Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

Diploma Thesis

Colour Movies Scratch Restoration

Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics

Academy of Sciences of the Czech Republic
Institute of Information Theory and Automation
Pattern Recognition Department

filipj@utia.cas.cz

January 2002

Jiří Filip

Abstract

Digital restoration of the scratches in image sequences is essential for recovering of the old movies as well as for online processing in scanning and duplicating machines.

The first part of this thesis describes review of digital movie representation. Next is introduced the survey of contemporary methods for motion estimation and scratch restoration in image sequence. The results of two simple motion estimation methods are discussed.

The main contribution of this thesis is a new algorithm for the scratch restoration in multi-spectral image sequence based on causal adaptive multidimensional prediction by 3D and 3.5D causal autoregressive models. The predictor use available information from corrupted pixel surrounding due to spectral, spatial and temporal correlation in multispectral image data, and adaptively updates it's parameters. The model assumes white Gaussian noise in each spectral layer, but layers can be mutually correlated. Experimental results captivate that proposed method easily outperforms any mentioned classical scratch restoration method.

Abstrakt

Digitální restaurace filmových sekvencí je nezbytná pro záchranu archívních filmů, opravu chyb způsobených duplikací originálu nebo digitálních dat znehodnocených během snímání.

První část této práce shrnuje přehled používané reprezentace digitálních filmových dat. Stejně tak je zmíněn přehled současných metod detekce pohybu a restaurace poškozených dat v obrazové sekvenci. Jsou zde zhodnoceny výsledky dvou jednoduchých metod detekce pohybu.

Hlavním přínosem této práce je nový algoritmus pro odstranění poškozených dat v barevné filmové sekvenci, založený na výpočtu kauzální, adaptivní, vícedimenzionální predikce pomocí 3D a 3.5D kauzálního autoregresivního modelu. Prediktor používá pro odhad svých parametrů dostupnou informaci o spektrální, prostorové a časové závislosti v datové struktuře v okolí každého poškozeného pixelu. Na základě těchto znalostí si adaptivně upravuje své parametry. Navržený model předpokládá bílý Gausovský šum v každém spektrálním pásmu, přičemž data v jednotlivých pásmech mohou být vzájemně závislá. Z experimentálních výsledků vyplývá, že navržená metoda má nejlepší výsledky v porovnání s klasickými metodami restaurace obrazu, popsanými v práci.

Acknowledgements

I'd like to thank to my tutor ing.Michal Haindl, DrSc. for his advises as well as for patient repairing of my often mistakes. He was always willing to help me and without his help it wouldn't be possible to reach any part of my task in this thesis. I'd like also to thank Pattern Recognition department at Institute of Information Theory and Automation AS CR that they enabled me to work in the Institute during carrying out of this thesis.

Poděkování

Rád bych poděkoval mému školiteli panu ing.Michalu Haindlovi, DrSc., za jeho vždy ochotnou, trpělivou pomoc a cenné rady, stejně tak jako za pečlivou opravu tohoto textu. Bez jeho pomoci by nebylo možné realizovat žádný z úkolů mé diplomové práce. Rád bych také poděkoval pracovníkům Oddělení rozpoznávání obrazů v Ústavu teorie informace a automatizace AV ČR za jejich vlídné přijetí a za to, že mi umožnili pracovat v Ústavu na této diplomové práci.

Declarations

I declare that I carried out this diploma thesis myself and mentioned entire used information sources in bibliography.

I declare that I don't have any objections against borrowing of this diploma thesis with agreement of Department of Cybernetics and it's publishing under my name.

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně, pouze s použitím literatury zmíněné v citacích.

Prohlašuji, že nemám námitek proti zapůjčování této diplomové práce se souhlasem Katedry kybernetiky, ani proti zveřejnění diplomové práce nebo její části pod mým jménem.

V Praze dne 12. ledna 2002

List of Notations and Acronyms

r_1, r_2, r_3, r_4	row, column, spectral and time index
r	multiindex $r = \{r_1, r_2, r_3, r_4\}$
s_1, s_2, s_3, s_4	row, column, spectral and time shift
$\bullet$	means all possible values of the corresponding index
$\mathbf{d}_{r_1,r_2,r_4}$	displacement vector between frames $r_4, r_4 - 1$ on frame coordinates r_1, r_2
d	the number of spectral bands
I_r	causal contextual neighbourhood
η	cardinality of causal contextual neighbourhood I_r
γ	predictor parameter matrix
Y_r	predicted values on location defined by multiindex r
X_r	data vector on location defined by multiindex r
$\beta(r)$	the number of model movements on image plane
σ	length of model history
ρ	exponential forgetting factor
ξ	the number of quantization levels in each spectral band
$\mathcal{S}$	scratch coordinates set (corrupted data)
MAD	**M**ean **A**bsolute **D**ifference
MAD_A	**M**ean **A**bsolute **D**ifferencee averaged in all spectral bands

2D, 3D, 3.5D, 4D	two, three, three and half, four **D**imensional
BM	**B**lock **M**atching
CAR	**C**ausal **A**uto**R**egressive
CN	**C**ontextual **N**eighbourhood
GB	**G**radient **B**ased
MC	**M**otion **C**ompensation
ME	**M**otion **E**stimation
MSE	**M**ean **S**quared **E**rror

Contents

Chapter 1

Introduction

From the dawn of cinematography[1] till today, movies have been recorded on media which suffer by different degradation processes during longtime storage. In records on celluloid or magnetic material defects often occur caused by extensive usage of film material as well as defects caused by mold or humidity in improper storage environment. This corruption occurs also when copying film negatives.

A short overview of defects [30] which can appear in historical and sometimes also in contemporary film materials follows.

Dust and dirt are very often defects in historical movie. They are caused mostly by the pollution during the movie acquisition or the duplication process. The pollution is local on the film material, thus the visible effect in a movie is the appearance of bright or dark spots only on some frames of the movie.

Scratches appear in the direction of the film strip over more than one frame of the film. Scratches are caused by film transport or by the developing process when there are particles in the developer's machine. Scratches caused by the film transport are exactly parallel to the film-strip perforation, but scratches caused by mentioned particles can change their position up to 5% of the film width a few frames of the movie sequence.

Missing or heavily corrupted frames are result of improper storage environment of the film material and the long storage time. In some papers we can find notion *blotch* which stands usually for large corrupted area.

Mold results from an improper storage environment for the film materials. It appears periodically every twenty to thirty frames and produces observable brightness or colour variations. Different chemical reactions cause different types of mold. There is possible to obtain lightening, darkening and local discontinuous defects caused by the mold.

[1] the first film projection was performed in Paris on 28th December 1895 by Auguste and Louise Lumière

Flickering is visible global brightness or colour variation between successive movie frames. Inside a frame the variation is homogeneous. This defect occurs mostly in historical film and it is caused by the different exposure time during the movie acquisition. Flickering can be caused in modern cameras by interference between the lighting and the exposure.

Jittering is defect which appears during conversion of the analogue video signal into a digital form. During sampling the frame grabber must synchronise to the incoming lines and frames. In case that incoming video signal is a noisy one or is affected by timing distortion, the frame grabber is unable to locate start and end of lines and frames in this video signal. Hence a digitised image is obtained where the lines are displaced relative to each other due to bad line synchronisation.

Image vibrations originate in lack mechanical accuracy of film-transporting systems in old movie cameras or duplication equipment. Image vibration can originate also from unstable camera attachment during the movie acquisition.

Captions or subtitles appears on a part of the movie and are created in the duplication step.

To save historical materials in the film and TV archives, original films have to be recorded on time-stable media (usually with digital supports). The constant growth of communication media (satellite and cable TV, video on demand, etc.) widing a new market for this movie resources. In order to keep a copy of the film sequence as close as possible to the original one, a restoration phase is necessary. It is necessary to employ digital film restoration. The digital film restoration has to solve several kinds of defects frequently encountered in film sequences, as was mentioned above. For each kind of the defect usually different kind of restoration algorithm is needed.

There are already several manual and semiautomatic systems for digital film restoration as the Cineon system by Kodak, the Domino system by Quantel, the Flame system by Discreet logic, the Matador system by Avid, but they are aimed mainly for generating of visual effects and movie editing with some restoration skills but their ability for on-line restoration are still constrained.

Between years 1994-1999 there was carried out an international project AURORA (AUtomatic Restoration of ORiginal film and video Archives) [28, 7]. The main aim of the AURORA project was an automatic real time movie restoration with control of level of correction by the user. The project is considered to be very successful. The original goal, of having a complete television archive restoration system has been reached but significant improvements and developments are still progressing. In 1999 started the next project BRAVA (Broadcast Archives Restoration Through Video Analysis) [29] which aims at developing further the results of the AURORA project.

Digital sequence restoration is quite new area for study. It's importance grew with coming of more powerful computers in late nineteens, hence it was possible to apply it for recovering of corrupted movies [32]. Before this date most of the researchers in image processing has concentrated on stills (except video compression research). The first approach in sequence restoration was only motion compensated median filtering. This method is fast and enables many modifications by the filter masks shape [17, 18, 3]. Later researchers tried to treat image sequences as a spatio-temporal signal. The model based methods of reconstruction were developed. The 3D autoregressive modelling for suppressing of scratches in black&white movies was introduced also in [17, 18] as well as filtering by the Wiener filter. More sophisticated model-based methods use most often Markov random field type of models either in the form of wide sense Markov (regressive models) or strong Markov models. Next approach in interpolation of missing data in image sequences based on combination of autoregressive models and Markov-random fields is in [34, 19]. Another Monte Carlo methods as Gibb's sampler and Markov chains applied for missing data reconstruction in colour image sequences are mentioned in [4, 10]. These methods have the main problem in time consuming iterative solution and have to solve also iteration stopping problem. Additional details of some contemporary missing data restoration methods for image sequences are mentioned in section 4.3.

In this thesis we present a scratch removal algorithm. The scratch defect don't follow exactly the above proposed scratch definition. So by the scratch notion, in this thesis, we mean every coherent region with missing data (simultaneously in all spectral bands) in a movie frame.

A typical image sequence restoration system would involve usually 3 steps: motion estimation, detection of the missing data regions, restoration of detected regions. Our field of concern in thesis is mainly the restoration step by model-based approach.

The modelling of colour image sequence isn't easy task, because it requires four independent indices. Horizontal and vertical spatial coordinates in the frame, spectral band and time coordinates. This leads to a 4D model of colour image sequences, where processed data are spectrally, spatially and temporally correlated. Such approach has unfortunately huge computational demands, requires a huge training data set and leads to non-linear parameter estimation, hence it has to solve problem when to stop iterative process and algorithm is then sometimes numerically unstable. Due to this reasons the 4D model does not have analytic solution. To overcome this drawback the 3.5D model is suggested. The assumption for the analytical solution of the 3.5D model is white Gaussian noise in each spectral layer. The reason for "3.5D" notion is caused by the fact, that we neglect the part of information about temporal correlation between data in consecutive frames in the image sequence. Thanks to this we can treat data from different frames (specified by the contextual neighbourhood) in the same way, so we attach to each data information about its shift according to predicted pixel placement. The problem is then how to determine the spatial shift of moving regions between successive frames, because the same region hasn't the same placement in both frames due to its motion during the image sequence acquisition. To solve this task some motion estimation in colour image sequence is necessary. However developing of motion estimation method wasn't goal of this thesis, so an simple block matching motion estimation method and also its iterative improvement [37] were implemented. We tried to employ these methods for motion compensation in the proposed restoration method.

The proposed new type of scratch removal algorithm is based on a causal adaptive multidimensional prediction. The predictor use available information from the failed pixel surrounding due to spectral and spatial correlation of multispectral data but not any information from failed pixel itself. Predictor parameters cannot be directly identified so a special approximation is introduced.

The thesis is divided to chapters as follows.

Chapter 2: Digital Movie Representation A short review of common digital movie formats is introduced and is mentioned difference between lossless and lossy compression methods.

Chapter 3: Methods for Motion Estimation This chapter describe principle of common Motion Estimation methods (Block Matching and Gradient methods) and discuss their properties. Image sequences for testing are described. The results of two implemented simple motion estimation methods discussed. Finally applying of motion compensation for the scratch restoration is described.

Chapter 4: The Scratch Restoration Methods Overview A simple methods for scratch restoration as averaging, Median filtering, Linear and Quadratic interpolation are introduced. These simple methods as well as other previously published methods for the scratch restoration are discussed.

Chapter 5: The Scratch Restoration by the Causal Autoregressive Model A new method based on causal adaptive multidimensional prediction by 3D and 3.5D CAR model is proposed. The employing of this method for the scratch restoration is introduced.

Chapter 6: Results and Discussion Used testing criteria are described. Obtained results of 3D and 3.5D CAR model are compared with the classical method's performance and properties of proposed method are discussed.

Chapter 7: Conclusions and Further Research This chapter summarises achieved results of image restoration based on 3D and 3.5D CAR model and mention also problems in image sequence restoration by subject of a further beyond scope of research.

Chapter 2

Digital Movie Representation

Digital video processing and storage requires a digital video representation. Several video formats were developed during last period. A short overview of the most important formats which are popular for digital movie presentation and processing is subject of this chapter. This and the additional information in this field are in [9, 35, 33, 20].

Digitising a video sequence results in extremely high data rates. For example, a television image with a resolution of 720 x 576 pixels and a colour depth of 16 bits produces a data stream of 1.35 MB per individual frame. As 25 frames per second are required to avoid jumpy video scenes, a gigantic data flow of 33.75 MB/s is produced! For this reason, it is absolutely inevitable that video signals are compressed so they can remove or reorganise data in order to reduce the size of digital files. One distinguishes between lossless compression methods and lossy compression.

2.1 Compression

2.1.1 Lossless compression

Lossless compression retains the original data so that the individual image sequences remain the same after compression. Most lossless compression techniques use run length encoding that removes images areas that use the same colour. However, the compression rate is not better than 3:1, depending on the complexity of individual images. In practise, lossless methods play a low-key role due to their low compression rates. There are two common code optimisation techniques as follows.

Run Length Encoding Run Length Encoding works by grouping elements that repeatedly occur and by encoding them with a count value. As the counter also requires space, elements that occur twice or three times remain uncoded. This type of compression is used in the graphics field, for example to display smooth surfaces with a minimum byte count.

Huffman Encoding The Huffman method encodes often-repeated elements with a few bits and rare ones with more bits. The number of times the elements occur is used to determine the respective bit encoding method.

2.1.2 Lossy compression

Lossy compression methods attempt to remove image information that is unlikely to be noticed by a human observer. These methods do not retain the original data and some image information is lost. The volume of data lost depends on the degree of compression. In practise, time compression is gaining in importance. With this method, the resulting data volume for each individual video sequence is optimised. Typical folder of lossy compression is MPEG format, which is described in section 2.2.3.

2.2 Movie Formats

2.2.1 The AVI and ASF Format

One of the oldest formats in the computer world is AVI. The abbreviation 'AVI' stands for “Audio Video Interlaced”. This video format was created by Microsoft, which was introduced along with Windows 3.1. AVI, the proprietary format of Microsoft's “Video for Windows” application, merely provides a framework for various compression algorithms such as Cinepak, Intel Indeo, Microsoft Video 1, Clear Video or IVI. In its first version, AVI supported a maximum resolution of 160 x 120 pixels with a refresh rate of 15 frames per second. The format attained widespread popularity, as the first video editing systems and software appeared that used AVI by default. However, there were a number of restrictions leading from format incompatibility, because each manufacturer adapted the open AVI format according to his own requirements. AVI is also subject to additional restrictions under Windows 95, which make professional work at higher resolutions more difficult. Despite its age and numerous problems, the AVI format is still used in semi-professional video editing cards. Many TV cards and graphic boards with a video input also use the AVI format. These are able to grab video clips at low resolutions (mostly 320 x 240 pixels). Colour depth specification: Microsoft Video 1 (CRAM) depth 8 and 16. SuperMac Cinepak (CVID) depth 24. Uncompressed (RGB) depth 8. Run length encoded (RLE8) depth 8. File extension: avi.

The ASF stands for “Active Stream Format”. An active stream format is defined and adopted for a logical structure that encapsulates multiple data streams. The data streams may be of different media. The data of the data streams is partitioned into packets that are suitable for transmission over a transport medium. The packets may include error correcting information. The packets may also include clock licenses for dictating the advancement of a clock when the data streams are rendered. The format of ASF facilitates flexibility and choice of packet size and in specifying maximum bit rate at which data may be rendered.

Error concealment strategies may be employed in the packetization of data to distribute portions of samples to multiple packets. Property information may be replicated and stored in separate packets to enhance its error tolerance. The format facilitates dynamic definition of media types and the packetization of data in such dynamically defined data types within the format. File extension: asf.

The inner structure of AVI and ASF format isn't published by one can assume that it is based on MPEG technology.
Advantages: Wide popularity all over the world.
Disadvantages: AVI does not use a single common time-line for playing images and sounds so these movies may sometimes exhibit some audio/video synchronisation problems. Restricted resolution.

