. Being able to remove them from the image with a few clicks saves a lot of time and effort, and may produce better results than manual retouching. Advanced image manipulation systems already provide crude tools with such features.

Final Remarks

Currently, film restoration algorithms have a limited, local view on the input data, both spatially and temporally, mainly for reasons of computational efficiency. The shortcomings of such an approach are obvious. A part of these limitations will be surpassed by designing algorithms which keep in memory some information about past artefacts or about undegraded content. This will result in restoration processes

which are more consistent over time, and would allow a fine tuning of their parameters. In some cases, superior reasoning will be needed in order to cope with special situations, such as the complex events mentioned in Chapter 2.

It has been shown that restoration is closely related to other image processing areas such as motion estimation, segmentation, and pattern recognition. It is then natural to state that a comprehensive algorithm in any of the aforementioned areas will have to be able to cooperate with algorithms from the other areas. Such an interdisciplinary approach will benefit all parties involved.

Many problems still remain to be solved in the restoration domain. There are artefacts and contexts for which the quality expectations are only partially fulfilled. These, together with the new artefacts that will appear with each new broadcast/storage medium or data format, will trigger further research on restoration.

Appendix A

Basic Types of Pathological Motion

Estimating motion is still an open research subject [1, 3, 4], especially in case of *pathological motion*. The pathological motion represents complex, non-linear motion that cannot be easily modeled by current motion estimation techniques. It can be observed in areas with fast or irregular motion, occlusions, scene entrances and/or other specific circumstances. Objects performing pathological motion (PM) behave in many ways, either from the point of view of human perception, or from the point of view of the numerical contents of the affected areas. Based on these aspects, the following PM taxonomy was introduced [2]:

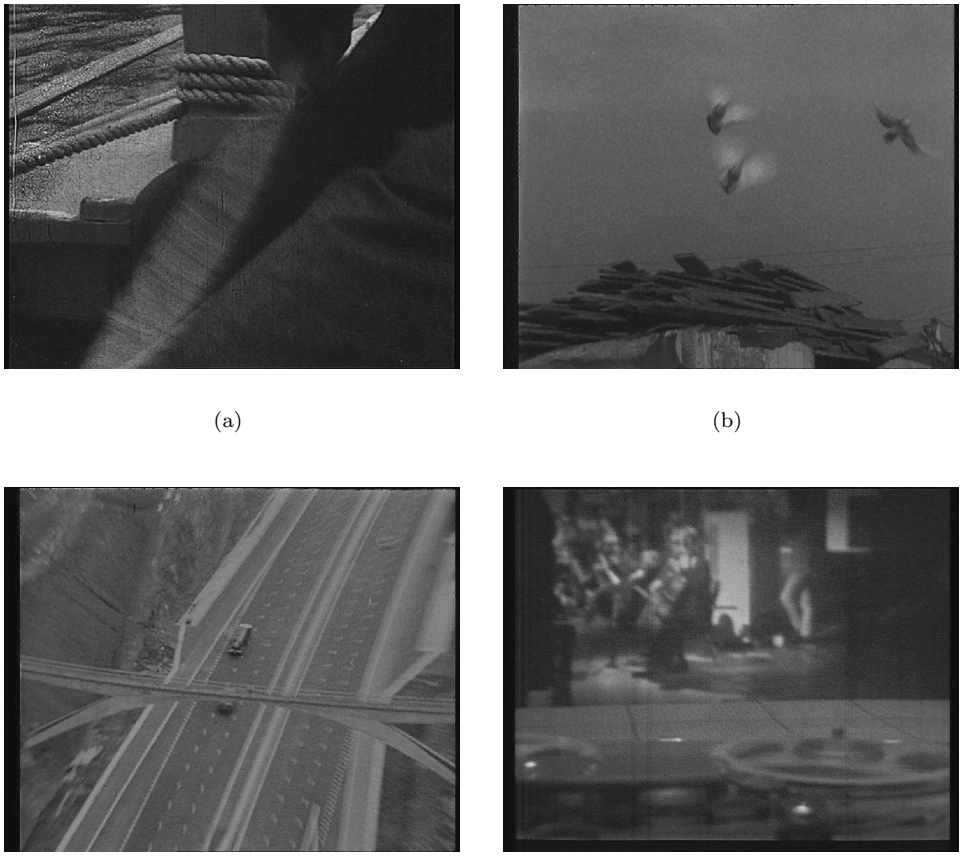
- *Blur*. This phenomenon is caused usually by a combination of factors: low sensitivity of the film/sensors array and improper shutter speed, compared to the relative movement between them and the objects, or improperly focused lens. As a consequence, objects look smeared, unclear (see Fig. A.1). Motion blur, blur caused by strong zooming, or out-of-focus blur are different from the point of view of the point spread function associated with them.
- *Occlusion and uncovering*. The occlusion, either partial or total, may also cause the failure of a motion estimator (see Fig. A.2). The uncovering represents the same case as the occlusion, if we take the frames in a reversed chronological order.
- *Large displacement* may render objects untrackable by common motion estimators usually because their displacement is larger than the search window (Fig. A.3).
- *Strong zooming* represents a problem for mainly the same reasons as a large displacement (Fig. A.4).

- *Rotation*, if above a small angle, poses a problem to most motion estimators, especially to block-based ones (see Fig. A.5).
- *Intermittent motion* is an interesting aspect that we hardly observe when looking at image sequences displayed at normal speed. This kind of (semi-) repetitive motion is more visible when playing the sequences frame by frame. The *blinking* subtype of intermittent motion, shown in Fig. A.6 (a-d), happens when an object, or a part of it, is visible every other frame (e.g. the propellers of a plane). The *alternating* subtype shows up when different sides of an object are displayed in a repetitive way across the frames. Such is the case of a pigeon flapping its wings, which are dark on one side, and bright on the other, as shown in Fig. A.6 (e-h).
- *Image overlap* happens when the image of an object is a combination of at least two other images (Fig. A.7). This ranges from simple cases, like superposition of shadows or lights, to more difficult ones, in which an object is (semi-) transparent, or it reflects another image. Shot changes of "dissolve" type also give rise to overlapped images.
- *Erratic motion*, presented in Fig. A.8, is a very irregular motion that we can best describe by giving examples. A strongly waving flag is such an example. It cannot be classified as occlusion/uncovering because many parts of it appear only in one frame, in several consecutive frames. It cannot be classified as intermittent motion either because it is not repetitive. Another example is the image of a fire. Here, the flames can split, merge, and change their shapes very fast. Wave crests or explosion-like phenomena also fall in this category.

