2009

Adaptive deinterlacing of video sequences using motion data

Elham Shahinfard
University of Windsor

Adaptive Deinterlacing of Video Sequences Using Motion Data

by
Elham Shahinfard

A Dissertation
Submitted to the Faculty of Graduate Studies
through the Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada
2009

Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-57655-7
Our file Notre référence
ISBN: 978-0-494-57655-7

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published or submitted for publication.

I certify that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances to my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

Abstract

In this work an efficient motion adaptive deinterlacing method with considerable improvement in picture quality is proposed. A temporal deinterlacing method has a high performance in static images while a spatial method has a better performance in dynamic parts. In the proposed deinterlacing method, a motion adaptive interpolator combines the results of a spatial method and a temporal method based on motion activity level of video sequence.

A high performance and low complexity algorithm for motion detection is introduced. This algorithm uses five consecutive interlaced video fields for motion detection. It is able to capture a wide range of motions from slow to fast. The algorithm benefits from a hierarchal structure. It starts with detecting motion in large partitions of a given field. Depending on the detected motion activity level for that partition, the motion detection algorithm might recursively be applied to sub-blocks of the original partition. Two different low pass filters are used during the motion detection to increase the algorithm accuracy. The result of motion detection is then used in the proposed motion adaptive interpolator.

The performance of the proposed deinterlacing algorithm is compared to previous methods in the literature. Experimenting with several standard video sequences, the method proposed in this work shows excellent results for motion detection and deinterlacing performance.

To Payman

Acknowledgments

Here, I wish to express my appreciation to all the wonderful people who have brightened my professional and personal life in the past few years.

I thank my advisor, Professor Sid-Ahmed for his guidance and support. Prof. Sid-Ahmed allowed me to work independently and explore various options. Meanwhile he was always available for discussing problems and providing me with valuable insights. Not only was he a great advisor in my research but he was also a supportive father far from home who led me through many challenges during my first years in Canada.

I am grateful to Prof. Ahmadi for his contributions in my Ph.D. studies and his extreme patience and insightful ideas. His support and encouragement have truly made a difference in my life. He works hard for maintaining a wonderful environment in the department. His close and friendly interaction with graduate students is inspiring and has benefited all of us.

I would like to thank Prof. J. Wu and Prof. J. X. Chen for dedicating their time to be on my Ph.D. committee and for providing me with valuable suggestions for the progress of my research work.

I acknowledge Prof. M. R. El-Sakka for kindly accepting to serve on my Ph.D. defense committee and reading this thesis. I am thankful to Prof. M. Wang for kindly accepting to be on my Ph.D. defense committee.

I would like to acknowledge and thank our department staffs, Ms. Shelby Marchand, Ms. Andria Ballo, Mr. Don Tersigni, and Mr. Frank Cicchello, for their professional assistance on administrative and technical matters. My special thanks go to Ms. Andria Ballo who has also been a true friend to me. I am extremely thankful to her for her kindness and care.

I would like to give my sincere thanks to my dear friends who have helped me to extend my vision and have added enthusiasm, happiness and comfort to my life. In particular, I would like to express my gratitude to Pegah, Mahzad, Mojgan, Leila, Neda, Arash, Hani and Khorameh who have helped me through my professional and personal life in the past few years.

My warmest thanks go to my lovely family for their unconditional love and support. I am grateful to my parents who have done everything in their power to help me reach my goals. I thank my sisters and brother for being with me through struggles and successes and sharing my disappointments and joys.

My deep love and appreciation goes to Payman; my true friend and my love. I thank him for being full of surprises and for sharing his exceptional and beautiful view of life with me. I am grateful to him for the valuable discussions we had about my research on the phone while I was in Windsor. I am indebted to him for the time he has spend criticizing my research and proofreading my papers. I dedicate this thesis to him as a sincere expression of appreciation and love.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AC-3	Adaptive Transform Coder 3
ACATS	Advisory Committee on Advanced Television Service
ATEL	Advanced Television Evaluation Laboratory
ATRC	Advanced Television Research Consortium
ATTC	Advanced Television Test Center
DTV	Digital TV
EDTV	Enhanced Digital TV
EIA	Electronic Industries Association
FCC	Federal Communication Commission
GA-HDTV	Grand Alliance of HDTV
GI	General Instrument Corporation
HDTV	High Definition TV
MIT	Massachusetts Institute of Technology
MPEG	Moving Picture Experts Group
MUSE	Multiple sUb-Nyquist Sampling Encoding system
NHK	Nippon Hodo Kyokai
NTSC	National Television System Committee
OSI	Open System Interconnection
VLSI	Very Large Scale Integration
VT	Vertical-Temporal

Chapter 1

Introduction

1.1 Motivation

High definition TV has attracted considerable attention in recent years. It provides stunning picture quality and details with vivid colors. It is going to serve as an ultimate replacement for current analog TV system.

Canada and U.S. have adopted the Advanced Television Systems Committee standard

(A/53) as their future TV standard [1-2]. The standard offers a fully digital system with options for both narrow-screen and wide-screen as well as “low quality” and “high quality” formats [3]. However it seems that progressive high quality format is going to be the most preferred format for the future. Since, it expands some potential for interactivity with computer and data services besides improving picture quality.

Old analog television systems use interlaced video format in their standards as a compression format. Interlacing benefits from the fact that human visual system is less sensitive to flickering details than to large-area flicker [4-5]. However, with recent advances in video compression and telecommunication interlaced video format is not the preferred format anymore. Accordingly providing efficient high quality deinterlacing methods for converting interlaced video to progressive format is becoming more challenging.

Over the last few decades many deinterlacing methods have been proposed. These algorithms could be categorized into three main categories: spatial methods, temporal methods and hybrid methods.

Spatial methods interpolate the missing lines of a field using the data exclusively from that same field. They employ different types of spatial interpolations ranging from simple line repetition to edge dependent interpolation methods [5]. Spatial methods are independent of motion activity in the video sequence. These methods ignore the high level of correlation that typically exists between successive video fields; as a result their performance is not optimal. Spatial methods are also known as intrafield methods.

Temporal methods such as field insertion interpolate the missing lines of a field by using temporal data from previous and/or subsequent fields [6]. As a result of high temporal correlation that typically exists between successive fields of a video sequence, the performance of a temporal method improves in static regions. However the improvement happens in static regions only and overall results are vulnerable to motion. Being vulnerable to motion and ignoring spatial correlation are two drawbacks of temporal methods resulting in suboptimal performance.

The idea of hybrid deinterlacing methods is to exploit both spatial and temporal correlations data for deinterlacing. By using temporal as well as spatial data for deinterlacing the performance of a deinterlacing method may improve substantially.

A wide range of hybrid methods are available in literature. The most advanced hybrid deinterlacing methods are motion compensated and motion adaptive methods [5].

In a motion compensated method the motion data is virtually removed from the video signal prior to interpolation. Once the motion is removed a temporal deinterlacing method will be applied for interpolation purpose. Motion data will be added to the video at a later stage [5],[7-10]. Motion compensated methods require the use of complicated motion estimation algorithms since their performance is highly dependent on successful motion removal. However it may still suffer from some common motion estimation obstacles such as object deformation, appearance and disappearance of objects, scene changes and sub-pixel motions [10-12]. In overall, a motion compensated deinterlacing method is perfect for high quality deinterlacing of videos taken from a panning camera while the objects have translational motions but it does not work as good for all types of video.

Motion adaptive methods are among the most popular deinterlacing methods. They adapt themselves to motion activity level of the video sequence. Depending on the detected level of motion activity they may use a simple temporal method in static parts of the video or a spatial method in highly dynamic parts or a combination of both [13-16]. This thesis focus on motion adaptive deinterlacing methods due to many potential advantages including: automatic handling of scene changes, automatic optimization for object deformation or appearing and disappearing of objects and most importantly high quality deinterlacing results.

1.2 Objectives

Main objective of this research has been proposition of a deinterlacing method which produces high quality progressive video. After extensive literature survey and implementation of several deinterlacing algorithms, motion adaptive deinterlacing has been selected as the most proper category. A high quality motion adaptive deinterlacing method is proposed. It includes a motion detection algorithm and an interpolation algorithm. Motion detection algorithm works on different color components separately. Motion detection results of the three color component are combined in a later stage to improve output uniformity. Proposed interpolation algorithm uses motion detection results to interpolate missing lines.

The main features which have been sought in the proposed algorithm are:

1. Accurate detection of motion in video sequences
2. Selection of a suitable temporal (interframe) method
3. Selection of a suitable spatial (intraframe) method
4. Appropriate combination of chosen interframe and intraframe deinterlacing methods
5. Obtaining high quality deinterlaced video in the output

1.3 Contributions

This thesis proposes a motion adaptive deinterlacing method with an accurate motion detection algorithm. Being adaptive to motion, proposed interpolation algorithm adaptively changes its coefficients to obtain optimum interpolation results.

This thesis includes three dominant contributions. The first one is the hierarchical motion detection algorithm. The second one is threshold optimization method. The third one is the corresponding motion adaptive interpolation algorithm which is used to estimate the missing lines of interlaced video signal.

Motion activity level in video sequence is measured by the proposed hierarchical motion detection algorithm. Motion detection algorithm uses five consecutive fields of interlaced video for motion detection. It is able to capture a wide range of motions from slow to fast. The hierarchical structure starts with detecting motion in large partitions of a given field. Depending on the detected motion activity level, the motion detection algorithm might recursively be applied to sub blocks of the original partition.

Two low pass filters are used in the motion detection algorithm to increase the algorithm accuracy. Proposed motion detection algorithm has proven to obtain accurate results in experimental simulations.

The same motion detection algorithm is applied to each color component of the video signal. Their results are compared in a later stage to guarantee the uniformity of the final results.

Motion detection results are then used to calculate adaptation factor for interpolation step. The method we have proposed for calculation of adaptation factor uses a mean square optimization technique to adaptively optimize the threshold values used in calculation of adaptation factors.

The proposed motion adaptive interpolation uses motion detection results for adaptive adjustment of its coefficients, where the ultimate goal is high quality deinterlacing results. Excellent experimental results are obtained for motion detection and deinterlacing performance.

1.4 Outline of the thesis

The motivation for the proposed research, the overall objectives of designed algorithms and contributions were outlined in this chapter.

A review of TV industry evolution and the principles of High Definition TV are addressed in Chapter 2.

Mathematical definitions of video signals and different scanning formats are briefly reviewed in Chapter 2. The most popular categories of deinterlacing methods are surveyed in Chapter 4. Theoretical approach to motion detection and several well know motion adaptive methods are explained in Chapter 5.

Proposed motion detection algorithm which is the first contribution of this thesis is explained in Chapter 6, this Chapter concludes with presenting motion detection simulation results. Chapter 7 explains the proposed threshold value optimization method followed by the proposed motion adaptive deinterlacing method.

Chapter 8 starts with description of simulation setup and evaluation criteria used in this research. Chapter 8 continues with presenting the proposed algorithm simulation results, followed by comparison results. Chapter 9 concludes the thesis.

Chapter 2

Digital High Definition TV

Digital high definition television was introduced in 1998 in the US and is replacing analog television that was introduced in 1930's. It is considered as one of the most exciting industry innovation which is going to change the meaning of TV as an standalone device.

High definition TV (HDTV) is also influencing the video communication applications including video phones and video conferencing on top of information infrastructure.

This chapter presents an introduction to HDTV standard. History of TV industry since its early invention is reviewed briefly in section 2.1. Section 2.2 explains history of HDTV in North America and the formation of the Grand Alliance of HDTV. High Definition TV features are explained in section 2.3. Section 2.4 explains HDTV video formats.

2.1 History of TV Industry

The origins of the TV industry possibly will be traced back to the discovery of Selenium photoconductivity in 1873[18], and then by the invention of a scanning disk in 1884 by Paul Nipkow [17]. Nipkow's scanning disk became the main part of the first electromechanical television system in the world in 1884 [17-19].

The next progress in the field could be considered the invention of an image dissector camera tube in 1927 by Philo Farnsworth [18]. The invention of this camera tube was a step toward electronic systems. In 1929, Philo Farnsworth broadcasted a motion picture film on air by an electronic television system. By further improvement to his system, Philo Farnsworth demonstrated the first all-electronic television system in the world in 1934. It was a live presentation of moving half-tone images publicized in Philadelphia Franklin institute [17-18].

Later in 1940, the first electronically scanned color TV was successfully developed, where images from two picture tubes were combined into a single projection screen [18].

The first public demonstration of color TV was initiated by Colombia Broadcasting System (CBS) broadcasting Inc on January 12th 1950. The demonstration was given in a public building in Washington, D.C., on eight 16-inch color television sets. The program was running for an hour each day between January 12th and January 30th of 1950. That successful demonstration created an enormous amount of potential customers for the industry and huge benefits for CBS broadcasting system [18].

CBS system was approved as the U.S. color broadcasting standard on October 11 1950. It was the first all-electronic color broadcasting standard in the world. However, the CBS system was not compatible with existing black and white television sets and the number of available color receivers was extremely limited. Therefore they had limited number of viewers as a result of limited number of receivers [18].

In 1953, U.S. National Television System Committee (NTSC) presented a new color system which was compatible with existing black and white TV sets. NTSC system was accepted by Federal Communication Commission (FCC) on December 17 1953. The amount of available color programming and the number of commercially available TV sets was also going up after 1953 [18].

In 1946, only 0.5% of American household had a TV set but the number increased to 55.7% in 1954 and 90% by 1962. These days there are at least one TV set in each household. TV has been an exciting device for many people since its invention and the industry has been growing a lot. The quality of the analog TV has improved greatly since its invention; however it is not acceptable to any further extent and the analog TV era is over [20].

Television technology has changed in the digital age. Digital TV is replacing the analog TV standard in all parts from end user TV sets to transmission lines and production studios.

Industries and broadcasters all over the world have started planning for life after analog [20],[23]. The governments have started setting exact dates for discontinuing all analog TV facilities. In Germany, Berlin has shut off analog in 2003 and the rest of the Germany is scheduled to follow it by 2010. In the United States, the congress has set June 12th 2009 as the firm date for complete terrestrial analog TV shutoff [24]. In France, 2010 is set as end-of-analog date. Canadian Radio-television and Telecommunications Commission has set Aug 31 2011 as analog TV shut off date [1]. Japan is following the suit by 2011. The United Kingdom is following the analog phase out by 2012 and the same trend is followed by the rest of the world. Digital TV will be the only option for TV viewers after these shut off dates [28].

Viewing high quality programs is the first benefit of digital HDTV. Another advantage of digital TV is its interoperability with PCs. In addition, digital TV will free up some frequency spectrum which was previously used by analog TV. This vacant spectrum band will open a great opportunity for new era in data and cellular phone services.

The history of HDTV in North America and the formation of the Grand Alliance of HDTV are explained in section 2.2.

2.2 HDTV in North America

High Definition TV research and development started in Japan in 1968. The Japanese HDTV system was an analog system called multiple sub-Nyquist sampling encoding system (MUSE) developed by Japan Broadcasting Corporation Nippon Hodo Kyokai (NHK). Their system was developed using digital signal compression, analog transmission and satellite delivery to consumers [20][23][50].

MUSE system was demonstrated in US early 1987 as the first working HDTV system in the world. The demonstration was successful and initiated US HDTV research and development.

However, the MUSE system had some drawbacks. It had motion artifacts, was vulnerable to ghosting and required a large bandwidth. High bandwidth demand was an obstacle which makes it inappropriate for terrestrial broadcast television systems.

In September 1987, federal communications committee formed an advisory committee on Advance Television Service to resolve these problems. The initial plan was to secure a set of terrestrial broadcast standard for Advanced TV [2].

Advance television system developers were required to submit their proposals to Advisory Committee on Advanced Television Service (ACATS) for evaluation. FCC advisory committee was responsible to review different advanced television scenarios and proposals and advise the FCC on related issues.

An Advanced Television Test Center (ATTC) was established in Alexandria, VA to test these proposals. ATTC was established by broadcasters and Electronic Industries Association (EIA). Cable-related testing facilities were supported by cable television industry. Test and evaluation of advanced television proposals started under the supervision of FCC advisory committee in 1989.

Canadian governmental Communication Research Center established an Advanced Television Evaluation Laboratory (ATEL) in Ottawa. Subjective evaluations of picture quality and test of transmission impairment effect on picture quality were conducted in ATEL.

Advisory committee on advance television service received 23 advance television proposals by 1989. However the field was overridden in 1990 by receiving the Chicago's General Instrument Corporation (GI) proposal which was the first all-digital HDTV system.

Second all-digital proposal was submitted by Advanced Television Research Consortium (ATRC: teamwork of the National Broadcasting *Co.* New York City, N.Y., Philips Electronics North America Corp., New York City, N.Y., the David Sarnoff research

Center, Princeton, N.J., Thomson Consumer Electronics Inc., Indianapolis, Ind.)

Zenith and AT&T teamed up with each other to propose the third one. MIT proposed another all digital system in 1991 [30-32].

All HDTV proposals were tested at the ATTC and ATEL during 1991 and 1992. However the main challenging proposals were the all-digital proposals. All-digital HDTV idea changed the HDTV standard trend toward an all-digital standard and FCC publicized its preference for digital HDTV rather than Enhanced Digital TV (EDTV).

Accordingly, simulcast approach was chosen by FCC for the transition time during which both HDTV and NTSC exist. By simulcast approach, HDTV channel will transmit its signal separately so it does not depend on NTSC and does not need to be compatible with NTSC. HDTV and NTSC channels transmit their signals simultaneously for their compatible receivers.

Meanwhile, four available all-digital proposals were effectively tested at the ATTC and ATEL during 1991 and 1992. After successfully passing the tests the advisory committee decided to chose an all digital system as next TV standard and these four competitors were advised to make further improvement and perform more comprehensive tests.

These competitors were negotiating for a possible merge to improve their system. Their plan was to utilize the best features of each proposal into a unique solution which has "the best of the best" characteristics. In May 1993 AT&T, GI, MIT, Philips, Sarnoff, Thomson and Zenith announced the formation of digital HDTV Grand Alliance.

HDTV Grand Alliance prototype system was matured in 1994 and sent to ATTC and ATEL for further tests.

The prototype system video encoder is constructed by AT&T and GI. Its video decoder is built by Philips. Transport system is the result of Thomson and Sarnoff cooperation. The digital modulation system is developed by Zenith and AC-3 digital sound system is provided by Dolby Laboratories Inc of San Francisco. System integration was done in Sarnoff.

The Grand Alliance has considered several layers of interoperability for the HDTV standard including layered digital system architecture for compatibility with OSI (Open System Interconnection) data communication standard. The final proposal of the Digital HDTV Grand Alliance has been approved by Advisory Committee on Advanced Television Service and recommended to FCC in November of 1995. HDTV features are explained in section 1.3.

2.3 HDTV Features

HDTV offers outstanding image quality using MPEG2 video standard [52] and wonderful sound feature using Dolby AC-3 digital audio standard [53]. HDTV has improved TV viewing experience by benefiting from the most recent advances in digital signal processing, digital compression, telecommunications, consumer electronics and very large scale integration (VLSI).

It delivers higher resolution and digitally clean video with less image artifacts. It does not have the flicker and snow noise, ghost and chrominance artifacts and color shade errors which are common artifact in analog TV standards such as NTSC (National Television System Committee), PAL (Phase Alteration Line) or SeCAM (Sequentiel Couleur Avec Memoire).

Pictures are presented in a panoramic horizontal to vertical aspect ratio of 16:9 which is a big improvement on the 3:4 aspect ratios of analog TV standards. Motion scenes are smooth for sport fans and computer graphics specialists and details are fine enough even on very large TV screens.

2.4 HDTV Formats

Advisory Committee on Advanced Television Service has derived HDTV standard specifications from the Digital HDTV Grand Alliance proposal. These specifications, which have been used by the Alliance for constructing their prototype system, include five subsystems: scanning format and image aspect ratio, video compression, audio compression, transport protocol and transmission protocol [3], [32].

Scanning format subsystem: The standard is mainly focused on progressive scanning format but it also support interlace format for the highest resolution case as its progressive implementation may not be practical at the time. Panoramic aspect ratio of 9:16 has been adopted for all scanning formats. The scanning specifications supported by the standard are shown in Table 2-1.

Table 2-1 HDTV video formats.

Vertical lines	Horizontal pixels per line	Aspect Ratio	Progressive/ Interlaced	Frame per Second
720	1280	16:9	Progressive	24fps,30fps, 60fps
1080	1920	16:9	Progressive	24fps,30fps
1080	1920	16:9	Interlace	60fps

Video compression subsystem: MPEG-2 parameters including “B-frames” have been adopted for video compression sub-system [52].

Audio compression subsystem: Standard 5.1 channel Dolby Surround Adaptive Transform Coder 3 (AC-3) Digital technique has been employed for audio compression subsystem [53].

Transport protocol subsystem: Packet data transport protocol with MPEG-2 transport layer features has been adopted for data transport protocol.

Transmission subsystem: 8-level Vestigial Sideband Modulation (8-VSB) has been adopted for terrestrial broadcasting. 16-level Vestigial Sideband Modulation (16-VSB) and Quadrature Amplitude Modulation (QAM) are chosen for digital cable TV transmission [54].

Although progressive format has been chosen by the Advisory Committee in most HDTV scanning formats but interlace scanning format is still remaining an important video compression technique. There is no evidence in Advisory Committee reports that either of the scanning formats may be dropped in early future. Therefore the problem of converting interlaced and progressive video formats to each other using efficient techniques remains an ongoing practice.

Mathematical representation of video sequences and definition of interlacing and deinterlacing are explained in Chapter 3. Theoretical complications of converting progressive video format to Interlaced video interlaced video format to progressive video are also explained in Chapter 3.

Chapter 3

Video Scanning Formats

There are two types of video scanning formats, interlaced and progressive. Interlaced scanning format is used by most of the analog TV standards for video transmission and display. Progressive scanning format is used in PC monitors and HDTV.

This chapter briefly reviews the mathematical basis of video format conversion. For the purpose of this review, the mathematical representation of a video sequence is explained in section 3.1. Mathematical explanations of interlaced and progressive scanning formats are defined in section 3.2. The theoretical problem of interlaced to progressive and progressive to interlaced conversion is explained in section 3.3.

3.1 Digital Video Sequence

A video sequence is a series of pictures of the same size demonstrated at rigid time intervals. Each picture in a video sequence is called a video frame.

A video frame could be represented by a two dimensional matrix where each element of the matrix, represents the intensity value of the corresponding pixel. Consequently a video sequence could be represented by a three dimensional matrix in the horizontal, vertical and temporal dimensions. Intensity value of a pixel, at horizontal position x, vertical position y and time n, could be represented by $F(x,y,\ n)$ in this three dimensional matrix. A sample video sequence is shown in Figure 3-1, where y component corresponds to the line number in a single frame. y starts from 0 all throughout this thesis.

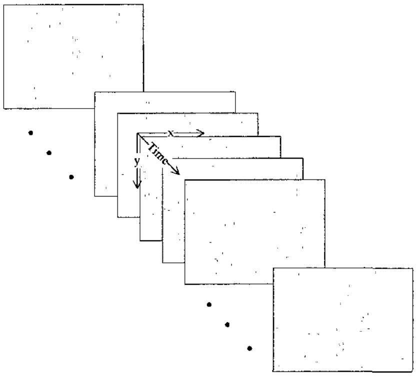
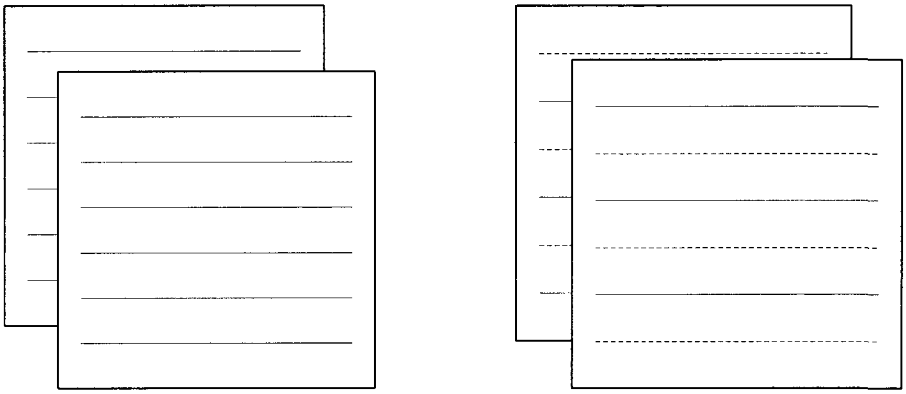


Figure 3-1 A sequence of video frames.

3.2 Interlaced and Progressive Scanning Formats

A video scan format refers to the method which is used for scanning the pixels of that video for demonstrations. There are two common scanning formats which are called *progressive scanning format* and *interlaced scanning format.*

In progressive scanning format, all lines of video frame are scanned in each frame. Therefore, in a progressively scanned video $F(x, y,\ n)$ is defined for all integer values of x, y, n within valid width, height and duration of video sequence. Two consecutive video frame of a progressive video are shown in Figure 3-2 (a).



(a) Progressive sequence

(b) Interlaced sequence

Figure 3-2 Different scanning formats for a video sequence. (a) In a progressive sequence both even and odd lines are scanned in each frame. (b) In an interlaced sequence either even or odd lines are scanned in each field.

In interlaced scanning format, only even or odd lines of the corresponding video frame are scanned each time and each of them is called an interlaced video *field*. Accordingly, the term, *field* is used to represent an instance of an interlaced video field at a time, while the term *frame* describes each instance of a progressively scanned video at a time. Figure 3-2 (b) shows two consecutive fields of an interlaced video sequence. As a standard convention, the fields with existing even lines are called *even fields* and the fields with existing odd lines are called *odd fields*.

Considering $F_p(x, y, n)$ as an original progressive video format, interlaced video sequence generated form $F_p(x, y, n)$ could be represented by $F_i(x, y, n)$ in Equation (3.1).

$$\boldsymbol{F_i(x,y,n)} = \begin{cases} \boldsymbol{F_p(x,y,n)} & \textbf{where } \boldsymbol{mod(n,2) = mod(y,2)} \\ \boldsymbol{\emptyset} & \textbf{otherwise} \end{cases} \qquad (3.1)$$

mod operator, in Equation (3.1), is the modulus operator. Modulus operator is defined in Equation (3.2).

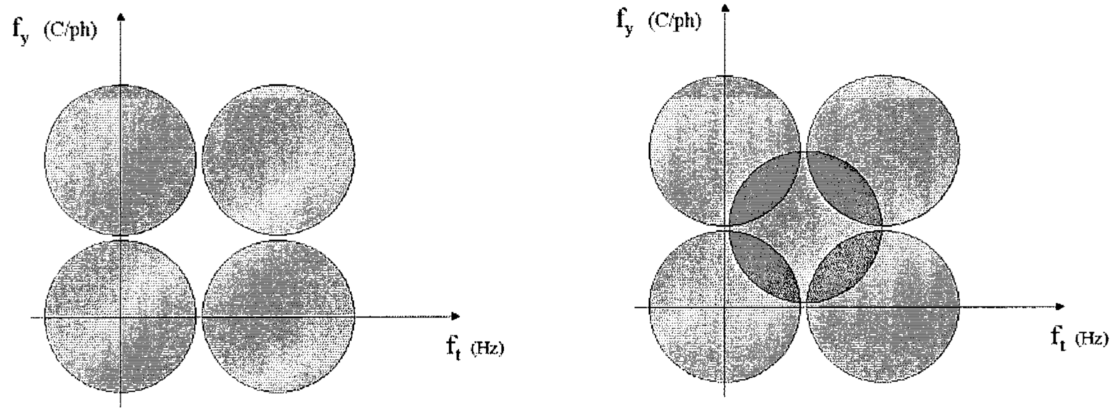
$$\boldsymbol{mod(n,2)} = \begin{cases} \mathbf{0} & \boldsymbol{n} \textbf{ even} \\ \mathbf{1} & \boldsymbol{n} \textbf{ odd} \end{cases} \qquad (3.2)$$

The null symbol, $\emptyset$, in Equation (3.1) represents missing pixels in interlaced scanned video. $F_i(x,y,n)$ represents the intensity value of a pixel in interlaced video sequence; in spatial location (x,y) and temporal location n. While $F_p(x,y,n)$ represents the intensity value of a pixel in the same location in the progressive video sequence [5].