2.2.2 Apple's Format

The MOV format which originated in the Macintosh world, was also ported to x86 based PC's. It is the proprietary standard of Apple's Quicktime application that simultaneously stores audio and video data. Between 1993 and 1995, Quicktime (ISO standard for digital media) was superior to Microsoft's AVI format in both functionality and quality. The functionality of the latest generation (Quicktime 4.0) also includes the streaming of Internet videos (the real-time transmission of videos without the need to first download the entire file to the computer). QuickTime is not really a movie format, but is a software development package that allows the synchronisation of multiple media streams (video, text, sound and music). Despite this, Apple's proprietary format is continually losing popularity with the increasing use of MPEG. Video clips coded with Apple's format are still used because of Quicktime's ability to run on both Macintosh and x86 computers. Format specifications: Apple Graphics (RLE) depth 1,8,16 and 24. Apple Animation (SMC) depth 8. Apple Video (RPZA) depth 16. SuperMac Cinepak (CVID) depth 24. Supports multiple video tracks. Supports animations with multiple codecs. File extensions: qt, mov.
Advantages: Allows the synchronisation of multiple media streams and streaming of Internet videos. Ability to run on Macintosh and x86 computers.
Disadvantages: Decreasing support of this format with the increasing use of MPEG.

2.2.3 MPEG Formats

The MPEG formats are by far the most popular standards. The abbreviation MPEG stands for "Motion Picture Experts Group" - an international organisation that develops standards for the encoding of moving images. In order to attain widespread use, the MPEG standard only specifies a data model for the compression of moving pictures and for audio signals. It means that it is not a definition of coding standard, it is assumed to be only constraint on the data bit stream. In this way, MPEG remains platform independent. One can currently differentiate between five standards: MPEG-1,MPEG-2, MPEG-4, MPEG-7, and

MPEG-21. Let's take a brief look how MPEG Compression works.

How MPEG Compression works

Subdivision into Macro Blocks The image is split into macro blocks with a size of 8x8 or 16x16 pixels that are separately handled. In the next step, the difference between the macro block in image N and the moved macro block in image N+1 is established (see figure 4). This error image has to be coded and saved along with the displacement vector in order to monitor the subsequent error accumulation. Memory space requirements are minimised if the difference between the moved macro blocks is so small that it's possible to completely forget encoding the difference.

In order to code the images (the next step), the following macro block types, shown in Fig.2.1, are used: the “I” frames are images which are saved in the equivalent JPEG format and are not dependent on the previous or subsequent frames. Only the “I” frame (intra coded image) permits direct access to the individual sections or still frames in a clip. In contrast, “P” frames (predicted image) are predicted from the previous “I” frame. The most universal image types are the ”B” frames (bi-directionally interpolated image) that are interpolated from the previous, or following P or I frame.

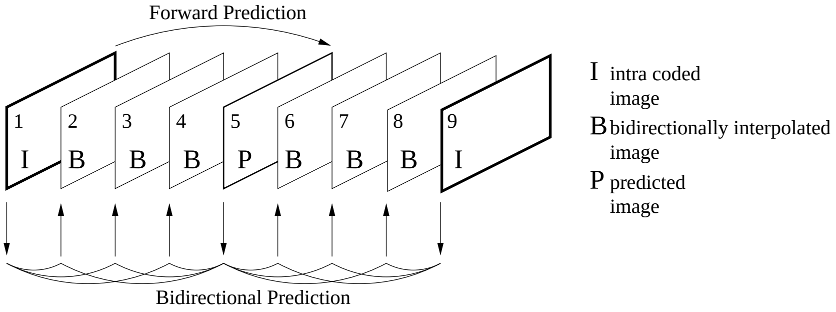

Figure 2.1: Interframe coding within MPEG format.

The key role in prediction and interpolation play successful estimate of a motion vectors. For fast motion prediction is commonly used Block Matching (BM) motion estimator. The motion vector is obtained by minimising a cost function measuring the mismatch between a block and each predictor candidate.

Motion-compensated interpolation (bidirectional prediction) is a multiresolution technique. A sub-signal with low temporal resolution is coded and the full-resolution signal is obtained by interpolation of the low-resolution signal and addition of a correction term. The signal to be reconstructed by interpolation is obtained by adding a correction term to a combination of a past and a future reference. The motion vector is obtained by minimising a cost function measuring the mismatch between a block and each predictor candidate.

Discrete Cosine Transformation (no data loss) The block 8x8 is assigned to a colour value matrix that is used for the discrete cosine transformation (DCT). DCT suppresses highly frequent image parts that aren't apparent to the human eye. DCT is based on Fourier transformations that present any signal as merged (superimposed) sine signals of different frequencies and amplitudes. The Fourier transformation yields frequency and amplitude distribution values from the location of pixel values in an image. This means that large, regular areas in the image are then represented more in the lower frequency parts, whereas finer details are in the higher range. In our concrete example, DCT transforms the displayed 8x8 macro block into an 8x8 coefficient matrix. The value in the upper left corner of the coefficient matrix contains the lowest frequency parts. This coefficient at location 0,0 is normally referred to as the DC coefficient, whereas the remaining 63 coefficients are termed AC coefficients (AC = Amplitude Coefficient). As there is normally a solid relationship between the DC coefficients of two subsequent 8x8 blocks, the DC coefficient is encoded as the difference to the predecessor. The remaining 63 AC coefficients are sorted according to a set pattern.

DCT concentrates the signal energy of a block in the lowest coefficients, especially in the DC coefficient. The higher AC coefficients are normally 0 or almost 0, because the main part of the visual information of an image lies in a continuously distributed range of values in the lower frequency area. Edges normally only constitute a small part of an image. After the discrete transformation, the coefficients are quantised in order to attain an additional compression improvement.

Quantising (high data loss) Quantising is a process that involves adapting the data encoding precision to the capacity of human perception. Due to the fact that the eye is not able to monitor changes to fine details very well, the observer will not notice the slightly reduced display precision. Thus quantization of the DCT coefficients is important because the combination of quantization and run-length coding contributes the most of the compression. Finally, adaptive quantization is one of the key tools to achieve visual quality.

MPEG's formats overview

MPEG-1 (ISO/IEC 11172-3) was released in 1993 with the objective of achieving acceptable frame rates and the best possible image quality for moving images and their sound signals for media with a low bandwidth (1 Mbit/s up to 1,5 Mbit/s). The design goal of MPEG-1 is the ability to randomly access a sequence within half a second, without a noticeable loss in quality. For most home user applications and business applications (image videos, documentation), the quality offered by MPEG-1 is adequate. MPEG-1 reduces the original data volume to about 1:35. File extensions: mpg, mpeg.

MPEG-2 (ISO/IEC 13818-3) has been in existence since 1995 and its basic structure is the same as that of MPEG-1, however it allows data rates up to 100

Mbit/s and is used for digital TV, video films on DVD-ROM and professional video studios. MPEG-2 allows the scaling of resolution and the data rate over a wide range. Due to its high data rate compared with MPEG-1 and the increased requirement for memory space, MPEG-2 is currently only suitable for playback in the home user field. The attainable video quality is noticeably better than with MPEG-1 for data rates of approximately 4 Mbit/s. File extensions: m2v, mpeg2.

Differences between MPEG-1 and MPEG-2

Although the MPEG-2 format is a more current technology, this format doesn't present a major technical improvement over MPEG-1 as far as the basic principles are concerned. However, some differences have resulted due to the extension of the specification as well, as changes made to match the requirements of digital television and future high-resolution television. The most important details changed are:

- Increase the precision of motion vectors to half-pixels
- Extended error redundancy due to special vectors in interpolated frames
- Selectable precision of discrete cosine transformation
- Further prediction modes and macro blocks
- Scalability (different quality levels in a single video stream)
- Global Motion Compensation

MPEG-4 (ISO/IEC 14496-3) is one of the latest video formats (1999) and its objective is to get the highest video quality possible for extremely low data rates in the range between 10 Kbit/s and 1 Mbit/s. Furthermore, the need for data integrity and loss-free data transmission is paramount as these play an important role in mobile communications. Something completely new in MPEG-4 is the organisation of the image contents into independent objects in order to be able to address or process them individually. So it doesn't encodes only rectangular pixels or blocks but also individual objects of the scene. MPEG-4 is used for video transmission over the Internet for example. Some manufacturers transmitting moving images to mobile phones. MPEG-4 is intended to form a platform for this type of data transfer. The new mobile radio standard UMTS is based on MPEG-4 technology. In Tab.2.1 is comparison of different MPEG formats.

MPEG-7 (Multimedia Content Description Interface) is a standard to describe multimedia data and can be used independently of other MPEG standards. MPEG-7 aims at offering a comprehensive set of audio-visual descriptions, which will form the basis for applications enabling the needed quality access to content, which implies good storage solutions, high-performance content identification,

Table 2.1: Comparison of compression formats MPEG-1, MPEG-2, and MPEG-4.

	MPEG-1	**MPEG-2**	**MPEG-4**
Standard available since	1992	1995	1999
Max. video resolution	352x288	1920x1152	720x576
Default video resolution (PAL)	352x288	720x576	720x576
Default video resolution (NTSC)	352x288	640x480	640x480
Max. audio frequency range	48kHz	96kHz	96kHz
Max. number of audio channels	2	8	8
Max. data rate	3Mbit/s	80Mbit/s	5 to 10Mbit/s
Regular data rate used	1380 Kbit/s (352x288)	6500Kbit/s (720x576)	880Kbit/s (720x576)
Frames per second (PAL/NTSC)	25/30	25/30	25/30
Video quality	satisfactory	very good	excellent
HW requirements for encoding	low	high	very high
HW requirements for decoding	very low	medium	high

proprietary assignation, and fast, ergonomic, accurate and personalised filtering, search and retrieval.

MPEG-21 development of this format started recently. It should be a standard that aims at creating a *Multimedia Framework* taking into consideration the different components involved in the delivery of content from the creator to the user.

DVD video

One example of exploiting of the MPEG format is DVD video (Digital Versatile Disc). In last years becomes DVD video very popular for a digital movie. DVD has the capability to produce near studio quality video and better than CD quality audio. DVD video is usually encoded from digital studio master tapes to MPEG-2 format. The resulting video, especially when it is complex or changing quickly, may sometimes contain visual flaws, depending on the processing quality and amount of compression. At average video data rates of 3.5 to 5 Mbit/s, compression artifacts may be occasionally noticeable. Higher data rates can result in higher quality, with almost no perceptible difference from the master at rates above 6 Mbit/s. As MPEG compression technology improves, better quality is being achieved at lower rates.

Major DVD video technology features:

- Over 2 hours of high-quality digital video (a double-sided, dual-layer disc can hold 8 hours of high-quality video, or 30 hours of VHS quality video).
- Support for wide-screen movies on standard or wide-screen TVs (4:3 and 16:9 aspect ratios).

- Up to 8 tracks of digital audio (for multiple languages, DVS, etc.), each with as many as 8 channels.
- Up to 32 subtitle/karaoke tracks.
- Automatic "seamless" branching of video (for multiple story lines or ratings on one disc).
- Up to 9 camera angles (different viewpoints on a scene can be selected during playback).
- It has built-in copy protection and regional lockout. It covers encryption, watermarking, protection of analog and digital outputs, and so on. There are many forms of content protection that apply to DVD.

Advantages: Wide popularity thanks to many application of this format in digital communication industry (video phones, DVD, digital TV, etc.).
Disadvantages: Lossy compression.

2.2.4 The MJPEG Format

The abbreviation MJPEG stands for "Motion JPEG". This format is practically an intermediate step between a still image and video format, as an MJPEG clip is a sequence of JPEG images. This is one reason why the format is often implemented by video editing cards and systems. MJPEG is a compression method that is applied to every image. Implementation of this format reduce the resulting data stream of a standard television signal from approximately 30 MB/s to 6 MB/s (MJPEG file). This corresponds to a compression ratio of 5:1. However, a standard for the synchronisation of audio and video data during recording has not been implemented in the MJPEG format so that the manufacturers of video editing cards have had to create their own implementations.
Advantages: Higher quality in comparison with MPEG, because all frames are original (not interpolated or bidirectional).
Disadvantages: Lossy compression. Much lower compression rate than MPEG. It is not very used worldwide.

2.2.5 The H.261 and H.263 Protocol

The H.261 standard is designed for video-conferences and video telephony via an ISDN network. H.261 enables the image quality to be adapted to the bandwidth of the transmission line. In addition, entire images from a sequence can be omitted during playback in order to improve image quality. Transmission can occur at a bit rate of 64 Kbit/s or 128 Kbit/s (grouping of two ISDN channels). The successor standard H.263 implements a higher precision for motion compensation in comparison to H.261. Also, other image formats (MPEG-4) are supported in order to accommodate for different application fields such as gate monitoring systems and wide screen video-conferences.

Advantages: The format was developed for special application (video-conferences, etc.) and in this field is widely extended.
Disadvantages: Data flow in this format is constrained by ISDN bit rate (max. 128 Kbit/s).

2.2.6 Another movie formats

There are also another movie formats which are not so widely used. It can be caused that they are for special purpose or they are easily out of date.

anm (DeluxePaint Animation file) by Deluxe Paint. File extensions: anm

fli (Flick movie file) by Autodesk Animator Pro. File extensions: fli, flc

flp (Wavefront Flipbook file) by Wavefront Technologies. File extensions: flp

gif (CompuServe Graphics Interchange Format file). It supports single and multiple images, GIF89a animation extensions. File extension: gif

grasp (GRAphical System for Presentation movie file) by IBM. File extension: gl

iris (SGI IRIS Movie Format file). File extension: mov

pfx (Amiga PageFlipper Plus F/X file). File extension: pfx

Chapter 3

Motion Estimation

Good scratch restoration algorithm would take advantage of both spatial and temporal information. Due to motion of objects in scene (i.e. corresponding regions in an image sequence) the same region don't occurs in the same place in previous frame as in current one. Hence for proper function of the proposed scratch removal algorithm, is favourable to estimate future motion in observed image sequence as accurate as possible. The motion typically encountered in an image sequence have three possible forms: translation, rotation and zoom of objects in the scene. The motion in image sequence can be modelled by equation

$$Y(r_1, r_2, r_4) = Y((r_1, r_2, r_4 - 1) - \mathbf{d}_{r_1, r_2, r_4}) \tag{3.1}$$

where $Y(r_1, r_2, r_4)$ is intensity of pixel with a spatial coordinates r_1, r_2 in the r_4-th frame. Parameter $\mathbf{d}_{r_1, r_2, r_4}$ represents displacement (estimated motion vector) of this pixel between frames $r_4, r_4 - 1$ from the image sequence.

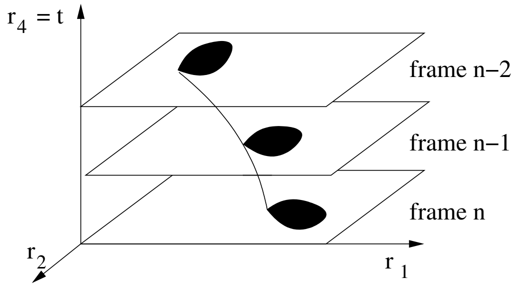

Figure 3.1: Motion trajectory of region moving in image sequence.

Regions in image are moving along their motion trajectories (see Fig.3.1) through the image sequence. So it is necessary to find similar areas in neighbouring frames and to treat them together. Solution is to find motion of the regions in each frame and then restore missing information in some frame by filtering along motion trajectories. The motion estimation (ME) is essential tool for determining of motion trajectories. It gives us motion vector of each pixel or block of pixels. Unfortunately no ideal motion estimation has been found yet. The more accurate is the motion estimation, the better image sequence restoration we can reach.

We can discriminate motion in a sequence into a two kinds. The first one is *local motion* - it means the movement of a object on the scene owing to theirs background. The second one is *global motion* - it represents movement of whole view on the scene. As a matter of fact, the majority of developed motion estimation methods make no distinction between global and local motion. There were already a lot of the motion estimators published and we can divide them, according to their principle, into a several basic groups. The estimation methods based on block correspondence matching *Block Matching methods* (BM) or computing of a gradient in image *Gradient Based methods* (GB) are mostly used. There are also other ME methods based on Markov random fields (MRF) or on spatio-temporal wavelet transformation.

For fast motion estimation the multi-resolution pyramid (Fig.3.2) is appropriate to apply. It is based on under-sampling of processed frames from image sequence to obtain smaller duplicates. The number of under-sampling steps corresponds with number of layers in pyramid. At the beginning is necessary to initialise motion vector on the coarsest level of the pyramid (top). After this we apply ME method on less and less under-sampled frames till we reach original frame resolution. This stops iteration process and final motion vector is obtained.

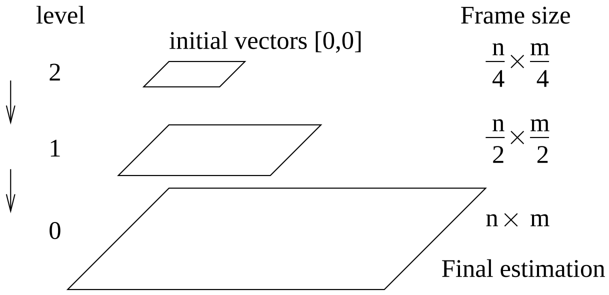

Figure 3.2: Multi-resolution pyramid principle.