Although the PM categories were made as independent as possible, it is certain that in some cases the detected PM will fall under two or more categories. This may happen because some of the aforementioned categories have, from certain points of view (e.g. color histogram), similar characteristics. Moreover, the detected PM may represent a combination of two or more categories (e.g. blur + large displacement).

Statistical Properties of the Pathological Motion Areas

The term *"Complex Events"*, as defined in Chapter 2, means either *pathological motion* (PM) and/or *artefacts*. These represent the two main reasons for which motion estimation can fail. Handling PM would be virtually impossible if we would not know its characteristics. We have therefore performed in [2] a series of tests on areas with complex events, to see whether we can discriminate between artefacts and PM. The tests consisted of histograms of areas with complex events in each of the Red, Green and Blue channels. The tests comprised two categories of PM: *blurred*



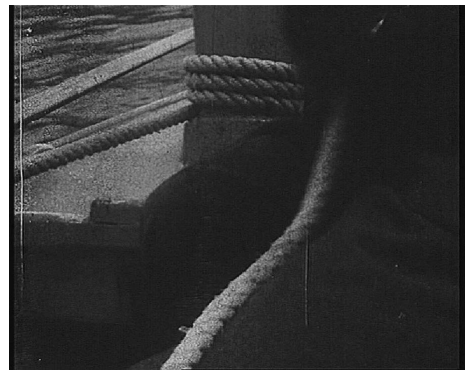
(a) (b)

(c) (d)

Figure A.1: Examples of blur. a-b) Motion blur: a fast-moving rope, and pigeons taking off from a roof, respectively; c) Zoom blur (mostly visible towards the image margins): a bridge stretching over a highway; d) Out-of-focus blur (both in the foreground and in the background): band playing behind a tape recorder.

(a)

Figure A.2: Examples of occlusion and uncovering. a) Woman walking behind a tree; b) Helicopter behind a crane, and crane behind stairs.

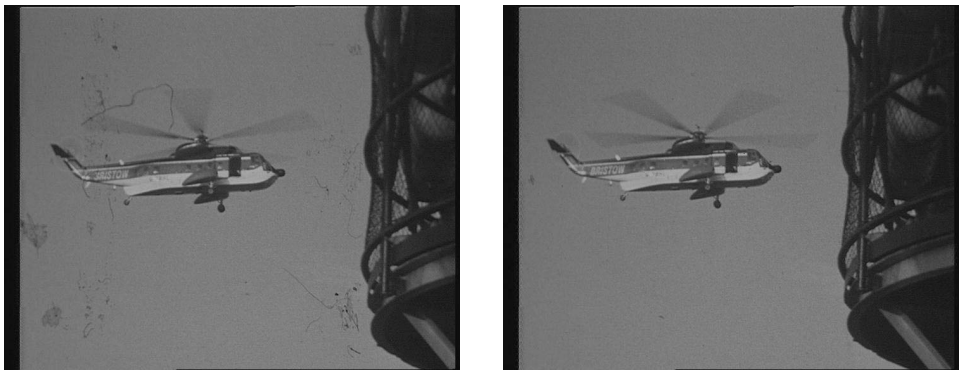


(a) (b)

Figure A.3: Example of large displacement: a fast-moving rope.

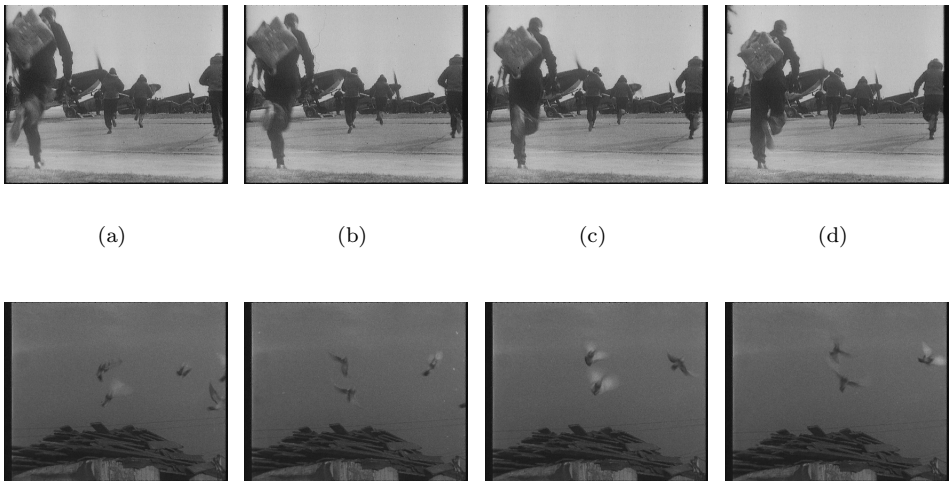
(a) (b) (c)

Figure A.4: Example of strong zooming: a bridge stretching over a highway.



(a) (b)

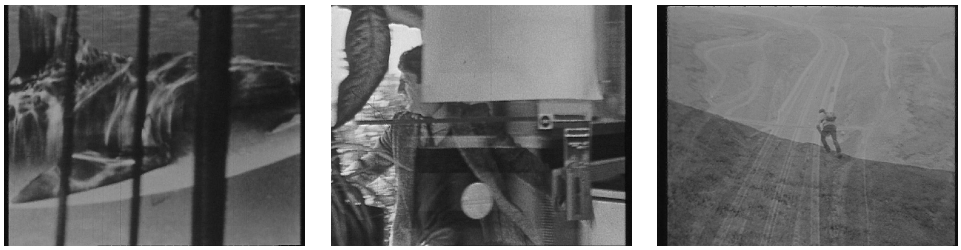
Figure A.5: Example of fast rotation: the helicopter blades.



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

(e) (f) (g) (h)

Figure A.6: Examples of intermittent motion. a-d) The blinking subtype: the plane propellers appear in the image every other frame; e-h) The alternating subtype: the wings of the pigeons are, alternatively, white or black every other frame.