3.3 Video Interlacing

Converting a progressive video sequence to an interlaced video sequence is called interlacing. Interlacing is removing either even or odd lines from the corresponding progressive video frame; shown by Equation (3.1). Hence interlacing is a vertical down-sampling which reduces signal bandwidth by half.

Figure 3-3 shows the spectrum of a progressive and an interlaced video sequence; the interlaced video is generated from the same progressive video. As shown in this figure, the down-sampling process in interlacing method does not maintain Nyquist sampling rate and causes aliasing effect. Nevertheless, interlacing is used as a major compression technique in analog TV standards benefiting from the fact that “The human visual system is less sensitive to flickering details than to large-area flicker.” [4]



(a) Progressive video

(b) Interlaced video spectrum

Figure 3-3 vertical Temporal spectrum of video sequences. (a) Vertical-temporal spectrum of a typical progressive video sequence. (b) Vertical-temporal spectrum of interlaced video generated from the same progressive video sequence [5].

3.4 Video Deinterlacing

Deinterlacing a video sequence is the process of converting an interlaced video to a progressive video. Therefore, deinterlacing converts each interlaced video field to a progressive frame. Ideally this should be done by a vertical upsampling which doubles the vertical sampling rates.

However Figure 3-3 shows that the original down-sampling does not fulfill the demands of the sampling theorem and includes aliasing. As a result of existing aliasing effect, perfect deinterlacing is not theoretically possible and deinterlacing methods are estimations of real pixel values [5].

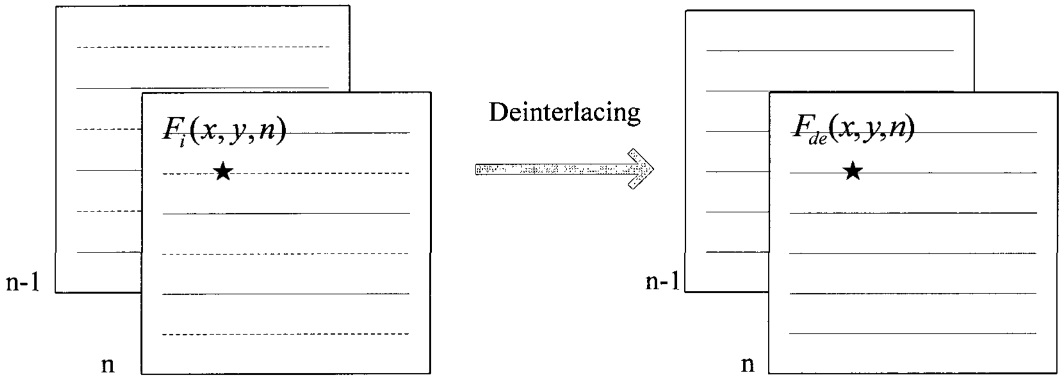


Figure 3-4 Deinterlacing.

Figure 3-4 shows the deinterlacing process. The output of a deinterlacing algorithm is shown by $F_{de}(x,y,\ n)$ in this thesis and can be defined by Equation (3.3)

$$\boldsymbol{F_{de}(x,y,n)} = \begin{cases} \boldsymbol{F_i(x,y,n)} & \textbf{where } \boldsymbol{mod(n,2) = mod(y,2)} \\ \boldsymbol{\hat{F}_i(x,y,n)} & \textbf{otherwise} \end{cases} \qquad (3.3)$$

Where $F_i(x,y,\ n)$ are the intensity values of the pixels in the input interlaced sequence and $\hat{F}_i(x,y,\ n)$ are the estimations of missing values calculated by the deinterlacing algorithm.

As Equation (3.3) shows, existing lines in interlaced input fields are exactly delivered to output deinterlaced frames and missing lines are estimated using existing lines of interlaced input. The best deinterlacing algorithms are therefore, the algorithms with minimum estimation errors for a given situations. Chapter 4 introduces the most well-known deinterlacing algorithms available in literature and review their advantages and disadvantages.

Chapter 4

Deinterlacing Algorithms

Converting interlaced video format to progressive video format has got more attention in the past few years. Recent advances in electronics and computational power of the processors have helped to the development of accurate deinterlacing methods.

A wide variety of deinterlacing methods is available in the literature featuring different applications. Although these methods are varying in complexity and performance, yet they can be divided into two main categories, namely intra-frame and inter-frame methods. Intra-frame methods use only current field of video data to interpolate the missing lines. Whereas inter-frame methods use more than one field of video to interpolate the missing lines.

Therefore, intra-frame methods are spatial methods in essence while inter-frame methods can be temporal or spatio-temporal. Spatio-temporal methods, furthermore called hybrid methods, make the most sophisticated and accurate deinterlacing methods by utilizing both temporal and spatial data for interpolating missing lines of interlaced video.

This chapter explores a number of well-known deinterlacing methods and investigates their performance when applied to test video sequences. Section 4.1 looks at intra-frame methods. Section 4.2 investigates inter-frame methods. Motion adaptive methods explored in this section are investigated in more details in Chapter 5.

Deinterlacing methods investigated in this chapter are not a complete investigation of all available methods. However, their implementation provides a fair comparison ground for evaluating the deinterlacing method proposed in this thesis.

4.1 Intra-frame Methods

Deinterlacing algorithms which use only current field of video for their interpolation are called intra-frame methods. Intra-frame deinterlacing methods are mathematically defined by Equation (4.1) and a choice of a spatial domain filter.

$$\boldsymbol{F_{de}(\vec{X},n)} = \begin{cases} \boldsymbol{F_i(\vec{X},n)} & \textbf{where } \boldsymbol{mod(n,2) = mod(y,2)} \\ \sum_{\vec{k}} \boldsymbol{F_i\,(\vec{X} - \vec{k},n)h(\vec{k},n)} & \textbf{otherwise} \end{cases} \tag{4.1}$$

In Equation (4.1):

$\vec{X} = (x, y)$ represents the location of a pixel in the field.

$\vec{K} = (k_1, k_2)$ is a spatial displacement vector.

$h(\vec{k},n)$ is the impulse response of the deinterlacing filter, sections 4.1.1 and 4.1.2 provide two examples of such filters. Different choices of $h(\vec{k},n)$ may change the deinterlacing algorithm performance and complexity. For an intra-frame deinterlacing $h(\vec{k},n)$ is independent to n therefore it should be an all pass filter in time domain [5], [55].

As a result intra-frame methods do not need additional frame storage as they are inherently spatial methods. Since the memory required for storing extra frames was not cheap until recently, most primitive deinterlacing methods are intra-frame methods. Line repetition, linear averaging and parametric image modeling are few intra-frame methods, which are explained in sections 4.1.1, 4.1.2, and 4.1.3.

4.1.1 Line Repetition

Line repetition is one of the first and simplest available deinterlacing algorithms, where the missing lines are generated by repeating the lines exactly below or above them in the following manner. In an even field, where odd lines are missing, each missing line is generated by copying the line exactly above it. In an odd field, where even lines are missing, each missing line is generated by copying the line exactly below it.

In mathematical explanation using Equation (4.1), line repetition could be implemented by choosing a spatial deinterlacing filter whose impulse response is expressed in Equation (4.2.a) and frequency response is given by Equation (4.2.b).

$$h(\vec{k},n) = \begin{cases} 1 & (mod(n,2) = 0 \,\&\, k_2 = 1) \\ 1 & (mod(n,2) = 1 \,\&\, k_2 = -1) \\ 0 & \text{otherwise} \end{cases} \tag{4.2.a}$$

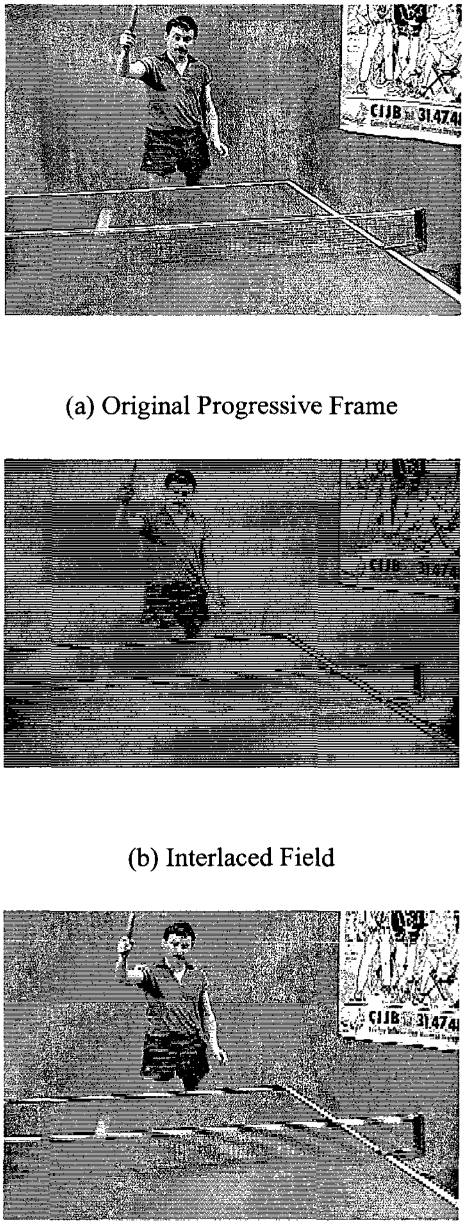
$$H(f_x, f_y, f_t) = |\cos \pi f_y| \tag{4.2.b}$$

Therefore, line repetition algorithm could be represented by Equation (4.3) for an even field and Equation (4.4) for an odd field.

$$F_{de}(x,y,n) = \begin{cases} F_i(x,y,n) & mod(y,2) = 0 \\ F_i(x,y-1,n) & \text{otherwise} \end{cases} \tag{4.3}$$

$$F_{de}(x,y,n) = \begin{cases} F_i(x,y,n) & mod(y,2) = 1 \\ F_i(x,y+1,n) & \text{otherwise} \end{cases} \tag{4.4}$$

Figure 4-1 shows a sample simulation results for line repletion algorithm. It shows an overall poor performance. The main problem with a line repetition is vertical detail distortion because of deinterlacing filter impulse response. This distortion may possibly be unnoticeable in some cases but cause annoying results in high quality videos.



(a) Original Progressive Frame

(b) Interlaced Field

(c) Deinterlaced Frame

Figure 4-1 Simulation results using Line Repetition deinterlacing method. (a) original progressive frame, (b) Interlaced field, (c) Line repetition deinterlacing output. Original image resolution is $\mathbf{240 \times 352}$.

4.1.2 Linear interpolation deinterlacing

Linear interpolation is a little more advanced inter-frame deinterlacing algorithm. In linear interpolation method, missing lines are estimated by finding the average value of the lines directly above and below them.

In mathematical explanation using Equation (4.1), linear interpolation method could be implemented by choosing a spatial deinterlacing filter whose impulse response is expressed in Equation (4.5.a). The frequency response of this filter could be expressed by Equation (4.5.b)

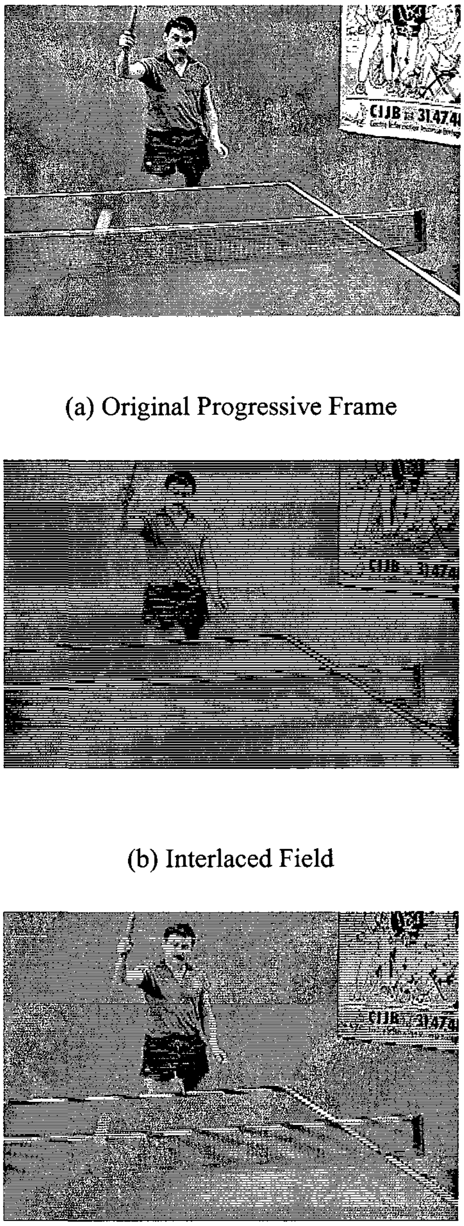
$$h(\vec{k},n) = \begin{cases} 1/2 & (k_2 = 1) \\ 1/2 & (k_2 = -1) \\ 0 & \text{otherwise} \end{cases} \tag{4.5.a}$$

$$H(f_x, f_y, f_t) = (1 + \cos 2\pi f_y)/2 \tag{4.5.b}$$

Therefore, linear interpolation algorithm is expressed by Equations (4.6).

$$F_{de}(x,y,n) = \begin{cases} F_i(x,y,n) & \text{where } mod(y,2) = mod(y,2) \\ (F_i(x,y+1,n) + F_i(x,y-1,n))/2 & \text{otherwise} \end{cases} \tag{4.6}$$

Figure 4-2 shows a sample simulation results for linear interpolation algorithm. It shows a poor performance however, its performance is slightly better than line repetition algorithm. These results are smoother than line repetition but shares aliasing and jitter effect with line repetition method. The problem is mainly cause by deinterlacing filter impulse response.



(a) Original Progressive Frame

(b) Interlaced Field

(c) Deinterlaced Frame

Figure 4-2 Simulation results using Linear Interpolation deinterlacing method. (a) original progressive frame, (b) Interlaced field, (c) Linear interpolation deinterlacing output. Original image resolution is 240 × 352.

4.1.3 Non-linear spatial interpolation

More sophisticated intra-frame deinterlacing methods may utilize advanced spatial filters, to improve their performance. Modeling a small region of an image to find the maximum correlation and adapting the spatial filter coefficients to those results is one approach.

Another approach might utilize a set of parameters and basis equations to model a small region of image seeking a contour of a particular shape in the image. Missing lines could be interpolated along that contour in search of minimum interpolation error.

Although some of these advanced intra-frame methods may improve the performance, their quality could never be high enough as they are ignoring a great amount of correlation that typically exists in the video stream.

Main advantage of intra-frame methods is their motion robustness. Intra-frame methods results are independent of both motion activities in the video data and video recording rate. This characteristic is the result of using a single field of video data for deinterlacing which makes them a proper choice for highly dynamic videos.

Another important advantage of intraframe methods is their algorithm simplicity, they do not need any extra frame storage and normally they are linear spatial filters. However, despite these attractive advantages, their performance is not optimal and they could not reach to high quality deinterlaced videos. Inter-frame methods, explained at section 4.2, seek to improve their performance by using temporal data for deinterlacing.

4.2 Inter-frame deinterlacing methods

Inter-frame deinterlacing methods employ more than one frame of video sequence for estimating the missing lines of interlaced video. By using the information from the previous and/or subsequent frames, an inter-frame method seeks useful correlation between adjacent frames to reduce estimation error. Considering one perfectly static scene in a video sequence, is an example of a case that interframe deinterlacing could obtain perfect estimation values by combining even and odd fields.

Inter-frame deinterlacing methods are mathematically defined by Equation (4.7) and a choice of a temporal or tempo-spatial filter.

$$\boldsymbol{F_{de}(\vec{X},n)} = \begin{cases} \boldsymbol{F_i(\vec{X},n)} & \textbf{where } \boldsymbol{mod(n,2) = mod(y,2)} \\ \boldsymbol{\sum_{\vec{k},m} F_i(\vec{X}-\vec{k},n+m)h(\vec{k},m)} & \textbf{otherwise} \end{cases} \tag{4.7}$$

In Equation (4.7):

$\vec{X} = (x,y)$ represents the location of a pixel in the field.

$\vec{K} = (k_1, k_2)$ is a spatial displacement vector.

m is a temporal delay element representing the effect of previous and/or subsequent fields.

$h(\vec{k},m)$ is the impulse response of the deinterlacing filter. Different choices of $h(\vec{k},m)$

may change the deinterlacing algorithm performance and complexity sections 4.2.1, 4.2.2, 4.2.3 provides sample choices of this filter [5], [55].

Inter-frame methods are using tempo-spatial filters to improve their performance. Field insertion 4.2.1 is the simplest inter-frame deinterlacing method, which could obtain perfect deinterlacing results in static scenes. Bilinear field interpolation and vertical temporal median filtering are two other interframe methods which are explained in sections 4.2.2 , 4.2.3.

However the most advanced interframe deinterlacing methods are motion compensated deinterlacing methods, explained in section 4.2.4, and motion adaptive deinterlacing methods explained in section 4.2.5. Each of motion compensated deinterlacing and motion adaptive deinterlacing methods are a category of methods which are the main focus in advanced deinterlacing methods.

4.2.1 Field Insertion

Field insertion is the simplest inter-frame deinterlacing method. In field insertion method, the values of missing lines are generated by copying the lines from the same vertical position in the previous filed. Therefore, field insertion will provide perfect estimation in stationary scenes of a video sequence [6].

Accordingly, mathematically definition of the appropriate temporal filter needed for implementation of field insertion method, that satisfies Equation (4.7), is represented by Equation (4.8); where (4.8.a) is the impulse response of the filter and (4.8.b) is its frequency response.

$$h(\vec{k},n) = \begin{cases} 1 & (k_1 = k_2 = 0, m = -1) \\ 0 & \text{otherwise} \end{cases} \qquad (4.8.a)$$

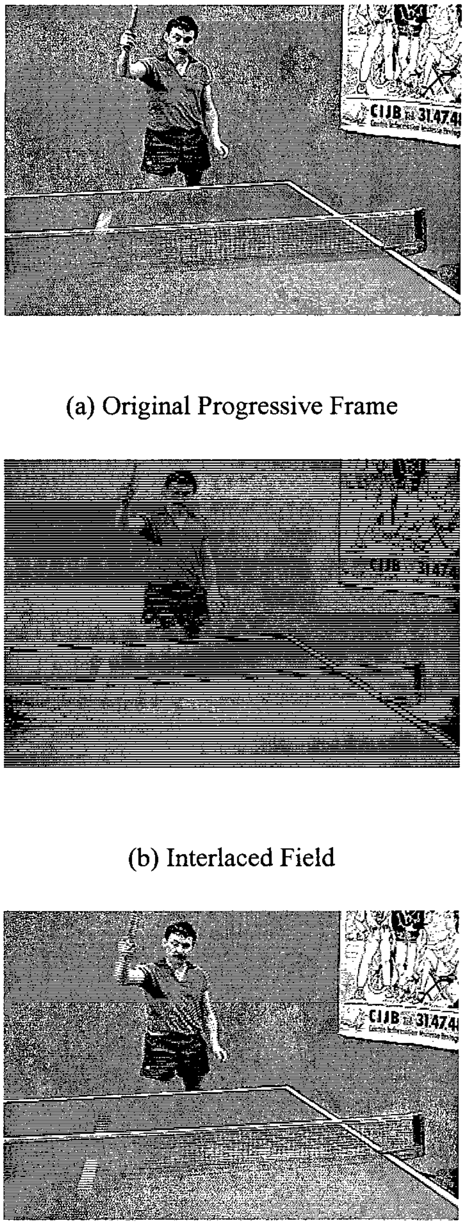
$$H(f_x, f_y, f_t) = |\cos \pi f_t| \qquad (4.8.b)$$

Therefore, field insertion method is represented by Equations (4.9).

$$F_{de}(x,y,n) = \begin{cases} F_i(x,y,n) & \text{where } mod(y,2) = mod(y,2) \\ F_i(x,y,n-1) & \text{otherwise} \end{cases} \qquad (4.9)$$

Field insertion method would obtain perfect deinterlacing results in static parts of a video sequence. Conversely, it may generate severe distortion in dynamic parts of a video sequence. Figure 4-3 shows field insertion results for a sample frame. It shows perfect deinterlaced background but blurred and jagged edges in moving parts.

In high frame rates, where temporal correlation between adjacent fields may go high, field insertion may obtain satisfactory results with minor blurring effect. However, its results are not satisfactory for high quality videos.



(a) Original Progressive Frame

(b) Interlaced Field

(c) Deinterlaced Frame

Figure 4-3 Simulation results using Field Insertion deinterlacing method. (a) original progressive frame, (b) Interlaced field, (c) Field insertion deinterlacing output. Original image resolution is $\mathbf{240 \times 352}$.

4.2.2 Bilinear Field Interpolation

Bilinear field interpolation method utilizes two field of a video sequence for estimating the missing lines value. One of the two fields is the field exactly before the current field and the other one is the field exactly after the current field. The value of each missing pixel is generated by finding the average values of the pixels with same spatial position in previous and subsequent fields.

Accordingly, mathematically definition of the appropriate temporal filter needed for implementation of bilinear field interpolation, that satisfies Equation (4.7), is represented by Equation (4.10); (4.10.a) shows the impulse response and (4.10.b) shows frequency response.

$$h(\vec{k},n) = \begin{cases} 1/2 & (k_1 = k_2 = 0, m = -1) \\ 1/2 & (k_1 = k_2 = 0, m = 1) \\ 0 & \text{otherwise} \end{cases} \tag{4.10.a}$$

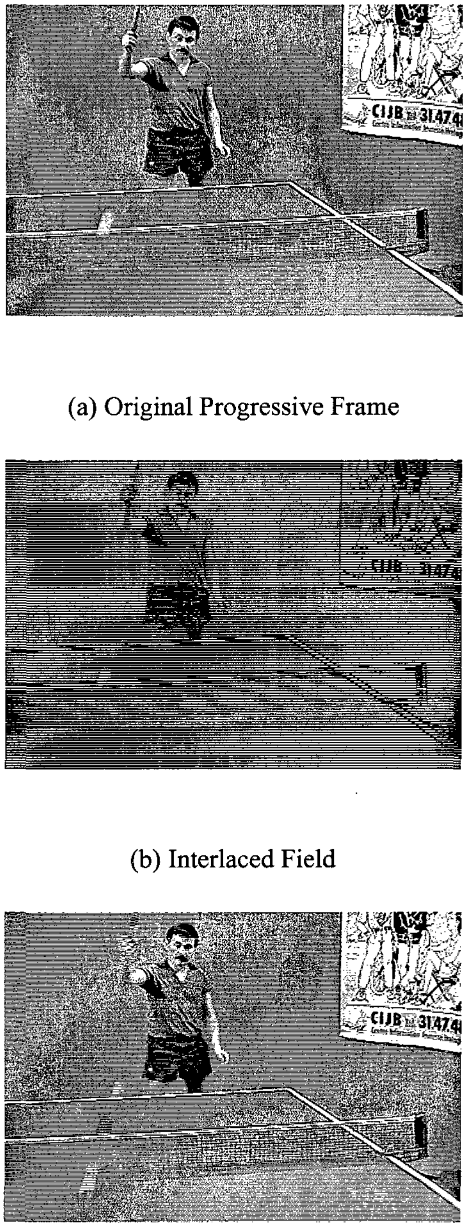
$$H(f_x, f_y, f_t) = (1 + \cos 2\pi f_t)/2 \tag{4.10.b}$$

Therefore, bilinear field interpolation method is represented by Equation (4.11).

$$F_{de}(x,y,n) = \begin{cases} F_i(x,y,n) & \text{where } mod(y,2) = mod(y,2) \\ \dfrac{\big(F_i(x,y,n+1) + F_i(x,y,n-1)\big)}{2} & \text{otherwise} \end{cases} \tag{4.11}$$

Bilinear field interpolation works similar to field insertion method and share same issues with field insertion. Its results are reasonable in static sections of video and unacceptable in moving parts.

Nevertheless, because of combining two fields in interpolation, bilinear field interpolation results may not be as perfect as field insertion results. However, it may reduce blurring effect, in dynamic parts, owing to its inherent temporal low pass filter. Figure 4-4 shows sample result of bilinear field interpolation method. It proves that blurring effect is serious in moving parts of video and the method is not suitable for high quality deinterlacing applications.



(a) Original Progressive Frame

(b) Interlaced Field

(c) Deinterlaced Frame

Figure 4-4 Simulation results using Bilinear Field Interpolation deinterlacing method. (a) original progressive frame, (b) Interlaced field, (c) Bilinear Field Interpolation deinterlacing output. Original image resolution is 240×352.

4.2.3 Vertical Temporal Median Filtering

As shown in sections 4.2.1 and 4.2.2, a merely temporal deinterlacing method may have advantages over purely spatial deinterlacing methods in special cases but neither of them could obtain satisfactory results.

This effect is because of ignoring a great amount of correlation that typically exists in a video sequence, mutually in temporal and spatial domains.

A more advanced category of inter-frame deinterlacing is the tempo-spatial category. A tempo-spatial inter-frame deinterlacing method tries to improve its performance over intra-frame methods by using temporal correlation available between adjacent frames, and over purely temporal deinterlacing methods by using spatial correlation available between adjacent pixels of the current frame.

One of the simplest tempo-spatial interframe methods is Vertical-Temporal (VT) median filtering. It uses both temporal and spatial data to interpolate missing lines of video data. In vertical-temporal median filtering method, a median operation is used to interpolate the missing lines of interlaced video.

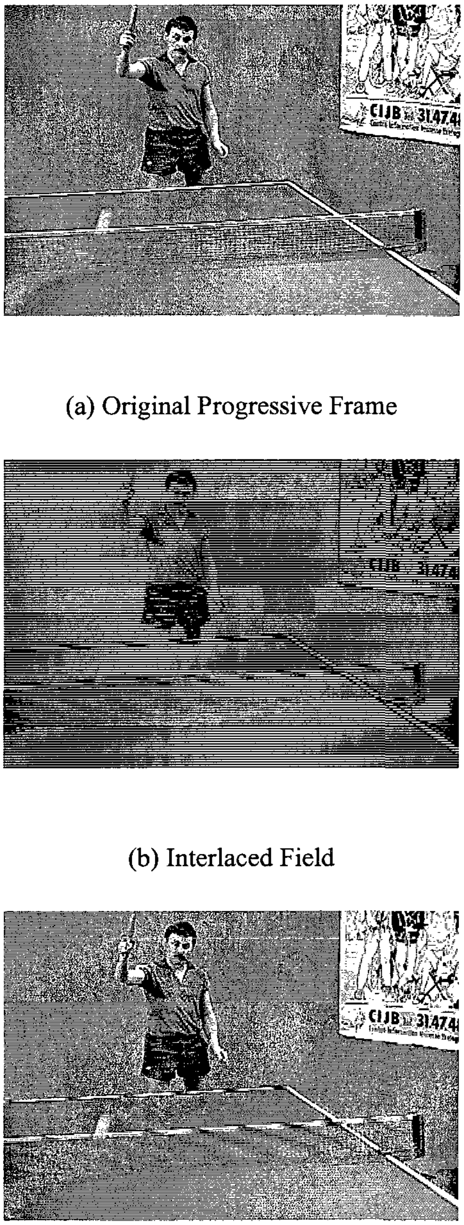
Median operation is easy to implement and possess a good performance. Therefore, vertical-temporal median filtering has become a popular deinterlacing method. A number of vertical-temporal median methods are suggested in literature which have different tap number and different weigh factors. The simplest of which is 3-tap median filtering.