The ME methods described in this chapter works only with monospectral movies. Thus to obtain motion vector for multispectral ones it is necessary to apply these methods for each spectral band separately.

3.1 Block Matching methods

Block matching (BM) is very popular and robust type of methods for motion estimation, which are widely adopted in various coding standards, such as H.261, H.263, MPEG-1, MPEG-2, MPEG-4. It is used for fast motion estimation of the blocks, into which is usually evaluated image divided. Principle of these

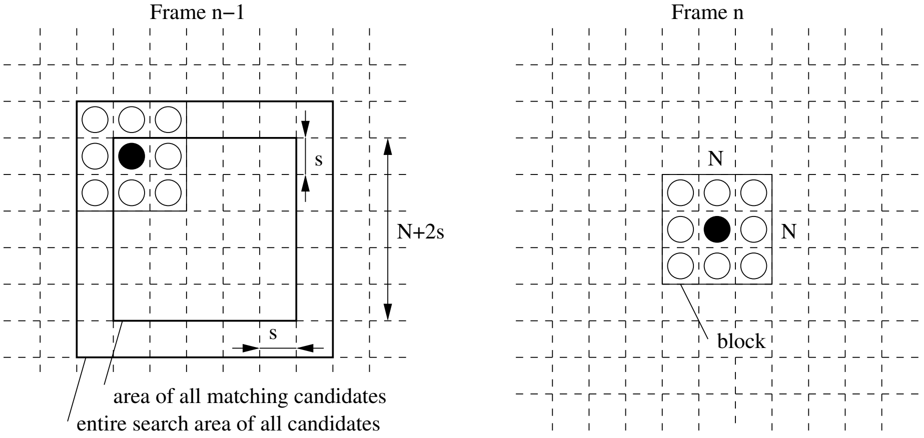
Frame n−1
Frame n
s
N+2s
s
N
N
block
area of all matching candidates
entire search area of all candidates

Figure 3.3: Principle of Block Matching methods.

methods [17] is depicted in Fig.3.3. For block in actual frame (frame n) are computed Mean Squared Error (MSE) or Mean Absolute Difference (MAD) of pixels, between this block and all candidates block in previous frame (frame $n-1$). The amount of the candidates is arbitrary and specifies the search area. As a final motion vector is chosen the candidate which gives the lowest MSE or MAD value. The more candidates is chosen, the more computational demanding the motion estimation process will be. If we use block of size $N \times N$ and size of candidate space $(N + 2s) \times (N + 2s)$ (Fig.3.3), we have to perform $N^2(2s + 1)^2$ operation for Motion Estimation of one block. Thus many approaches were performed to reduce this computational requirements. Some work and improvements in this field was made in [6, 5]. In [31] the neighbouring blocks are overlapped and interact with one another to achieve higher performance of the method.

Interesting approach using iteration principle was created by Zhu S. and Ma K. [37] called *Diamond search* (DS). The shape of the search pattern is depicted in Fig.3.4-a. Evaluated pixel is placed in the center of this pattern. They are reducing computation in this pattern only into direction in which is the computed MAD value the lowest. The MAD is computed in mentioned Search Pattern between evaluated pixel in actual frame and pixel lying in distance defined by attained motion vector in previous frame. When the new direction is chosen (by the lowest MAD), the center of the Search Pattern is moved to this direction

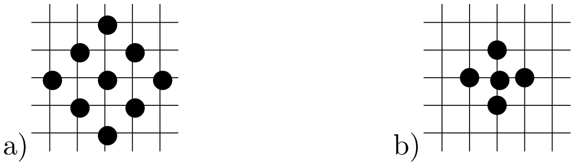

Figure 3.4: The "DS" algorithm: (a)large neighbourhood (b)small neighbourhood.

and computation continues again. This is made iteratively till no direction is chosen (no movement of the shape from the central point). After this the smaller neighbourhood (Fig.3.4-b) is used to compute a final value of motion vector. The example of iterative obtaining a motion vector is shown in Fig.3.5.

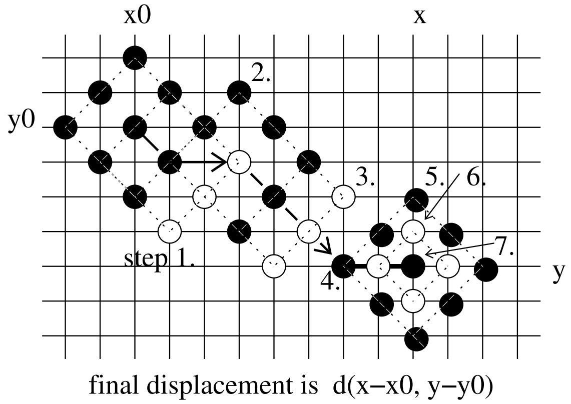

Figure 3.5: An example on principle of "DS" algorithm.

We found this algorithm fast and easy to implement, so we used it for motion estimation in image sequence for proposed restoration method in Chapter 5. Results of the tests of implemented algorithm are in Figs.3.8-e,f , 3.9-e,f.

3.2 Gradient Based methods

These methods were developed independently by workers in Video Coding and Computer Vision in seventies. They tried to find a motion estimator which requires less computation than BM [17].

The Gradient Based methods (GB) iteratively computing update vector of motion vector $\mathbf{d}$

$$\mathbf{u}_{r_4} = \mathbf{d} - \mathbf{d}_{r_4} \tag{3.2}$$

where $\mathbf{d}_{r_4}$ is current estimate of $\mathbf{d}$. The goal is to minimise the size of update vector and Mean Square Error expressed by

$$E[(\mathbf{u}_{r_4} - \hat{\mathbf{u}}_{r_4})^T(\mathbf{u}_{r_4} - \hat{\mathbf{u}}_{r_4})] \ . \tag{3.3}$$

To have direct access to the motion parameter d we have to rearrange the equation (3.1). This equation can be linearised using a Taylor series expansion

$$Y_{r_4}(r_1, r_2) = Y_{r_4-1}(r_1, r_2) + \mathbf{d}^T \nabla Y_{r_4-1}(r_1, r_2) + e_{r_4-1}(r_1, r_2) \tag{3.4}$$

where $e_{r_4-1}(r_1, r_2)$ represents the higher order terms of the Taylor expansion and ∇ is multidimensional gradient operator. After rearranging we obtained expression involving the Displaced Frame Difference with zero displacement d and substitution of displacement d by its update u_{r_4}.

$$DFD((r_1, r_2), 0) = Y_{r_4}(r_1, r_2) - Y_{r_4-1}(r_1, r_2) = \mathbf{u}_{r_4}^T \nabla Y_{r_4-1}(r_1, r_2) + e_{r_4-1}(r_1, r_2) \tag{3.5}$$

This we can express by equation

$$\mathbf{z}_{r_4} = \mathbf{G}\mathbf{u}_{r_4} + e \ . \tag{3.6}$$

Let have size of movie frame $N \times N$. $\mathbf{z}_{r_4}$ is vector containing $DFD((r_1, r_2), \mathbf{d}_{r_4})$ for each pixel ($(r_1, r_2) = 1...N^2$) and $\mathbf{G}$ is matrix of gradients in both directions (r_1, r_2) for each pixel. The guess of update vector is then

$$\hat{\mathbf{u}}_{r_4} = [\mathbf{G}^T\mathbf{G} + \mu\mathbf{I}]^{-1}\mathbf{G}^T\mathbf{z} \tag{3.7}$$

where μ is something like damper in system preventing the inverse $[\mathbf{G}^T\mathbf{G} + \mu\mathbf{I}]^{-1}$ from being unstable when the matrix $\mathbf{G}^T\mathbf{G}$ is ill conditioned.

After obtaining update vector u_{r_4}, new motion vector is computed

$$\mathbf{d}_{r_4+1} = \mathbf{d}_{r_4} + \mathbf{u}_{r_4} \ .$$

Update $\mathbf{u}_{r_4}$ is then computed iteratively till the convergence is reached ($\mathbf{u}_{r_4}$ is smaller then some threshold). Because the displacement vector $\mathbf{d}$ after estimation can contain fractional number some interpolation into pixel grid is necessary.

There is a lot of modification of this method, which are described in [17]. The fast GB method based on coarse initial estimation and the hierarchical implementation is proposed in [8].

3.3 Block Matching vs. Gradient Based motion estimators

The Block Matching methods are more computational demanding than the Gradient Based ones, but they are still very popular because of simple implementation and ability to handle with any displacement size (it depends only on the chosen search area). Gradient Based techniques can't handle large motion, because for the large displacements, there is no relation between two regions being compared in two frames. Hence the Taylor serial expansion of the region in the previous frame won't have connection with the current region.

Another drawback of Gradient Based methods is possibility of ill-conditioning of the computation when regions in the frame have insignificant texture.

The advantage of the Gradient Based methods is variable resolution of the motion vector. With using of multi-resolution pyramid we can obtain variable resolution also with Block Matching techniques. But Block Matching techniques gives us only integer accurate motion estimation. For fractional accuracy we need extract blocks between location on the image grid and perform some interpolation.

3.4 Other Motion Estimation Methods

There are also other ME methods. As an example can be mentioned methods based on Markov Random Fields (MRF) [36]. These methods have the best results, but their applying for online processing is more difficult due to their huge computational complexity.

Other new methods estimate motion in wavelet domain [24, 26]. They are based on the spatio-temporal continuous wavelet transformation. The wavelet transformation decomposes a non-stationary signal into a set of multiresolutional wavelet coefficients where each component becomes relative more stationary and easier to describe.

The approach for ME in colour image sequences based on Maximum-Likelihood formula using chrominance information is introduced in [21, 22].

The motion field interpolation method for minimising the effect of transmission error during broadcasting is in [23].

3.5 Test Data Description

Testing of the proposed ME and scratch restoration algorithms requires appropriate image sequences. First of all we searched for some standard test sequence with some defined corruption. Unfortunately we didn't succeeded in it, because each researcher use different one. Finally we used two following sequences for testing.

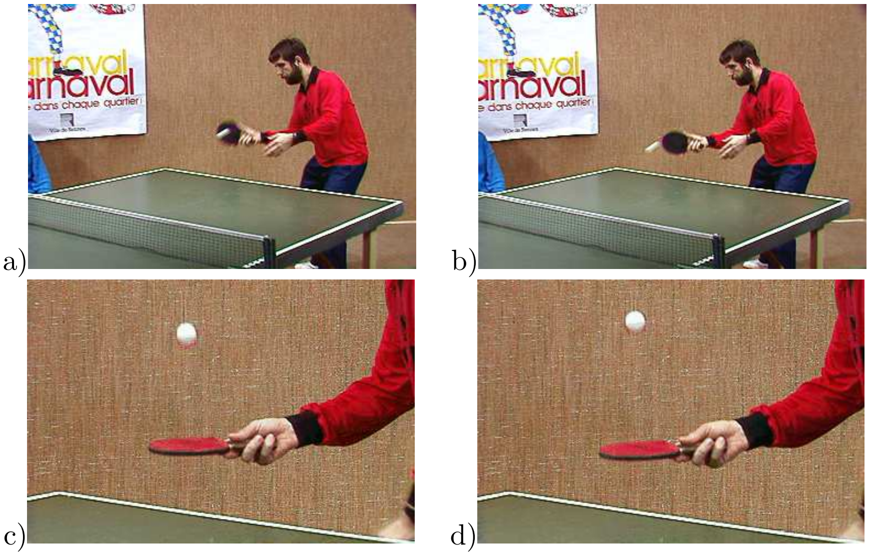

Figure 3.6: Selected frames from the "tennis" sequence: (a)frame No.77 (b)frame No.78 (c)frame No.0 (d)frame No.1.

- **"tennis"** is standard test sequence obtainable from public domain WWW page [25]. It introduces players during their table tennis play and includes various motion characteristics such as camera panning, zooming, and motion of human body. The examples of frames from "tennis" sequence used for testing are presented in Fig.3.6. This sequence includes several seams with different view on the scene. We selected from this sequence two parts. First with player hitting a ball (Fig.3.6-a,b) and second which contains detail of racket in player's hand with ball moving in the air (Fig.3.6-c,d).

- **"cars"** is the second sequence we used. We needed to test proposed algorithm also on sequence including fast motion. Fast means, in this thesis, more than 10 pixel between successive frames. We didn't found an appropriate colour sequence so we created it ourself. The requirement on motion was that it should be easy to estimate manually, to verify results of ME algorithm. For this purpose the pure translation of the object was satisfactory. Hence we took several pictures of moving car-toys by digital camera. The two obtained frames are shown in Fig.3.7.

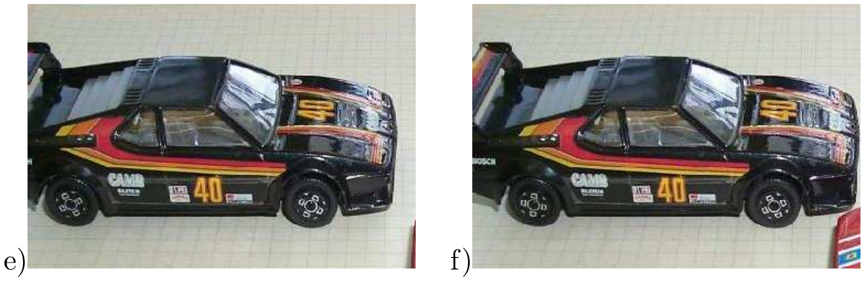

Figure 3.7: Selected frames from the "cars" sequence: (a)frame No.76 (b)frame No.77.

3.6 Tested Motion Estimation Methods Results

For aim of comparison the two ME methods are implemented. The classical simple BM ME method Fig.3.3 [17] (further referred as BM1) and its iterative improvement "Diamond search" Figs.3.4,3.5 [37] (further referred as BM2). Both methods are tested on an image sequences "tennis" and "cars". The original frames No.0 and No.1 from "tennis" sequence are shown in Fig.3.8-a,b and motion vectors in horizontal and vertical direction estimated by method BM1 are in Fig.3.8-c,d and by method BM2 in Fig.3.8-e,f. In this test the results of both method are satisfactory. Method BM2 is even able to detect slow (less than 5 pixels/frame) global motion (zooming) in the image sequence. This is observable in Fig.3.8-e,f in region where the tennis table appears.

The "cars" sequence involve fast motion (about 18 pixels/frame), hence it is more difficult to estimate it by BM1. The BM1's performance here depends mainly on the size of area of all matching candidates blocks. So if we increase this size, the method complexity increase quadratically. On the other hand the

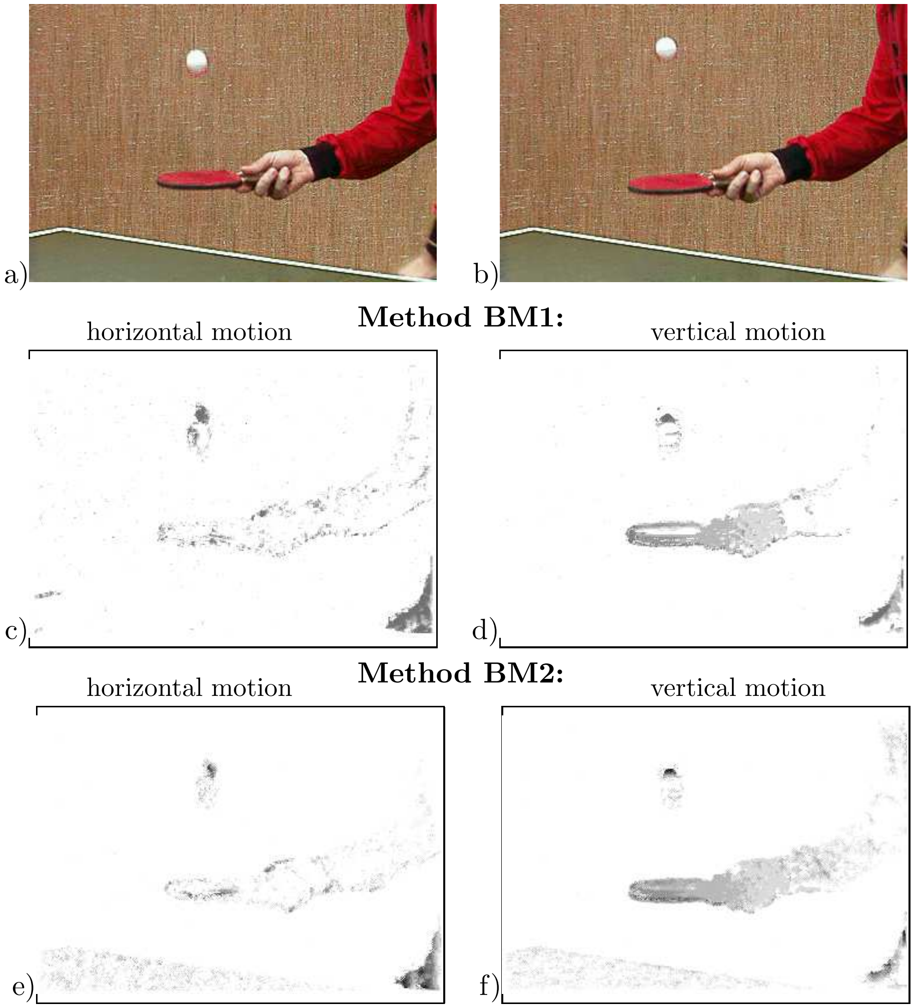

Figure 3.8: Motion estimation between frames (a)No.0 (b)No.1 from "tennis" sequence by method BM1 (c)(d) and by method BM2 (e)(f) in horizontal (c)(e) and vertical (d)(f) direction respectively.

iterative method BM2 can be often trapped into local maximum solution during computation of an optimal motion vector.