(a) (b) (c)

Figure A.7: Examples of image overlap. a) Lights and shadows on the back of a shark ; b) A person opening a transparent door; c) Shot change of “dissolve” type.

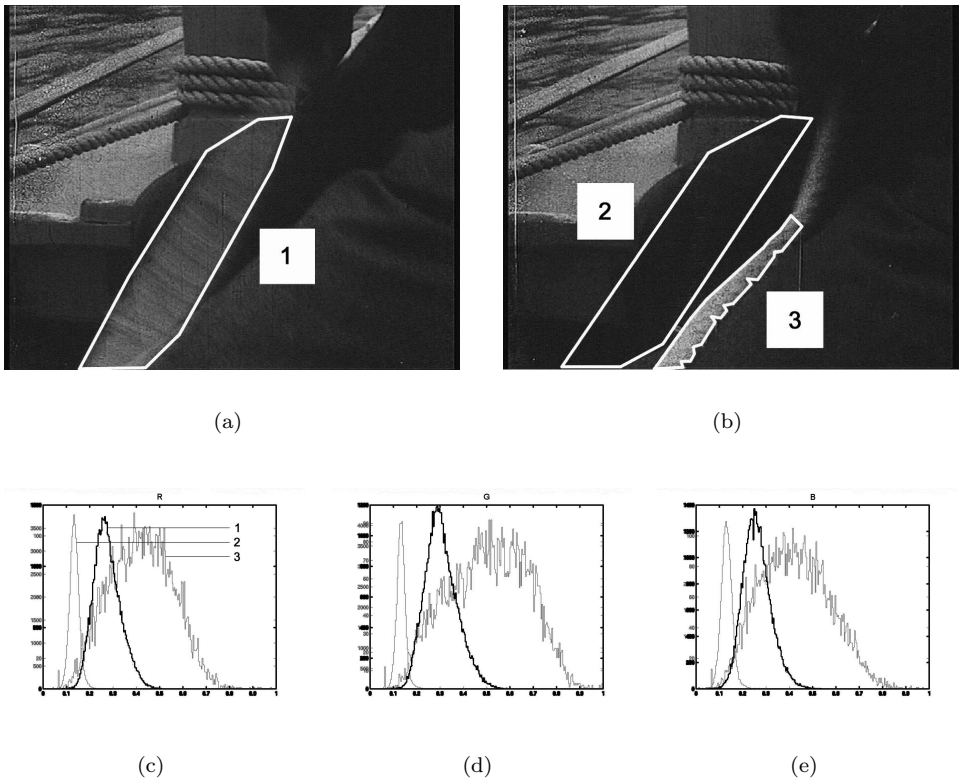
(a) (b) (c)

Figure A.8: Examples of erratic motion. a) Flags by strong wind; b) Fire flames; c) Clutter produced by a shark biting his pray;

motion, and *erratic* motion. In Fig. A.9, we can see a rope moving very fast. As a result, the rope is completely smeared over the background. By taking the statistics of the blurred rope, on one hand, and those of the separate, clearly distinguishable rope and background in the next frame, on the other hand, we can see that the histograms of the blurred rope will fall between the histograms of the clear rope and background. This observation holds for all channels of an RGB image. It gives us an important piece of information when trying to classify CE areas.

In Fig. A.10, we can see the same rope, this time only partially blurred by movement. The statistics of the blurred rope fall again between the statistics of the clear rope and background. This time, however, biased towards the foreground object (the rope). The underlying reason is that the foreground object is only partially blurred, and therefore it "melts" with the background only around its edges.

A problem occurs where artefacts are present together with PM. Here, we consider only artefacts like scratches or blotches. Their appearance is known to be placed at the extremes of the color scale. In general, they are either white or black (semi-transparent blotches are rare). Actually, some of these artefacts can be detected even if they are not completely black or white. Fig. A.11 shows a rope that is blurred, and partially deteriorated by some blotches at the same time. The blotches are an interesting combination of bright and dark areas. In the example at hand, the blotches were actually caused by the vinegar syndrome in combination with some emulsion melting, which manifested as a loss of colour in some of the film layers. As such, the bright areas inside the blotches are not white, but bluish-green. The histograms of the rope and impaired areas together show a main component in each of the RGB channels. These components lie between the components of the clear rope and background (not shown here), indicating a blurred motion. Apart from this, we can observe some "spikes" at the extremes of the histograms. Two such spikes appear on the right end of the G and B channels, corresponding to the bright



(a) (b)

(c) (d) (e)

Figure A.9: Frames from the sequence “rope3”. a) Frame 50, with the rope smeared over the background; b) Frame 51, with clear rope and background; c-e) Histograms of the selected objects for the R, G and B channels.