As vertical-temporal median filtering, is not a linear operation it could not be represented in the form of Equation (4.7). Mathematical representation of a 3-tap median filtering method is described by Equation (4.12).

$$\boldsymbol{F_{de}(x,y,n)} \tag{4.12}$$

$$= \begin{cases} \boldsymbol{F_i(x,y,n)} & \textbf{where } \boldsymbol{mod(y,2) = mod(y,2)} \\ \boldsymbol{median(F_i(x,y+1,n), F_i(x,y-1,n), F_i(x,y,n-1))} & \textbf{otherwise} \end{cases}$$

Figure 4-5 shows sample result of vertical-temporal median filtering method using a 3-tap median filter. Vertical-temporal median operation works on a pixel by pixel basis and acquires improved deinterlacing results.



(a) Original Progressive Frame

(b) Interlaced Field

(c) Deinterlaced Frame

Figure 4-5 Simulation results using Vertical Temporal Median Filtering deinterlacing method. (a) original progressive frame, (b) Interlaced field, (c) Vertical Temporal Median Filtering deinterlacing output. Original image resolution is 240 × 352.

4.2.4 Motion Compensated Deinterlacing

Motion compensated deinterlacing is one of the most advanced deinterlacing methods. In motion compensation, motion data is virtually removed from the video data in order to take advantage from the usage of the group of temporal deinterlacing methods which perform very well on stationary data [5], [7-12], [55].

By removing motion data, a moving sequence will be virtually converted to a stationary sequence. Consequently, any deinterlacing method that performs better in static areas could benefit from motion compensation to achieve high performance deinterlacing results. Motion data will be subsequently added to the deinterlaced video.

Motion compensation is done using motion estimation methods. There are a wide variety of available motion estimation methods such as the methods explained in [11], [69-70] and [76]. A success factor of a motion compensated deinterlacing is the use of an accurate motion compensation method which could remove the motion from the video sequence perfectly. However it is theoretically impossible to remove motion data completely as motion estimation methods are offering only an estimate of the real motion. Therefore, a good motion compensated deinterlacing method should be robust to motion estimation error.

Block matching algorithms [11] are the most common motion estimation methods. In a block matching algorithm each image is segmented into a number of blocks with a specified size; 8x8 blocks are the most common ones. Then each block in the current field is compared with the same- size blocks of a given search area in the previous and/or subsequent fields. A block in the search area, which most closely matches the original block, is consequently chosen as the matching block. Motion vector will be equal to the displacement vector between the original and matching blocks.

Mean square error and mean absolute error are the typically used matching criterion. However different motion estimation algorithms may use different matching criteria, different search regions or different search methods.

In any case, motion compensation methods always suffer from a degree of inaccuracy, which is inherent to motion estimation algorithms. For instance, if an object moves beyond the search region, motion estimation algorithm will be unable to find the real motion vector. Another instance could be while the object movements in the video include appearing and disappearing circumstances. In both these cases motion estimation algorithm finds the closest match but not the real one. Another instance, which is particularly important in deinterlacing, is while pixels show sub-pixel motions. In this instance, motion estimation results should point between the pixels or at the missing lines, which is not applicable to the case of interlaced video [55].

Consequently, motion compensation can be used to compensate translational moving of solid objects, which their shape is constant during their motion. Therefore motion compensated deinterlacing can be very effective in these cases, for instance a video obtained from a panning camera.

Some non-motion compensated deinterlacing methods which, may be converted to effective motion compensated deinterlacing methods are field insertion deinterlacing and bilinear field interpolation deinterlacing [5], [55]. Motion compensated field insertion deinterlacing and motion compensated bilinear field interpolation deinterlacing methods using block based motion estimation are implemented in this thesis for comparison purpose. Their simulation results are given in Chapter 8.

4.2.5 Motion Adaptive Deinterlacing

As shown in previous sections, an interlaced video sequence could be perfectly reconstructed using an interframe deinterlacing method in the absence of motion; while an intraframe method performs better in the presence of motion. Therefore if a video sequence could be divided to moving and static regions it may benefit from a intraframe method in moving parts and a interframe method in static parts.

This idea is the basic concept behind motion adaptive deinterlacing. In a motion adaptive deinterlacing method a motion detector is used to detect motion in the video sequence. The video sequence is consequently divided to dynamic and static sections. Accordingly, an interframe method will be applied to static parts while an intraframe method reconstructs dynamic parts. These results will be added to each other using different combination methods which may decide on their combination factor based on the amount of motion or other combination criteria.

Groundwork literature survey indicated that motion adaptive deinterlacing have the best potential for a high quality deinterlacing method. Therefore they were studied more closely in this research. References [13-16], [55] and [67] propose a number of motion adaptive deinterlacing methods. We have also proposed a motion adaptive deinterlacing method in this thesis. As all motion adaptive methods use some motion detection algorithms, Chapter 5 examines a number of motion detection methods in detail.

Chapter 5

Motion Detection Algorithms

5.1 Introduction

The main objective of a motion detection algorithm is to monitor the presence of motion in a sequence of images.

A motion detection algorithm may only detect the presence or absence of motion in the video sequence; however some motion detection algorithm may measure the motion activity level of video sequence as well.

By detecting the presence of motion in a section of video sequence a motion detection algorithm can split a video sequence to moving and stationary regions. These results are used in motion adaptive deinterlacing approach to improve deinterlacing performance. References [76], [80], [83-84] provides a number of motion detection algorithms.

Motion detection in progressive video sequence is briefly explained in section 5.2. Motion detection in interlaced video sequence and their special challenges are explained in section 5.3.

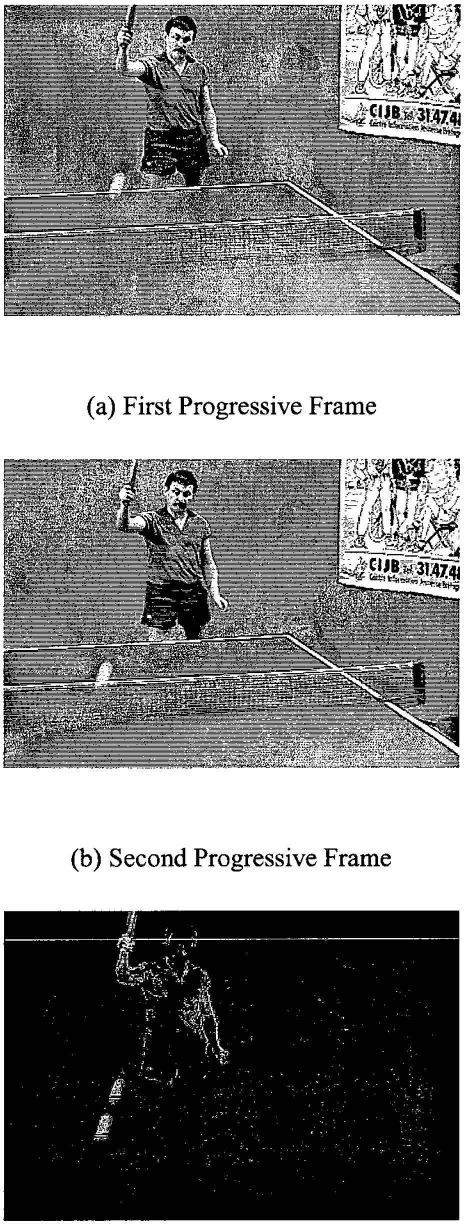
5.2 Motion Detection in Progressive Video

Detecting the presence of motion in a progressive video sequence is a straightforward process. Finding the absolute spatial difference between two consecutive frames of the sequence provides a primitive motion detection results which could be improved by proper pre/post processing.

A noise removal filter is one proper choice for pre processing which can improve motion detection accuracy. Solid thresholding is a post processing option to improve the reliability of difference signal and therefore accuracy of motion detection.

Figure 5-1 shows a simple motion detection results which is obtained by finding the absolute difference between pixel values of two consecutive frames. Figure 5-1(a) and Figure 5-1(b) are two consecutive frame of progressive video and Figure 5-1(c) is motion detection result. In interpretation of motion detection results, it should be considered that each dark pixel represent a static pixel while a bright pixel represent a dynamic pixel.

Table Tennis standard video has been used for implementation of all addressed motion detection algorithms in this chapter because it is assumed to be an appropriate combination of both fast and slow moving objects. However several more implementation results are presented in Chapter 6 as an evaluation tool for proposed motion detection method.



(a) First Progressive Frame

(b) Second Progressive Frame

(c) Motion Detection Result

Figure 5-1 Motion detection result for progressive video. (a) First progressive frame, (b) Second progressive frame, (c) Motion detection result.

5.3 Motion Detection in Interlaced Video

Motion detection in interlaced video is much more complicated than motion detection in progressive video. The main source of complications emerges from the fact that the position of missing lines in two consecutive fields of interlaced video are different. Therefore their existing pixel's location is not the same.

Consequently, finding spatial difference between two consecutive fields does not furnish any valuable information and some other tricks must be applied to detect motion in an interlaced video sequence. Sections 5.3.1, 5.3.2, 5.3.3, and 5.3.4 address some well-known motion detection algorithms and discuss their positive and negative points based on their theoretical approach and implementation results.

5.3.1 Two-Field Motion Detection

In a two-field motion detection algorithm, the goal is to detect the motion using only two consecutive fields of an interlaced video sequence. Figure 5-2 shows two consecutive fields of interlaced video sequence rendering the relative position of their missing lines. Figure 5-3 represents an edge view the two consecutive fields of interlaced video plotted in Figure 5-2, to provide more clear view of the relative position of missing pixels in two consecutive fields with respect to each other.

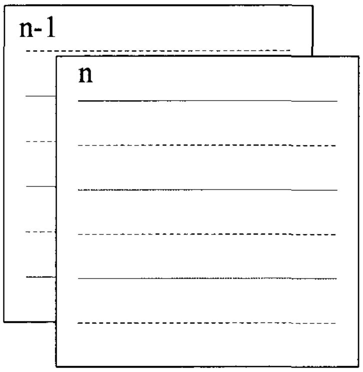


Figure 5-2 Two consecutive fields of an interlaced video. *n* represents the current field and *n-1* represents its previous field. Each solid line represents an existing line while each dashed line represents a missing line.

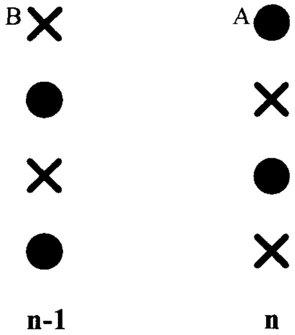


Figure 5-3 Edge view of two consecutive interlaced video fields. The right column represents the current field which is an odd field; the left column represents the previous field which is an even field in Figure 5-2. In each column, a dark circle represents an existing pixel and a cross represents a missing pixel.

As shown in Figure 5-3, an algorithm which seeks a pixel in the same spatial location as pixel *A*, in the current field, will find a missing pixel in the previous field, called *B*.

Therefore two field motion detection in interlaced video could not be a simple extension of the same approach in progressive video.

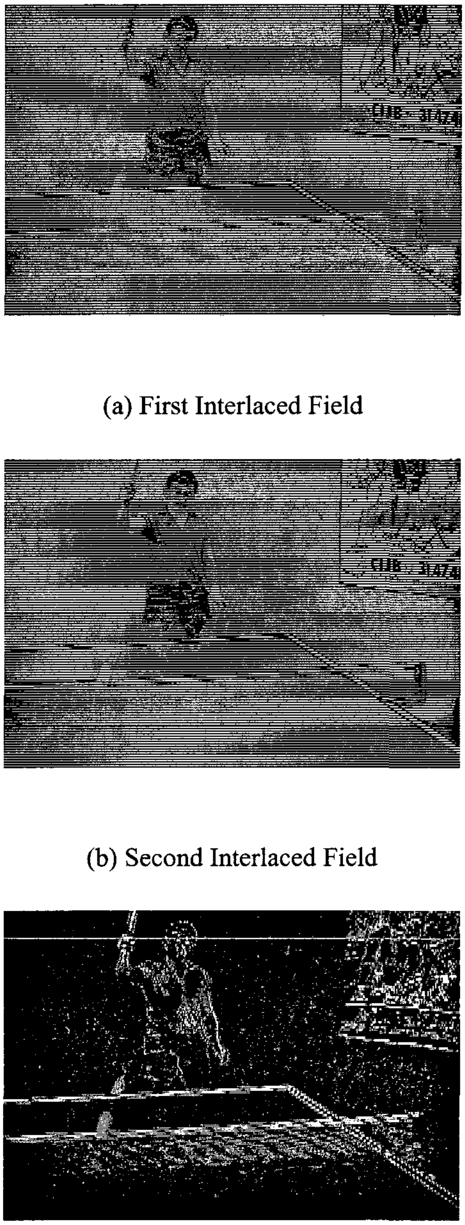
However, emulating the motion detection approach in progressive video format, two-field motion detection algorithm, interpolates the missing lines of the interlaced video field in the first stage. Subsequent to interpolation, the concept of using spatial differentiation result for motion detection will be applied to the originally interlaced video fields.

Any spatial or temporal deinterlacing method may be used in the first step of a two-field motion detection algorithm. In the second step, the absolute difference between two consecutive fields will be calculated and used as a measure of motion [5], [55], [67].

Figure 5-4 shows two consecutive fields of Table Tennis video followed by two-field motion detection results as suggested by Haan et al. [67]. Motion detection results shows motion in the background and the table edges which are not real motions.

Comparing this result by results obtained from progressive motion detection proves that some static regions are detected as dynamic by this algorithm. The problem is caused by the fact that using any spatial or temporal deinterlacing method in the first stage will introduce some error to the system which could be detected as motion.

Therefore the interpolation error created in the first stage decreases the accuracy of motion detection. For example, using a spatial deinterlacing method in the first stage may distort vertical details in the video, this distorted details will be detected as motion in the second stage leading to false motion detection results.



(a) First Interlaced Field

(b) Second Interlaced Field

(c) Motion Detection Result

Figure 5-4 Two field motion detection results. (a) First interlaced field (b) Second interlaced field (c) Motion detection result.

5.3.2 Three-Field Motion Detection

As shown in section 5.3.1, two-field motion detection does not provide acceptable results and the error created by interpolation step causes lots of false error detections. The goal in a three-field motion detection algorithm is to avoid interpolation error by using more than two fields of interlaced video.

Figure 5-5 shows three consecutive fields of interlaced video sequence rendering the relative position of their missing lines. The three interlaced video fields shown in this figure are current field and its previous and subsequent fields. Figure 5-6 presents an edge view of the three consecutive fields plotted in Figure 5-5. This figure provides more clear view of the relative position of missing pixels in three consecutive fields with respect to each other.

As shown in Figure 5-6, an algorithm which seeks a pixel in the same spatial location as pixel B, in the current field, will find missing pixels, either A or C, in the previous and subsequent fields. However, the same algorithm will reach to the valid pixels D and F if it searches for the pixels with the same spatial position of its missing pixel E. This approach is used in a three field motion detection algorithm to avoid interpolation error.

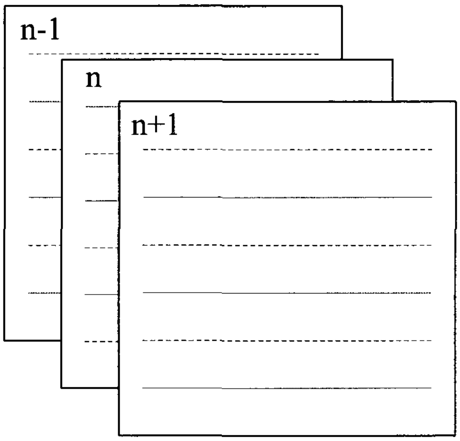


Figure 5-5 Three consecutive fields of an interlaced video. *n* represents the current field, *n-1* represents its previous field, and *n+1* represents its consequent field. Each solid line represents an existing line while each dashed line represents a missing line.

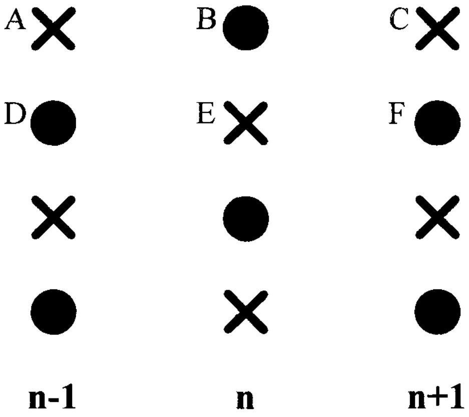


Figure 5-6 Edge view of three consecutive interlaced video fields. Column *n* represents the current field, column *n-1* represents its previous field and column *n+1* represents its subsequent field as shown in Figure 5-5 . In each column, a dark circle represents an existing pixel and a cross represents a missing pixel.

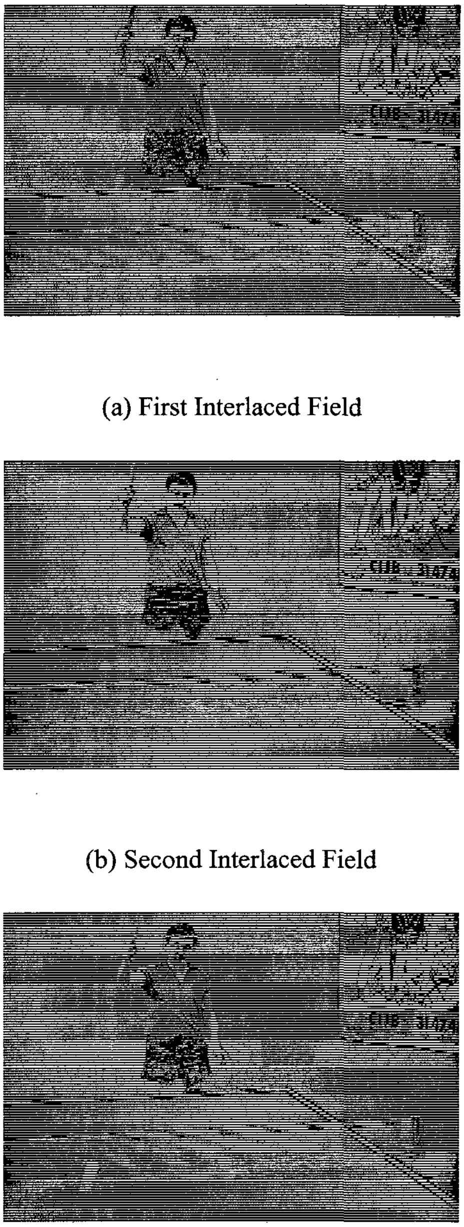
By using the absolute difference between pixels D and F as a measure of motion activity level for pixel E, a three-field motion detection algorithm avoids interpolation error. Therefore motion detection results for each field, is the absolute difference between its previous and subsequent fields.

Figure 5-7 shows three consecutive field of Table Tennis video sequence. Figure 5-8 shows motion detection results using three consecutive fields demonstrated in Figure 5-7. In this implementation, second field in Figure 5-7 is considered as current field, first field is considered as its previous field and third field is considered as its subsequent field.

The interlaced video fields shown in Figure 5-7 demonstrate a fast motion in the movement of tennis ball. The ball is in three different locations in the consecutive fields.

Three-field motion detection algorithm compares first and third fields and detects the motion between these two but it has no indication of the location of the ball in the current field. This situation is the source of an inevitable error in a three-field motion detection algorithm.

This characteristic causes lots of missed detection errors in three-field motion detection approach.



(a) First Interlaced Field

(b) Second Interlaced Field

(c) Third Interlaced Field

Figure 5-7 Three consecutive fields of Table Tennis interlaced video.

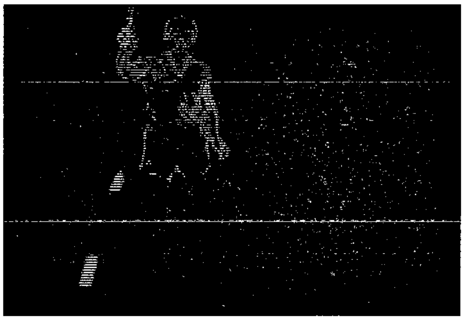


Figure 5-8 Three field motion detection implementation result.

5.3.3 Four-Field Motion Detection

Although three-field motion detection approach improves false motion detection results available in two-field motion detection but its results are not satisfactory because of their inability in catching fast motions. In order to improve motion detection performance references [5], [13], [55] propose four-field motion detection approach.

Figure 5-9 shows four consecutive fields of interlaced video sequence exposing the relative position of their missing lines. The four interlaced video fields shown in this figure are current field and its two previous fields and one subsequent field. Figure 5-10 presents an edge view of the four consecutive fields plotted in Figure 5-9 providing more clear view of the relative position of missing pixels with respect to each other.

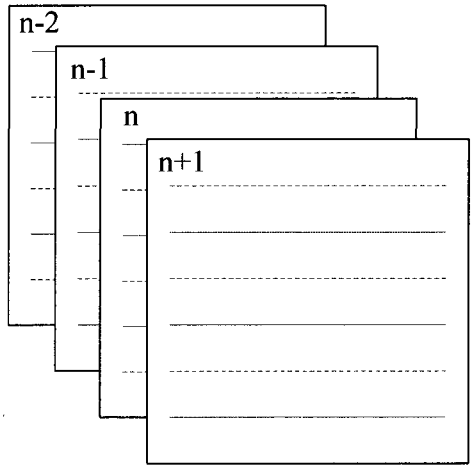


Figure 5-9 Four consecutive fields of an interlaced video. *n* represents the current field, *n-1* and *n-2* represent its previous fields, and *n+1* represents its consequent field. Each solid line represents an existing line while each dashed line represents a missing line

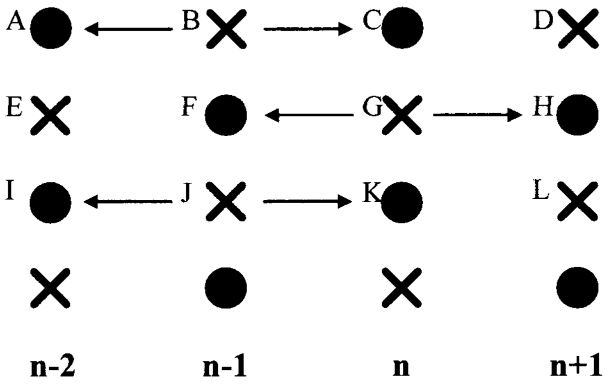


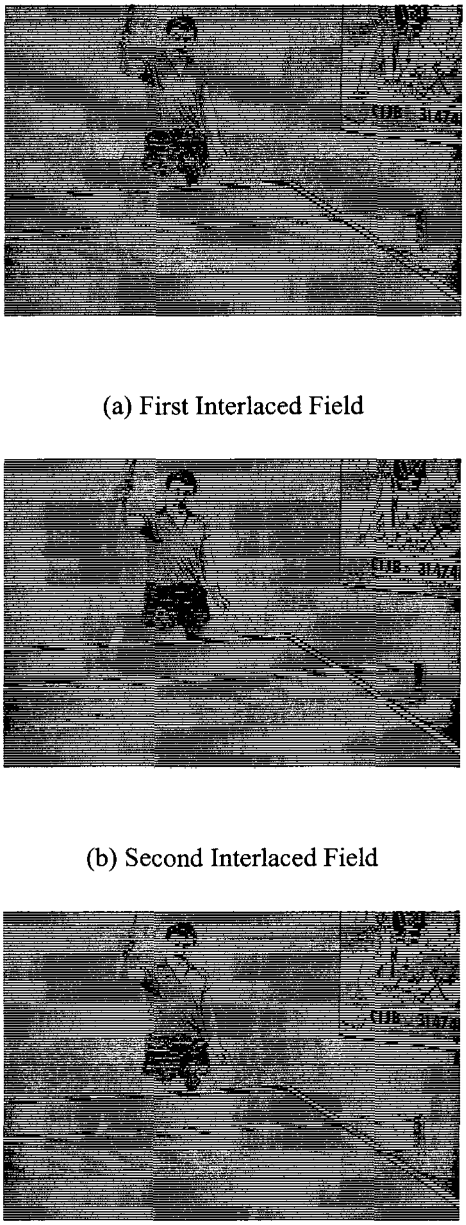
Figure 5-10 Edge view of four consecutive interlaced video fields. Column *n* represents the current field, columns *n-1* and *n-2* represent its previous fields and column *n+1* represents its subsequent field as shown in Figure 5-9. In each column, a dark circle represents an existing pixel and a cross represents a missing pixel.

In four-field motion detection proposed in [5] and [13] instead of using one difference value available in three-field motion detection, three difference values will be calculated and compared.

These difference values are calculated between corresponding pairs (A, C), (F, H) and (I, C). These corresponding pairs are pointed out in Figure 5-10 by arrows. The maximum value of these three pixel differences will be considered as motion detection results. Using three pixel differences instead of one decreases the possibility of missing fast motions available in video sequence.

Figure 5-11 shows four consecutive field of Table Tennis video Sequence followed by four-field motion detection results. In the implementation of motion detection algorithm, third interlaced field is considered as current field, first and second interlaced fields are considered as its previous fields and fourth field is considered as its subsequent field.

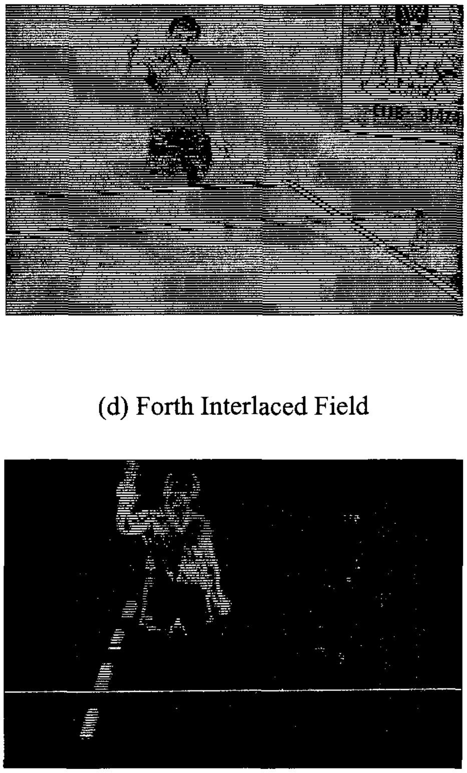
The algorithm implementation result shows that the four-field motion detection algorithm is capable of catching fast motion available in the movement of ball however it includes lots of false motion detections results such as the ones detected in ball location on other fields and player's hand.



(a) First Interlaced Field

(b) Second Interlaced Field

(c) Third Interlaced Field



(d) Forth Interlaced Field

(e) Four-field motion detection

Figure 5-11 Four-field motion detection result. (a) First interlaced field, (b) Second interlaced field, (c) Third interlaced field considered as current field, (d) Forth interlaced field, (e) Four-field motion detection results using third field as current field, first and second fields as its previous and forth field as its subsequent field.

5.3.4 Five-Field Motion Detection

Five-field motion detection approach is one step forward in improving motion detection results obtained from utilizing four consecutive field of interlaced video.

Figure 5-12 shows five consecutive fields of interlaced video sequence rendering the relative position of their missing lines. Interlaced video fields shown in this figure are current field marked as *n*, its two previous fields marked as *n-1* and *n-2* and its two subsequent fields marked as *n+1* and *n+2*. Figure 5-13 represents an edge view of the five consecutive fields plotted in Figure 5-12. It provides a more clear view of the relative position of missing pixels with respect to each other.

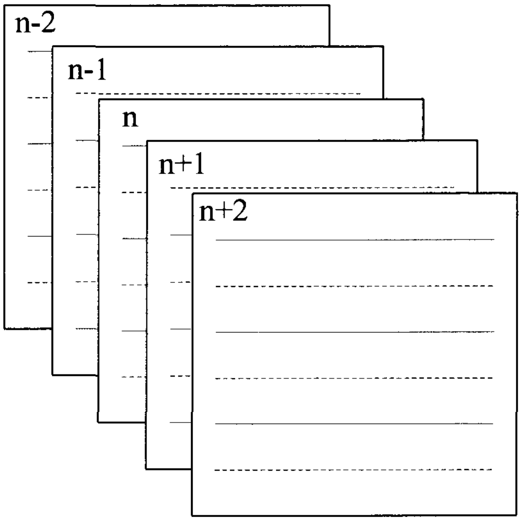


Figure 5-12 Five consecutive fields of an interlaced video. *n* represents the current field, *n-1* and *n-2* represent its previous fields, while *n+1* and *n+2* represent its consequent fields. Each solid line represents an existing line while each dashed line represents a missing line.