The tested frames No.76 and No.77 from "cars" sequence are in Fig.3.9-a,b. The corresponding motion vectors in horizontal and vertical direction estimated by method BM1 are in Fig.3.9-c,d and by method BM2 are in Fig.3.9-e,f. The both methods suffer here by false estimation in regions where no motion appears, while method BM1 gives visually better results. We have to remark, that illustrated ME results represents motion vectors averaged in all spectral bands. Obtained results denounce that tested methods aren't ideal, but they give us at least an estimate of motion tendency of observed region in the image sequence.

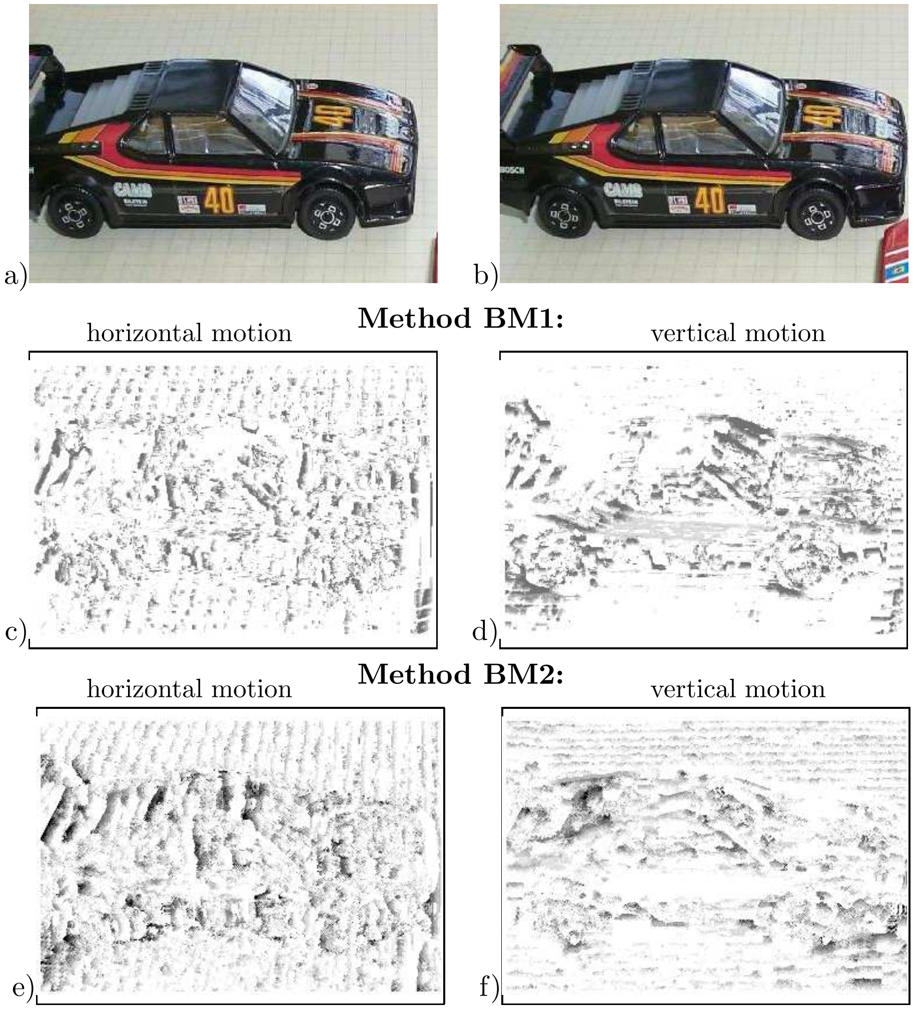

Figure 3.9: Motion estimation between frames (a)No.76 (b)No.77 from "cars" sequence by method BM1 (c)(d) and by method BM2 (e)(f) in horizontal (c)(e) and vertical (d)(f) direction respectively.

3.7 Employing of Motion Compensation for Scratch Restoration

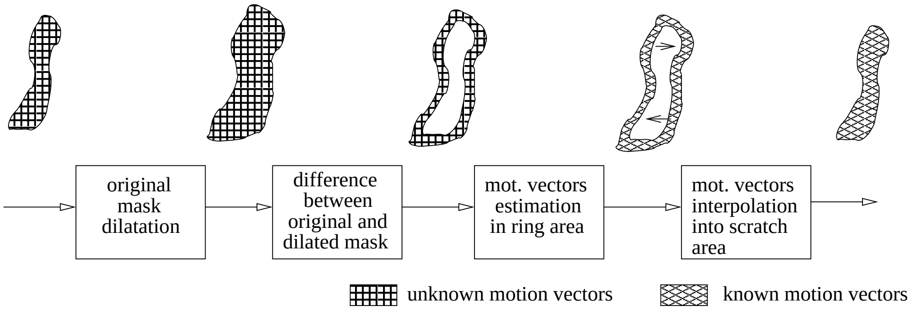
original mask dilatation
difference between original and dilated mask
mot. vectors estimation in ring area
mot. vectors interpolation into scratch area
unknown motion vectors
known motion vectors

Figure 3.10: Principle of motion vectors interpolation in the scratch area.

Above mentioned ME methods assume known data from two processed frames in image sequence. However this condition in not fulfilled in the frame where the scratch occurs. During the scratch restoration the motion vectors mostly from the pixels placed in the scratch area are needed. Thus some method propagating the known motion vectors from the scratch surrounding into the scratch area is essential. Efficient technique for motion interpolation based on Gibbs Energy prior in Markov Random Fields was introduced in work of Kokaram [18, 19]. In this thesis is this problem solved by simple linear interpolation of the motion vectors from the scratch surrounding (Fig.3.10). This surrounding is obtained by morphological dilatation of the original scratch mask. The obtained results are used as displacement motion vectors for restoration method proposed in Chapter 5.

Chapter 4

The Scratch Restoration Methods Overview

4.1 Scratch Detection Methods

The basic assumption for the successful scratch restoration, is knowledge about location of corrupted area with missing pixels in the processed frames from the image sequence. There are several scratch detection methods with different complexity [18].

In this thesis we mention the simplest method for the corrupted area detection is SDI detector (Spike Detection Index) [18]. The method assumes that dirt and sparkle represents regions of temporal discontinuity in intensity along motion trajectories (Fig.3.1). It have several adaptations. The simplest detector SDIa is proposed by equations (4.1), (4.2). When the observed pixel is corrupted the E_- and E_+ takes a high magnitude.

$$\begin{aligned} E_+ &= Y(r_1, r_2, r_4) - Y((r_1, r_2, r_4 - 1) + \mathbf{d}_{r_1,r_2,r_4}) \\ E_- &= Y(r_1, r_2, r_4) - Y((r_1, r_2, r_4 - 1) + \mathbf{d}_{r_1,r_2,r_4}) \end{aligned} \tag{4.1}$$

Unfortunately this occurs also when motion discontinuities (occlusion and uncovering) occur. This drawback can be overcome if the detector will be able to recognise if temporal discontinuity occurs only in one direction along the motion trajectory or not. The SDI detector flags missing data if both E_- and E_+ have higher values then user defined threshold value and is 0 otherwise.

$$b_{SDIa}(r_1, r_2) = \begin{cases} 1 & \text{for } (|E_+| > \alpha) \wedge (|E_-| > \alpha) \\ 0 & \text{otherwise} \end{cases} \tag{4.2}$$

where α is user defined threshold.

Improvement of the method, called SDIb, is in equation (4.3). It constraints the ability of missing data detection but the results are then more reliable.

$$b_{SDIb}(r_1, r_2) = \begin{cases} 1 & \text{for } (|E_+| > \alpha) \wedge (|E_-| > \alpha) \\ & \wedge (sign(E_+) = sign(E_-)) \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

From the text above it is clear that efficiency of the scratch detectors rely on good ME method estimating displacement vector $\mathbf{d}_{r_1,r_2,r_4}$. Other missing data detectors are described in [18].

4.2 Classical Restoration Methods

For the scratch restoration the below mentioned basic restoration methods are commonly used. These methods are mostly very simple, but can help us as a basic quality measure for comparison with the next proposed restoration method. In this thesis we called these methods as “classical”. In our case the mentioned classical methods works only with the static images, so they don’t’ use temporal information in the image sequence. That’s why for image sequences it is necessary to apply these methods on every frame separately. Also all spectral layers in frame are processed separately.

4.2.1 Averaging

Averaging is the simplest method which replaces missing pixel value by mean value computed from known pixels of local support set. The shape and size of the support set is arbitrary and we tried simply square one. The best results on our test data corrupted by a scratch (Fig.6.1-b) are obtained with it’s size 5×5. The unknown pixels on a scratch are computed consecutively rightward row-wise and moved with this support set through the scratch in order to obtain most of known data in support set. This approach cause linear dependency, which is sometimes observable in restored image like straight bright or dark artifact, mostly if corrupted area is too wide.

4.2.2 Filtering by median

Median is a robust statistic. The method finds in random population outliers, eliminates them and substitute the remaining values by their typical value. It takes reconstructed pixel surroundings in appropriate support area (we used simply square). Then all pixels are sorted according to their intensity from the biggest one to the smallest one. Optimal value is then in the middle of the sorted array. We tested this method, on our test data corrupted by a scratch (Fig.6.1-b), for several sizes of square support set and the best results gives size 7×7.

4.2.3 Linear and Quadratic regression

In these methods the pixels on scratch are interpolated by using of parameters computed from known pixels around the scratch on every line where the scratch occurs. Amount of known pixels which are taken into an account from every side of scratch is arbitrary and represents surrounding which is used for parameters computing. Then this parameters are used to obtain appropriate interpolation by outline and parabola. The equations for a pattern and a coefficients computation are (4.4),(4.5).

$$y_{LR} = b_0 + b_1 x \qquad y_{QR} = b_0 + b_1 x + b_2 x^2 \tag{4.4}$$

The solution is based on least squares estimate, which is explained in [27]. It is also possible to interpolate missing data by higher order curves. Universal equation for model parameters computing is

$$\begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{pmatrix} = \begin{pmatrix} n & \sum_{i=1}^n x_i & \dots & \sum_{i=1}^n x_i^k \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 & \dots & \sum_{i=1}^n x_i^{k+1} \\ \vdots & \vdots & & \vdots \\ \sum_{i=1}^n x_i^k & \sum_{i=1}^n x_i^{k+1} & \dots & \sum_{i=1}^n x_i^{2k} \end{pmatrix}^{-1} \begin{pmatrix} \sum_{i=1}^n Y_i \\ \sum_{i=1}^n x_i Y_i \\ \vdots \\ \sum_{i=1}^n x_i^k Y_i \end{pmatrix} \tag{4.5}$$

where x_i is position of known value Y_i.

4.3 Advanced Restoration Methods

In this section we discuss recent solutions of scratch restoration in image sequences. Unfortunately we can't compare our results also with these methods, because their authors rarely mention any test criteria and compared their results usually only in terms of the subjective visual quality. Surprisingly the number of published algorithms in the area of movie restoration is rather small probably because of computational complexity of most restoration approaches. A considerable amount of these methods is concerned in a noise and blur filtering. Recent overview of the image sequence restoration methods was introduced in work of Kokaram [17]. The several methods for the suppressing local distortion in image sequence were introduced. Other model-based methods use most often Markov random fields type of models either in the form of wide-sense Markov (AR models) or strong Markov models.

A short overview of recent developed scratch restoration methods follows.

4.3.1 Spatio-Temporal Median Filtering

Proposed filters depicted in Fig.4.1 use data from three frames to achieve filtering. The center window in each window-set refers to a window in the current frame. An example of the filter is described by equation (4.1).

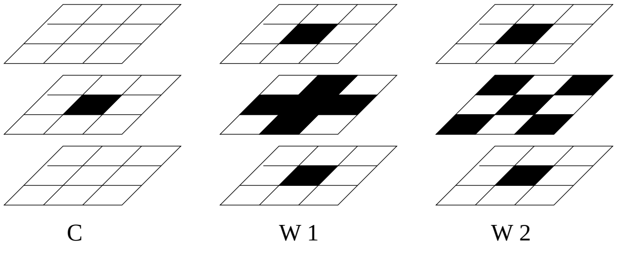

Figure 4.1: Example of sub-filter masks for median filter introduced in [1].

$$
\begin{aligned}
z_l = median[W_l], \qquad 1 \leq l \leq 2, \\
Filter\ output = median[z_1, z_2, C]
\end{aligned}
\tag{4.6}
$$

There are many possible improvements by filters shapes and their amount [1, 2]. The method is assumed for dirt and sparkle removing and is not suitable for large scratch removal thanks to linear dependence on previously restored pixels.

4.3.2 Non-causal 3D AR Modelling

The proposed non-causal 3D AR model handle with mono-spectral image sequence. Model compute prediction of a pixel as a weighted linear combination of pixels in predefined contextual neighbourhood. The model is defined as follows:

$$
\begin{aligned}
Y(r_1, r_2, r_4) \quad = \quad & \sum_{s_1, s_2, s_4 \in I_{r_1, r_2, r_3}} a_{s_1, s_2, s_4} Y(r_1 + s_1 + d_{1;r_4, r_4+s_4}, r_2 + s_2 + d_{2,r_4, r_4+s_4}, \\
& r_4 + s_4) + \epsilon(r_1, r_1, r_4)
\end{aligned}
\tag{4.7}
$$

Where: $Y(r_1, r_2, r_4)$ is intensity of the pixel at the position (r_1, r_2, r_4) where (r_1, r_2) is horizontal and vertical position in frame r_4.

a_{s_1, s_2, s_4} are model coefficients.

$[s_1, s_2, s_4]$ is the vector offset in the k-th predefined non-causal contextual neighbourhood I_{r_1, r_2, r_4} .

$\mathbf{d}_{r_1, r_2, r_4} = [d_{1;r_4, r_4+s_4}, d_{2;r_4, r_4+s_4}]$ is the motion vector between frames r_4 and $r_4 + s_4$.

$\epsilon(r_1, r_2, r_4)$ is the error representing difference between model prediction and actual pixel intensity.

Motion displacement is estimated by modification of the GB ME method introduced in section 3.2 on page 17.

The coefficients of AR model are estimated by minimising of the square error ϵ and it leads to a Normal equations. For coefficients computation the correlation function of the image sequence is necessary.

An improvement of 3D AR model employing a hierarchical GB ME, robust in regions of corrupted data with Markov random field motion prior, was published in [19].

4.3.3 Controlled Pasting Scheme for Missing Data Interpolation

This solution employing 2D AR model and Markov random field techniques, based on controlled pasting scheme is proposed in [34]. The pixel intensity from the reference image frame that gives the smallest AR prediction error is pasted into the current frame.

4.3.4 Bayesian Approach with Vector Median Colour Image Model

Another approach for scratch restoration in colour image sequences was introduced by Armstrong [3]. It is based on n-dimensional vector median image model. Where n represents amount of spectral bands in image. This is combined with a strategy that first detect scratches by simple SDI detector and then the scratch area is restored using motion-compensated model-based Markov Chain Monte Carlo sampling techniques.

The vector median of a set of n-dimensional vectors $X = x_q : q = 1, \cdots, N$ is defined in

$$x_{med} = arg \min_{x_j \in X} \sum_q |x_q - x_j| \ . \tag{4.8}$$

The Gibbs sampler is used to iterative samples drawing for the missing data and the noise process for each channel is modelled with a zero-mean Laplacian distribution. Motion compensation is performed by multidimensional implementation of GB ME method.

4.3.5 The JOMBADI Algorithm

The last and the most complex method for missing data restoration based on non-causal 3D AR modelling is described in [18]. Algorithm name stands for “JOint Model BAsed Detection and Interpolation”. The algorithm includes GB ME method and Bayesian framework for joint detection (by SDI detector) and restoration of the missing data. Solution is based on Gibbs sampler and it’s adaptations.

Algorithm was applied during developing of automatic digital restoration system AURORA [28].

4.3.6 Causal Autoregressive Models

The majority of mentioned scratch restoration algorithms have the main problem in time consuming iterative solution and have to solve also iteration stopping problem. This drawbacks are eliminated by the method described in [14, 15]. The method for scratch restoration in monospectral image sequences is based on causal 2.5D autoregressive model which has analytic solution instead of iterative one. The method was further improved in [16] to select a locally optimal predictor from two mutually competing symmetrical adaptive predictors for each pixel to be reconstructed. We extended this model into 3D and 3.5D one in [12].

Chapter 5

The Scratch Restoration by the Causal Autoregressive Model

Modelling of colour image sequence requires in general four dimensional data model where data are spectrally, spatially and temporally correlated.