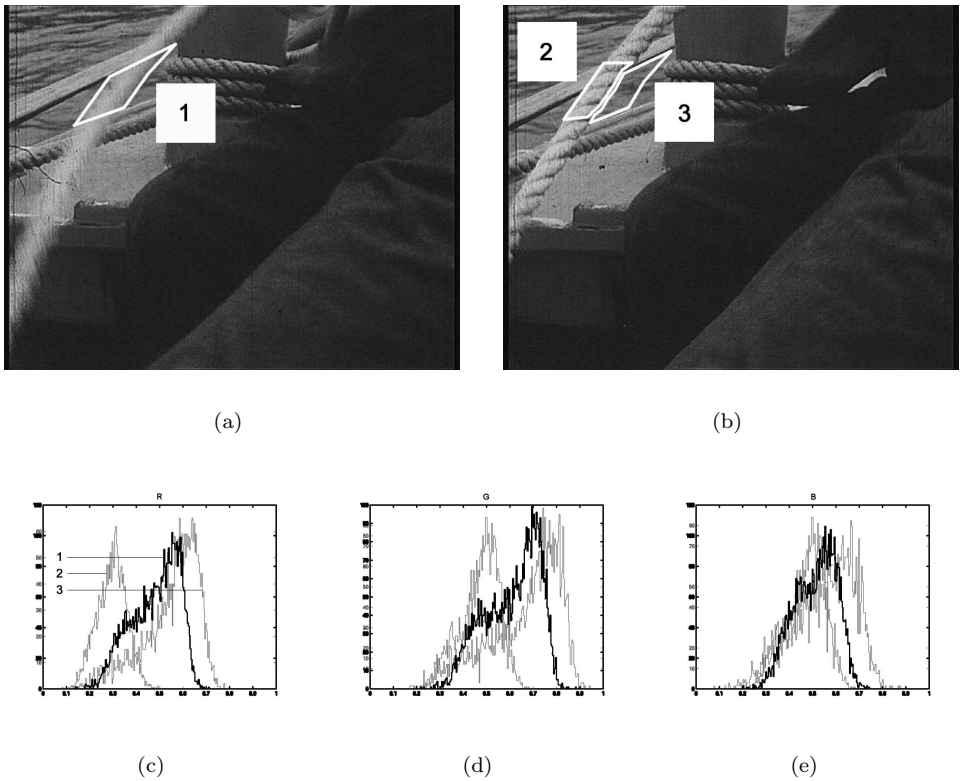
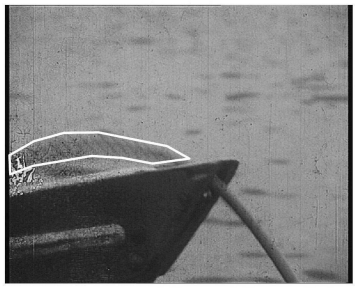
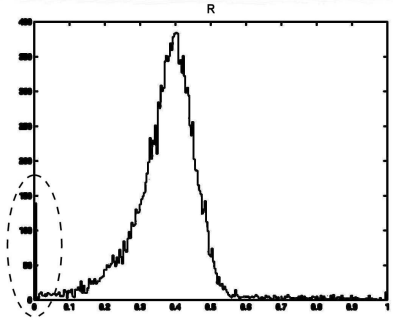


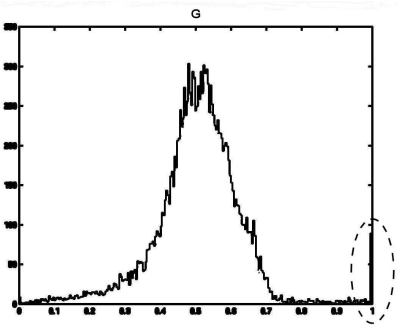
Figure A.10: Frames from the sequence "rope3". a) Frame 48, with the rope smeared over the background; b) Frame 47, with clear rope and background; c-e) Histograms of the selected objects for the R, G and B channels.



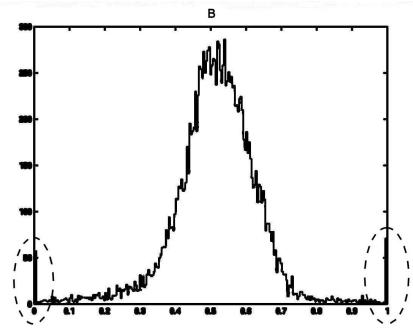
(a)



(b)



(c)



(d)

Figure A.11: Sequence "rope1" (frame 57). a) Rope smeared over the background, with artefacts overlay. The region of interest is marked with a white contour; b-d) Histograms of the region of interest for the R, G and B channels. The contribution of artefacts is marked with dashed ellipses. Artefacts show up only in some of the channels.

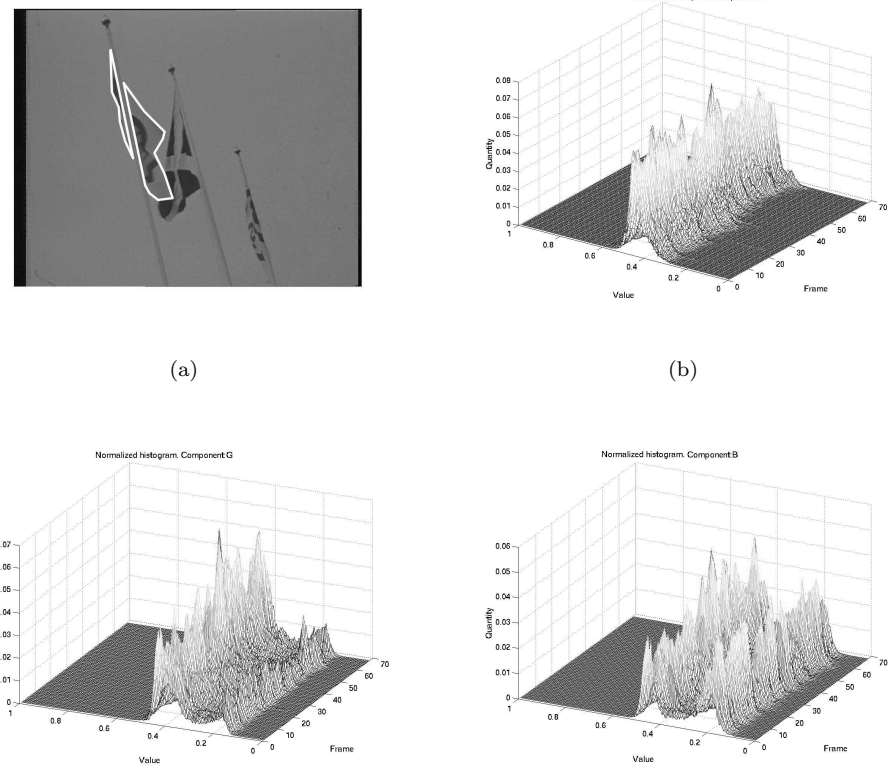
blotches, while the right end of the R channel shows no significant values. These spikes explain the bluish-green appearance of some of the blotches. At the other extreme of the channels, the spikes indicate the presence of the dark blotches (in our case, the dark spikes are more evident in the R and B channels).

We have also performed similar measurements over objects performing erratic motion. The histograms that were extracted are shown in Fig. A.12 and show a high degree of similarity over the entire duration of the erratic motion.

The results presented here indicate that, given a complex events area, blotches and scratches can be separated from pathological motion by carefully studying the histograms in each of the R, G and B channels, or by means of other measurements specific to the given type of pathological motion.