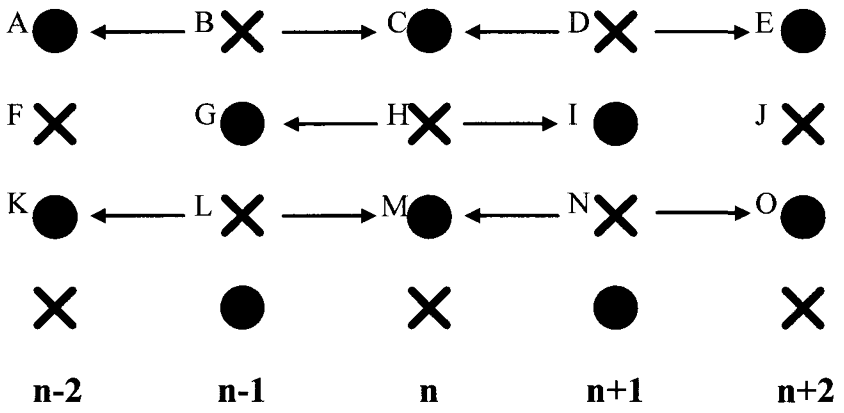


Figure 5-13 Edge view of five consecutive interlaced video fields. Column *n* represents the current field, columns *n-1* and *n-2* represent its previous fields while columns *n+1* and *n+2* represent its subsequent fields shown in Figure 5-13. In each column, a dark circle represents an existing pixel and a cross represents a missing pixel.

As shown in Figure 5-13 one major difference in five-field motion detection which may improve the performance is the possibility of looking for motion in both forward and backward directions.

Grand Alliance of HDTV has suggested a five-field motion detection algorithm [51]. In the first stage of this algorithm, the absolute difference between corresponding pixel pairs (A, C), (C, E), (G, I), (K, M), (M, O) are calculated, Equations (5-1) to (5-5) are related to this stage. The motion detection result is then calculated using Equation (5-6).

$$m_1 = |A - C| \tag{5-1}$$

$$m_2 = |C - E| \tag{5-2}$$

$$m_3 = |G - I| \tag{5-3}$$

$$m_4 = |K - M| \tag{5-4}$$

$$m_5 = |M - O| \tag{5-5}$$

$$motion_{detected} = max\left(\frac{m_1 + m_4}{2}, m_3, \frac{m_2 + m_5}{2}\right) \tag{5-6}$$

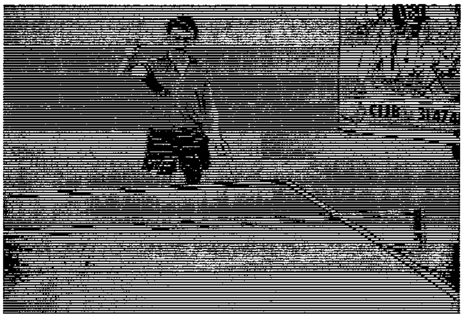
Figure 5-11 shows five consecutive field of Table Tennis video Sequence followed by five-field motion detection implementation results. In the implementation of motion detection algorithm, third interlaced field (c) is considered as current field, first (a) and second (b) interlaced fields are considered as its previous fields while fourth (d) and fifth (e) fields are considered as its subsequent fields.

(a) First Interlaced Field

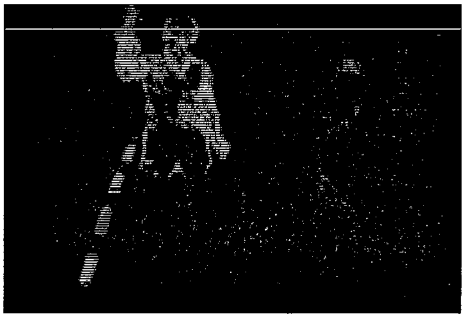
(b) Second Interlaced Field

(c) Third Interlaced Field

(d) Forth Interlaced Field



(e) Fifth Interlaced Field



(e) Five-field motion detection

Figure 5-14 Five field motion detection result. (a) First interlaced field, (b) Second interlaced field, (c) Third interlaced field (current field), (d) Forth interlaced field, (e) Fifth interlaced field, (f) GA-HDTV proposed five-field motion detection results.

Five-field motion detection result depicted in Figure 5-14 shows some improvement compared to four-field motion detection results shown in Figure 5-11. The improvements are partially the results of looking for motion in both forward and backward directions however there are some existing false detections which could deteriorate the motion adaptive deinterlacing results.

5.4 Conclusion

The overall concept of motion detection was briefly reviewed in this chapter. It followed by reviewing the principal differences between motion detection in progressive and interlaced videos.

Several well-known motion detection approaches for interlaced video were introduced in sections 5.3.1 to 5.3.4 . Each section includes sample implementation results as a performance evaluation tool.

In general it could be concluded that using odd number of fields in motion detection has advantage over using even number of them. The advantage is the result of the symmetry which exists between the positions of missing lines in an arrangement of odd number of consecutive interlaced fields.

Comparing the implementation results over different sets of video sequences, it could be concluded that the performance of five field motion detection method proposed by Grand

Alliance of HDTV has superiority over the other explored methods. However its results are not accurate enough. In this thesis we have tried to find a more accurate motion detection algorithm.

Although first improvement idea might be increasing the number of fields in motion detection, using seven or nine fields in motion detection has not proved any performance improvement. Experimental results provided in Chapter 6 proves that temporal correlation between consecutive fields generally drops after few fields and it is not reasonable to look forward or backward for more than few fields.

A new motion detection algorithm is introduced in Chapter 6. The proposed method utilizes a hierarchical structure and several filters to improve motion detection accuracy. Implantation results on a number of video sequences are provided as well.

Chapter 6

Proposed Motion Detection Algorithm

6.1 Introduction

Some dominant motion detection methods for interlaced video were reviewed in Chapter 5. Examining their simulation results proved that five-field motion detection algorithm suggested by grand Alliance of HDTV has better performance compared to other available methods. However its motion detection results are not accurate enough.

In this thesis we have tried to examine different approaches to find a better motion detection algorithm with higher accuracy. As a result we are proposing a new hierarchical motion detection algorithm with improved accuracy. This algorithm is introduced in this chapter.

Section 6.2 of this chapter investigates the result of increasing the number of fields in motion detection. Section 6.3 introduces the structure of the new motion detection algorithm and provides some implementation and evaluation results.

6.2 Increasing Number of Fields

The first investigated approach, for improving motion detection accuracy, has been studying the result of utilizing more interlaced video fields for motion detection. However increasing the number of fields does not provide a great deal of algorithm improvement under different arrangements.

To find a theoretical justification for this, time correlation between pixel intensity values of consecutive frames of a video sequence has been calculated for a number of video sequences. The overall results are depicted in Figure 6-1. Figure 6-1 shows the average time correlation value between ten consecutive fields of a video sequence.

Video sequences used in this experiment have been chosen from different video categories, from panning camera to sports. No scene change has been included in the partial sets under test, where a set means ten consecutive fields. Table 8-2, Table 8-3 and Table 8-4 provide detailed technical information about each video sequence.

As shown in Figure 6-1, time correlation between consecutive fields of video drops sharply after few fields. It drops to a value below 0.7 after five fields; which means that video sequences are normally uncorrelated after five fields. Therefore it is not reasonable to include more than five field of video sequence in motion detection.

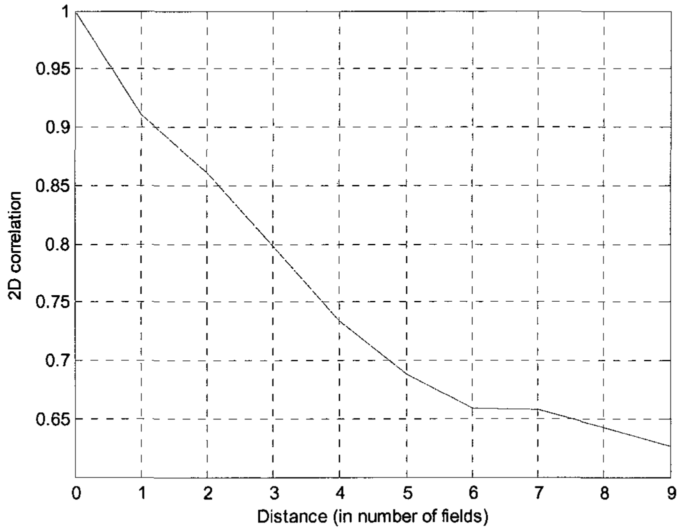


Figure 6-1 Average time correlation values, between consecutive fields of an interlaced video sequence.

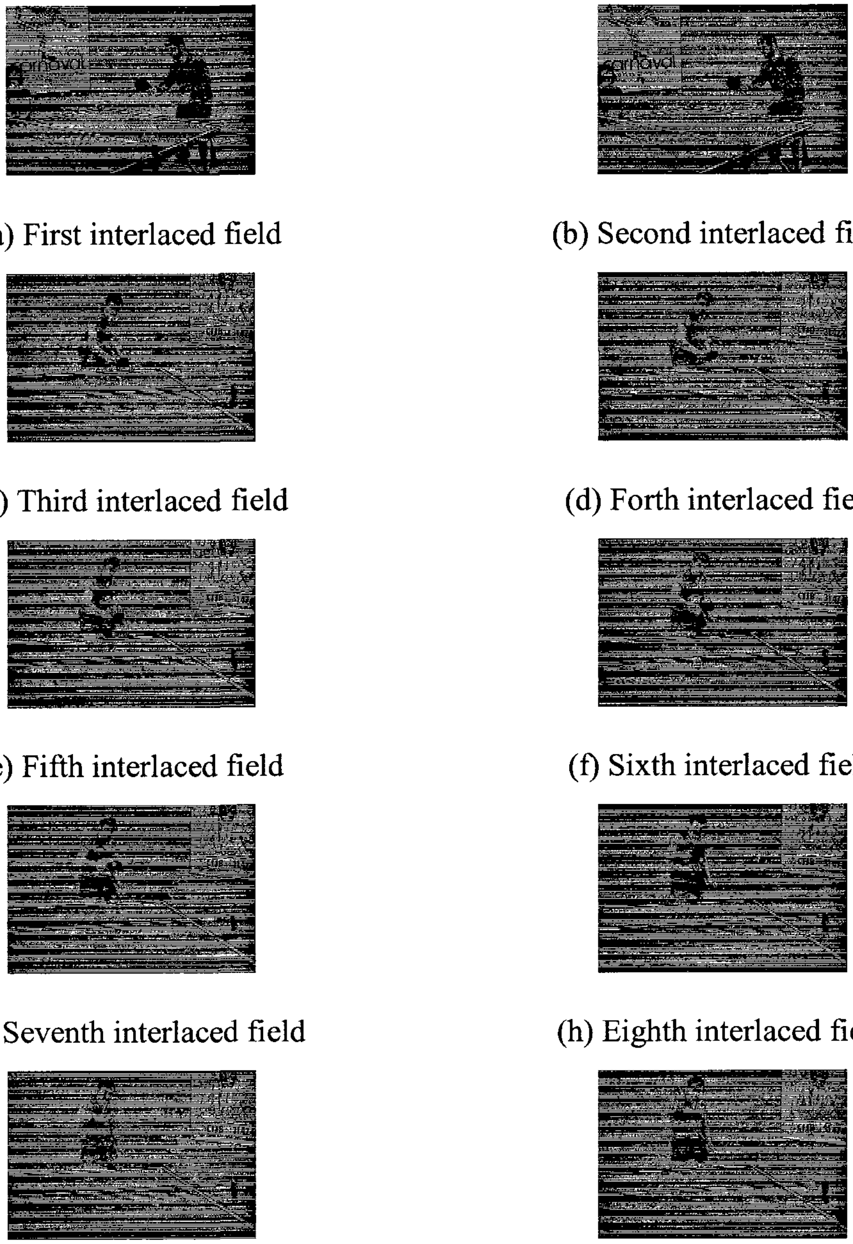
Some other tests have been run to examine the result of scene change in time correlation values between consecutive fields.

Figure 6-2 shows ten consecutive field of an interlaced video where a scene change occurs after second field. Figure 6-3 shows the time correlation values between first field shown in Figure 6-2 and its consecutive fields. It shows a sharp correlation drop after second field which is coincident with scene change.

Including a scene change in the test will cause a sharp drop in time correlation values. Considering scene change as a normal incident in a video sequence, this result is one more evidence proofing that using more than five consecutive fields of video sequence for motion detection is not beneficial.

Therefore it can be concluded that the optimum number of correlated fields to be used in a motion detection algorithm is five. Using less than five fields is not broad enough therefore does not achieve best possible accuracy while using more than five field does not provide more valuable information compared to five fields and is not reasonable.

Proposed motion detection algorithm uses five consecutive fields of interlaced video for motion detection with a superior structure which provides high motion detection accuracy.



(a) First interlaced field

(b) Second interlaced field

(c) Third interlaced field

(d) Forth interlaced field

(e) Fifth interlaced field

(f) Sixth interlaced field

(g) Seventh interlaced field

(h) Eighth interlaced field

(i) Ninth interlaced field

(j) Tenth interlaced field

Figure 6-2 Ten consecutive fields of an interlaced video sequence; a scene change occurs after second field.

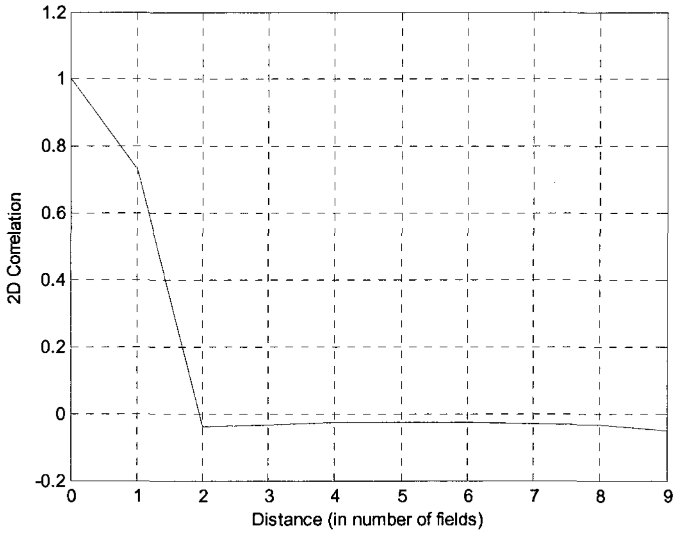


Figure 6-3 Time correlation between video sequence shown in Figure 6-2

6.3 Proposed Motion Detector

A motion detection method with high detection accuracy is proposed in this section. The proposed method has a hierarchical structure which uses five consecutive fields of video data for motion detection. The proposed algorithm is capable of catching a wide range of motions from slow to fast. Evaluation results confirm that the proposed algorithm accuracy is higher than the other available methods.

The proposed method works on a set of five consecutive interlaced video fields at each given time. Section 6.3.1 explains a standard arrangement of five consecutive interlaced fields which is representative of relative position of missing lines in each standard input to the algorithm.

6.3.1 Standard Input of Proposed Motion Detector

Proposed motion detection algorithm uses five consecutive fields of interlaced video data to increase the ability of detecting fast motions and consequently the overall algorithm performance.

Figure 6-4 shows a sequence of five interlaced video fields which will be transferred to motion detector at each time. In this figure, the field marked as "*n*" represents the current field, the fields marked by "*n+1*" and "*n+2*" represent its two subsequent fields, while the fields marked by "*n-1*" and "*n-2*" represent the two fields prior to the current field.

As shown in Figure 6-4 the pairs (n-2, n) (n-1, n+1) and (n, n+2) have the same missing lines and the same time difference. This observation is more plainly illustrated in Figure 6-5. Two fields of each pair are linked to each other by arrows in Figure 6-5; three different colors used for arrows are representatives of the three analogous pairs.

Therefore each five consecutive fields of interlaced video will form three analogues pair of fields which will be fed to the motion detector simultaneously. Section 6.3.2 introduces the block diagram representation of motion detector.

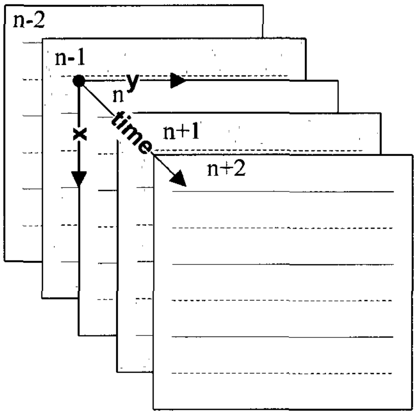


Figure 6-4 Standard input of proposed motion detector; five interlaced video fields.

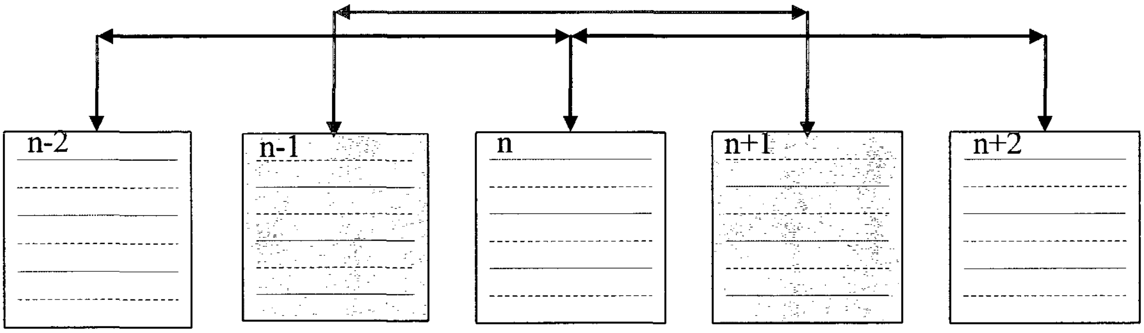


Figure 6-5 Relative position of missing lines in five consecutive fields; each connected pair of fields has same missing lines and same time difference as the other connected pairs.

6.3.2 Block Diagram of Proposed Motion Detector

Block diagram of proposed motion detector is shown in Figure 6-6, where it receives five consecutive fields of interlaced video as input.

The spatial differences between three pairs of fields mentioned in Figure 6-5 are calculated in the first stage of motion detection.

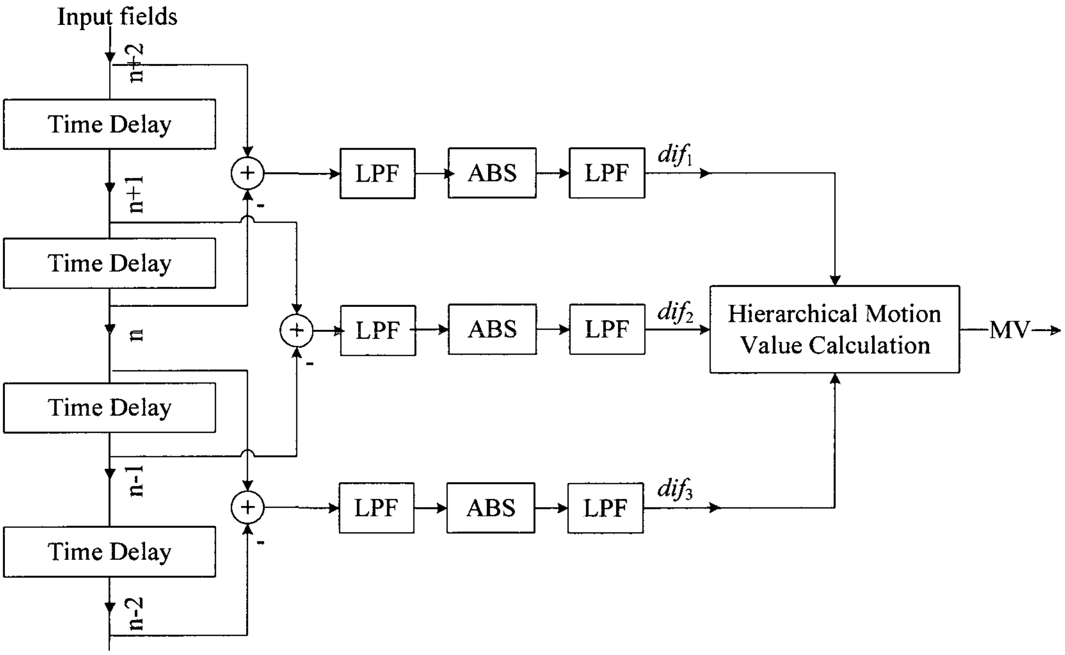


Figure 6-6 Block diagram of the proposed motion detection algorithm.

To increase the accuracy of the proposed motion detector, the following common assumptions are made[1]:

1. Signal is large and noise is small
2. The low frequency energy in signal is greater than low frequency energy in noise and alias

To profit from these assumptions, the spatial field differences calculated in the first stage are low-pass filtered in the second stage of the algorithm. This low pass filtering will reduce high frequency noise from the spatial difference signals.

Several combination of low pass filtering has been studies; including various versions of Two-dimensional (2-D) Gaussian filtering, 2-D median filtering and 2-D spatial averaging. 2-D spatial averaging filters have shown to achieve best results in most cases.

In the third stage of the algorithm, the absolute values of these filtered signals are obtained. Finding the absolute value is justified by considering the fact that motion detection goal is to detect the presence of motion rather than finding its direction or estimating motion value.

The rectified signals are then low-pass filtered again to improve the consistency of the outputs based on the assumption that moving objects are large compared to a pixel.

The assumption which justifies the existence of this filter helps in choosing the best

[1] These assumptions may not be valid in some rare cases. However those rare cases are not the concern of this thesis.

candidate for this filter as well. By choosing an $m \times m$ median filter for this stage, it is possible to improve output consistency while preserving the edges.

The last block in Figure 6-6 is hierarchical motion value calculator. It receives low-pass filtered signals dif_1, dif_2 and dif_3 as input. Subsequent to some calculation it delivers final motion detection result to its output. The final output is a matrix called Motion Value (MV), with the same size as the original interlaced fields. The hierarchical structure of this block increases the accuracy of motion detector and reduces its complexity. Section 6.3.3 explains the structure of this block.

6.3.3 Hierarchical Motion Value Calculator

As shown in Figure 6-6, dif_1, dif_2 and dif_3 are delivered to the hierarchical motion calculation module. This module partitions each input signal into $n \times m$ data blocks. Subsequent to that, the average intensity value of each data block will be calculated

Considering three corresponding average values as the average values of data blocks with the same size and spatial location, the maximum of each three corresponding averages value is then compared with a predefined threshold value.

If the maximum of the three is smaller than threshold, the block is considered as a static part of video and by setting its motion value equal to zero no further processing will be applied to that block.

If the maximum of the three average intensity values is greater than threshold, the block of data is considered as a dynamic block and will be recursively partitioned to smaller blocks and the same procedure continues. This procedure may continue to pixel level, where each data block is one pixel.

The final output of hierarchical structure is a motion value matrix with the same size as the original video fields.

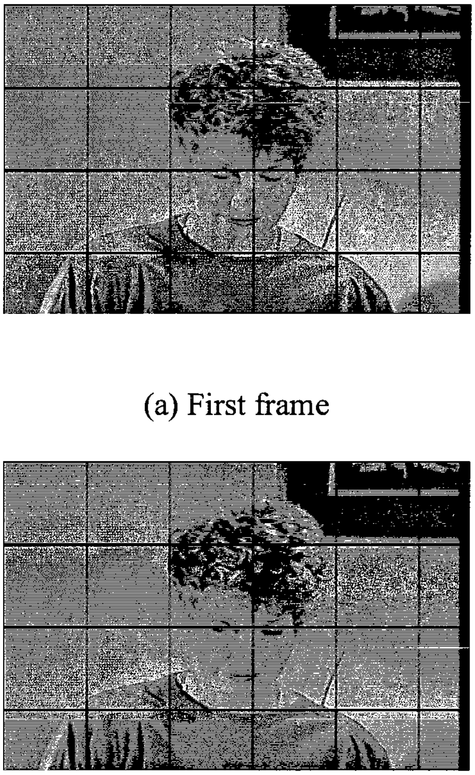
The next section explains how the threshold values are defined and adjusted in each level of the hierarchical procedure.

6.3.4 Finding Optimal Values for Predefined Thresholds

The hierarchical algorithm utilizes some predefined threshold values to determine whether a block of data is static or dynamic. The optimum values for these thresholds are found trough some trial and error experiments which are briefly explained in this section.

Since motion detector compares average intensity values with predefined threshold values to choose whether a block of data is static or dynamic, threshold values should have some common characteristics with average intensity values. These threshold values should also be within the same range as average intensity values.

The standard Mom video sequence has been chosen as a starting point for experimental test. This video sequence has slow motions in head and face area and no motion in most other regions. Two consecutive frames of this video have been partitioned to 64 × 64 data blocks. These two frames are shown in Figure 6-7.

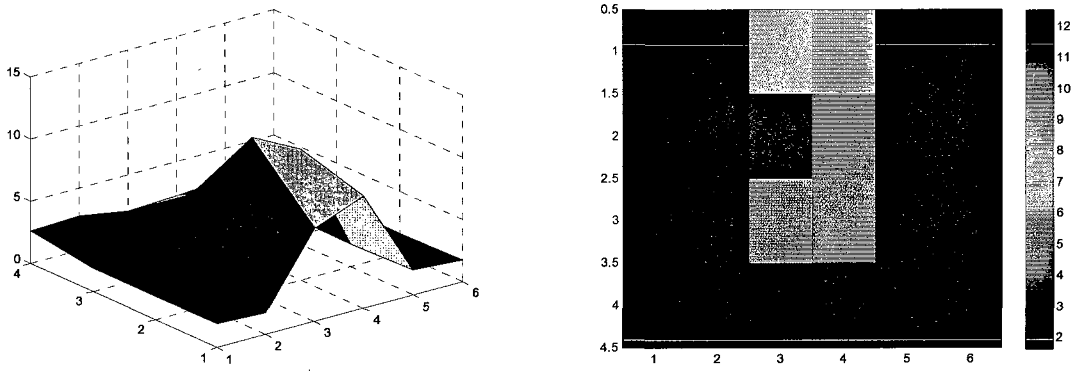


(a) First frame

(b) Second frame

Figure 6-7 Two consecutive frame of MOM video partitioned to 64 × 64 blocks.

The average value of absolute intensity differences has been calculated for each 64 × 64 data block. Figure 6-8 (a) shows a three-dimensional plot of these average values. These average values are also plotted in Figure 6-8 (b) as an image object. It shows that the slight motion of the face causes the average value of its corresponding blocks to be larger than the average values of data blocks with no existing motion.



(a) (b)

Figure 6-8 Average Intensity values for each 64 × 64 block of data shown in Figure 6-7 . (a) Three-dimensional plot (b) The average value of each block as an image object in the same position.

In a subsequent step, each 64×64 data block in Figure 6-7 has been partitioned to 16×16 data blocks. The average value of absolute intensity differences has been calculated for each 16×16 data block as well. These average values are plotted in Figure 6-9 as an image object.

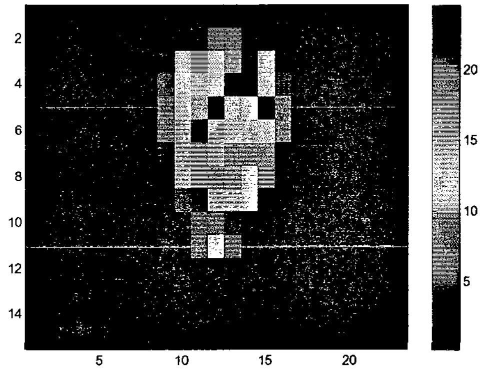


Figure 6-9 Average intensity values for 16 × 16 data blocks of Figure 6-7.