Unfortunately the 4D model has huge amount of parameters to estimate and has to represented by a tensor. Therefor the 4D model has huge computational demands, requires a huge training data set, leads to non-linear parameter estimation, hence it has to solve problem when to stop iterative process. Due to constrained numerical precision it is possible that the model is numerically unstable.

Due to these reasons the 4D model does not have analytic solution and some simplifying steps are necessary. The number of dimensions can be reduced if we are able to factorise input data space and finally more less-dimensional models are used. This is possible if data in each dimension aren't mutually correlated. This condition is never fulfilled but we can perform data space decorrelation by some statistical method as for example Karhun-en-Loeve transformation, etc. Another approach of 4D model simplifying is employing of the next proposed 3.5D model.

5.1 Image 3.5D Causal Autoregressive Model

Suppose Y represents a digitised colour movie defined on a finite rectangular four dimensional $N \times M \times d \times \tau$ underlying lattice I, where $N \times M$ is the frame size, d is the number of spectral bands (i.e., $d = 3$ for usual colour movies) and τ is the overall number of frames in the film to be reconstructed. Corresponding pixel multiindex $r = \{r_1, r_2, r_3, r_4\}$ has the row, columns, spectral and time indices, respectively.

All image data are assumed to be known except a set ($\mathcal{S}$) of unobservable multispectral pixels from some frame belonging to a scratch. The missing scratch data reconstruction from the topologically nearest known data in the lattice I using temporal and spatial correlation in the neighbourhood generally requires a 4D model. Unfortunately parameters of such a AR model cannot be estimated analytically.

However if we neglect mutual temporal correlations, i.e.,

$$E\{e_{r_1,r_2,r_3,\bullet} e^T_{s_1,s_2,s_3,\bullet}\} = \; diag[\sigma^2_{r_3},\dots,\sigma^2_{r_3}] \qquad \forall s : s_1 = r_1,\; s_2 = r_2,\; s_3 = r_3$$

but nevertheless use different temporal multispectral pixels in a 3D AR model we obtain the 3.5D AR model which can be solved analytically under some additional acceptable assumptions. The notation $\bullet$ has the meaning of all possible values of the corresponding index. Suppose further that the multispectral multitemporal movie data can be represented by an adaptive causal 3.5 dimensional simultaneous autoregressive model:

$$\begin{aligned} Y_{r_1,r_2,\bullet,r_4} &= \sum_{\{s_1,s_2,s_4\} \in I_{r_1,r_2,r_4}} A_{s_1,s_2,s_4} Y_{r_1-s_1,r_2-s_2,\bullet,r_4-s_4} + e_{r_1,r_2,\bullet,r_4} \qquad \forall r \in I \\ &= \gamma\, X_{r_1,r_2,\bullet,r_4} + e_{r_1,r_2,\bullet,r_4} \;\; , \qquad (5.1) \end{aligned}$$

where $Y_{r_1,r_2,\bullet,r_4}$ is a multispectral $d \times 1$ vector corresponding to a single multispectral pixel in the r_4-th frame. I_{r_1,r_2,r_4} is a causal index shift set including index shifts $s1, s2, s4$. I_{r_1,r_2,r_4} specifies shape of the contextual neighbourhood (CN) around the actual index r_1, r_2, r_4. Causality is fulfilled when all data obtained from CN are known (not missing pixels). The example of causal and non-causal CN is depicted in Fig.5.1.

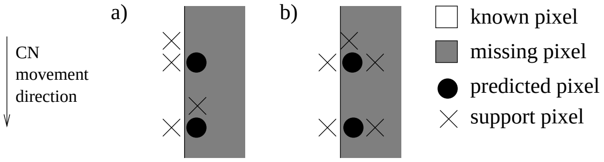

Figure 5.1: The example of two causal (a) and two non-causal (b) contextual neighbourhoods.

From this causal contextual neighbourhood the known data are arranged into a vector:

$$X_{r_1,r_2,\bullet,r_4} = [Y^T_{r_1-s_1,r_2-s_2,\bullet,r_4-s_4} : \; \forall \{s_1,s_2,s_4\} \in I_{r_1,r_2,r_4}]^T \;\; . \qquad (5.2)$$

The expression

$$\gamma = [A_1, \dots, A_\eta] \;\; , \qquad (5.3)$$

is a $d \times d\eta$ parameter matrix with

$$A_i = \begin{pmatrix} a^i_{1,1} & \dots & a^i_{1,d} \\ \vdots & \ddots & \vdots \\ a^i_{d,1} & \dots & a^i_{d,d} \end{pmatrix} \qquad \forall i \in \{1 \dots \eta\} \;\; , \qquad (5.4)$$

where

$$\eta = \; card\{I_{r_1,r_2,r_4}\} \enspace .$$

If all matrices A_i are diagonal (i.e. $A_i = diag[a^i_{1,1}, \ldots, a^i_{d,d}]$) then the 3.5D model reconstruction is identical with separately applied 2.5D model reconstruction on every monospectral scratch pixel component.

Data ordering in $X_{r_1,r_2,\bullet,r_4}$ (5.2) corresponds to the arrangement of parameters in (5.3). The noise vectors $e_{r_1,r_2,\bullet,r_4}$ are assumed to be mutually uncorrelated zero mean white Gaussian, i.e.,

$$E\{e_{r_1,r_2,\bullet,r_4} e^T_{s_1,s_2,\bullet,s_4}\} = \begin{cases} 0 & \text{if } r \neq s \\ \Sigma & \text{otherwise} \end{cases} \quad ,$$

where Σ is the noise covariance $d \times d$ matrix which is assumed to be constant but unknown to us.

The missing scratch data will be reconstructed from the topologically nearest known data in the lattice I using temporal and spatial correlation in the neighbourhood. Scratch pixels are replaced by the one-step-ahead predictive ones using the conditional mean predictor

$$E\left\{Y_{r_1,r_2,\bullet,r_4} \,|\, Y^{(r-1)}\right\} \enspace , \tag{5.5}$$

where $Y^{(r-1)}$ is the known process history

$$\begin{aligned} Y^{(r-1)} &= \{Y_{r-1}, Y_{r-2}, \ldots, Y_1, X_r, X_{r-1}, \ldots, X_1\} \\ &= \{Y_{r-1}, Y_{r-2}, \ldots, Y_1\} \enspace . \end{aligned} \tag{5.6}$$

Simplified notation $r, \; r-1, \ldots$ denotes the multispectral process position in I, i.e., $r = \{r_1, r_2, \bullet, r_4\}$, $r-1$ is the location immediately preceding $\{r_1, r_2, \bullet, r_4\}$, etc. A direction of movement on the underlying image sub-lattice corresponding to a corrupted frame is chosen in a way to erode the frame scratch, i.e., $r-1 = (r_1 - \triangle_1, r_2 - \triangle_2, \bullet, r_4)$, $r-2 = (r_1 - 2\triangle_1, r_2 - 2\triangle_2, \bullet, r_4), \ldots$.

The estimator of unknown model parameter matrix γ (5.3) is

$$\hat{\gamma}^T_{r-1} = V^{-1}_{xx(r-1)} V_{xy(r-1)} \enspace . \tag{5.7}$$

The following notation is used in (5.7):

$$\tilde{V}_{vw(r-1)} = \sum_{k=1}^{r-1} V_k W_k^T \tag{5.8}$$

and

$$V_{vw(r-1)} = \tilde{V}_{vw(r-1)} + V_{vw(0)} \enspace ,$$

where V_k, W_k are either X_k or Y_k .

The number of model movements on image plane is

$$\beta(r) = \beta(0) + r - 1 = \beta(r-1) + 1 \ , \tag{5.9}$$

where

$$\beta(0) > \eta - 2 \tag{5.10}$$

and

$$\lambda_{(r)} = V_{yy(r)} - V^T_{xy(r)} V^{-1}_{xx(r)} V_{xy(r)} \ . \tag{5.11}$$

See [14] for more details.

The model adaptation is introduced using the standard exponential forgetting factor technique in parameter learning part of the algorithm [11]. The exponential forgetting factor is stated by parameter ρ and afterwards the equation (5.8) becomes

$$\tilde{V}_{vw(r-1)} = \rho \sum_{k=1}^{r-2} V_k W_k^T + V_{r-1} W_{r-1}^T \ . \tag{5.12}$$

The estimate of process-history-data covariance matrix is [14]

$$\Sigma_{r-1} = \frac{\lambda_{(r-1)}}{\beta(r)} \ . \tag{5.13}$$

Marginal density $p(\gamma_r | Y^{(r-1)})$ can be evaluated from

$$p(\gamma_r | Y^{(r-1)}) = \int p(\gamma_r, \Sigma_r^{-1} | Y^{(r-1)}) d\Sigma_r^{-1} \ . \tag{5.14}$$

The marginal density $p(\gamma_r | Y^{(r-1)})$ can be expressed analytically in matrix t distribution density

$$\begin{aligned} p(\gamma_r | Y^{(r-1)}) \quad &= \quad \frac{\prod_{i=1}^{d} \Gamma(\frac{\beta(r)+d+2-i}{2})}{\prod_{i=1}^{d} \Gamma(\frac{\beta(r)+d\eta+d+2-i}{2})} \pi^{-\frac{d^2\eta}{2}} |\lambda_{(r-1)}|^{-\frac{d\eta}{2}} |V_{xx(r-1)}|^{\frac{d}{2}} \\ & \quad\quad |I + \lambda^{-1}_{(r-1)} (\gamma_r - \hat{\gamma}_{r-1}) V_{xx(r-1)} (\gamma_r - \hat{\gamma}_{r-1})^T|^{-\frac{\beta(r)+d+1}{2}} \end{aligned} \tag{5.15}$$

with conditional mean value, which represents previous predictor parameter matrix

$$E\left\{\gamma_r | Y^{(r-1)}\right\} = \hat{\gamma}_{r-1} \tag{5.16}$$

and covariance matrix

$$E\left\{(\gamma^{-1} - E\{\gamma_r | Y^{(r-1)}\})^T (\gamma_r - E\{\gamma_r | Y^{(r-1)}\}) | Y^{(r-1)}\right\} = \frac{V^{-1}_{xx(r-1)} \lambda_{(r-1)}}{\beta(r) - d\eta} \ . \tag{5.17}$$

If we assume the normal-Wishart (or alternatively Jeffreys) parameter prior then it was proved in [13] that the one-step-ahead predictive posterior density to have the form of d-dimensional Student's probability density

$$p(Y_r \,|\, Y^{(r-1)}) = \frac{\Gamma(\frac{\beta(r)-d\eta+d+2}{2})}{\Gamma(\frac{\beta(r)-d\eta+2}{2}) \; \pi^{\frac{d}{2}} \; (1+X_r^T V_{xx(r-1)}^{-1} X_r)^{\frac{d}{2}} \; |\lambda_{(r-1)}|^{\frac{1}{2}}}$$

$$\left(1+\frac{(Y_r-\hat{\gamma}_{r-1}X_r)^T \lambda_{(r-1)}^{-1} (Y_r-\hat{\gamma}_{r-1}X_r)}{1+X_r^T V_{xx(r-1)}^{-1} X_r}\right)^{-\frac{\beta(r)-d\eta+d+2}{2}} , \qquad (5.18)$$

with $\beta(r)-d\eta+2$ degrees of freedom, if $\beta(r) > d\eta$ then the conditional mean value is

$$\tilde{Y}_{r_1,r_2,\bullet,r_4} = E\left\{Y_{r_1,r_2,\bullet,r_4} \,|\, Y^{(r-1)}\right\} = \hat{\gamma}_{r-1} X_{r_1,r_2,\bullet,r_4} \; . \qquad (5.19)$$

This equation is used for computation of new predicted value on indices $r_1, r_2, \bullet, r_4$ dependently on known process history $Y^{(r-1)}$.

The corresponding covariance matrix is

$$E\left\{(Y_r - E\{Y_r|Y^{(r-1)}\})^T (Y_r - E\{Y_r|Y^{(r-1)}\})|Y^{(r-1)}\right\} = \frac{1+X_r V_{x(r-1)}^{-1} X_r^T}{\beta(r)-d\eta} \lambda_{(r-1)} \; . \qquad (5.20)$$

5.1.1 Optimal Model Selection

Let us assume a set of AR models (5.1) $M_1, M_2, \ldots$ which can differ either in the contextual neighbourhood I_{r_1,r_2,r_4} or / and in their exponential forgetting factor ρ. The optimal decision rule for minimising the average probability of decision error chooses the maximum a posterior probability model, i.e., a model whose conditional probability given the past data is the highest one. The presented algorithm can be therefore completed [16] as:

$$\tilde{Y}_r^i = \tilde{\gamma}_{r-1}^{i,T} X_{i,r} \qquad \text{if} \quad p(M_i|Y^{(r-1)}) > p(M_j|Y^{(r-1)}) \qquad \forall j \neq i \qquad (5.21)$$

where $X_{i,r}$ are data vectors corresponding to $I^i_{r_1,r_2,r_4}$. Following the Bayesian framework used in our paper, choosing uniform a priori model in the absence of contrary information, $p(M_i|Y^{(t-1)}) \sim p(Y^{(t-1)}|M_i)$, and assuming conditional pixel independence, the analytical solution has the form [16]

$$p(M_j|Y^{(r-1)}) = k \; \exp\{D_j\} \; ,$$

where

$$\begin{aligned} D_j \;\; &= \;\; -\frac{d}{2}\ln|V_{x(r-1)}| \; - \frac{\beta(r)-d\eta+d+1}{2}\ln|\lambda_{(r-1)}| \; + \frac{d^2\eta}{2}\ln\pi \qquad (5.22) \\ &\sum_{i=1}^{d}\left[\ln\Gamma(\frac{\beta(r)-d\eta+d+2-i}{2}) - \ln\Gamma(\frac{\beta(0)-d\eta+d+2-i}{2})\right] \end{aligned}$$

and k is a common constant. All statistics related to a model M_j $\tilde{V}_{xy(r-1)}$, $\tilde{V}_{xx(r-1)}$, are computed from data in $X_{j,r}$. The determinant $|V_{xx(r)}|$ as well as λ_r can be evaluated recursively see [14].

5.2 The Scratch Restoration Algorithm

The task consists in question how to apply the proposed 3.5D CAR model for corrupted area restoration. The basic idea of the approach is a scratch pixel replacement with it's corresponding 3.5D CAR model prediction. The model, specified by contextual neighbourhood $I^c_{r1,r2,\bullet,r4}$ is moving towards the scratch and adaptively updating its parameters during this process. After reaching a corrupted pixel its prediction is computed from accumulated data as well as actual data in the neighbourhood of the corrupted pixel. The length of model history σ in pixels corresponds to the number of updates before applying the prediction. The parameter σ is important for restoration and it's value affects obtained results as can be seen in the Fig.6.11.

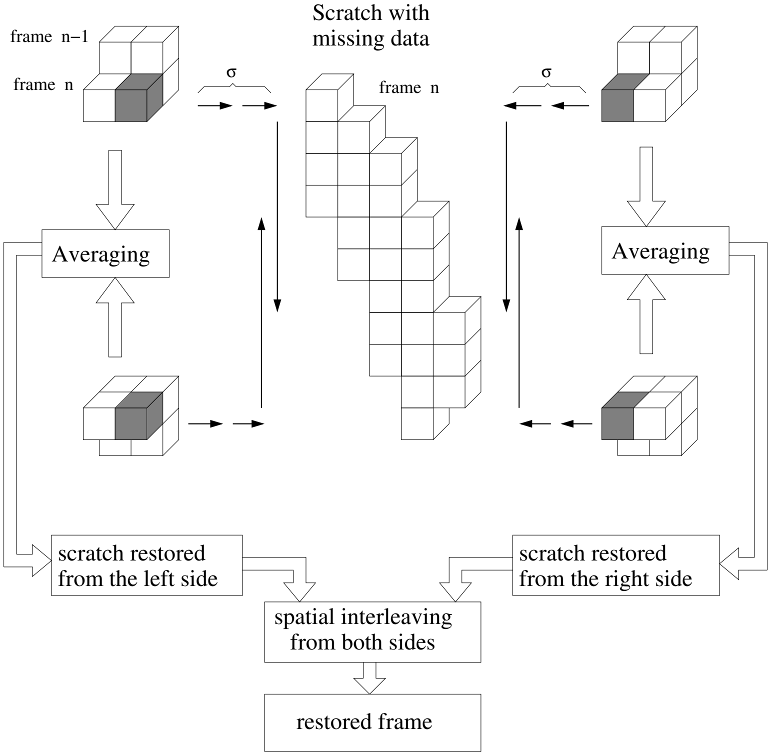

Figure 5.2: The restoration method scheme for vertical scratch.

The restoration algorithm is depicted in Fig.5.2. Let's assume the rightwards movement of the model. When the model reaches the scratch, the corrupted pixel prediction is evaluated. This is performed for each line in the scratch from top and bottom edge using symmetrical downwards and upwards moving models and their results are averaged. This helps as to overcome artificial restriction on contextual neighbourhood which have to be causal. The action is performed till there aren't unrepaired pixels in the scratch. In this moment we get the scratch pixel repaired by the data predictions from its left side. Similarly the restoration is done also by the data predictions from the right side. Now it's necessary to link together two possible predictions for each unknown pixel. Simple averaging isn't appropriate, because we need to consider distance of the predicted data from the last known original data. Hence linear interleaving was used to weight the influence of data from both sides of the scratch dependently on the horizontal placement of the predicted pixel on a line of the scratch. Even better results were obtained if exponential interleaving instead linear one was used. This solution gives bigger weight to predictions which are closer to their last known uncorrupted data than linear interleaving variant.