Acknowledgements

The images presented in this Appendix are courtesy of INA (Institut National de L'Audiovisuel), BBC (British Broadcasting Corporation) and RTP (Radiotelevisão Portuguesa).



(a) (b)

(c) (d)

Figure A.12: a) Frame excerpt from a sequence containing flags with erratic motion. The object of interest is marked with a white contour; b-d) 3D plot showing the object's histogram evolution in time for the R, G and B channels.

Bibliography

[1] V.K. Madisetti, D.B. Williams, editors, "The Digital Signal Processing Handbook", Section XI "Image and video processing", CRC Press and IEEE Press, 1998, ISBN 0-8493-8572-5.

[2] A. Rareş, M.J.T. Reinders, J. Biemond, "Statistical analysis of pathological motion areas", The 2001 IEE Seminar on Digital Restoration of Film and Video Archives, pp. 8/1-8/28, London, UK, January 16, 2001.

[3] M.I. Sezan, R.L. Lagendijk, editors, "Motion Analysis and Image Sequence Processing", Kluwer Academic Publishers, 1993, ISBN 0-7923-9329-5.

[4] J. Watkinson, "The Engineer's Guide to Motion Compensation", Snell & Wilcox Handbook Series, 1994.

Summary

The progressive degradation of current film archives poses a serious threat to the preservation of our cultural and technical heritage. Transferring the content onto new film rolls, as well as classical photochemical restoration techniques represent only a short-term solution and have limited effectiveness. Digitization and digital restoration are currently the most viable solutions for long-term preservation and high-quality restoration of filmed material, respectively. They also open the way to more effective searching, reuse and distribution of the archived content.

This thesis covers various aspects of digital restoration applied to archived film. The main focus here lies on blotch restoration in conditions of difficult object motion. In such conditions, wrong motion vectors are extracted that appear as impulsive temporal events. This is also the behaviour of an important group of artefacts: blotches. As a result, false artefacts are flagged in valid content areas. Furthermore, the temporal restoration of artefacts (be they real or false) in an area with difficult motion fails because of the wrong motion information. Currently, the image areas containing difficult motion and/or severe artefacts are protected against restoration, since the restored content would look worse than the original. In chapter 2, we concentrate on the identification of one type of difficult motion, namely motion blur. This is statistically modelled over image segments as linear combinations of the foreground and background objects from adjacent frames. The only artefact detection carried out in these areas uses the extreme (dark or bright) luminosity which also characterizes most blotches. The resulting masks are then used to better separate valid content from artefact, allowing restoration to take place only in the relevant places. Experimental results are presented which confirm the feasibility of the proposed method.

Because of its unreliability in and around objects moving in a complicated way, motion information cannot be used to support the restoration process. We therefore developed in Chapter 3 a restoration method that takes into account only spatial information surrounding blotch-like artefacts. This method identifies and reconstructs object edges interrupted by artefacts, by estimating the pairwise similarity of edges in terms of colour, shape and spatial order of the overall edge configuration. Edges are assumed to have constant curvature locally, resulting in an edge model based

on circular arcs. The recovered object shapes are used to guide a smooth object filling process which is able to reconstruct the specific transitional appearance for both sharp and smooth edges. Extensive experiments were carried out in order to compare our method with a related algorithm, and to validate its performance qualitatively and quantitatively.

In general, textured objects cannot be properly recovered by smooth inpainting. Special texture synthesis algorithms were developed for this purpose. These, however, do not preserve the smoothness of object edges. In Chapter 4, we developed a method which reconstructs smooth object shapes in a similar manner to the method described in Chapter 3, and then fills in the objects by means of a non-parametric texture synthesis algorithm. Based on the same constant curvature assumption, the shape reconstruction step uses intensity and gradient statistics within the objects, together with edge continuity and spatial order measures in order to reconstruct object edges. The texture synthesis algorithm that we used was adapted in our method to take into account the reconstructed object masks. These masks constrain the texture synthesis process such that missing object information is recovered only from the available areas of the same object. Our algorithm is compared with the original texture synthesis algorithm and the inpainting algorithm presented in Chapter 3. The results show the effectiveness of our combined approach.

If artefacts persist in the same place in consecutive frames, the usual temporal restoration algorithms are bound to fail. They would suffer from a lack of information source for the affected areas, i.e., a lack of information which could be pasted from neighbouring frames. Moreover, the motion vectors would be seriously disturbed by the presence of artefacts. In chapter 5, we developed a hybrid method which first uses a simplified version of the edge-based restoration algorithm from Chapter 3 in order to spatially restore each frame, and then uses the restored frames as input for a normal temporal restoration. The mask of artefacts in the temporal restoration step is the same as that in the spatial restoration step. The algorithm is compared with the spatial and temporal restoration algorithms representing each of its two steps. The comparison results show that the combined algorithm inherits the strengths of the two algorithms, while avoiding some of their weaknesses.

Every artefact type has its own characteristics. They are best restored by those algorithms that adapt to their specific properties. An artefact type which has been little studied in the past is the vinegar syndrome. It represents a major problem for all film archives, and its effects accelerate over time because it is an autocatalytic process. The vinegar syndrome is also an artefact which may have multiple appearances, depending on various factors such as the chemical composition of the film and the storage conditions. The method presented in Chapter 6 approaches for the first time a particular form of vinegar syndrome (in combination with emulsion melting), which shows up as a local, partial loss of colour in only a few of the colour layers. Our method uses the information remaining inside the artefact in the less affected layer in order to guide the restoration process, as well as the information surrounding

the artefact for recovering the original colours of the affected areas. Experimental results are presented in which heavily degraded images were successfully recovered.

One of the drawbacks of many temporal restoration algorithms is that they have to rely on motion information provided by general-purpose motion estimators. Usually, these motion estimators were not designed to be robust against incomplete input data, such as image sequences corrupted by artefacts. Also, many motion estimators do not consider multiple image features which can be extracted from the image. Chapter 7 proposes an object tracking algorithm based on multiple features: object position, colour, and Gabor wavelet responses. The algorithm can be used to extract motion vectors at object level. It does not need a complete object mask to extract the features, and uses the online Expectation Maximization algorithm to update its object descriptors from one frame to the other. Comparative experimental results are provided for a setup that includes object colour and position, and a setup that also includes textural features (the Gabor jets).