To be able to monitor the effect of block size on average intensity values, this test has been repeated for various block size on several images and the average values has been recorded each time.

For each $N \times N$ image, the size of the data blocks has been changed from $N/2 \times N/2$ to 1×1. The average intensity difference values for each block size has been calculated and the effect of the size of the block on its corresponding average intensity difference has been studied.

This experiment has been repeated for different types of video sequences with very different motion activity levels from slow motion to fast motions. The average intensity difference values have been monitored for dynamic and static parts of each video.

Accordingly, some preliminary estimation for threshold values for different types of videos and different block sizes have been obtained.

These initial values have been applied to the proposed motion adaptive deinterlacing method which uses the proposed motion detector. The resulting mean square errors have been calculated in each step; the mathematical definition of calculated mean square error is explained in Chapter 8. Threshold values have been recursively readjusted to improve the deinterlacing performance by reducing the calculated errors.

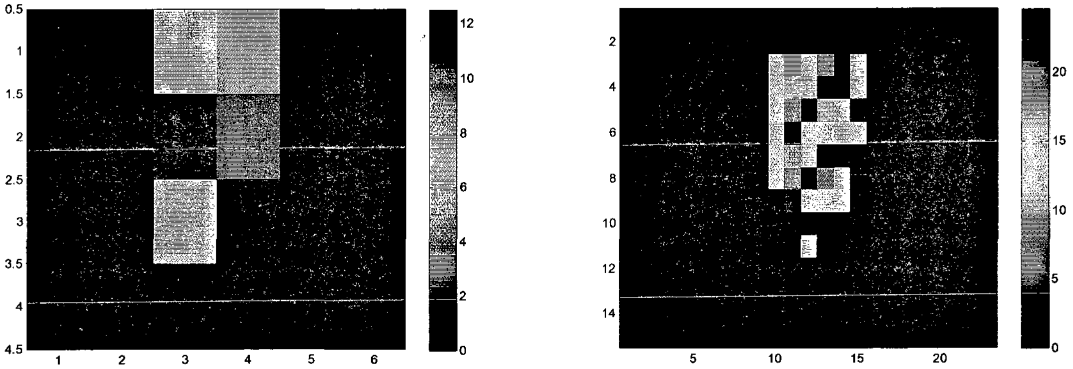
As a result, it has been found that intensity differences larger than 20, on a 0-255 scale for intensity values, generates noticeable motion artifacts. This value is found while working on pixel basis. For other block sizes, this value will be scaled down by a factor equal to inverse of the logarithm of the size of the block; size of a block is equal to the total number of pixels in that block.

Although the above mentioned arrangement for pixel level threshold equal to 20 and a scale down proportional to $1/\log_{10} n^2$ for an $n \times n$ block has proved to obtain best performance in most cases, one more level of fine tuning is possible if we know the type of video sequence in advance. The required values for fine tuning are found by repeating our test procedure for different types of video sequences. Table 6-1 provides optimum threshold values and adjustment factors for a few types of video sequences.

Table 6-1 Initial threshold values and adjustment factors for different video types.

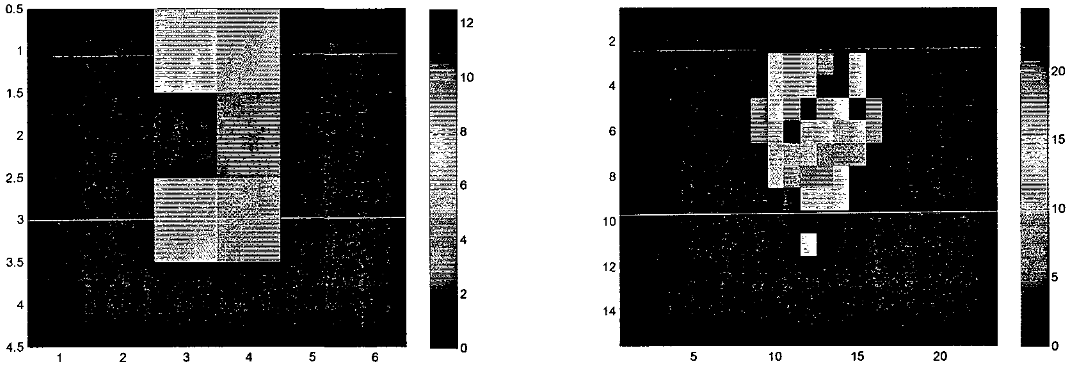
Video Type	Initial Threshold Value	Adjustment Factor for an $n \times n$ block
Type-0: General setup with no advance information about video type	20	$1/\log_{10} n^2$
Type-1: Slow motions such as a video taken from a person talking calmly	15	$1/\log_{10} n^2$
Type-2: Fast motions with fine details such as a video taken from a sport contest	27	$1/\log_2 n$
Type-3: Lots of moving objects with varying range of motion speeds and fine details such as a video obtained from a panning camera.	45	1

Threshold values found for general setup has been applied to average values plotted in Figure 6-8 and Figure 6-9 and the results are plotted in Figure 6-10. As MOM video sequence could be considered a Type-1 video according to Table 6-1, threshold values related to Type-1 video has been applied to these images. The corresponding intermediate results are plotted in Figure 6-11. It shows that using Type-1 threshold values provides more accurate intermediate results.



(a) 64 × 64 blocks of Figure 6-8 (b) 16 × 16 blocks of Figure 6-9

Figure 6-10 Intermediate results using Type-0 (general type) threshold setup from Table 6-1.



(a) 64 × 64 blocks of Figure 6-8 (b) 16 × 16 blocks of Figure 6-9

Figure 6-11 Intermediate results using Type-1 threshold setup from Table 6-1

6.4 Conclusion

Time correlation between consecutive fields of a video sequence was investigated in this chapter. As a result it was concluded that using five consecutive fields of video sequence for motion detection is the optimum case

A five field motion detection method was proposed and explained in detail. A sample simulation results using proposed motion detection method is shown in Figure 6-12.

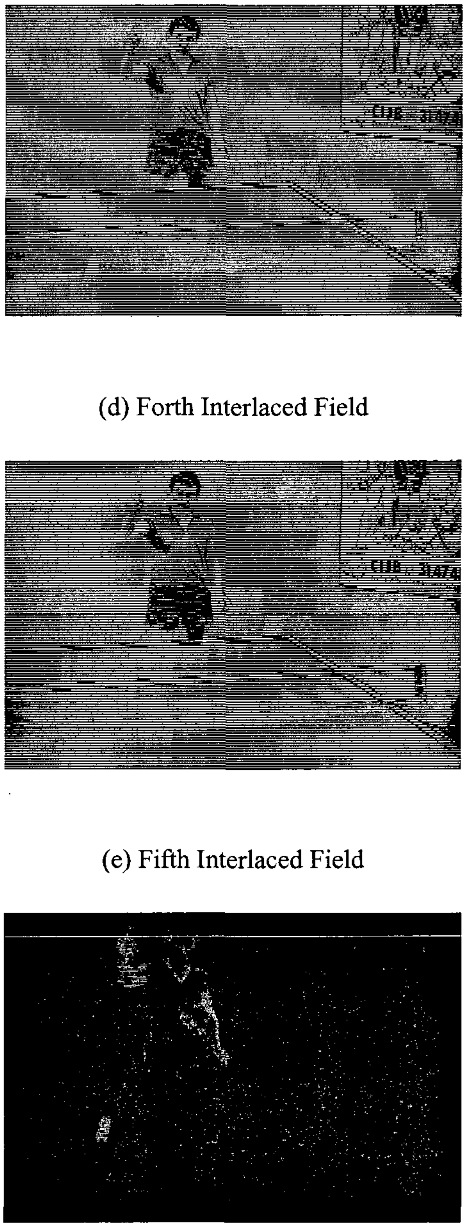
The first low pass filter in this implementation has been a 2×2 spatial averaging filter while the second low pass filter has been a 3×3 spatial median filter, and general threshold setup has been used.

Comparing motion detection result shown in Figure 6-12 (f) with motion detection result shown in Figure 5-14(f) shows that proposed motion detection has higher capability in detecting fast motions. More detailed simulation results are provided in Chapter 8 from which it can be concluded that proposed motion detection algorithm is capable of detecting a wide range of motions with a high accuracy.

(a) First Interlaced Field

(b) Second Interlaced Field

(c) Third Interlaced Field



(d) Forth Interlaced Field

(e) Fifth Interlaced Field

(e) Proposed motion detection

Figure 6-12 Implementation results for proposed motion detection; (a) First interlaced field, (b) Second interlaced field, (c) Third interlaced field (current field), (d) Forth interlaced field, (e) Fifth interlaced field, (f) Proposed motion detection results.

Chapter 7

Proposed Motion Adaptive Deinterlacing

7.1 Introduction

Motion adaptive deinterlacing methods are among the most popular deinterlacing methods, where the deinterlacing performance is improved by combining the benefits of temporal and spatial deinterlacing methods.

A temporal deinterlacing method exploits temporal data to interpolate missing lines of an interlaced video field. Temporal correlation is typically high in static parts of a video sequence therefore a temporal deinterlacing method obtains more valuable temporal data in static parts of video. Consequently, performance of a temporal deinterlacing method is high in static parts of a video sequence.

A spatial deinterlacing method exploits spatial data to interpolate missing lines of an interlaced video field. In a dynamic part of video sequence, spatial correlation is typically higher than temporal correlation. Consequently, a spatial deinterlacing method has a higher performance in dynamic parts of video sequence.

A motion adaptive deinterlacing improves its performance by applying a temporal deinterlacing method to static parts of video sequence and a spatial deinterlacing method to dynamic parts of a video sequence. A motion detection method will be used to divide a video sequence to dynamic and static parts.

The results from the static and dynamic parts will be combined to reconstruct the whole deinterlaced video sequence.

In this thesis we have proposed a high performance motion adaptive deinterlacing method. The proposed motion adaptive deinterlacing method, uses the hierarchical motion detection algorithm introduced in Chapter 6 to calculate motion activity level of an interlaced video sequence.

The motion detection results will be consequently used in proposed deinterlacing structure to interpolate the missing lines of interlaced video fields. This chapter explains the proposed deinterlacing method. Section 7.2 explains the structure of the deinterlacing method using a block diagram representation. Section 7.2.1 defines the nonlinear transform used in deinterlacing method and explains the procedure which is used to calculate its constant values.

The interpolation block is explained in section 7.2.2. A sample simulation result is given in conclusion section. However more detailed simulation and evaluation results are presented in Chapter 8.

7.2 Proposed Motion Adaptive Deinterlacing

Proposed motion adaptive deinterlacing method measures the motion activity level of video data using the proposed motion detection algorithm. The measured motion activity level is subsequently transformed to motion possibility value.

Motion possibility value will be used in interpolation algorithm as a weight factor for combining the results of a temporal deinterlacing and a spatial deinterlacing method. As a result of this combination, deinterlacing performance of the proposed method outperforms both temporal and spatial deinterlacing methods. Figure 7-1 shows a block diagram representation of the proposed motion adaptive deinterlacing method.

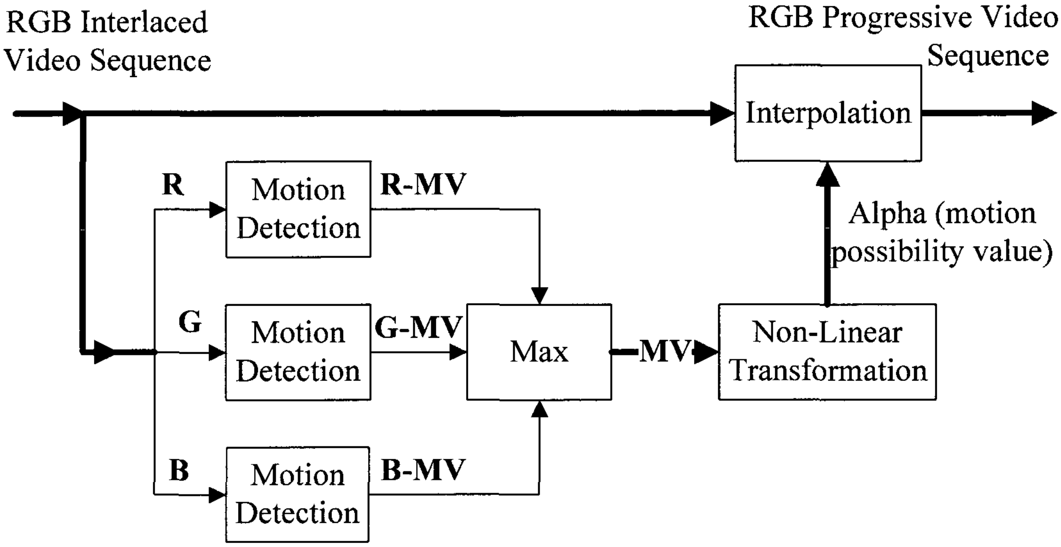

RGB Interlaced Video Sequence
RGB Progressive Video Sequence
Interpolation
R
Motion Detection
R-MV
Alpha (motion possibility value)
G
Motion Detection
G-MV
Max
MV
Non-Linear Transformation
B
Motion Detection
B-MV

Figure 7-1 Block diagram of proposed motion adaptive deinterlacing method.

As shown in Figure 7-1, each color component of video sequence is delivered to a motion detection block. Each motion detection block calculates motion values using proposed motion detection method. In Figure 7-1, motion detection results on color component *R* is called *R_MV*, motion detection results on color component *G* is called *G_MV* and motion detection results on color component *B* is called *B_MV*.

The result of motion detection on *R*, *G*, and *B* color components are then compared to each other. The maximum of three motion values is used as the motion value (MV) for all three color components.

Applying the motion detection algorithm to all color components reduces the possibility of error detection by reducing the number of missed detections. Comparing these values and using a single value for all three color components avoids distortion artifacts in the output.

The final motion value (MV*)* is then delivered to a non-linear transformation block. The implemented non-linear transformation is explained in section 7.2.1. It transforms the motion value to a number between zero and one. This value corresponds to the possibility of motion. It is called motion possibility value and represented by α.

The motion possibility value is consequently delivered to interpolation block as an auxiliary data. The interpolation algorithm uses motion possibility value to make a linear combination of a temporal and a spatial deinterlacing method applied to the RGB interlaced video. The preferred temporal and spatial deinterlacing methods as well as the combination method are explained in section 7.2.2. The output of the interpolation block is a RGB progressive video sequence.

7.2.1 Non-Linear Transformation

A motion activity value obtained from proposed motion detection algorithm may have any value between 0 and 255; on a 0-255 scale for intensity values. This motion value is converted to a motion possibility value using a non-linear transformation.

The non-linear transformation function is represented in Figure 7-2. It is a uniform function of motion value. The transformation result is the possibility of motion.

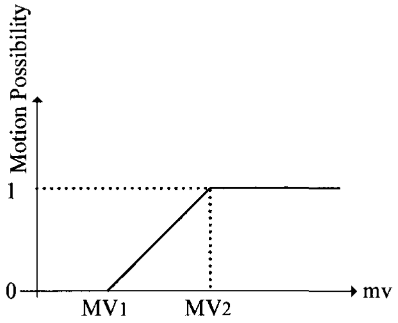


Figure 7-2 Non-linear transformation function.

MV_1 and MV_2 are two constant motion values. Motion possibility is zero if motion value is smaller than MV_1. Motion possibility is one if motion value is greater than MV_2 . For motion values between MV_1 and MV_2 motion possibility will be a number between zero and one.

MV_1 and MV_2 are found by recursive error optimization. For this procedure, both MV_1 and MV_2 have been changed by small steps in their valid range. The deinterlacing error has been calculated and recorded for each setup.

As MV_1 and MV_2 are two motion values their actual valid range is 0 to 255 on a 0-255 scale for intensity value. However, the only constrain is that MV_1 is always smaller than MV_2. Therefore the actual valid range for MV_1 is considered 0 to 254. While the actual valid range for MV_2 is considered 1 to 255. Figure 7-3 is a sample plot of MSE (Mean Square Error) vs. MV_2 while MV_1 has been set equal to 60.

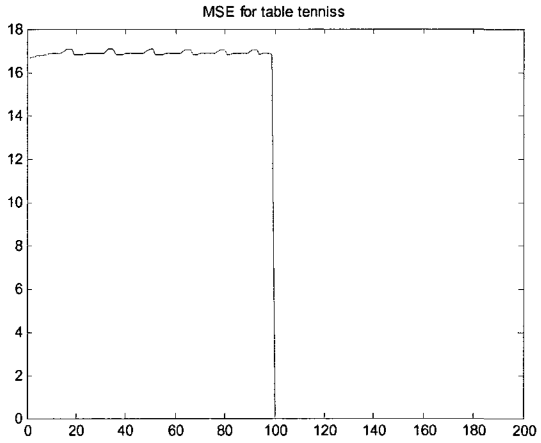


Figure 7-3 MSE of proposed deinterlacing method for various MV_2while $MV_1 = 60$

Based on minimum calculated error, optimum range of values for MV_1 and MV_2 has been found. Smaller step size has been used in the optimum range to find the best choice for minimum deinterlacing error. This procedure has been repeated on several video sequences and best estimation for MV_1 and MV_2 has been found based on minimum deinterlacing error.

As a result, optimum values has been found to be $MV_1 = 60$ and $MV_2 = 100$.

7.2.2 Interpolation Algorithm

In the proposed motion adaptive method simple line averaging is chosen as the spatial deinterlacing and median filtering is chosen as the temporal deinterlacing method. Motion possibility values, calculated after motion detection step, are used as the weighting factor for combining the results of spatial and temporal interpolations to improved performance. The interpolation step could be formulized by Equation (7-1) where α is the motion possibility value measured in motion detection algorithm.

$$F_o(X,n) = \begin{cases} F_i(X,n) & where \quad y\,mod\,2 = n\,mod\,2 \\ \alpha F_{spatial}(X,n) + (1-\alpha)F_{temporal}(X,n) & otherwise \end{cases} \tag{7-1}$$

Where:

$X = (x,y)$ represents pixel position

n represents field index

$F_o(X,n)$ represents the intensity value of a pixel in deinterlaced frame

$F_i(X,n)$ represents the intensity value of a pixel in original interlaced field

$F_{spatial}(X,n)$ represents the intensity value of a pixel using the utilized spatial deinterlacing method.

$F_o(X,n)$ represents the intensity value of a pixel using the utilized temporal deinterlacing method.

α represents motion possibility value.; α is a function of X and n

α is a function of X and n. For highly dynamic parts of the video, the value of α would be equal to unity; resulting in pure spatial interpolation, according to Equation (7-1). Temporal correlation is low in highly dynamic regions. Therefore deinterlacing performance of temporal methods drops in these regions. Converting to a pure spatial deinterlacing in highly dynamic regions is to keep the proposed algorithm performance high in these regions.

For highly static parts of the video signal α would be equal to zero; resulting in pure temporal interpolation, according to Equation (7-1). Temporal correlation is high in static regions and temporal deinterlacing performance is high as well. Converting to a pure temporal deinterlacing in static regions is to benefit from the high performance of temporal methods in these regions.

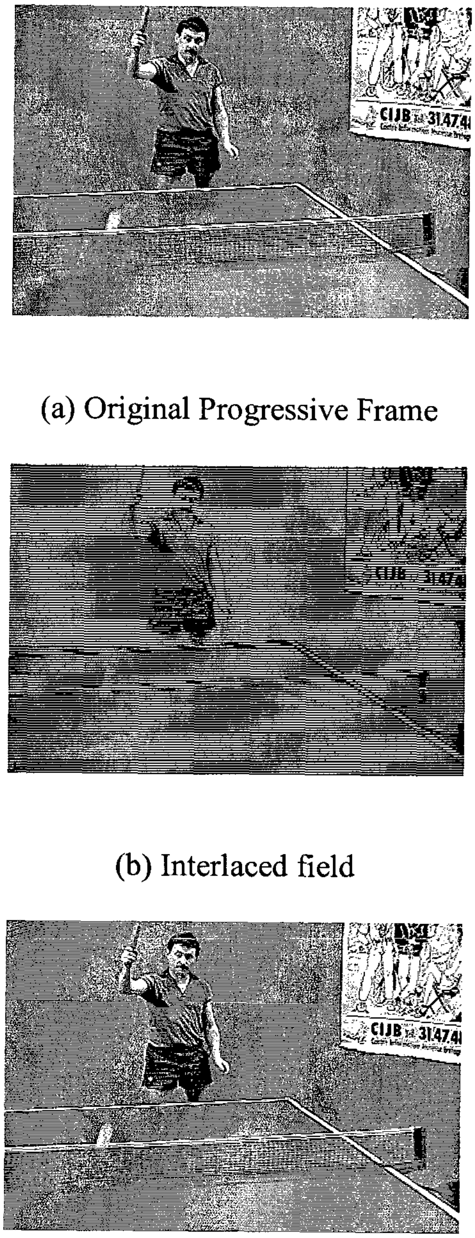
7.3 Conclusion

Proposed motion adaptive deinterlacing method was introduced in this chapter. The method uses the proposed motion detection algorithm which calculated the motion activity level of video sequence by a hierarchical structure.

In order to save the consistency of motion detection algorithm and to avoid introducing extra artifacts the motion detection algorithm is separately applied to all color components and the results are compared to each other to reach to a unique motion possibility result.

The interpolation step uses motion detection results to obtain best weighting factor for combining temporal and spatial interpolation values.

A sample implementation result is illustrated in Figure 7-4. A broad range of implementation and evaluation results are offered in Chapter 8. The overall results prove that proposed deinterlacing algorithm has outperformed other available deinterlacing methods.



(a) Original Progressive Frame

(b) Interlaced field

(c) Deinterlaced frame

Figure 7-4 Proposed motion adaptive deinterlacing result.

Chapter 8

Simulation Results

8.1 Introduction

Video quality is primarily a subjective matter however some objective measures are commonly used to reflect subjective impression of video quality. The objective measures are not always directly related to subjective quality but using them is a common practice in evaluating video processing algorithms.

Seeking an all-inclusive evaluation tools, both subjective and objective evaluations have been performed in this research. Performance of the proposed deinterlacing method is evaluated both subjectively and objectively. Several deinterlacing methods have been implemented and compared with the proposed method.

Section 8.2 explains our experimental setup. Section 8.2.1explains the objective performance evaluation criterion used in this thesis while section 8.2.2 explains the subjective evaluation criterion. Technical details of the test video sequences are provided in section 8.2.3. Section 8.2.4provides a list of the implemented deinterlacing algorithms

Experimental results are provided in section 8.3. Objective evaluation results are provided in section 8.3.1. Where section 8.3.1.1compares the performance of the proposed deinterlacing method with other deinterlacing methods. The overall performance improvement using this method is provided in this section. The robustness of the proposed deinterlacing method for various frame rates is examined and compared with other methods in section 8.3.1.2. Section 8.3.1.3provides some evaluation results for the proposed motion detection algorithm. Objective evaluation results are provided in section 8.3.2.

8.2 Experimental Setup

In order to evaluate the performance of the proposed motion adaptive deinterlacing method, the proposed method has been simulated using some standard video sequences. Since the original standard video sequences have been in progressive format, they have been interlaced in the beginning. Interlacing has been done by removing every other line from each frame of the progressive video sequence which is equal to vertical decimation. Figure 8-1 shows this procedure. As shown in Figure 8-1, one field of interlaced video will be generated from each frame of a progressive video sequence.

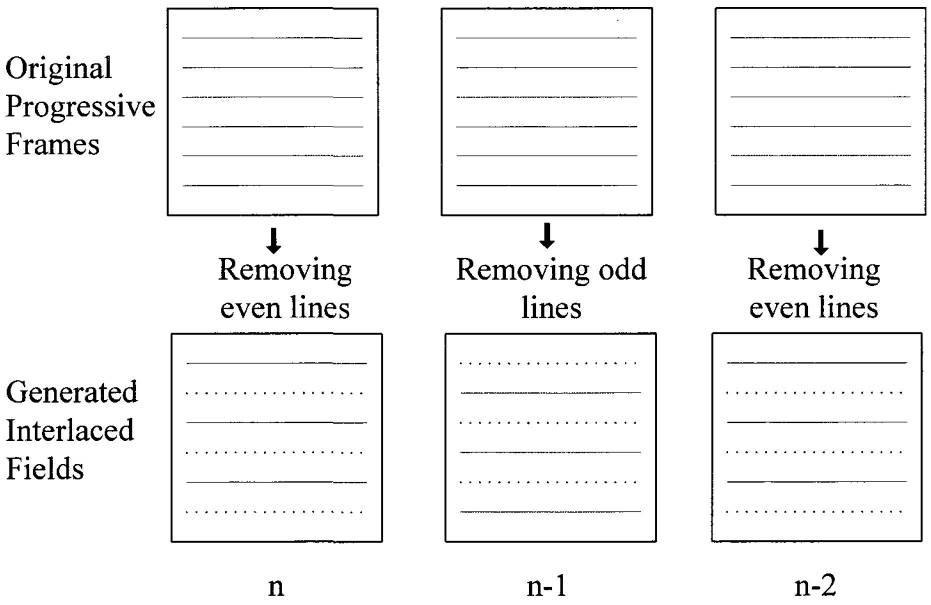


Figure 8-1 Interlacing; generating interlaced video sequences from progressive video sequences.

Deinterlacing methods will be applied to the interlaced video sequence. The deinterlaced video sequences are then compared with the original progressive video sequences to evaluate their performance. Figure 8-2 shows this procedure.

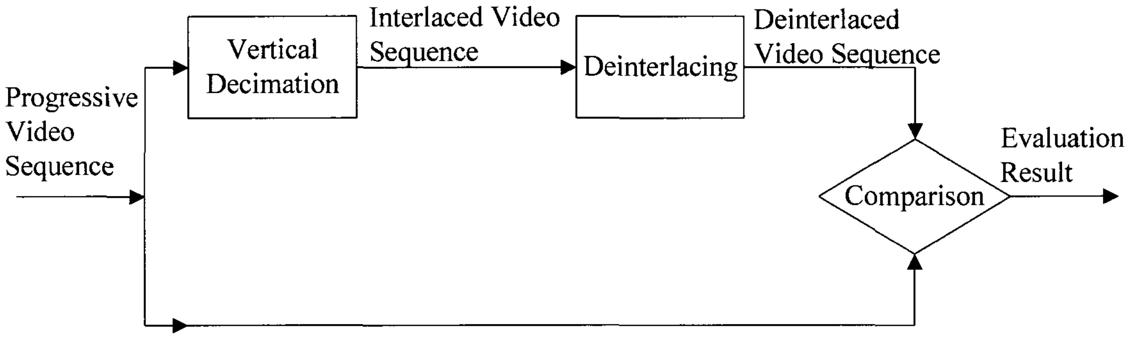

Progressive Video Sequence
Vertical Decimation
Interlaced Video Sequence
Deinterlacing
Deinterlaced Video Sequence
Comparison
Evaluation Result

Figure 8-2 Performance evaluation method.

8.2.1 Objective Evaluation Criterion

In literature on video deinterlacing, either Mean Square Error (MSE) or Peak Signal to Noise Ratio (PSNR) is used as an objective measure criterion.

MSE, defined by Equation (8.1), is the average square difference between the original progressive frame and the deinterlaced frame.

$$MSE = \frac{1}{N}\sum_{\vec{x}}\left(F_p(\vec{X},n) - F_{de}(\vec{X},n)\right)^2 \tag{8.1}$$

While the PSNR is calculated according to Equation (8.2)

$$PSNR = 10\ \log_{10}\left(\frac{255^2}{MSE}\right) \tag{8.2}$$

This thesis uses PSNR as an objective measure to compare deinterlacing methods.

8.2.2 Subjective Evaluation Criterion

Subjective test has been established according to ITU-R[1] BT[2].500-11[87]. ITU-R BT.500-11 is a methodology recommended by ITU-R for subjective assessment of the quality of the television pictures.

According to ITU-R BT.500-11 at least 15 observers should be used in subjective test and they should be non-expert viewers with normal visual acuity and color vision. Therefore we have asked 25 people to evaluate our sequences. They have been both male and female non-expert observers between ages 15 and 65.