The motion restoration algorithm was implemented in C++ on HP-UNIX operation system. To fulfil the method's requirements the libraries for motion estimation and linear and quadratic regression were developed.

Chapter 6

Results and Discussion

In this section we present results of proposed restoration method and compare them with classical method mentioned in section 4.2 on page 25.

6.1 Test Criteria

For our simulation, the selected frame in image sequence, is corrupted by known "artificial" scratch generated by arbitrary mask image. Thus we know data values in scratch area for comparison with restoration results. To prevent subjective visual comparison of results by different methods, some numerical metric is necessary for their evaluating. Quality of the restoration is measured by Mean Absolute Difference (MAD) between original frame and restored one:

$$MAD(r_3, r_4) = \frac{1}{\nu \xi_{r_3}} \sum_{\forall r \in S} |Y_{r_1,r_2,r_3,r_4} - \tilde{Y}_{r_1,r_2,r_3,r_4}| \qquad (6.1)$$

$\nu = \ card\{\mathcal{S}\}$ is the number of missing multi-spectral scratch pixels and ξ_{r_i} is the number of quantization levels in the i-th spectral band. For all our experimental movies $\xi_{r_i} = \xi = 255 \qquad i = 1, 2, 3$. MAD is computed for each spectral layer separately and thus results can provide also information about colour properties of the image. For faster comparison of results the MAD averaged in all spectral layers is computed

$$MAD_A(r_4) = \frac{1}{d} \sum_{r_3=1}^{d} MAD(r_3, r_4) \ . \qquad (6.2)$$

The used image sequences are described in section 3.5 on page 19.

6.2 The Classical Methods Results

We tested all classical methods on frame No.78 from "tennis" sequence corrupted by different scratches. Results of tests performed with scratch "tmask2"are in

Tab.6.1 and Fig.6.1. Generally the best results were obtained by Linear Regression with support surrounding of two pixel from both side of the scratch. Averaging and Median filtering suffer by blurring of the scratch area. The results of Linear regression method are blurred in direction of interpolation and Quadratic restoration method produce discernible columns in the scratch area.

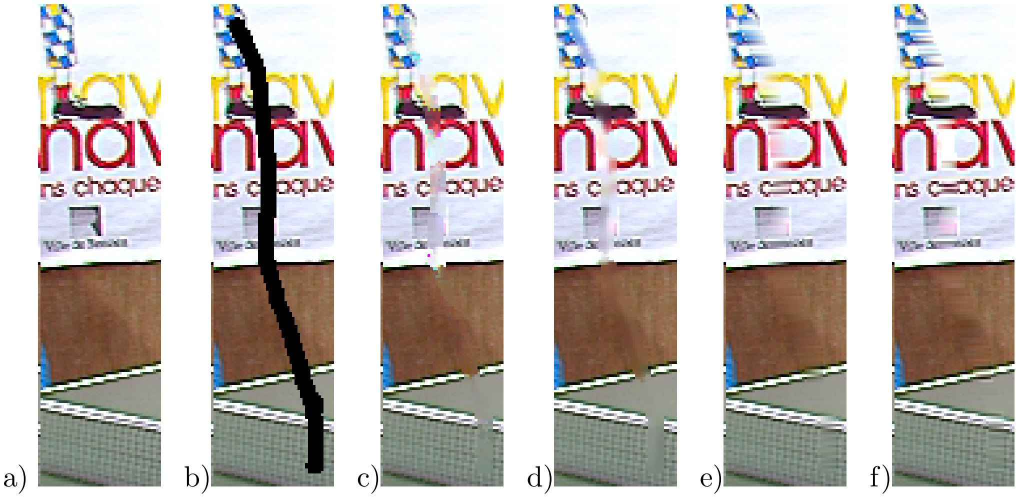

Figure 6.1: Results of restoration of the scratch "tmask2" (b) in the frame No.78 from "tennis" sequence: (a)original image (b)image corrupted by the vertical scratch (c)Averaging (d)Median filtering (e)Linear regression (f)Quadratic regression.

Table 6.1: Results of classical scratch restoration methods on frame No.78 in a "tennis" sequence corrupted by scratch "tmask2" (Fig.6.1-b).

MAD error in spectral band	method			
	Averaging filter	Median filter	Linear regression	Quadratic regression
Red	0.136	0.129	**0.123**	0.125
Green	0.155	0.142	**0.133**	0.140
Blue	0.159	0.142	**0.134**	0.141
average value	0.150	0.138	**0.130**	0.136

6.3 The Proposed Method Results

If we don't use time shift in used contextual neighbourhood (CN) $I_{r_1,r_2,\bullet,r_4}$ the 3.5D CAR model becomes only 3-dimensional. Thus missing data are predicted only from surrounding of the scratch in actual frame. Restoration results of 3D CAR and 3.5D CAR model are affected by spatial shape of the CN, by exponential forgetting factor ρ, and by the length of model history σ. As it was mentioned in Chapter 5 the CN has to be causal. It means that it can include only known or predicted data from previous model movements. We tested different shapes of the 3D causal CN (Fig. A.1 on page 57) on frame from image sequences "tennis" and "cars", corrupted by several scratches (Figs.6.5-b, 6.3-b, 6.4-b, 6.6-b, 6.8-b). The CNs with the best performance for the most of the scratches are marked by frame-box in Fig. A.1 on page 57.

There are also Bayesian methods available for optimal CN shape selection described in section 5.1.1 but their performing involve high computational requirements so they aren't appropriate for online processing. So constant shape of CN can be compromise between speed of the algorithm and quality of the restoration results.

To reach better performance of the method, the more information about missing data surrounding is necessary. This information is easy to obtain from previous or / and following frame(s) if we know their data values. Neighbouring frames we can index by appropriate time shift in CN and the model become 3.5D. Determining of the optimal shape of the CN is here more complicated than in previous case. Performance of the 3.5D CAR model depends except CN's shape also on the number of support pixels in the CN obtained from neighbouring frame(s) in comparison with the number obtained from the actual frame.

On the one hand the more pixels from previous (uncorrupted) frame we choose, the better results are reached. On the other hand the cardinality of CN have influence on computational demand of the whole algorithm so the less cardinality CN has, the faster whole method will be.

The experimentally found CN shapes with the best results for 3D and 3.5D CAR models are shown in Fig.6.2.

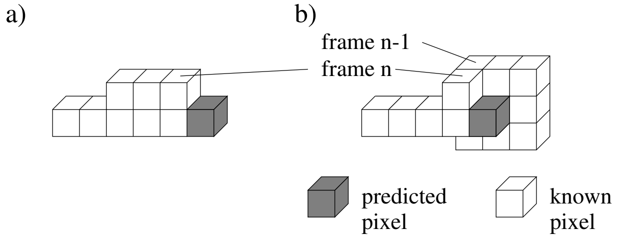

Figure 6.2: The best shapes of contextual neighbourhood I_r for (a)3D CAR model (b)3.5D CAR model. Note that each block in index shift contains data from all spectral layers (RGB).

Unfortunately we can't simply pick up the data from neighbouring frames, because of the motion in the image sequence. So the Motion Compensation (MC) is needed between successive frames. If speed of the object on the scene in the image sequence is less than about 3-4 pixels between neighbouring frames employing of the MC isn't essential and obtained results are satisfactory.

We perform several tests for comparison of the mentioned classical methods with 3D CAR model. Then also the performance of 3D CAR model in comparison with 3.5D CAR model is tested. We divide our test into three categories dependently on the motion velocity of corrupted regions in the image sequence.

During the all tests the forgetting factor ρ is set up to value 0.985 and the length of model history σ is 20 pixels.

6.3.1 Test 1: Slow Motion Velocity in Image Sequence

In these tests we don't perform motion compensation between neighbouring frames. The frames No.77 and No.78 from "tennis" sequence are used.

We tested proposed method on several shapes of the scratches with different placement in the frame.

tmask1 (Fig.6.3-b) presents vertical scratch with average width about 12 pixels. The corrupted area in image sequence is slowly moving body of a tennis player. The scratch is located in whole image sequence mostly on places where occlusion and uncovering appears. The best results from all the classical methods has Median filter in Fig.6.3-c. The results of 3D CAR model are in Fig.6.3-d while performance of 3.5D CAR model is in Fig.6.3-e. MAD values for mentioned methods are in Tab.6.2.

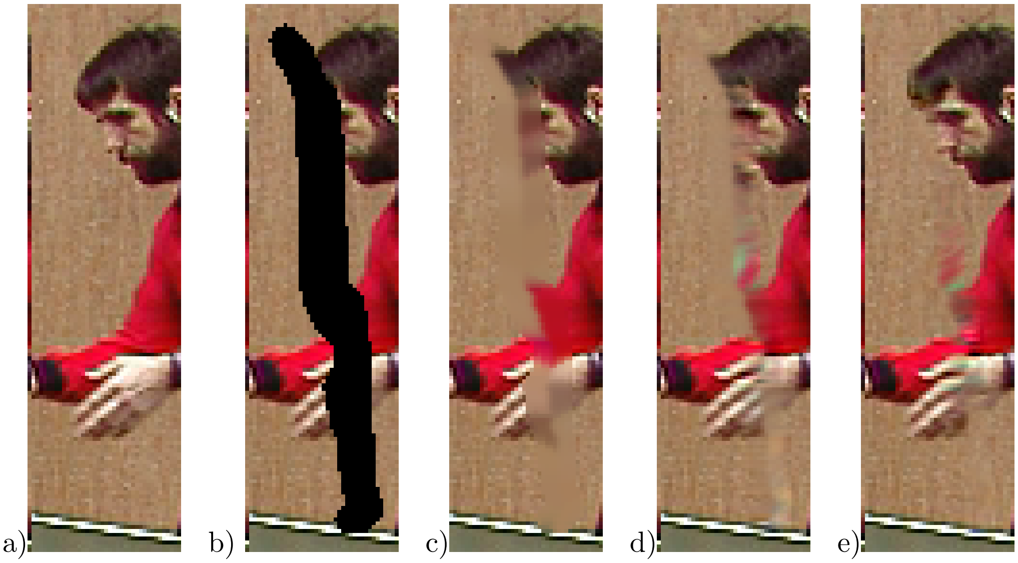

Figure 6.3: Scratch "tmask1" (b) restoration in frame No.78 from the "tennis" sequence (a)original frame-cut (b)corruption by the scratch (c)Median Filter (d)3D CAR model (e)3.5D CAR model.

tmask2 (Fig.6.4-b) is also vertical scratch little curved with average width about 6 pixels. The surrounding of the scratch in the image sequence is wall and the table in the room zoomed and panned in time. The best results has method based on the Linear regression (Fig.6.4-c). The results of 3D CAR model

Table 6.2: MAD computed for proposed method performed on scratch "tmask1" (Fig.6.3-b) in comparison with Median filter.

MAD error in spectral band	Median filter	3D CAR model	3.5D CAR model
Red	0.078	0.078	0.054
Green	0.095	0.090	0.060
Blue	0.084	0.075	0.053
average value	0.085	0.081	0.056

and 3.5D CAR model are illustrated in Fig.6.3-d, Fig.6.3-e respectively. MAD values for all methods are in Tab.6.3.

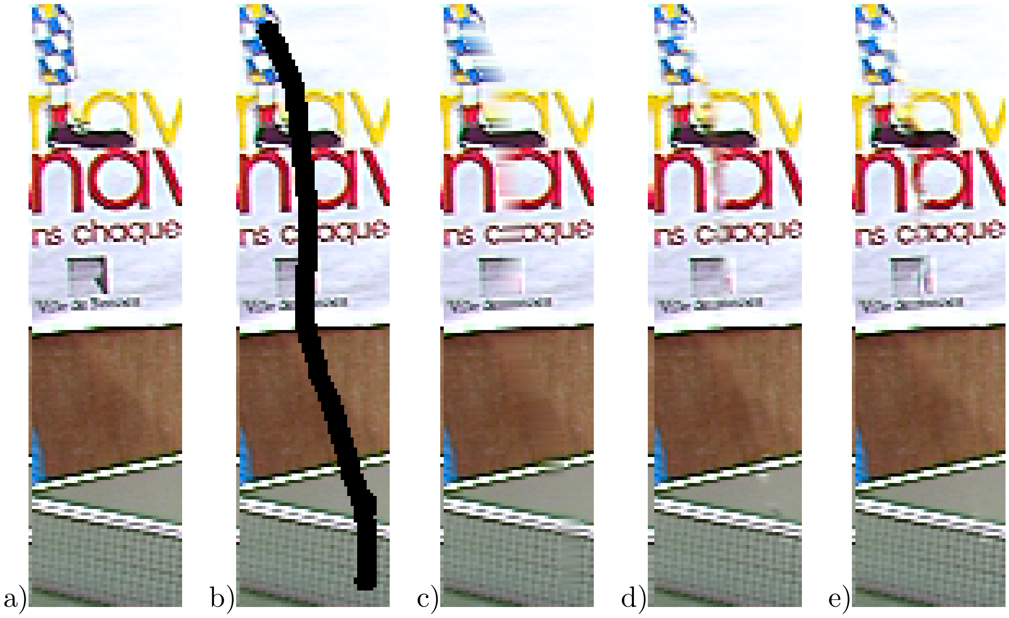

Figure 6.4: Scratch "tmask2" (b) restoration in frame No.78 in the "tennis" sequence (a)original frame-cut (b)corruption by the scratch (c)Linear Regression (d)3D CAR model (e)3.5D CAR model.

tmask3 (Fig.6.5-b) is large area scratch with average width about 20 pixels. The surrounding of scratch in the image sequence is similar as for "tmask2". It is zoomed and panned poster on the wall. The best classical method is again the Linear regression (Fig.6.5-c). Similar MAD values has also Averaging method but a visual appearance of its results is poor. The results of 3D CAR model and 3.5D CAR model are in Fig.6.5-d, Fig.6.5-e respectively. MAD values for all methods are in Tab.6.4.

The results of these tests are encouraging, and performance of 3D and also

Table 6.3: MAD computed for proposed method performed on scratch "tmask2" (Fig.6.4-b) in comparison with Linear Regression method.

MAD error in spectral band	Linear regression	3D CAR model	3.5D CAR model
Red	0.123	0.096	0.079
Green	0.133	0.112	0.088
Blue	0.134	0.117	0.093
average value	0.130	0.108	0.087

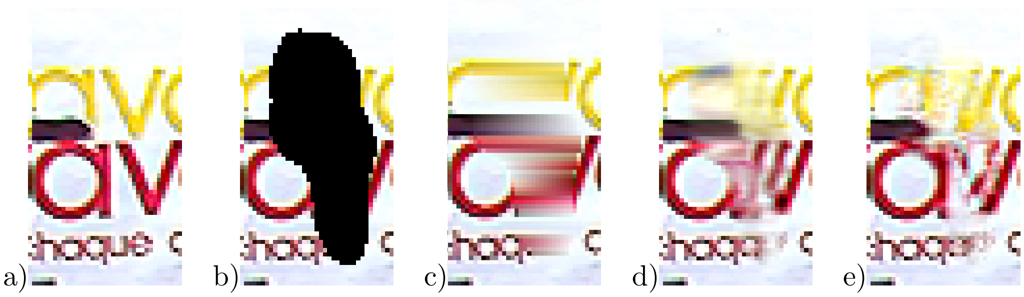

Figure 6.5: Scratch "tmask3" (b) restoration in frame No.78 from the "tennis" sequence (a)original frame-cut (b)corruption by the scratch (c)Linear Regression (d)3D CAR model (e)3.5D CAR model.

Table 6.4: MAD computed for proposed method performed on scratch "tmask3" (Fig.6.5-b) in comparison with Linear Regression method.

MAD error in spectral band	Linear regression	3D CAR model	3.5D CAR model
Red	0.120	0.119	0.088
Green	0.203	0.185	0.133
Blue	0.268	0.246	0.193
average value	0.197	0.183	0.138

3.5D CAR model is satisfactory even if we don't compensate the motion between pixels from CN in different frames. This is possible due to relative small motion speed (about 3-4 pixels/frame) in corrupted regions in the image sequence.

6.3.2 Test 2: Moderate Motion Velocity in Image Sequence

The frames No.0 and No.1 in "tennis" sequence (Fig.3.6-c,d) presents motion of human hand with moderate speed (about 4 pixels/frame). The sequence is corrupted by horizontal scratch **tmask6** (Fig.6.6-b) with average width about 11 pixels, which is placed in image sequence in the area of player's moving hand. Hence it is advisable to compensate the motion between the following frames.