All algorithms in this thesis were tested on real-life still images and image sequences. Visual results are presented in each chapter. For some restoration algorithms, artificial images and/or artificial artefacts were used as well.

Until now, regions with difficult object motion were not restored, as this would worsen the image quality. In this thesis we have shown that it is possible to restore such areas, with visually pleasing results. The diversity of motion contexts and artefact types existing in practice indicates that a considerable amount of research still needs to be carried out in order to cover the restoration problem in all major types of difficult motion, and to study the integration and reciprocal influence of the different processes involved: restoration, motion estimation, segmentation and pattern recognition.

Summary of the thesis: *"Archived Film Analysis and Restoration"*.

A. Rareş, Delft, March 2004.

Samenvatting

Ons culturele en technische erfgoed wordt ernstig bedreigd door gestage degradatie van de huidige filmarchieven. Het omzetten naar nieuw filmmateriaal of het gebruik van klassieke fotochemische restauratietechnieken biedt slechts kortetermijnoplossingen met beperkt effect. Digitalisering en digitale restauratie zijn momenteel de meest geschikte oplossingen voor langdurige conservering en hoogwaardige restauratie van filmmateriaal. Dit opent tevens de weg naar effectieve zoekmethoden en hergebruik en distributie van gearchiveerd beeldmateriaal.

Dit proefschrift gaat in op verschillende aspecten van de digitale restauratie van gearchiveerd filmmateriaal. Het accent ligt op de restauratie van kleine vlekken ("blotches") bij complexe bewegingen in het beeld. In dit geval worden foute bewegingsvectoren geëxtraheerd, die zich voordoen als impulsachtige temporele gebeurtenissen. Een belangrijke groep artefacten, de vlekken, vertoont hetzelfde gedrag. Delen in het beeld die complexe beweging vertonen worden hierdoor ten onrechte als artefact gedetecteerd. Bovendien faalt de temporele restauratie van artefacten (bij daadwerkelijke artefacten, maar ook bij ten onrechte gedetecteerde artefacten) in gebieden met complexe beweging vanwege de foutieve bewegingsinformatie. Thans worden delen van het beeld die complexe beweging en/of ernstige artefacten vertonen afgeschermd van restauratie. Met als reden dat de gerestaureerde beeldinhoud er slechter uitziet dan de originele. In hoofdstuk 2 concentreren we ons op de identificatie van één bewegingstype, de bewegingsvervorming. Dit type vervorming wordt statistisch en beeldsegmentgewijs gemodelleerd als een lineaire combinatie van voor- en achtergrondobject(en) in opeenvolgende beeldframes. De enige artefactdetectie die wordt uitgevoerd in deze gebieden is gebaseerd op extreme lokale intensiteiten (licht of donker), karakteristiek voor de meeste vlekken. De resulterende maskers worden vervolgens gebruikt om te komen tot een betere scheiding van artefact en beeld, zodat alleen restauratie plaatsvindt op de relevante plaatsen. Experimentele resultaten tonen het nut van de voorgestelde methode.

Zowel binnen als rondom objecten die gecompliceerde beweging uitvoeren kan geen gebruik worden gemaakt van bewegingsinformatie ter ondersteuning van het restauratieproces, dit vanwege de grote onbetrouwbaarheid. Daarom wordt in hoofdstuk 3 een restauratiemethode ontwikkeld die slechts spatiele informatie in beschouwing

neemt in de buurt van vlekachtige artefacten. De voorgestelde methode identificeert en reconstrueert objectranden die onderbroken zijn door artefacten door het schatten van de paarsgewijze overeenkomst in termen van kleur, vorm en spatiele volgorde van de totale randconfiguratie. Randen worden verondersteld lokaal een constante kromming te bezitten, wat resulteert in een op cirkelbogen gebaseerd randmodel. Een artefact kan nu gerestaureerd worden door tijdens het inkleuren op een gladde wijze rekening te houden met de gerestaureerde objectvormen. Hierdoor kunnen zowel scherpe als gladde randen gereconstrueerd worden. Uitgebreide experimenten zijn gedaan om de voorgestelde methode te vergelijken met een gerelateerd algoritme en om de kwaliteit te vergelijken, zowel in kwalitatieve als kwantitatieve zin.

Meestal kunnen textuurobjecten niet voldoende gereconstrueerd worden door een gladde manier van inkleuren. Daarom zijn voor dit doel speciale textuursynthesealgoritmen ontwikkeld. Deze houden echter geen rekening met de gladheid van objectranden. In hoofdstuk 4 wordt een methode ontwikkeld die gladde objectranden reconstrueert op een zelfde wijze als in hoofdstuk 3, en vervolgens de objecten opvult door middel van een niet-parametrisch textuursynthesealgoritme. De vormreconstructiestap maakt gebruik van dezelfde constante krommingaanname en van de statistiek binnen het object (intensiteit en gradiënt) om objectranden te reconstrueren. Daarnaast wordt gebruik gemaakt van randcontinuïteit en spatiele ordening. Het gebruikte textuursynthesealgoritme is aangepast aan onze methode om de gereconstrueerde objectmaskers in beschouwing te kunnen nemen. Deze maskers beperken het textuursyntheseproces op zodanige wijze dat ontbrekende objectinformatie slechts wordt hersteld op basis van de aanwezige gebieden van hetzelfde object. Het voorgestelde algoritme wordt vergeleken met het originele textuursynthesealgoritme en met het inkleuralgoritme uit hoofdstuk 3. De resultaten tonen de effectiviteit van de gecombineerde aanpak.

Indien artefacten zich op dezelfde plaats blijven voordoen in opeenvolgende frames, dan zijn de gebruikelijke algoritmen tot mislukking gedoemd. Ze hebben dan te lijden onder een gebrek aan informatie over de aangetaste gebieden. Bovendien kunnen de bewegingsvectoren aanzienlijk verstoord zijn door de aanwezigheid van artefacten. In hoofdstuk 5 wordt daarom een hybride methode ontwikkeld, die gebruik maakt van een vereenvoudigde versie van het randgebaseerde restauratiealgoritme uit hoofdstuk 3 om eerst elk frame op spatiele wijze te restaureren. Vervolgens worden de gerestaureerde frames gebruikt als input voor een standaard temporele restauratieslag. Het artefacten masker in de temporele restoratie stap is het zelfde als in de spatiële stap. Het algoritme wordt zowel vergeleken met het spatiele als met het temporele algoritme waaruit het algoritme bestaat. Het gecombineerde algoritme, zo blijkt, combineert de sterke kanten van beide algoritmen, terwijl een aantal nadelen wordt vermeden.