According to ITU-R BT.500-11 double-stimulus impairment scale (DSIS) method, the

[1] International Telecommunication Union-Radiocommunication sector

[2] Broadcasting Service (Television)

assessors have been first presented by an original progressive video sequence. Then they have been presented by the deinterlaced version of same video sequence. Following that they have been asked to vote on the deinterlaced video, keeping in mind the original progressive version. They have been asked to vote based on five-grade impairment scale given in Table 8-1.

Table 8-1 Grading scale for subjective evaluation.

Grade	Impairment level
5	Imperceptible
4	Perceptible but not annoying
3	Slightly annoying
2	Annoying
1	Very Annoying

The last stage in subjective test is to condense the recorded scores in graphical or numerical forms which summarize the performance of the deinterlacing algorithm. ITU-R BT.500-11 recommends the calculation of mean score and its associated confidence interval for analysis of the results of double-stimulus impairment-scale method. Mean score is calculated using Equation (8.3)

$$\bar{u}_j = \frac{1}{N}\sum_{i=1}^{N} u_{ij} \tag{8.3}$$

Where:

u_{ij} : Score of observer i for test sequence j

N : Number of observers

Similarly, overall mean score $\bar{u}$ could be calculated.

For the associated confidence interval it is recommended to use the 95% confidence interval which is given by Equation (8.4)

$$[\bar{u}_j - \delta_j \ \bar{u}_j + \delta_j] \tag{8.4}$$

Where

$$\delta_j = 1.96\frac{S_j}{\sqrt{N}}$$

$$S_j = \sqrt{\sum_{i=1}^{N}\frac{(\bar{u}_j - u_{ij})^2}{N-1}}$$

By using 95% confidence interval, with probability of 95%, the absolute value of the difference between the experimental mean score and the true mean score is smaller than the 95% confidence interval.

8.2.3 Test Video Sequences

This thesis has used a number of standard video sequences for performance evaluation. These video sequences are divided to three groups. The first group is made up of videos with slow motions, called Type-1 in Table 6-1. In particular they are taken from different individuals talking in front of camera. Properties of this group are given in Table 8-2.

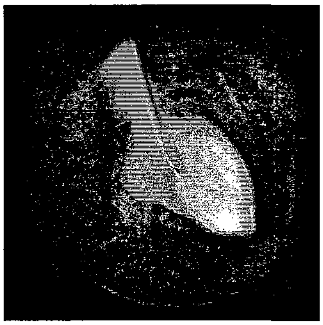
Table 8-2 Technical properties of first group of test sequences; Type-1 video.

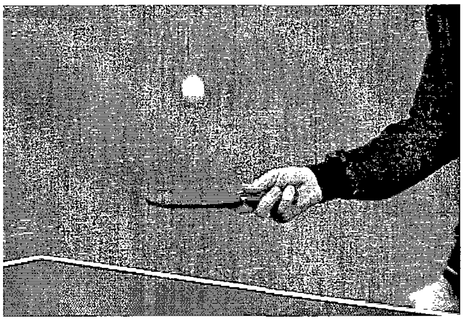
	Sequence Name: Grandmom Video Type: RGB Number of Frames: 150 Frame size: 240 × 360 Frame Rate: 60 frames/sec
	Sequence Name: Mom Video Type: RGB Number of Frames: 150 Frame size: 240 × 360 Frame Rate: 60 frames/sec

	Sequence Name: Momdaugther Video Type: RGB Number of Frames: 150 Frame size: 240 × 360 Frame Rate: 60 frames/sec

The second group is made up of videos with fast motions and fine details which requires high quality, called Type-2 in Table 6-1. They include sport videos or medical images. Properties of this group are given in Table 8-3.

Table 8-3 Technical properties of second group of test sequences; Type-2 video.

	Sequence Name: Heart Video Type: Gray scale Number of Frames: 38 Frame size: 256 × 256 Frame Rate: 60 frames/sec

	Sequence Name: Table tennis Video Type: RGB Number of Frames: 150 Frame size: 240 × 352 Frame Rate: 60 frames/sec

The third group is made up of videos with lots of moving objects with varying range of motion speeds and fine details, called Type-3 in Table 6-1. They are normally captured by a panning camera. Properties of this group are given in Table 8-4.

Table 8-4 Technical properties of third group of test sequences;Type-3 video.

	Sequence Name: Diskus Video Type: Gray scale Number of Frames: 124 Frame size: 288 × 352 Frame Rate: 60 frames/sec

	Sequence Name: Movi Video Type: Gray scale Number of Frames: 10 Frame size: 512 × 512 Frame Rate: 24 frames/sec
	Sequence Name: Flower garden Video Type: RGB Number of Frames: 150 Frame size: 240 × 352 Frame Rate: 60 frames/sec

8.2.4 Implemented Deinterlacing Algorithms

Thirteen deinterlacing methods have been implemented in this research. These methods are listed in Table 8-5. Specifically all methods surveyed in Chapter 4 have been implemented. The performances of these methods have been compared with the performance of the proposed deinterlacing method.

These methods are chosen from different deinterlacing categories. Line repetition and bilinear line averaging are chosen from spatial deinterlacing methods. Field insertion and bilinear field interpolation are chosen from temporal methods. Vertical Temporal median filtering and weighted and edge dependant median filtering [60] have been chosen from hybrid deinterlacing methods.

Motion compensated field insertion and motion compensated bilinear field interpolation have been chosen from motion compensated deinterlacing methods.

From motion adaptive methods, motion adaptive with 3-field motion detection and bilinear field interpolation and line averaging, motion adaptive with 4-field motion detection and field insertion and line averaging have been implemented. Motion adaptive method proposed in reference [77] and motion adaptive method proposed by GA-HDTV (available in [51}, [30]) have been implemented as well.

Table 8-5 Deinterlacing algorithms used for performance evaluation.

Algorithm Number	Algorithm Name
	Spatial deinterlacing methods
1	Line repetition (Li_Rep)
2	Bilinear line averaging (Li_Ave)
	Temporal deinterlacing methods
3	Field insertion (Fi_Ins)
4	Bilinear field interpolation (Fi_Ave)
	Hybrid deinterlacing methods
5	VT median filtering
6	Weighted and edge dependant median filtering [60]
	Motion Compensated (MC) deinterlacing methods
7	MC field insertion
8	MC bilinear field interpolation
	Motion Adaptive (MA) deinterlacing methods
9	MA with 3-field (MD), bilinear field interpolation and line averaging (Fi_Ave & Li_Ave)
10	MA with 4-field (MD), field insertion and line averaging (Fi_Ins & Li_Ave)
11	MA method proposed in [77]
12	MA method proposed by GA-HDTV
13	Proposed MA method

8.3 Experimental Results

8.3.1 Objective Evaluation Results

In this section we compare the objective quality of our proposed algorithm with several deinterlacing methods in the literature. For this purpose, the algorithms listed in Table 8-5 are applied to the video sequences mentioned in section 8.2.3. The average PSNR for each video sequence has been calculated using Equation (8-2).

The average PSNRs are shown in Figure 8-3 to Figure 8-21. Figure 8-3 to Figure 8-5 show the average PSNR for Type-1 video sequences which are Grandmom, Mom, and MomDaughter. Only a small portion of each video frame in this group is moving and the rest of the video frame is mainly static. Figure 8-6 shows the average PSNR for these three video sequences in the same plot. This plot shows that temporal methods perform better than spatial methods for this group of video sequences. This result supports the theoretical idea that temporal methods performs better in the absence of motion.

Figure 8-7 shows the average PSNR for all video sequences in this group. It shows that proposed deinterlacing methods outperforms all other methods.

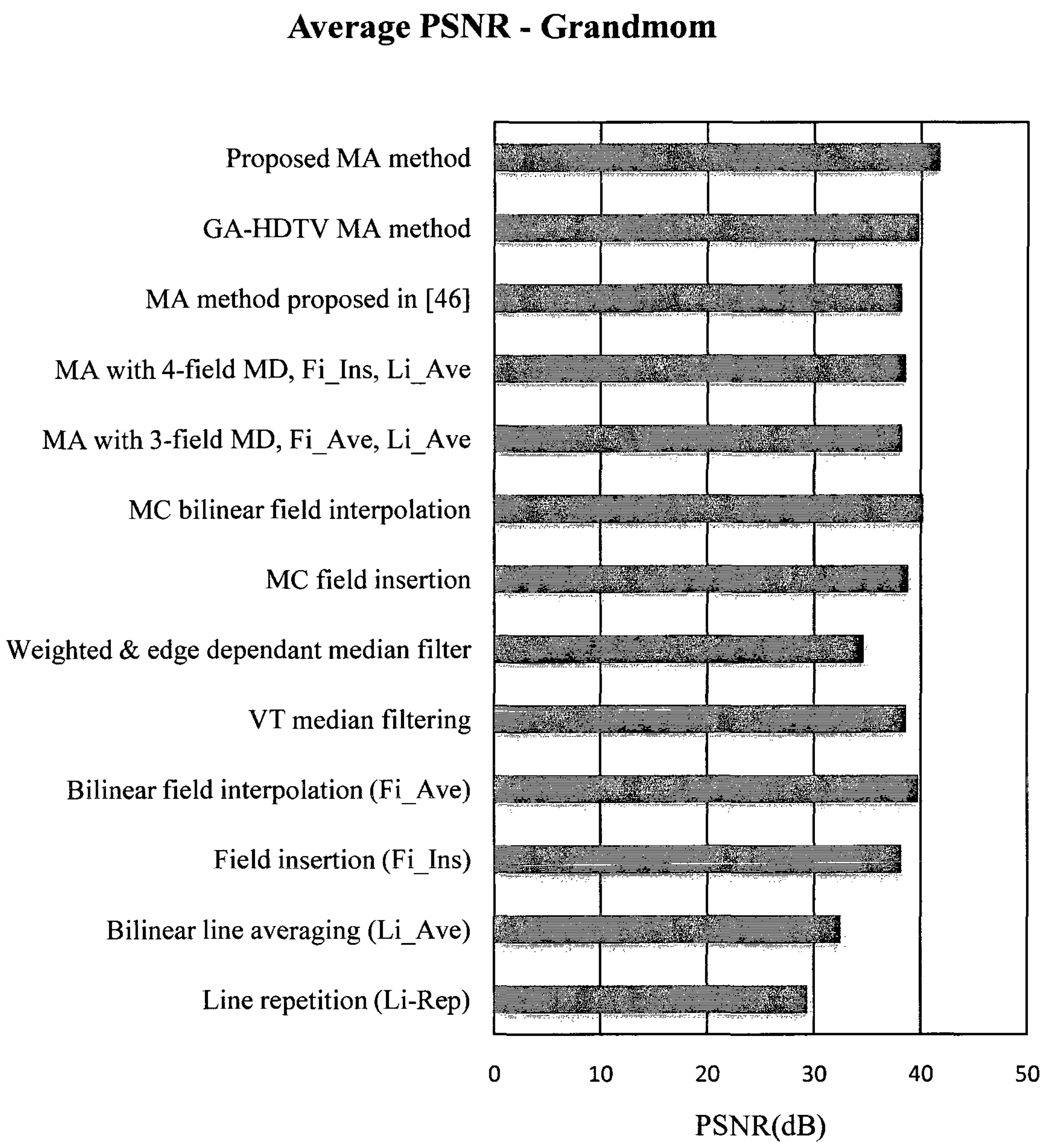


Figure 8-3 Comparing average PSNR values of various deinterlacing methods for Grandmom video sequence.

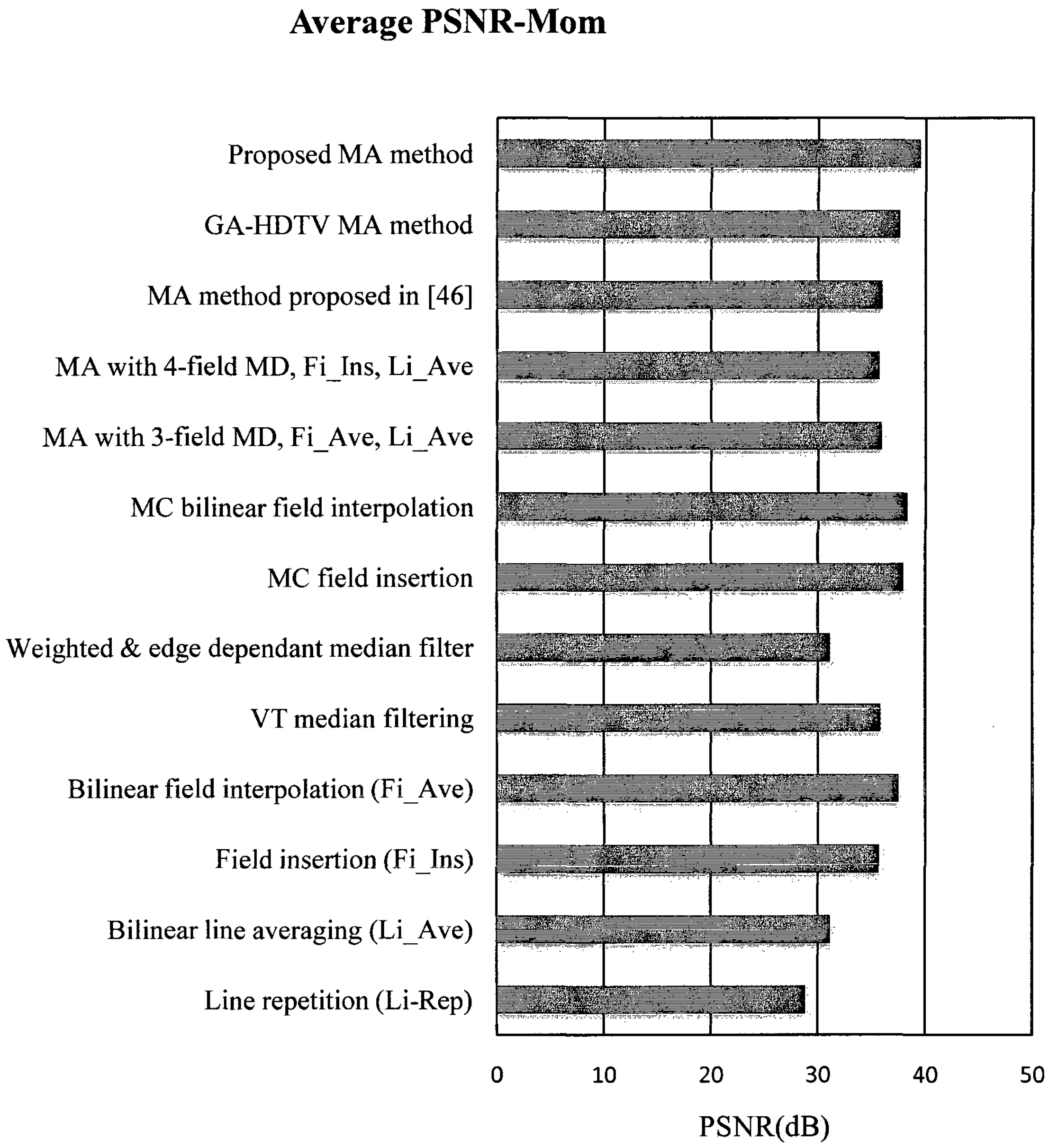


Figure 8-4 Comparing average PSNR values of various deinterlacing methods for Mom video sequence.

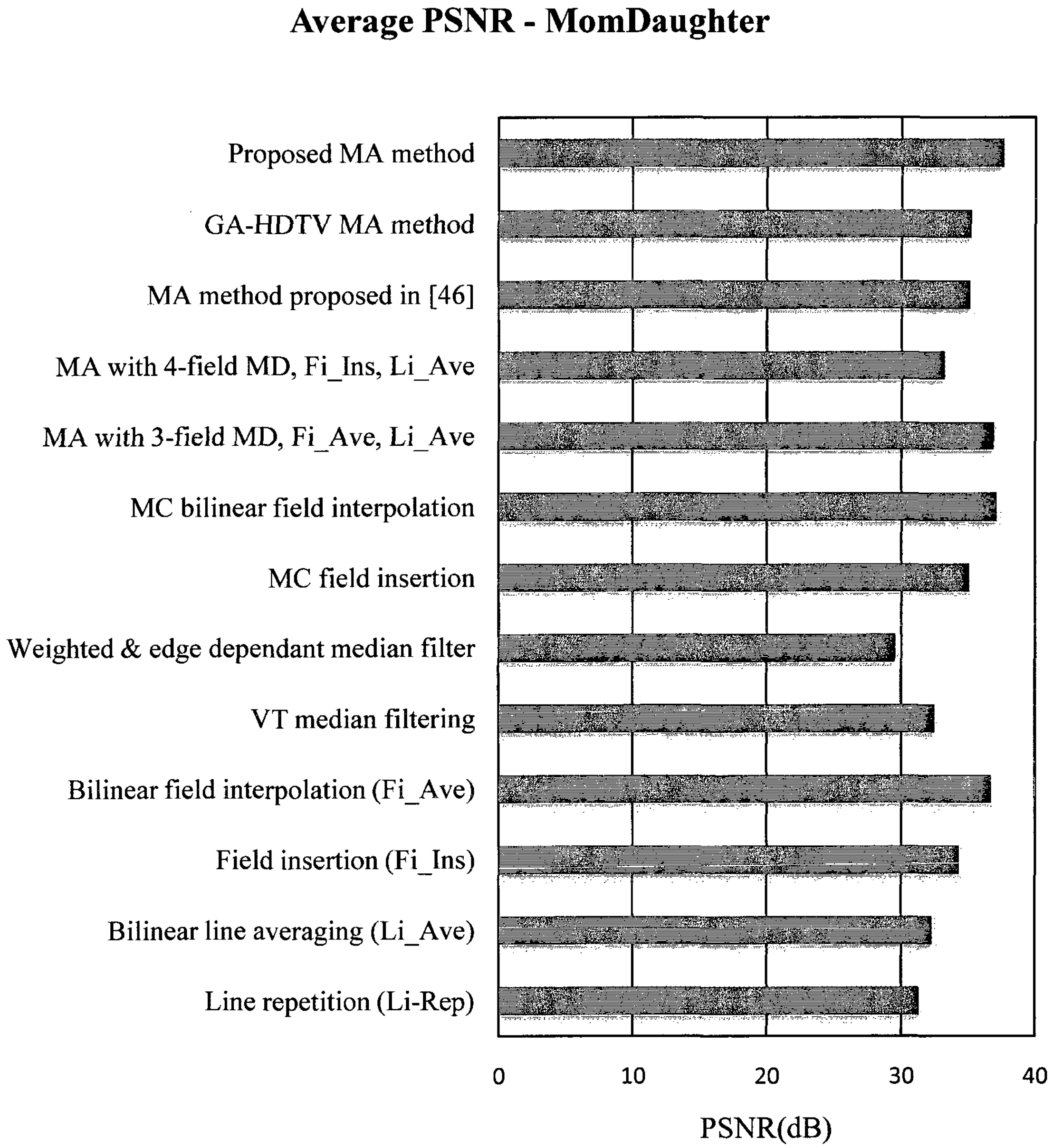


Figure 8-5 Comparing average PSNR values of various deinterlacing methods for MomDaughter video sequence.

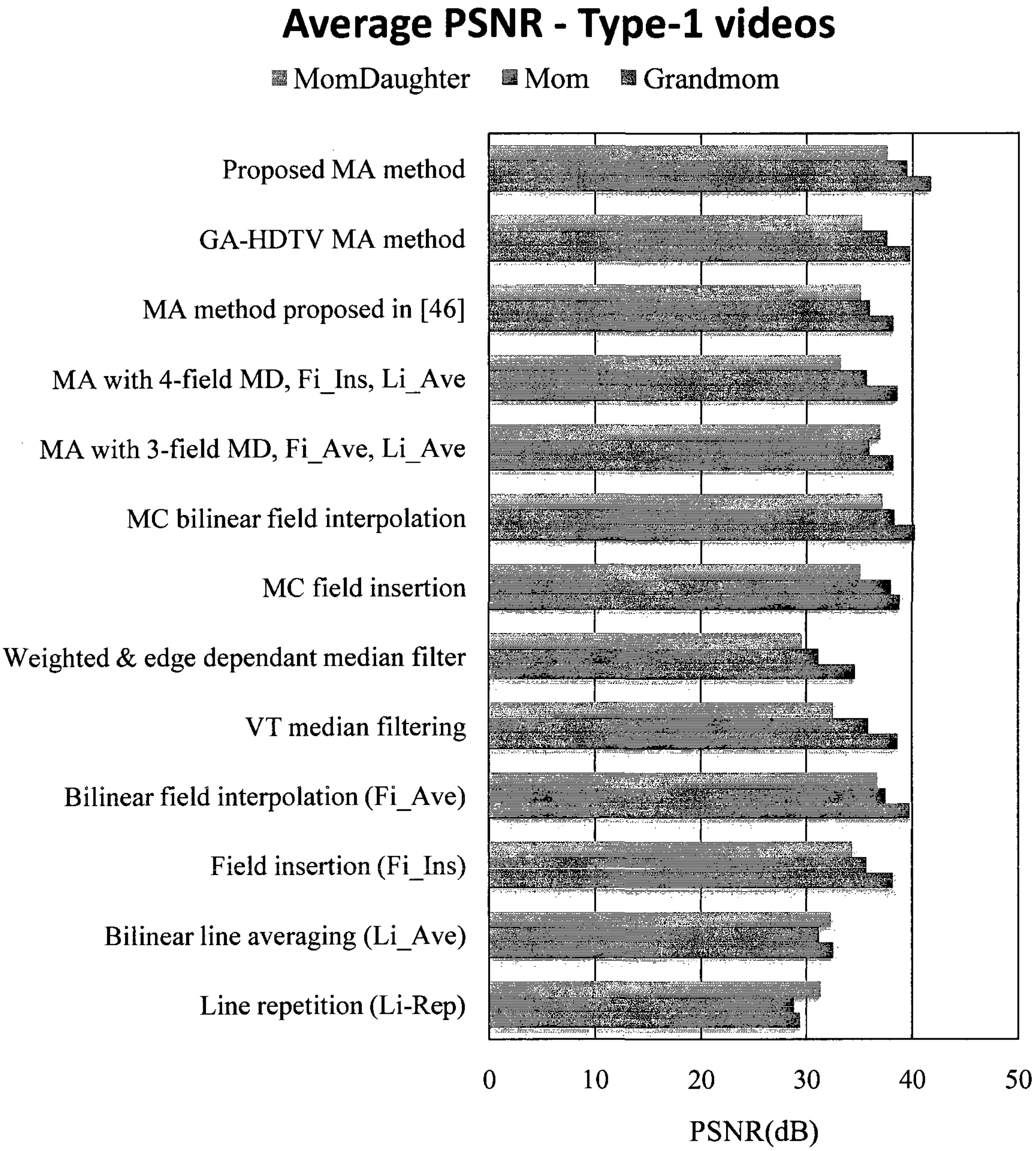


Figure 8-6 Comparing average PSNR values of various deinterlacing methods for Grandmomm, Mom, and MomDaughter video sequences on separate bars.

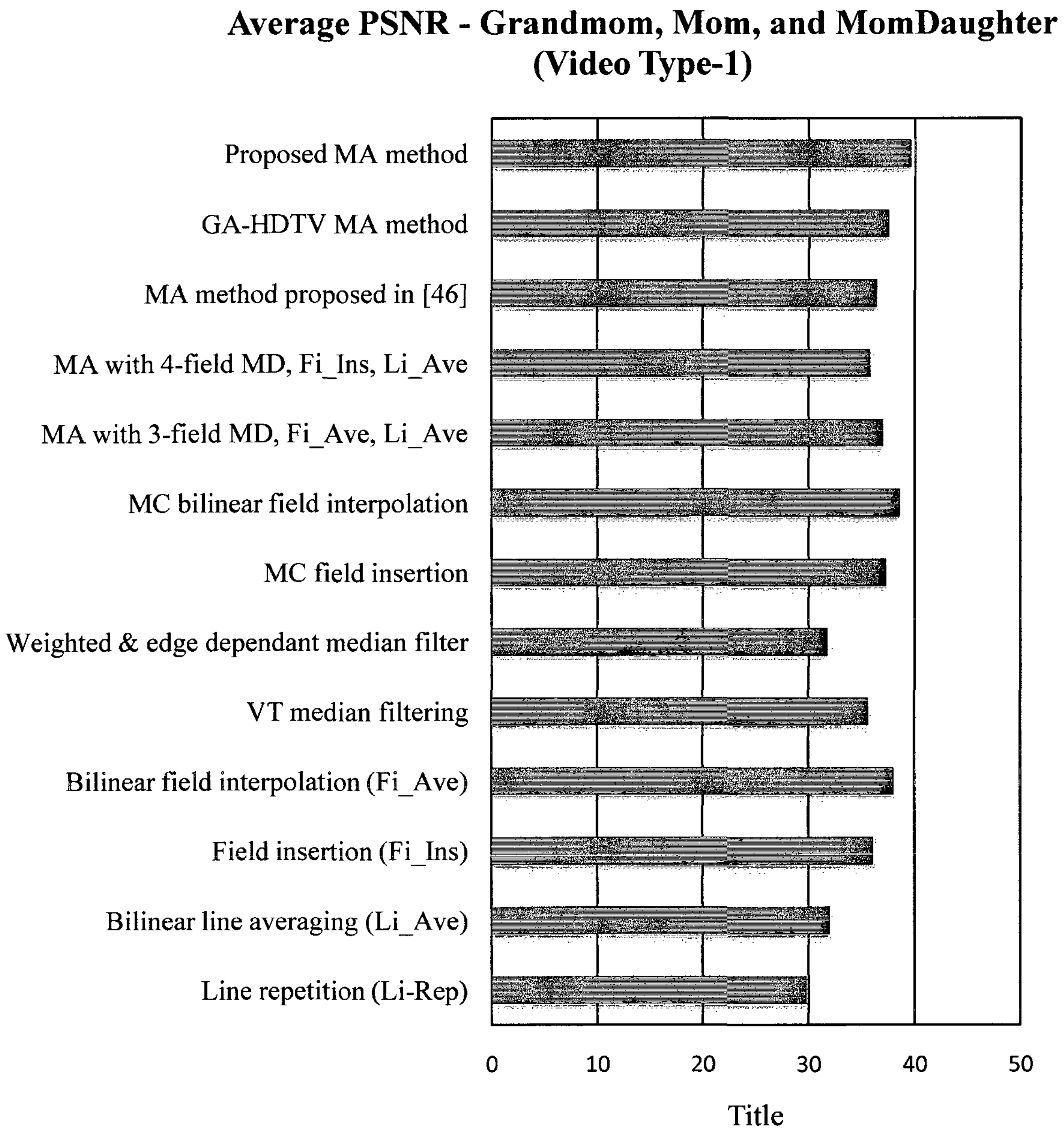


Figure 8-7 Comparing average PSNR values of various deinterlacing methods for video type-1 (Grandmomm, Mom, and MomDaughter) sequences.

Figure 8-8 and Figure 8-9 show the average PSNR for Type-2 video sequences which are Heart and Stennis. These video sequences have fast moving objects with fine details and require high quality deinterlacing results.

Figure 8-8 shows that motion compensated methods works fine for Stennis video sequence. This result is justified by the fact that tracking a moving object and estimating its motion vector could be done with high accuracy in this type of video sequences. However as shown in Figure 8-9 they don't work as good for Heart sequence.

Figure 8-10 shows the average PSNR for these two video sequences in the same plot. It shows that hybrid methods which combine the benefits of both temporal and spatial methods perform better for this group of video sequences. Among all hybrid methods, proposed deinterlacing method outperforms the others. Figure 8-11 confirms this result.