Restoration results without motion compensation are shown in Fig.6.6. Original frame is in Fig.6.6-a and frame corrupted by the scratch “tmask6” is in Fig.6.6-b . Linear regression is the best from the classical methods with results in Fig.6.6-c. After restoration by 3D CAR model we obtain Fig.6.6-d . After adding of information from previous frame (3.5D CAR model) we get Fig.6.6-e. Tab.6.5 offers exact MAD values in all spectral bands as well as MAD$_A$ during the tests.

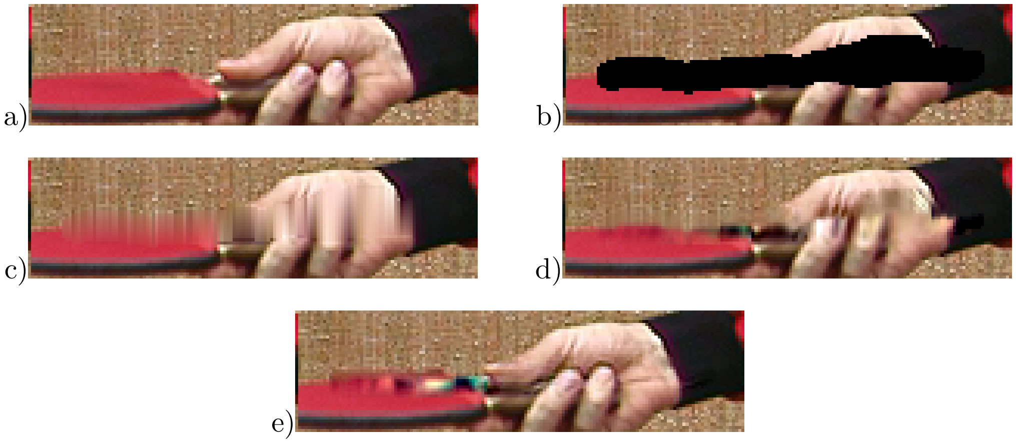

Figure 6.6: Scratch “tmask6” (b) restoration in frame No.1 from the “tennis” sequence (a)original frame-cut (b)corruption by the scratch (c)Linear Regression (d)3D CAR model (e)3.5D CAR model without MC.

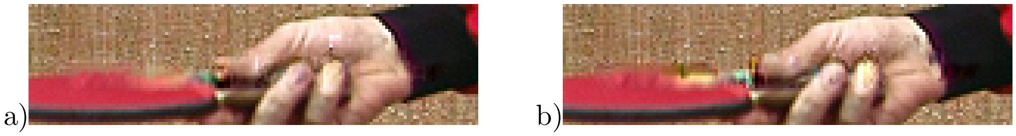

Figure 6.7: Scratch “tmask6” Fig6.6-b restoration in frame “cars77” (a)3.5D CAR model with MC by BM1 method (b)3.5D CAR model with MC by BM2 method.

Table 6.5: MAD computed for proposed method performed on scratch “tmask6” (Fig.6.6-b) in comparison with Linear Regression method.

MAD error in spectral band	Linear regression	3D CAR model	3.5D CAR model
Red	0.104	0.124	0.077
Green	0.123	0.113	0.076
Blue	0.116	0.111	0.065
average value	0.115	0.116	0.073

The 3.5D CAR model brings observable results improvements, in comparison with 3D CAR model only, but there are visible artifacts on the left side of the restored area (Fig.6.6-e) caused by tennis racket motion. To compensate this motion, ME methods described in section 3.1 are employed: the simple Block matching method (BM1) and it's iterative improvement (BM2) introduced in [37]. Results of both methods are illustrated in Fig.6.7. Surprisingly simple method BM1 (Fig.6.7-a) gives better performance then method BM2 (Fig.6.7-b), first of all in mentioned left part of the scratch area. Exact MAD values in all spectral bands as well as MAD_A value for both methods are introduced in Tab.6.6. In term of MAD the results of the both method are worse then 3.5D CAR model without MC, but visually the method BM1 brings observable improvement while method BM2 is of no help.

Table 6.6: MAD computed for 3.5D CAR model with MC method performed on scratch “tmask6” (Fig.6.6-b) in frame No.1 from the “tennis” sequence.

MAD error in	3.5D CAR model with	
spectral band	MC by BM1 method	MC by BM2 method
Red	0.076	0.104
Green	0.093	0.123
Blue	0.087	0.116
average value	0.085	0.115

6.3.3 Test 3: Fast Motion Velocity in Image Sequence

When the fast motion (more than 10 pixels/frame) occurs in an image sequence the MC between neighbouring frames is essential for proper function of 3.5D CAR model. We performed our tests on frames No. 76 and 77 in “cars” sequence (Fig.3.7) corrupted by the scratch **mask4** (Fig.6.8-b). This scratch has thick vertical shape with average width about 10 pixels and is placed in the image sequence on the horizontally moving car-toy.

Firstly we performed the tests without MC and results are in Fig.6.8. Original frame is in Fig.6.8-a and frame corrupted by the scratch “mask4” is in Fig.6.8-b. Linear regression provided again the best results from all the classical methods Fig.6.8-c. After restoration by 3D CAR model we obtain Fig.6.8-d. After adding of information from previous frame (3.5D CAR model) we get Fig.6.8-e. From the last image we can see obvious influence of the fast uncompensated motion, so it means that data included into model from previous frame do not match with those in current frame. Exact MAD values in all spectral bands as well as MAD_A value are introduced in Tab.6.7.

To overcome this poor 3.5D CAR model results the MC compensation is necessary. We performed ME by the same methods as in previous section. The results of method BM1 and BM2 are depicted in Fig.6.9-a, Fig.6.9-b respectively. There is only slight improvement in comparison with 3D model and uncompensated 3.5D CAR model, and the better method for fast motion estimation is essential. To prove that with better ME method the results can be satisfactory the motion is estimated manually as constant motion in one direction. This is easy task in "cars" sequence, due to the movement of whole scratch area only for 18 pixels to the left in horizontal direction between previous and restored frame. The result of this "ideal" ME method is depicted in Fig.6.9-c. The quality of restoration grew obviously. All numerical results of tested MC methods in terms of Mean Absolute Difference (MAD) are summed if Tab.6.8.

Also employing of this MC algorithm demands more computations and prolongs the total time for restoration in comparison with 3.5D CAR model without motion compensation.

The results of the whole restoration method during the fast motion can be improved in several ways. The often used approach is to include multi-resolution pyramid onto the ME scheme as was described in section 3.1. Next possible improvement can be performed by implementation of more sophisticated methods for the motion vectors interpolation into the scratch area from its surrounding.

Table 6.7: MAD computed for proposed method performed on scratch "mask4" (Fig.6.8-b) in frame No.77 from the "cars" sequence in comparison with Linear Regression method.

MAD error in spectral band	linear regression	3D CAR model	3.5D CAR model	Motion Compensated 3.5D CAR model
Red	0.210	0.149	0.182	0.086
Green	0.173	0.138	0.203	0.085
Blue	0.095	0.115	0.186	0.089
average value	0.159	0.134	0.190	0.087

Table 6.8: MAD computed for 3.5D CAR model with MC performed on scratch "mask4" (Fig.6.8-b) in frame No.77 from the "cars" sequence.

MAD error in	3.5D CAR model with		
spectral band	MC by BM1 method	MC by BM2 method	manual MC
Red	0.143	0.149	0.092
Green	0.131	0.136	0.087
Blue	0.110	0.116	0.071
average value	0.128	0.134	0.083

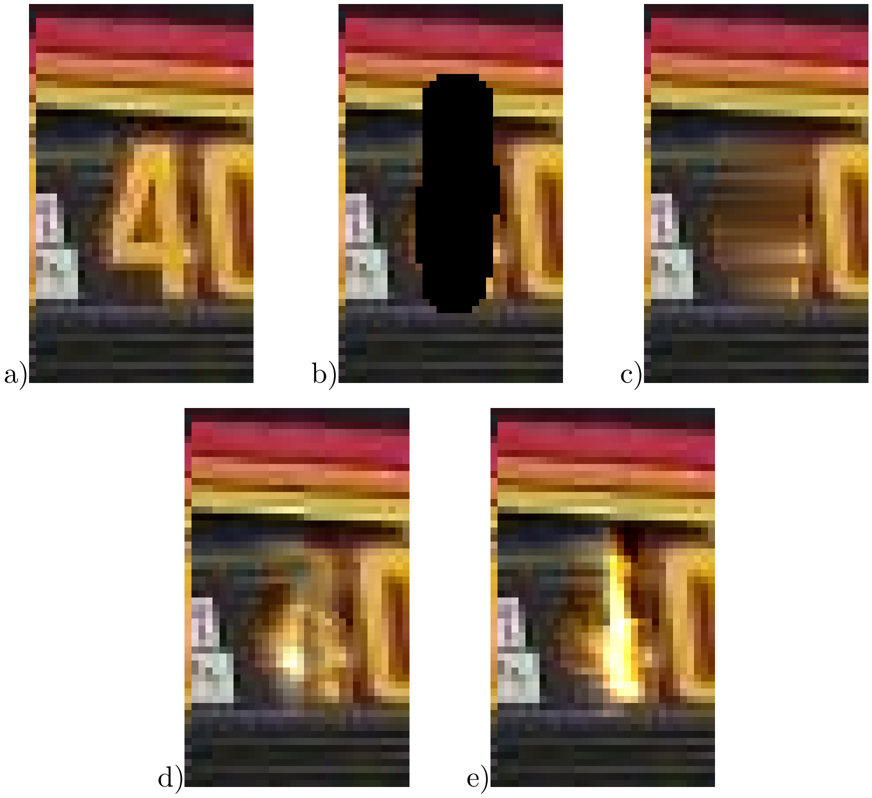

Figure 6.8: Scratch "mask4" (b) restoration in frame No.77 from "cars" sequence (a)original frame-cut (b)corruption by the scratch (c)Linear Regression (d)3D CAR model (e)3.5D CAR model without MC.

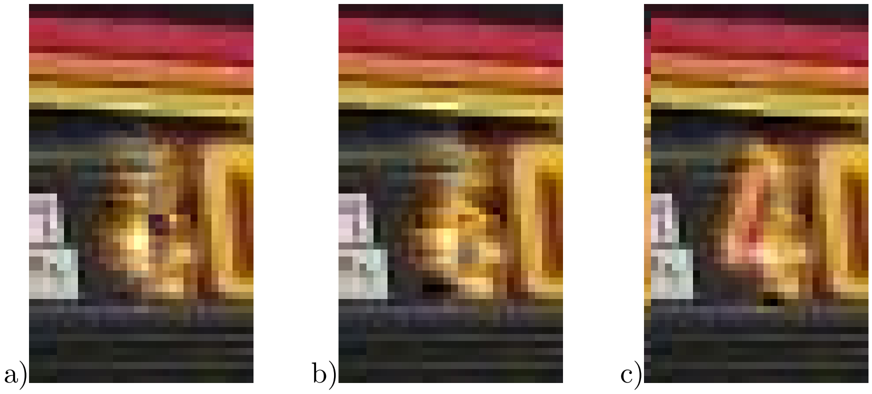

Figure 6.9: Scratch "mask4" (Fig6.8-b) restoration in frame No.77 from "cars" sequence (a)3.5D CAR model with MC by BM1 method (b)3.5D CAR model with MC by BM2 method (c)3.5D CAR model with manual MC.

Influence of parameters σ and ρ on results is described in Figs.6.11, 6.12. There are depicted these graphs for both 3D (Figs.6.11-a, 6.12-a) and 3.5D CAR model (Figs.6.11-b, 6.12-b). From dependency of average MAD on forgetting factor ρ (Fig.6.12) is obvious that for tested scratches are the best values $\rho = 0.980 - 0.995$. Dependency of average MAD on length of model history σ is more difficult to interpret. From the graph Fig.6.11 we conclude that appropriate σ value depends on the scratch width. So the larger the scratch is the bigger value of σ is necessary to get satisfactory results. On the other hand the long model history σ can negative affect the results when the predictor updates his parameters by improper data in further distance from the scratch and this data don't match with scratch tight surrounding. This occurs even if the forgetting of older data by exponential factor ρ is performed. Example in Fig.6.10-c shows bad prediction in the upper part of the scratch caused by this problem when $\sigma = 20$ pixels and the scratch width is about 6 pixels.

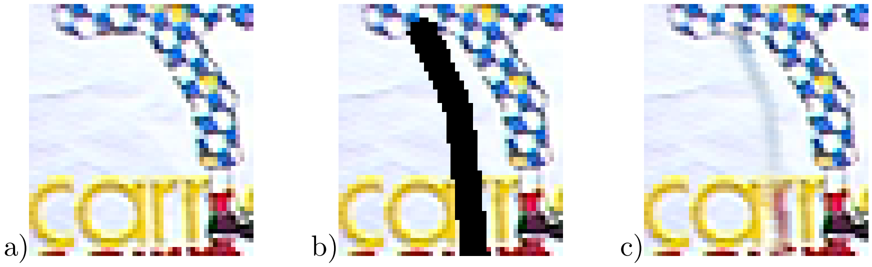

Figure 6.10: The example of improper data prediction when the model history σ is too long. (a)original (b)with scratch (c)restored by 3D CAR model $\sigma = 20$pixels.

Thus it is advisable to adapt σ value choosing dependently on the scratch shape and data pattern in the surrounding. From graph for 3D CAR model in Fig.6.11 we induce that appropriate value for scratches "tmask1" Fig.6.3-b and "tmask2" Fig.6.4-b is $\sigma = 5$ pixels and for scratch "tmask3" Fig.6.5-b $\sigma = 15$ pixels. After several experiments we conclude that appropriate equation for σ value choosing is

$$\sigma_{OPT} = \frac{\text{actual scratch width}}{2} + 5 \tag{6.3}$$

To prevent MAD error increasing when the σ is too small (see Fig.6.11) the minimal σ is constrained to 5 pixels (6.3). Thus the predictor have always certain information about character of the data in the past.

Final results are affected, except mentioned model parameters, also by kind of data pattern in the scratch surrounding. The method results are influenced also by number of CAR model parameters initialisations. However the influence isn't so significant thanks to forgetting factor ρ, which suppress the older data impact during predictor parameters computation.

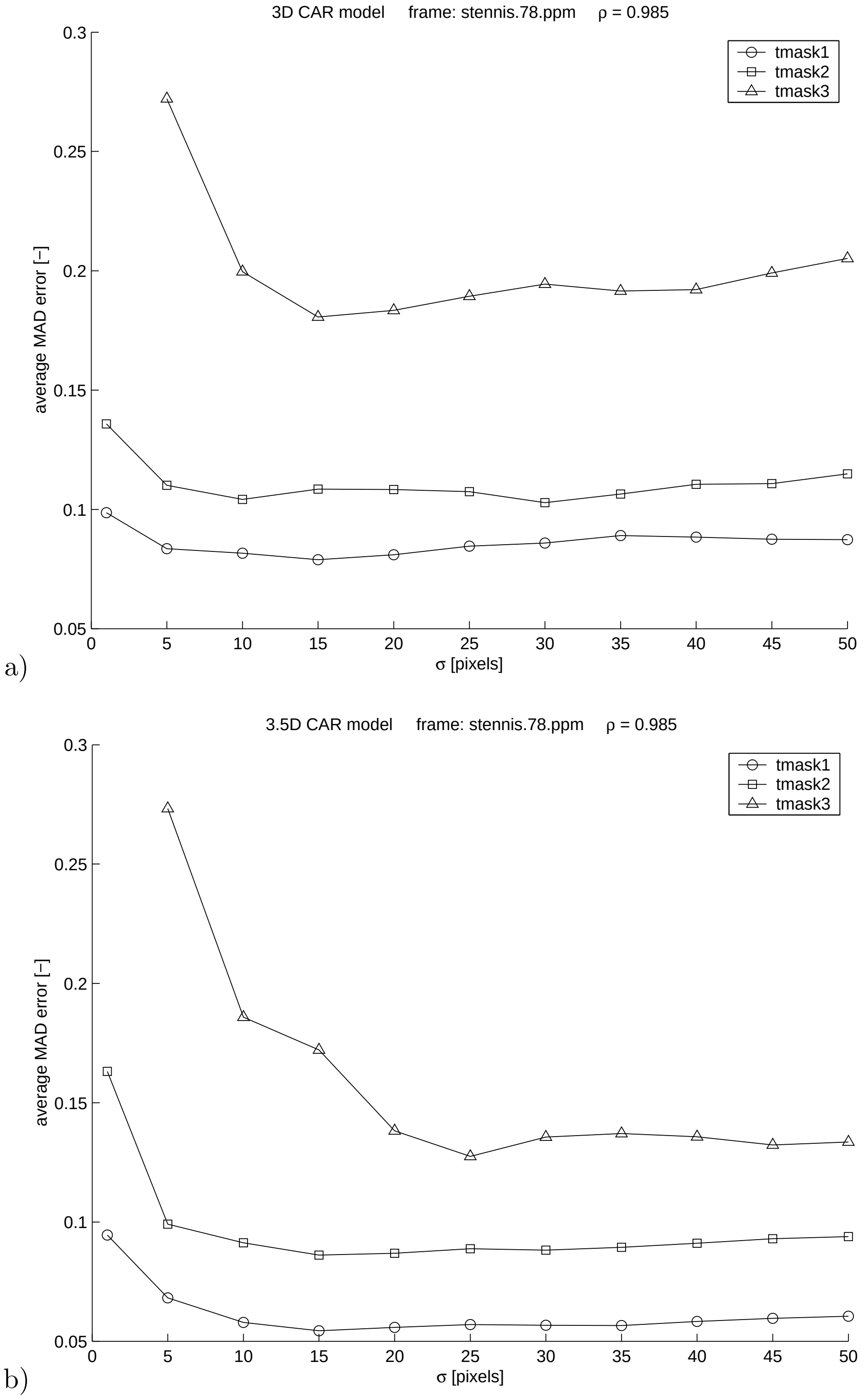

Figure 6.11: Dependency of average MAD value (MAD_A) on the length of model history σ during restoration of different scratches in frame No.78 from the "tennis" sequence (a) by 3D CAR model (b) by 3.5D CAR model.