Elk artefacttype heeft zo zijn eigen kenmerkende eigenschappen. Ze worden het best gerestaureerd door algoritmen die hier speciaal op zijn aangepast. Een artefact dat slechts heel beperkt is bestudeerd is het zogenaamde azijnsyndroom ("vinegar syndrome"). Het vertegenwoordigt een belangrijk probleem voor alle filmarchieven en

de effecten nemen met de tijd toe vanwege een autokatalytisch proces. Het azijnsyndroom is een artefact dat vele verschijningsvormen kent afhankelijk van factoren, zoals chemische samenstelling van de film en de opslagcondities. De methode die in hoofdstuk 6 wordt beschreven kan gezien worden als een eerste poging om een type azijnsymdroom (in combinatie met het smelten van de emulsielaag) aan te pakken, dat zich manifesteert als een lokaal effect. Namelijk het gedeeltelijke verlies van kleur in slechts een beperkt aantal kleurlagen. De voorgestelde methode maakt gebruik van informatie die nog binnen het artefact aanwezig is uit de minst aangetaste laag om het restauratieproces te begeleiden. Ook wordt informatie rondom het artefact gebruikt voor het herstellen van de originele kleuren van de aangetaste gebieden. Experimentele resultaten worden getoond waarin zwaar aangetaste gebieden in de beelden met succes worden hersteld.

Eén van de nadelen van de vele temporele restauratiealgoritmen is dat zij afhankelijk zijn van bewegingsinformatie die wordt geleverd door algemeen toepasbare bewegingsschatters. Gewoonlijk zijn deze bewegingsschatters niet ontworpen om robuust gedrag te vertonen in het geval van incomplete ingangsdata, zoals beeldreeksen behept met artefacten. Ook houden veel bewegingsschatters geen rekening met meerdere beeldkenmerken die geëxtraheerd kunnen worden uit het beeld. Hoofdstuk 7 introduceert daarom een objectvolgalgoritme dat gebaseerd is op meerdere kenmerken: objectpositie, kleur en Gabor wavelet responsies. Het algoritme kan worden gebruikt om bewegingsvectoren te extraheren op objectniveau. Het heeft geen compleet objectmasker nodig om de kenmerken te extraheren, doch maakt gebruik van het "Expectation Maximization" (EM) algoritme om de objectkenmerken te updaten in de opeenvolgende frames. Vergelijkende experimentele resultaten worden gegeven voor een opstelling met alleen kleur en positie en een opstelling die ook textuurkenmerken (Gabor jets) bevat.

Alle algoritmen in dit proefschrift zijn getest op natuurlijke beelden en beeldreeksen. Visuele resultaten worden in elk hoofdstuk getoond. Voor sommige restauratiealgoritmen is tevens gebruik gemaakt van kunstmatige beelden en/of artefacten.

Tot nu toe werden gebieden met complexe beweging afgeschermd van restauratie, als een soort vangnet om een grotere aantasting van de beeldkwaliteit te voorkomen. In dit proefschrift hebben we laten zien dat het wel degelijk mogelijk is om deze gebieden te restaureren met goede visuele resultaten. Het grote aantal mogelijke (complexe) bewegingsvormen en het grote aantal verschillende artefacten dat zich in de praktijk voordoet maakt duidelijk dat er nog een aanzienlijke onderzoeksinspanning nodig is op het gebied van de filmrestauratie. Ook mag hierbij niet vergeten te worden de integratie van de verschillende processen en de studie van hun onderlinge beïnvloeding, zoals restauratie, bewegingsschatting, segmentatie en patroonherkenning.

Samenvatting van het proefschrift: *"Archived Film Analysis and Restoration"*

A. Rareş, Delft, Maart 2004

Acknowledgements

This is not the easiest part of my thesis... I am so much indebted to so many people and in so many ways, that it would be impossible to name them all here. I just hope I manage to mention most of those who, in one way or another, have contributed to my thesis. Thank you, all of you.

First of all, I am grateful to my promoter, Jan Biemond, for his knowledgeable, kind and firm guidance, which ensured that my research stayed on track. After the first year of PhD research, he gave me the opportunity to continue my activities in the BRAVA project, and became my promoter. Thank you, Jan, for allowing me to discover what a beautiful subject film restoration is. I am also grateful to my co-promoter, Marcel Reinders, who supervised me for the entire PhD period. He contributed to my work with a lot of energy, inspiration and careful attention to "polishing" my articles. Marcel, I learned from you how to look at problems from many points of view, and I probably owe most of my scientific writing skills to you. Many thanks to Emile Hendriks, too, for supervising me in the first year, together with Marcel. Emile, I always learned something from our discussions.

Cees, we have been roommates for most of my PhD studies ... until you went to a brighter place. I enjoyed so much our French-Dutch-English discussions about dolphins, wavelets, and anything else. Most of all, *mon "vieux"*, I always felt you were the younger spirit in our room.

Bangjun, you have been my roommate for some time, and a nice friend always, together with Xiaohong. I am sorry I did not use your image in my thesis, as you did with mine. Actually... I tried to restore images of you, but they always came out as "Lena" :-) Ioana and Iwan, we have been roommates only for a short time, but just enough to discover your good-tempered natures and to enjoy our daily discussions. I am so much indebted to Erik Mouw, for his Linux tips and tricks, but especially for getting me out of technical troubles in key moments. Erik, your friendly help was priceless! I would like to thank the ICT staff: Annett, Anja, Jeanine and Sanne for their readiness to help with the bureaucratic documents and procedures; and Ben and Hans for making sure that my experiments went smoothly. Many thanks to Inald, Cor, Eugene and Ivo Shterev for our useful scientific discussions. Jesper, your advice on algorithms was highly valued, and your good mood was contagious...

maybe ;-) Everyone from the ICT group deserves thanks for contributing to the friendly atmosphere. I thoroughly enjoyed our dinners together, the group trips and all other common activities.