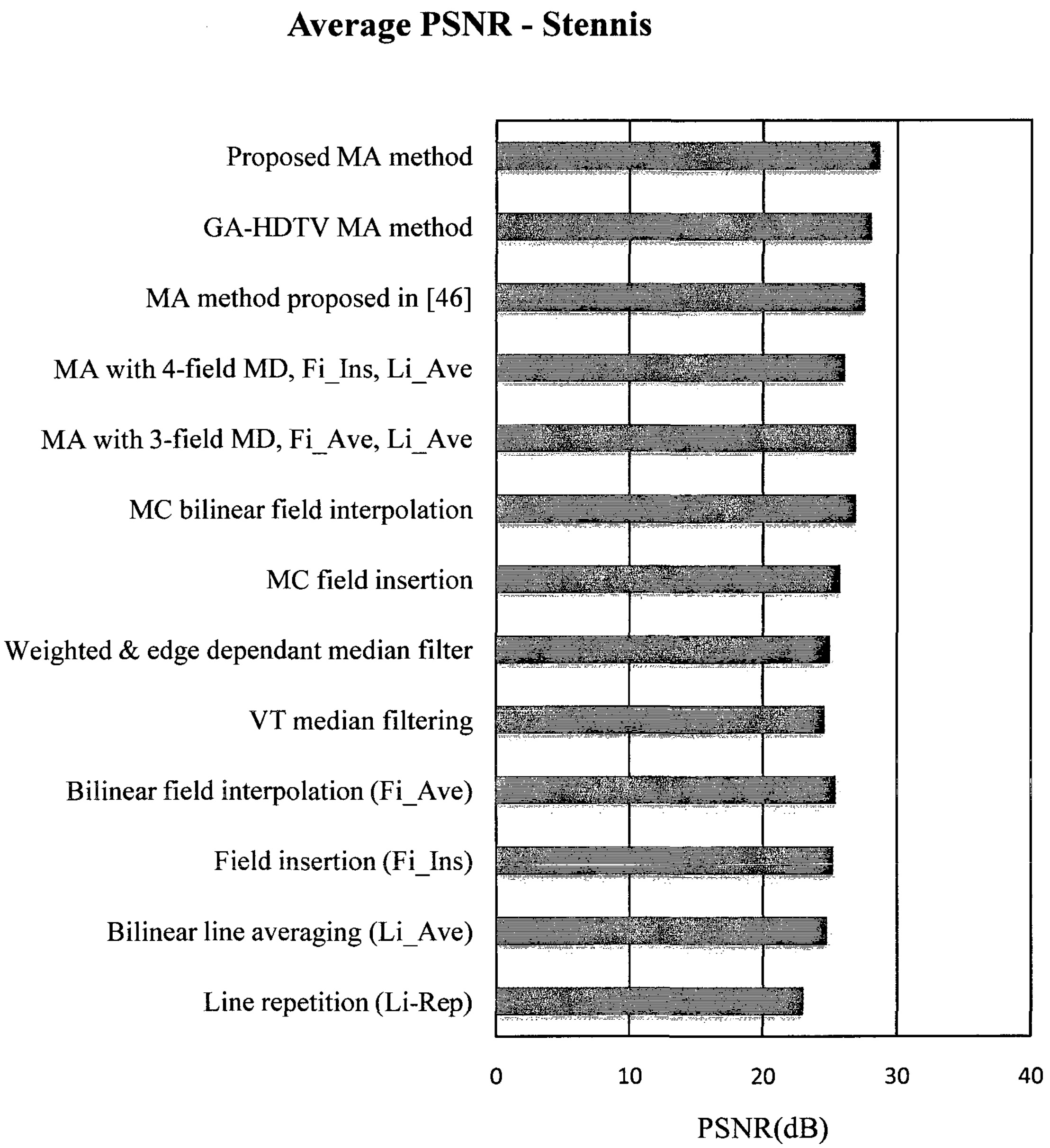


Figure 8-8 Comparing average PSNR values of various deinterlacing methods for Stennis video sequence.

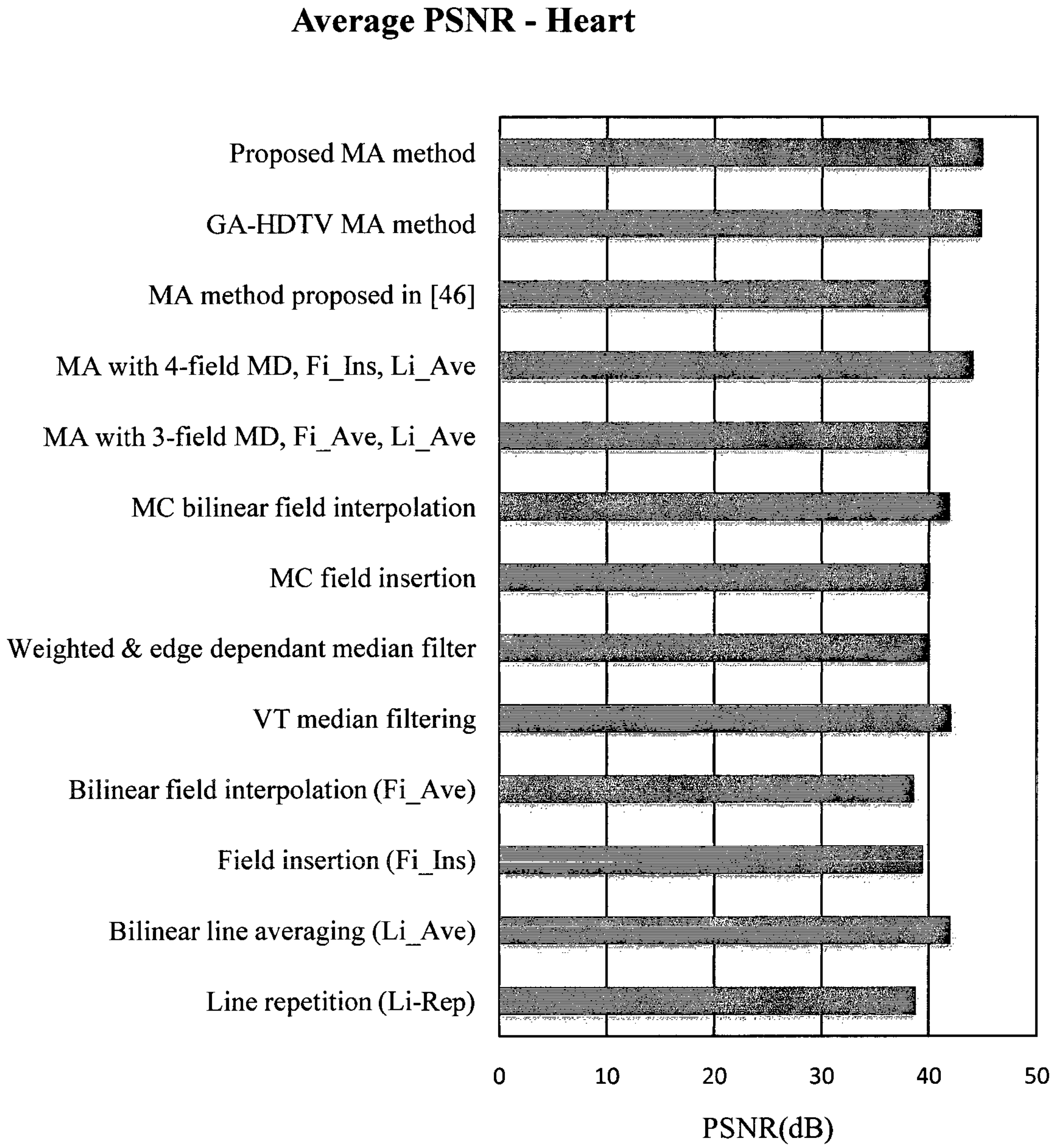


Figure 8-9 Comparing average PSNR values of various deinterlacing methods for Heart video sequence.

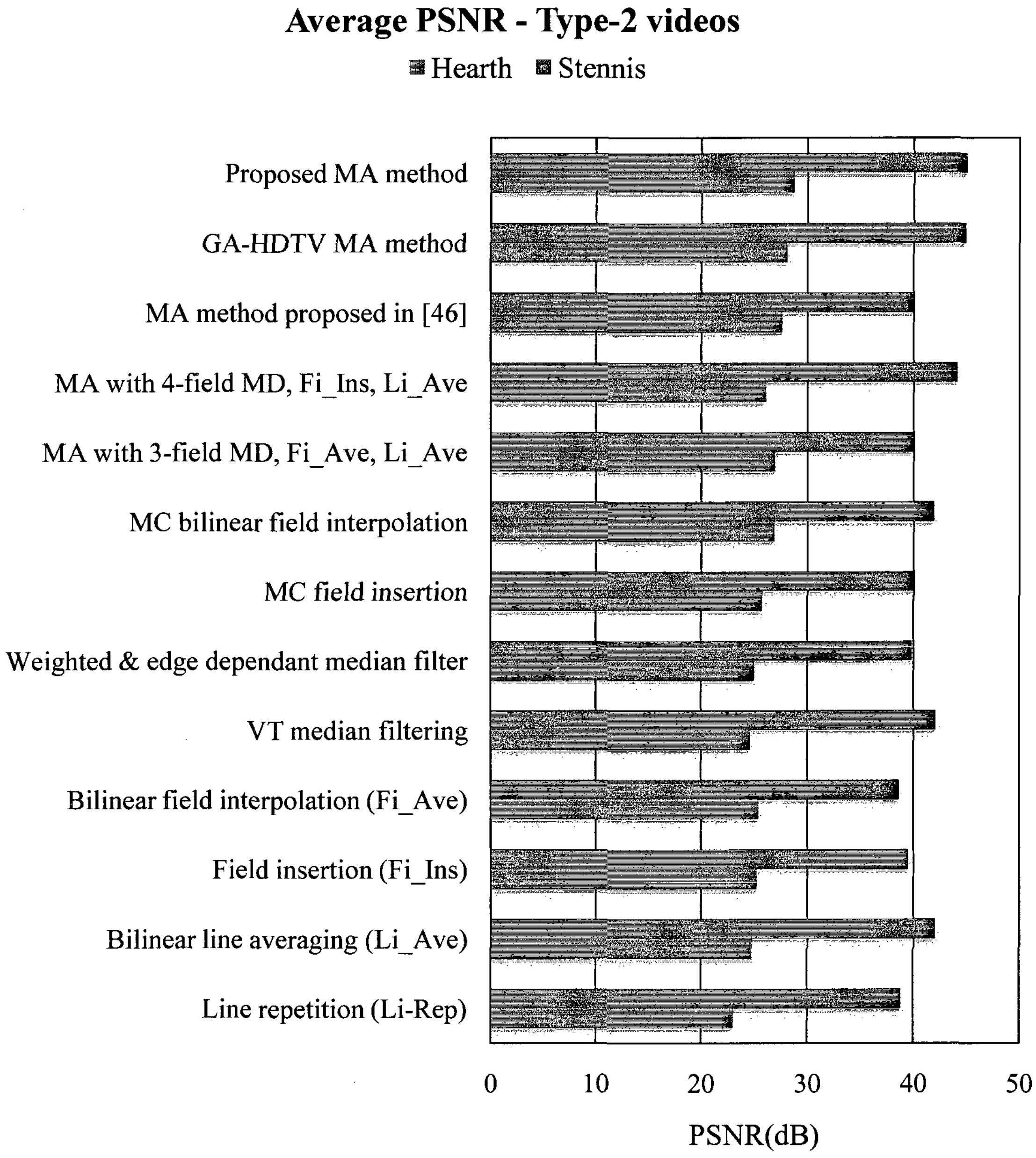


Figure 8-10 Comparing average PSNR values of various deinterlacing methods for Stennis and Heart video sequences combined in one plot.

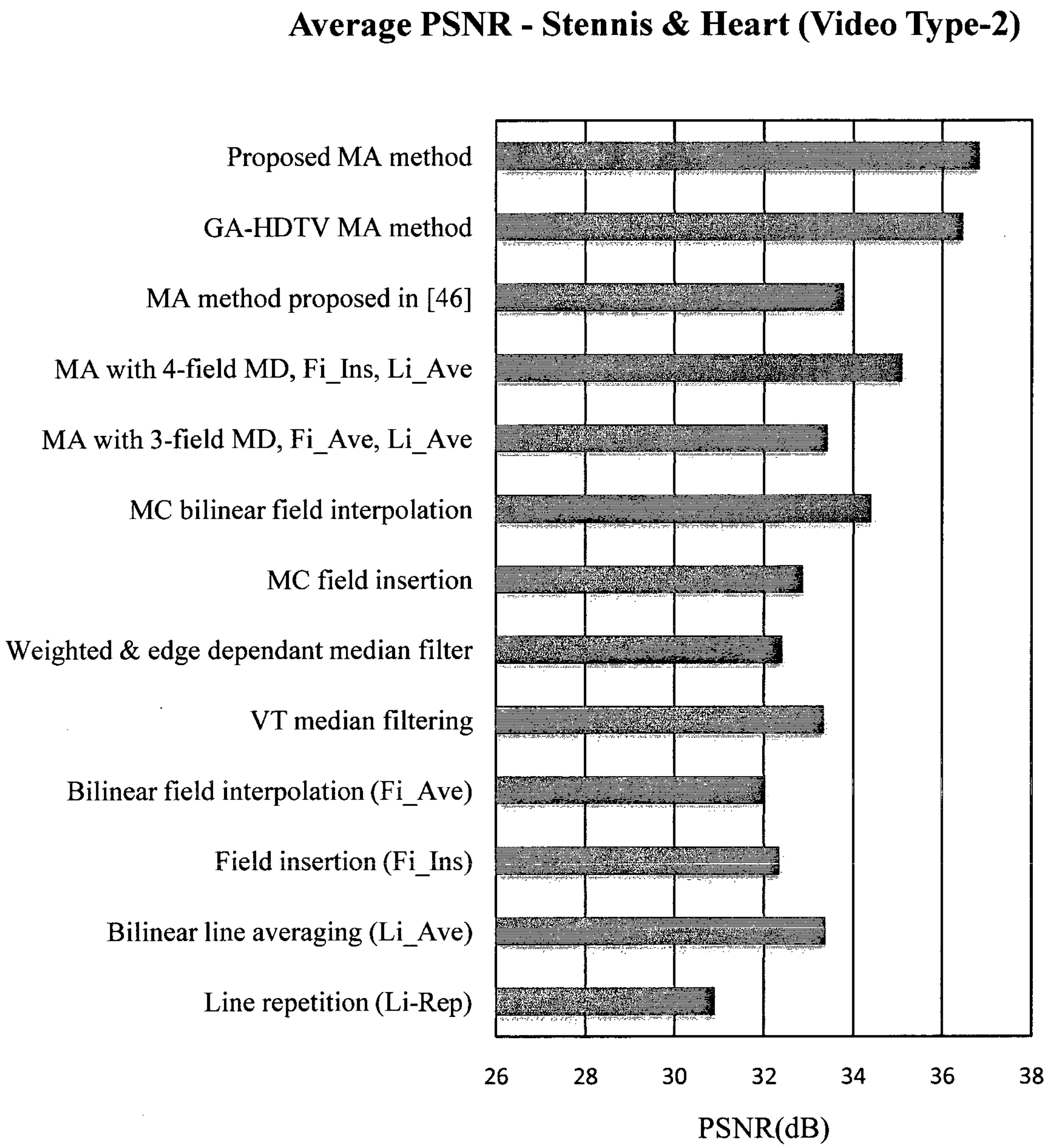


Figure 8-11 Comparing average PSNR values of various deinterlacing methods for video type-2 (Stennis, Heart) video sequences.

Figure 8-12 to Figure 8-14 show the average PSNR for Type-3 video sequences which are Sflowg, Movi, and Disku. These videos are obtained from moving cameras.

Figure 8-12 shows the average PSNR for Sflowg test sequence. Sflowg has been obtained by a panning camera from a flower garden with many fine details. As a result of large number of moving objects, spatial methods perform better than temporal method for this video sequence. However, motion compensated methods may benefit from the type of motion available in this video sequence. It is possible to detect accurate motion vectors for this video sequence; therefore MC field insertion performs better than some other methods. But motion adaptive method proposed by GA-HDTV and the motion adaptive method proposed in this thesis perform better than other methods.

Figure 8-13 shows the average PSNR for Movi test sequence. Movi is a black and white video obtained from a panning camera with fast circular motion around a building, therefore objects in this video are changing with a fast speed between consecutive frames which makes motion estimation hard. Therefore motion compensated methods does not make any improvement in this video. Weighted and edge dependant median filtering and proposed motion adaptive method performs better than other methods for this type of video sequences.

Figure 8-14 shows the average PSNR for Disku test sequence. Disku is a video taken from discussion panel where the video scene contains both zooming and panning. Proposed deinterlacing method performs better than all other methods for this video sequence.

While Figure 8-15 shows the average PSNR of this video type in the same graph; Figure 8-16 shows the average PSNR for these three video sequences in the same plot. These plots show that hybrid methods perform better than spatial or temporal methods for this group of video sequences. Proposed deinterlacing method provides the best deinterlacing results between all implemented methods.

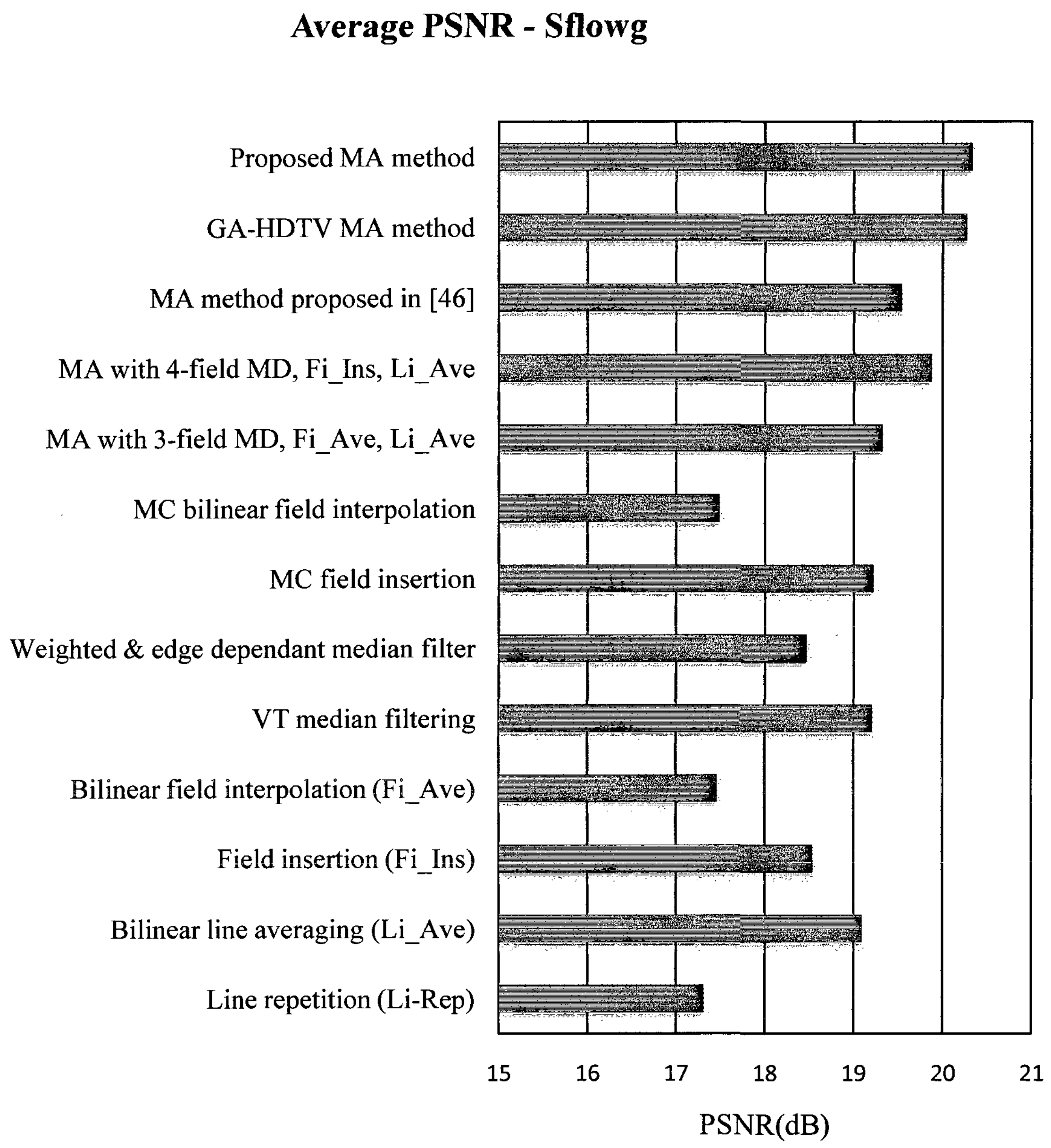


Figure 8-12 Comparing average PSNR values of various deinterlacing methods for Sflowg video sequence.

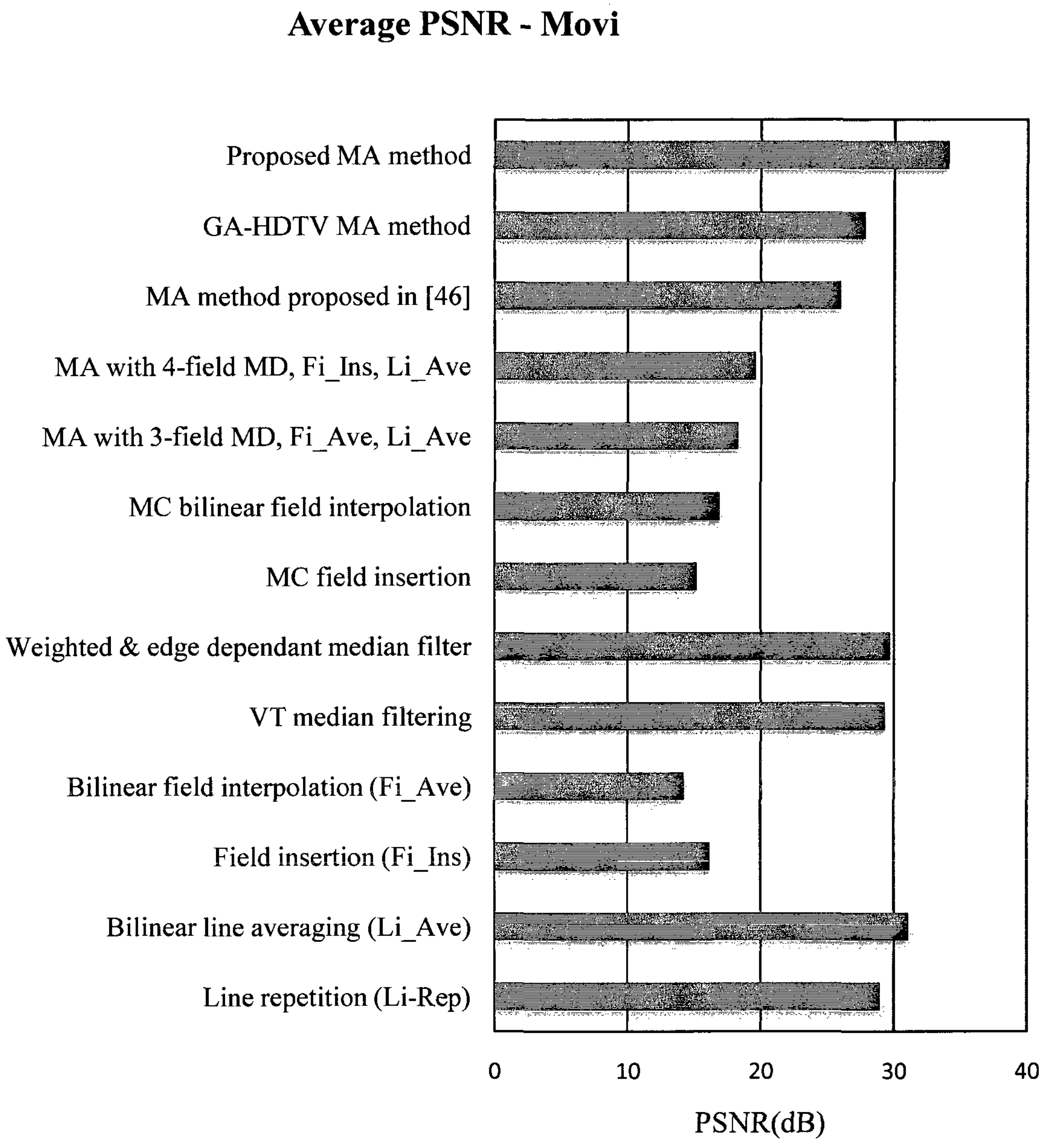


Figure 8-13 Comparing average PSNR values of various deinterlacing methods for Movi video sequence.

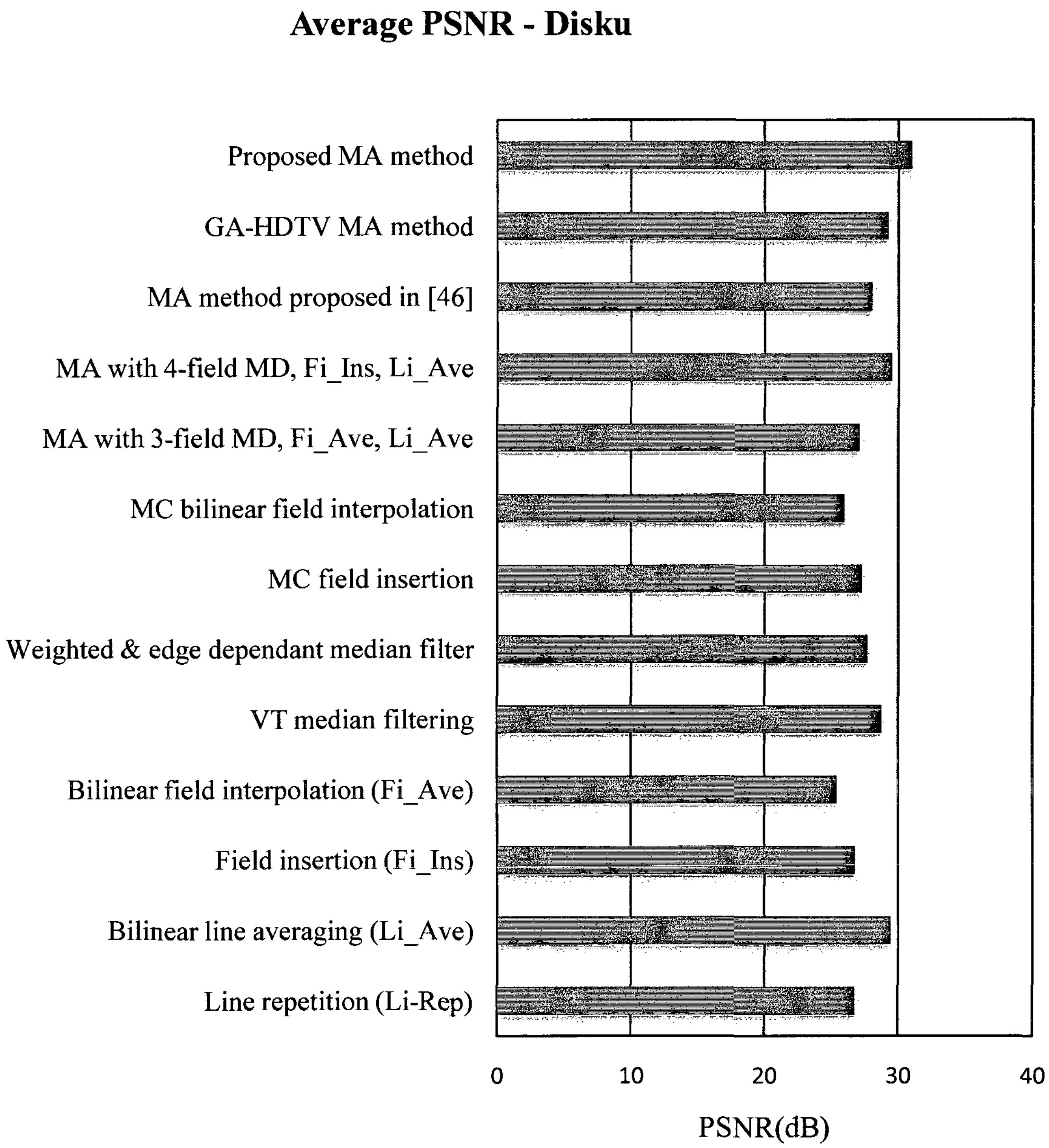


Figure 8-14 Comparing average PSNR values of various deinterlacing methods for Disku video sequence.