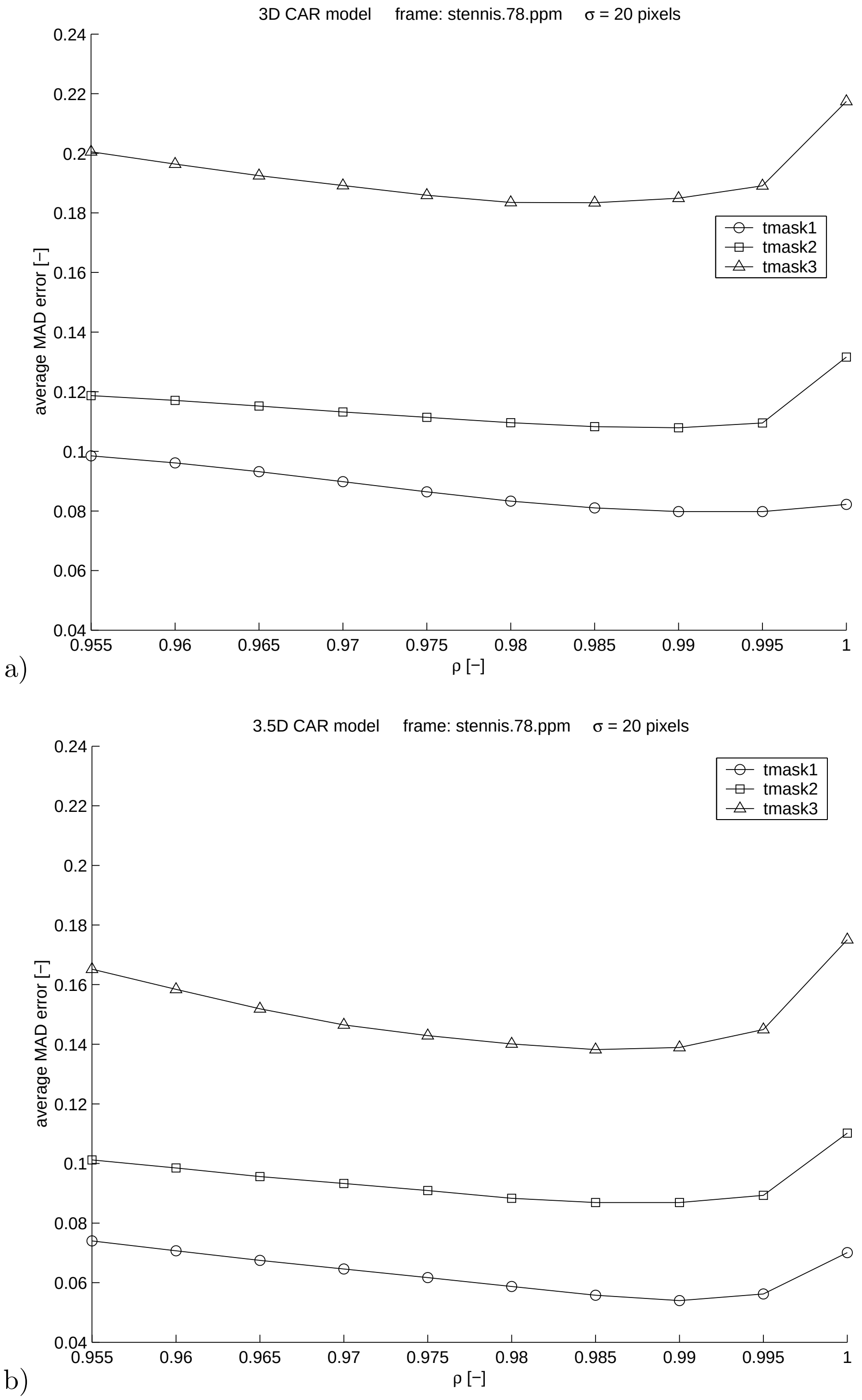

Figure 6.12: Dependency of average MAD value (MAD_A) on the forgetting factor ρ during restoration of different scratches in frame No.78 from the "tennis" sequence (a) by 3D CAR model (b) by 3.5D CAR model.

Chapter 7

Conclusions and Further Research

The results of our tests are encouraging. The proposed method has a superior performance to the classical methods in terms of the mean absolute difference (MAD) as well as the subjective visual quality. For sequences with fast moving objects, the method requires a fast and accurate motion detection. However even if the reliable motion estimation is missing the method can be easily restricted to single frame support data with slightly compromised restoration quality. The proposed method is fully adaptive, numerically robust and with moderate computation demands so it can be applied in an on-line restoration systems. The algorithm is not limited to motion pictures only. It can be applied on any degraded image sequence obtained by applications in remote sensing, astronomy, microscopy or medical field.

The way in further improvement of the method is in adaptive choosing of parameters ρ and σ dependently on the scratch size and data pattern which surrounds the corrupted area. There is also possible to test Jeffreys parameter prior instead of used normal-Wishart during computing of one-step-ahead prediction.

The proposed restoration method restores the data in horizontal or vertical direction. This can be drawback if some more difficult scratch shape occurs. The one solution is to divide the scratch into several more-or-less horizontal and vertical parts and restore them in a sequence. The complexity of the scratch shape can be easily evaluated if the scratch skeleton is computed. The scratch then can be simply segmented in places where the skeleton curve changes direction. The example of idea is depicted in Fig.7.1.

In this thesis was mentioned necessity of motion compensation for proper function of 3.5D CAR model. The second necessary application of motion compensation is scratch detection in image sequence. The shape of the scratches evaluated in this thesis is almost arbitrary so it is impossible to use methods for scratch detection assuming some apriori information about scratch shape or placement in image as for example Hough transformation for vertical line scratches [18]. So the universal scratch detection assumes knowledge about motion between two con-

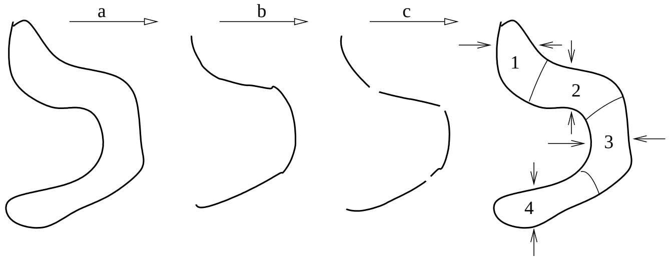

Figure 7.1: An idea of difficult scratch shape restoration using its skeleton.

secutive frames in image sequence. The pixel values in one frame are then shifted accordingly to the known motion vectors in pixel coordinates in second frame (motion compensation). Then the difference between both frames is computed and after the results thresholding the corrupted areas (scratch) are detected.

There are several remaining problems in the field of image sequences scratch restoration. One of them consists in the appropriate restoration method parameters adaptation according to known data in image sequence. The second task is robust and fast scratch detection, which is based mainly on proper motion estimation algorithm. So the fast and accurate motion estimation method development is the main encourage for the further research.

Bibliography

[1] B. Alp, P. Haavisto, T. Jarske, K. Öistämö, and Y. Neuvo. Median-based algorithms for image sequence processing. In *Proceedings of SPIE Visual Communication and Image Processing*, pages 122–133, 1990.

[2] G.R. Arce. Multistage order statistic filters for image sequence processing. *IEEE Transactions on Image Processing*, 39:1146–1161, May 1991.

[3] S. Armstrong, A.C. Kokaram, and P. Rayner. Reconstructing missing regions in colour images using multichannel median models. In *Proceedings of European Conference on Signal Processing (EUSIPCO '98)*, volume 2, pages 1029–1032, September 1998.

[4] S. Armstrong, A.C. Kokaram, and P.J.W. Rayner. A bayesian framework for reconstructing missing data in colour image sequences. In *Proceedings of SPIE Conference on Bayesian Inference for Inverse Problems*, pages 21–28, July 1998.

[5] Y.L. Chan and W.C. Siu. An efficient search strategy for block motion estimation using image features. *IEEE Transactions on Image Processing*, 10(8):1223–1238, August 2001.

[6] Y.S. Chen, Y.P. Hung, and Fuh C.S. Fast block matching algorithm based on the winner-update strategy. *IEEE Transactions on Image Processing*, 10(8):1212–1222, August 2001.

[7] J.H. Chenot. AURORA - summary of project results. *Project Number: AC072, CEC Deliverable Number: AC072-INA-RD-DS-P-D13-b1*, May 1999.

[8] F. Dufaux and J. Konrad. Efficient, robust, and fast global motion estimation for video coding. *IEEE Transactions on Image Processing*, 9(3):497–501, March 2000.

[9] B. Fisher and U. Schroeder. Video formats and compression methods. *URL: http://www6.tomshardware.com/video*, September 1999.

[10] S. Godsill and A.C. Kokaram. Restoration of image sequences using a causal spatio-temporal model. In *Proceedings of 17th Leeds Annual Statistics Re-*

search Workshop (The Art and Science of Bayesian Image Analysis), pages 189–194, July 1997.

[11] M. Haindl. An Adaptive Image Reconstruction Method. Technical Report CS-R9537, CWI, Amsterdam, 1994.

[12] M. Haindl and J. Filip. A Fast Model-Based Restoration of Colour Movie Scratches. Technical Report 2031, ÚTIA AV ČR, Praha, 2001.

[13] M. Haindl and S. Šimberová. A Regression Type Image Destriping Method. In *Biosignál '90*, pages 74–75, Brno, June 1990. Dům techniky ČSVTS.

[14] M. Haindl and S. Šimberová. *Theory & Applications of Image Analysis*, chapter A Multispectral Image Line Reconstruction Method, pages 306–315. World Scientific Publishing Co., Singapore, 1992.

[15] M. Haindl and S. Šimberová. A scratch removal method. *Kybernetika*, 34(4):423–428, 1998.

[16] M. Haindl and S. Šimberová. A high-resolution radiospectrograph image reconstruction method. *Astronomy and Astrophysics, Suppl.Ser.*, 115(1):189–193, January 1996.

[17] A.C. Kokaram. *Motion Picture Restoration.* PhD thesis, Cambridge University Engineering department, Trumpington Street, Cambridge CB2 1PZ, England, 1993.

[18] A.C. Kokaram. *Motion Picture Restoration.* Springer, London, Great Britain, 1998.

[19] A.C. Kokaram and S.J. Godsill. A system for reconstruction of missing data in image sequences using sampled 3d AR models and MRF motion priors. In *Proceedings of European Conference on Computer Vision*, pages 613–624, April 1996.

[20] D. Le Gall. MPEG - digital image and video standards. *Communication of the ACM*, 34(4):47–58, April 1991.

[21] J. Magarey, A.C. Kokaram, and N. Kingsbury. Optimal schemes for motion estimation on colour image sequences. In *Proceedings of IEEE International Conference on Image Processing*, pages 187–190, October 1997.

[22] J. Magarey, A.C. Kokaram, and N. Kingsbury. Robust motion estimation using chrominance information in colour image sequences. *Image Analysis and Processing, Springer-Verlag*, 2:486–493, 1997.

[23] M.E.AI. Mualla, C.N. Canagarajah, and D.R. Bull. Motion field interpolation for temporal error concealment. *IEEE Proc.-Vis. Image Signal Process.*, 146(5):445–453, October 2000.

[24] F.A. Mujica, J.P. Leduc, R. Murenzi, and M.T.J. Smith. A new method parameter estimation algorithm based on the continuous wavelet transform. *IEEE Transactions on Image Processing*, 9(5):873–888, May 2000.

[25] Ohio State University Page. Computer vision database. *URL: http://sampl.eng.ohio-state.edu/~sampl/database.htm*, April 2001.

[26] H.W. Park and Kim H.S. Motion estimation using low-band-shift method for wavelet-based moving-picture coding. *IEEE Transactions on Image Processing*, 9(4):577–587, April 2000.

[27] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recepies in C*. Cambridge University Press, 1992.

[28] The AURORA project description. *URL: http://www.ina.fr/Recherche/Aurora/aurora.en.html*, September 2000.

[29] The BRAVA project description. *URL: http://www.ina.fr/Recherche/Brava/index.en.html*, September 2001.

[30] P. Schallauer, A. Pinz, and W. Haas. Automatic restoration algorithm for 35 mm film. *Journal of Computer Vision Research*, 1(3):60–85, 1999.

[31] S.K. Su and R.M. Mersereau. Motion estimation methods for overlapped block motion compensation. *IEEE Transactions on Image Processing*, 9(9):1509–1521, September 2000.

[32] D. Suter and P. Richardson. Historical film restoration and video coding. In *Proceedings of PCS'96, Melbourne*, pages 389–394, 1996.

[33] J. Taylor. DVD frequently asked questions (and answers). *URL: http://www.dvddemystified.com/dvdfaq.html*, October 2001.

[34] P.M.B. van Roosmalen, A.C. Kokaram, and J. Biemond. Fast high quality interpolation of missing data in image sequences using a controlled pasting scheme. In *Proceedings of IEEE Conference on Acoustics Speech and Signal Processing (ICASSP '99)*, volume IMDSP 1.2, pages 3105–3108, March 1999.

[35] F. Völkel. MPEG-4 - copying a DVD video to CD-ROM. *URL: http://www6.tomshardware.com/video*, September 2000.

[36] J. Wei and Z.N. Li. An efficient two-pass MAP-MRF algorithm for motion estimation based on mean field theory. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(6):960–972, 1996.

[37] S. Zhu and K-K. Ma. A new diamond search algorithm for fast block-matching motion estimation. *IEEE Transactions on Image Processing*, 9(2):287–290, February 2000.

List of Figures

List of Tables

Appendix A

A.1 Tested Contextual Neighbourhoods

The CN with the best performance during the testing on “’tennis” and “cars” sequences are marked by dashed framebox.

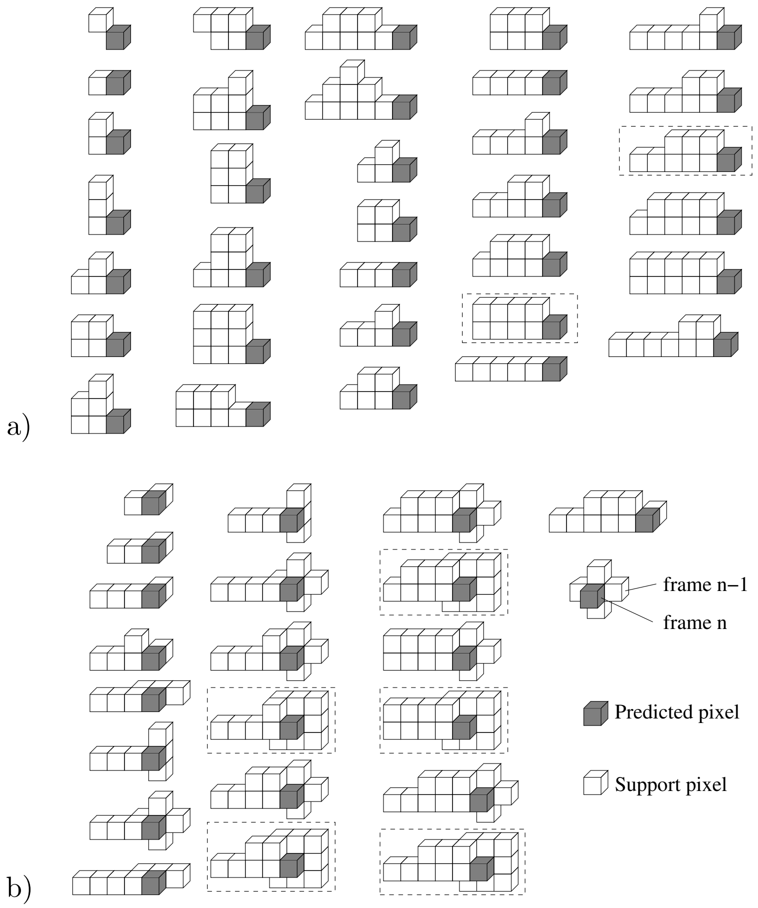

Figure A.1: An overview of all tested contextual neighbourhoods (a)for 3D CAR model (b)for 3.5D CAR model.

Appendix B

B.1 The Enclosed CD Contents

In this thesis is enclosed a CD with:

- animations of restored sequences,
- complete presentation in Windows PowerPoint,
- electronic version of this document in a PostScript format.