Working in a EU project like BRAVA has been a rewarding experience in many ways, and most of the work presented in this thesis is related to it. I would like to thank all people involved in this project for our fruitful interactions. Special thanks to Anil for the interesting and informal discussions during our meetings, and for his skillful guidance of the research activities in our project.

My new workplace, the LKEB group of the Leiden University Medical Center, has welcomed me with open arms. I would like to thank Hans Reiber and Gerhard Koning for allowing me to combine the work duties with the final thesis preparations in a flexible manner.

I would also like to thank my professors Nicolae Ţăpuş and Florica Moldoveanu from the Faculty of Automatic Control and Computers from Bucharest, who have paved the way towards this PhD.

My friends have offered me support and the much needed relaxation outside the working hours. Elena and Viorel, Gabi, Alina and Călin Zamfirescu, Călin Marioara, Cristina Enache, Oana, Cristina Soviany and her playful children Răzvan and Andrei, Monica and Stelian, Laura and the rest of my friends from the Netherlands, I enjoyed your company all these years. Florin and Amalia, you were far away, but always next to me when needed.

The community gathered around the Romanian Orthodox Church from the Netherlands has always provided a pleasant atmosphere.

I am especially grateful to all those who offered me precious guidance and advice in day-to-day life.

I would like to thank my sister for staying in close contact with me all these years. Cori, I love the many beautiful drawings you have sent to me.

Last but not least, I cannot thank my parents enough for their love and support, and for encouraging me to pursue this PhD. *Mulţumesc, mamă, mulţumesc, tată!*

Curriculum Vitae

Andrei Rareş was born on the 2^{nd} of July 1972 in Bucharest, Romania. He graduated from the Bucharest Informatics High School in 1991. He then went to study Computer Science at the Faculty of Automatic Control and Computers of the "Politechnica" University of Bucharest, from which he received the B.Sc. degree in 1996. The topic of his graduation project was a groupware development platform based on distributed objects, and the work was carried out during a four-month scholarship at INRIA (Institut National de Recherche en Informatique et en Automatique) Grenoble, France. In 1997, he received the M.Sc. degree from the same faculty, for a research project about unification algorithms for natural language grammars. In 1995 and 1996 he taught programming classes at the Bucharest Informatics High School. Between 1996 and 1999 he was a teaching assistant at the Faculty of Automatic Control and Computers.

In 1999, he started his PhD with the Information and Communication Theory Group of the Faculty of Electrical Engineering, Mathematics and Computer Science of the Delft University of Technology. In the first year, he worked in the MCCWS project ("Mediated Communication in Collaborative Work and Services"), under the supervision of Marcel Reinders and Emile Hendriks. The project was aimed at developing a multi-viewpoint videoconference prototype that would enable people to communicate with each other during collaborative activities. In 2000, he joined the BRAVA research project ("BRoadcast Archives restoration through Video Analysis") of the European Union, under the supervision of Professor Jan Biemond and Marcel Reinders. The focus of this project was the development of algorithms for the restoration of degraded old films and video. During his PhD, he also acted as a teaching assistant in the practicals of various courses for digital image and video processing. Since 2003, he has been working as a post-doc in the Laboratory for Clinical and Experimental Image Processing at the Leiden University Medical Center. Here, he has been developing algorithms for the detection, analysis and quantification of blood vessel stenoses in X-ray images.

Propositions

belonging to the thesis

Archived Film Analysis and Restoration

by Andrei Rareş

1. The most complex problems in video restoration can only be solved if the fields of restoration, motion estimation, object tracking, segmentation and pattern recognition cooperate intimately.

   This thesis, Chapter 8

2. Long-term memory should be an essential ingredient in the next generation of temporal restoration algorithms.

   This thesis, Chapter 8

3. Complex event detection should be part of any motion estimator.

4. The blank boundaries of film frames are as crucial for restoration as the frames themselves and should therefore be digitized as well.

5. Publishing new algorithms without describing their known limitations is like selling medicines without specifying their contraindications or side effects.

6. The search for life on Mars shows how little we care about life.

7. Children should learn to swim in the ocean of information.

8. Placing the Backspace button next to the Enter button is a mistake.

9. Romania's bet on information technology can only succeed if it includes a proper educational system.

10. The Dutch way of using bikes in day-to-day life should be copied in Romania.

These propositions are considered defendable and as such have been approved by the supervisor, prof.dr.ir. J. Biemond.

Stellingen

behorende bij het proefschrift

Archived Film Analysis and Restoration

door Andrei Rareş

1. De meest complexe problemen op het gebied van videorestauratie kunnen slechts worden opgelost als de onderzoeksgebieden restauratie, bewegingschatting, 'object tracking', segmentatie en patroonherkenning nauw samenwerken.

   Dit proefschrift, Hoofdstuk 8

2. Een lange termijn geheugen zou een essentieel onderdeel moeten zijn van de volgende generatie temporele restauratiealgoritmen.

   Dit proefschrift, Hoofdstuk 8

3. De detectie van complexe gebeurtenissen zou onderdeel moeten zijn van elke bewegingsschatter.

4. De lege randen van filmframes zijn even belangrijk voor de restauratie als de frames zelf en zouden dus ook gedigitaliseerd dienen te worden.

5. Het publiceren van nieuwe algoritmen zonder een beschrijving van bekende beperkingen is hetzelfde als het verkopen van medicijnen zonder het noemen van de contra-indicaties of neveneffecten.

6. Het zoeken naar leven op Mars laat zien hoe weinig we ons bekommeren om het leven.

7. Kinderen zouden moeten leren zwemmen in de informatieoceaan.

8. Het plaatsen van de 'Backspace' toets naast de 'Enter' toets is een vergissing.

9. De inzet op informatietechnologie door Roemenië heeft slechts kans van slagen indien een degelijk onderwijssysteem er deel van uitmaakt.

10. Het Nederlandse gebruik van de fiets in het dagelijkse leven zou moeten worden overgenomen door Roemenië.

Deze stellingen worden verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotor, prof.dr.ir. J. Biemond.