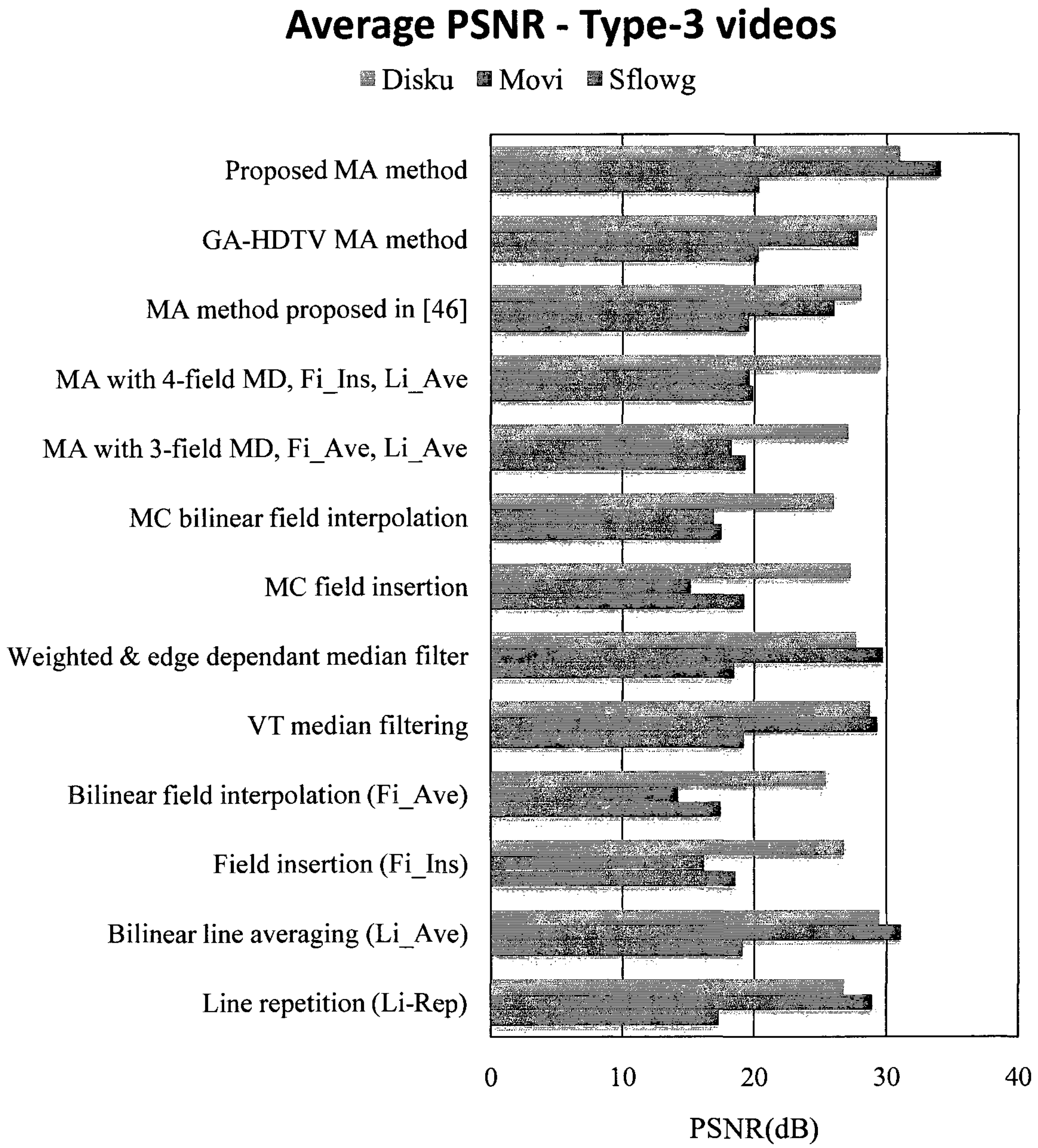


Figure 8-15 Comparing average PSNR values of various deinterlacing methods for Type-3 video sequences; Sflowg, Movi, and Disku.

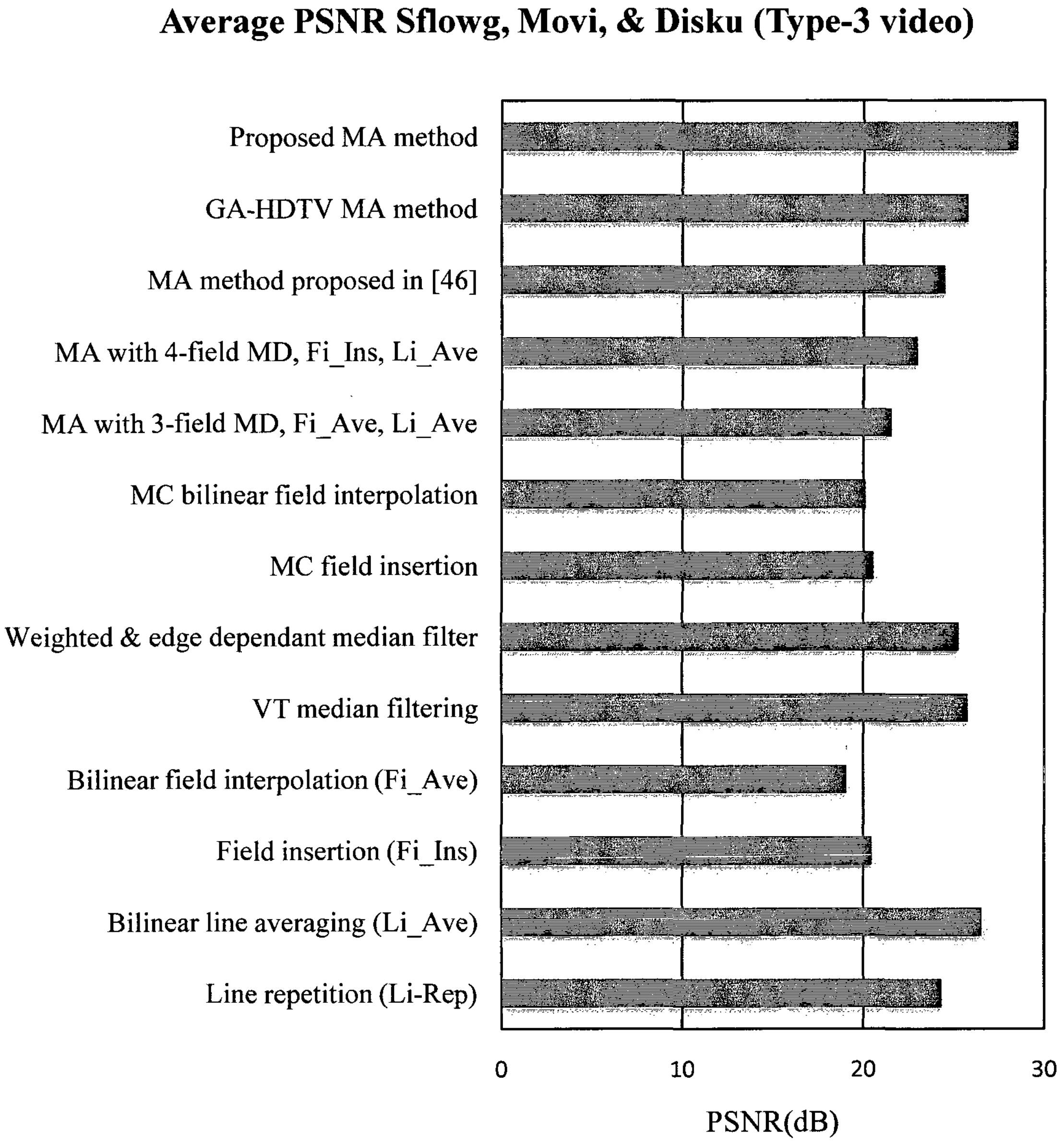


Figure 8-16 Comparing average PSNR values of various deinterlacing methods on Sflowg, Movi, and Disku video sequences.

Figure 8-17 shows average PSNR of all video groups in the same plot. It shows that average PSNR for type-1 videos is generally larger than average PSNR for other video types. This result is justified by the fact than only a small number of pixels are changing in each frame of these video sequences. Therefore even though that change might be a large number by itself the PSNR does not drop significantly.

For example the average PSNR of video type-1 for bilinear field insertion is higher than many other algorithms and close to the value obtained from GA-HDTV proposed method. But the subjective quality of deinterlaced frames doesn't match these results and they are unsatisfactory in moving parts of these videos.

Figure 8-20 show an illustrative explanation of this problem. Figure 8-18 shows five consecutive frame of Mom video where third frame is considered as current frame to be deinterlaced. Figure 8-19 shows the corresponding deinterlaced frame using bilinear field insertion, GA-HDTV proposed method and MA method proposed in this thesis, where all of them shows fairly acceptable quality.

A shown in Figure 8-18, the main motion in this part of video is occurring in the eyes area. Figure 8-20 shows the same frames as Figure 8-19 zoomed in the eyes area. This area is severely corrupted in bilinear field insertion results, while it is much better in GA-HDTV. The best result is achieved by the proposed deinterlacing method.

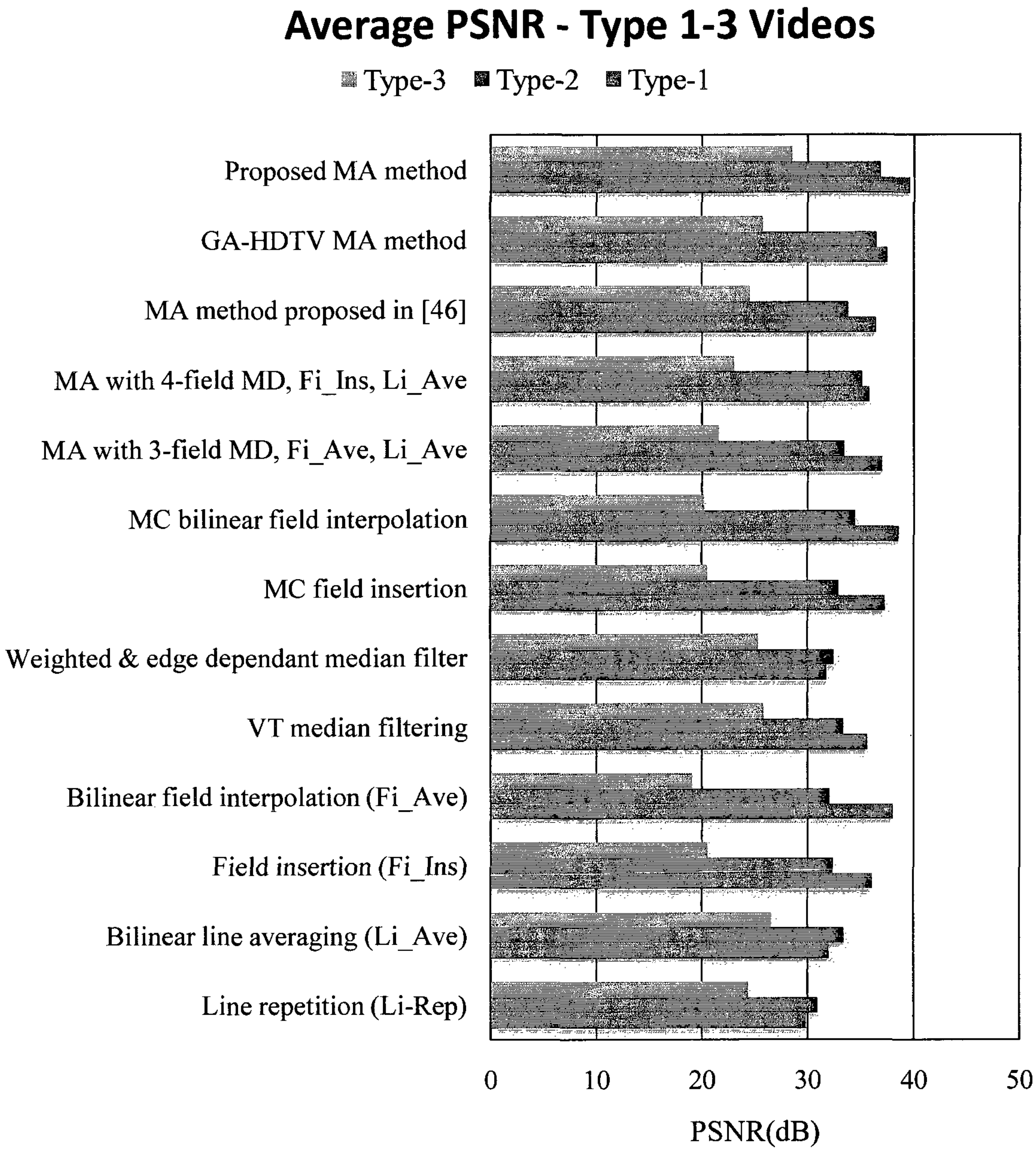
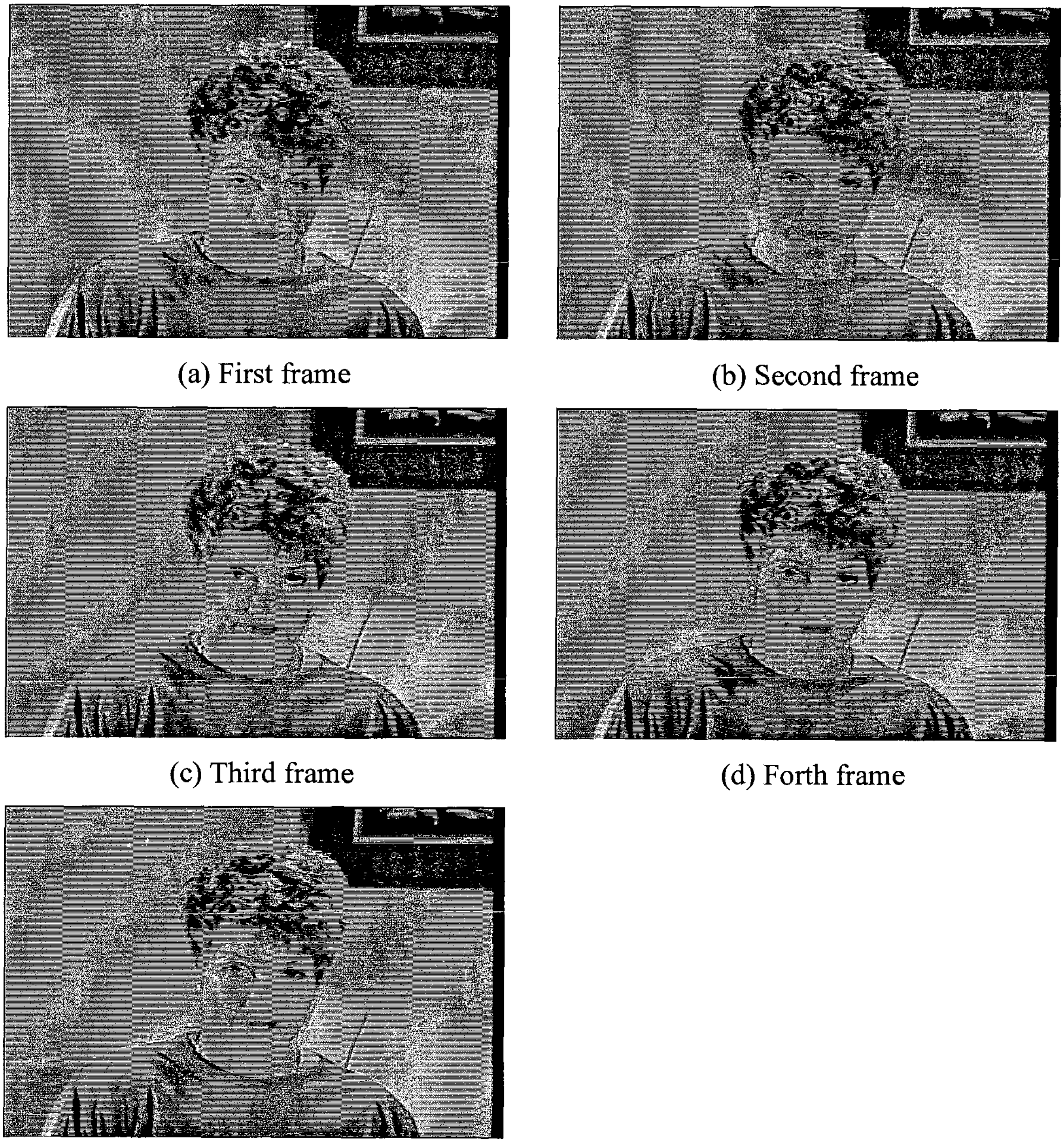


Figure 8-17 Comparing average PSNR values of various deinterlacing methods for all test video sequences;Type-1, Type-2 and Type-3.



(a) First frame

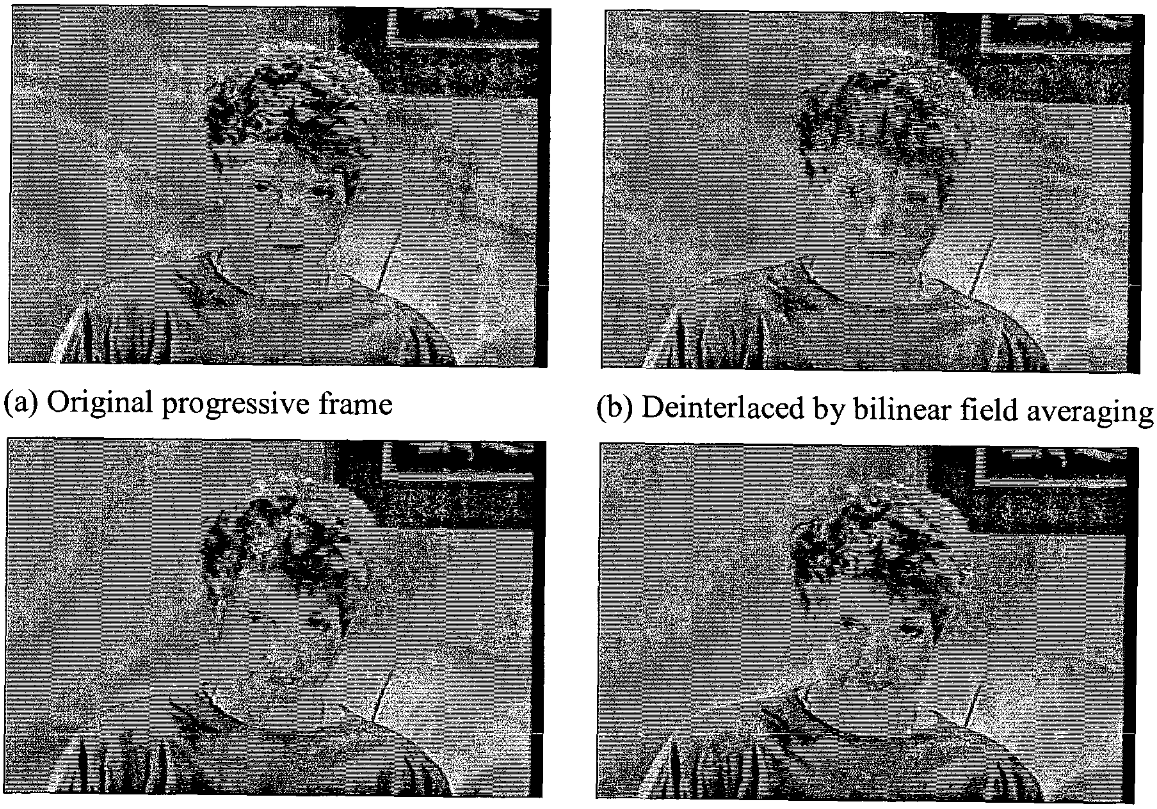
(b) Second frame

(c) Third frame

(d) Forth frame

(d) Fifth frame

Figure 8-18 Five consecutive frame of Mom video sequence, used for motion detection.



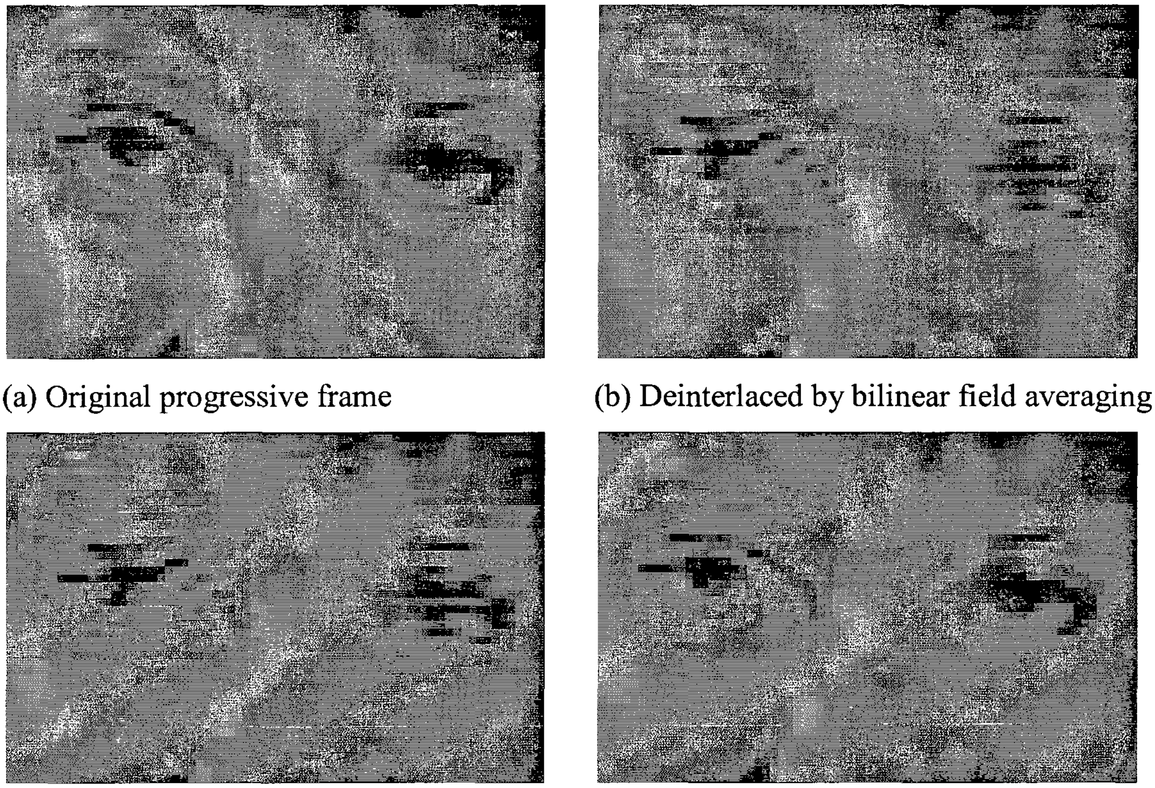
(a) Original progressive frame

(b) Deinterlaced by bilinear field averaging

(c) Deinterlaced by GA-HDTV method

(b) Deinterlaced by proposed MA method

Figure 8-19 Comparing single deinterlaced frame using bilinear field insertion, GA-HDTV method, and proposed deinterlacing method



(a) Original progressive frame

(b) Deinterlaced by bilinear field averaging

(c) Deinterlaced by GA-HDTV method

(b) Deinterlaced by proposed MA method

Figure 8-20 Scaled version of Figure 8-19 enlarged in a moving area.

As a final observation Figure 8-21 shows the average PSNR of all video sequences for different deinterlacing methods. It shows that proposed deinterlacing method outperforms all other implemented methods.

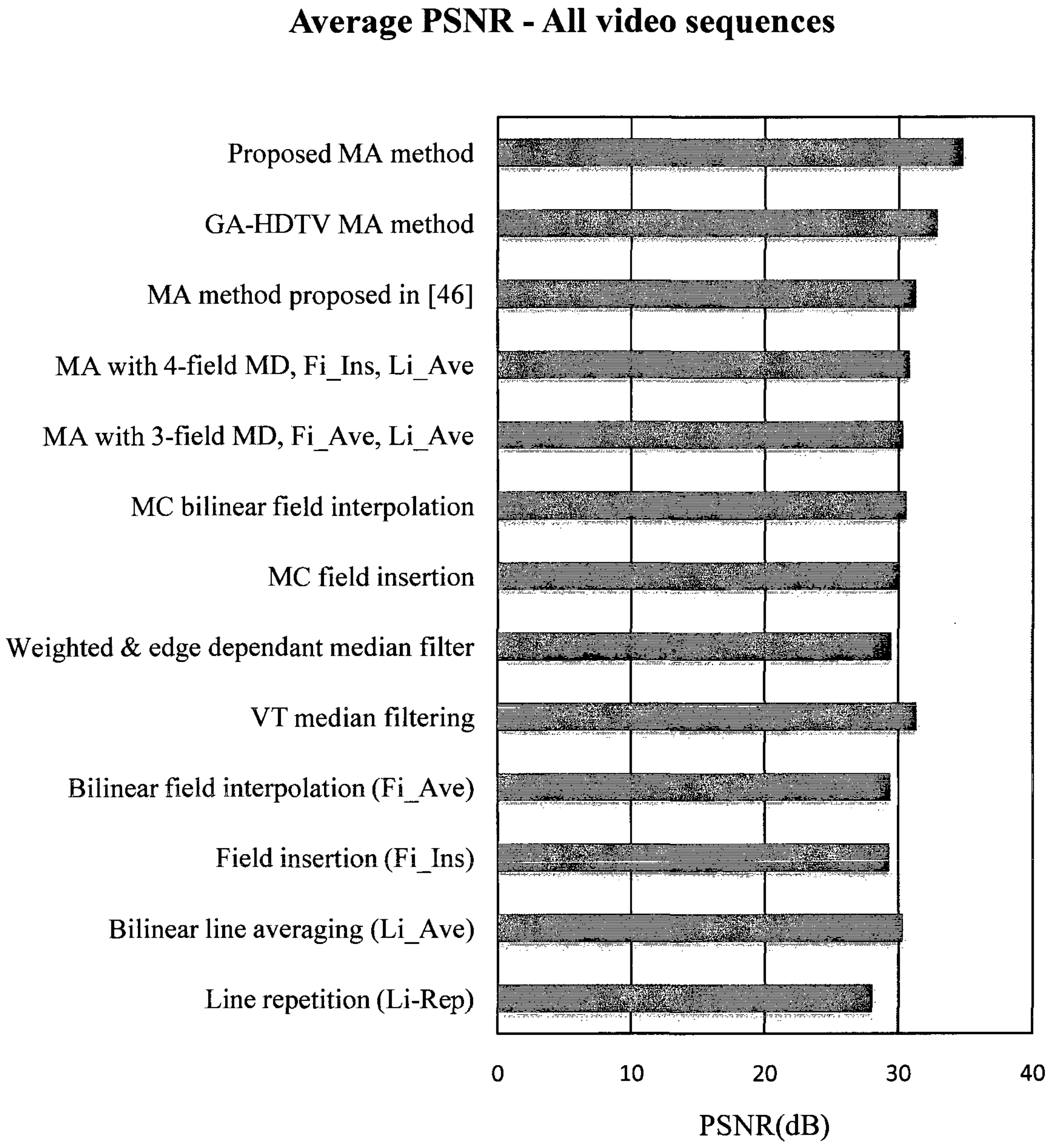


Figure 8-21 Comparing average PSNR of all video sequences for various methods.

8.3.1.1 Performance Improvement

As shown in Figure 8-21 proposed deinterlacing method outperforms all other implemented methods. Figure 8-22 shows the performance improvement obtained from proposed deinterlacing method over other implemented methods. Figure 8-23 shows the performance improvement for each video type.

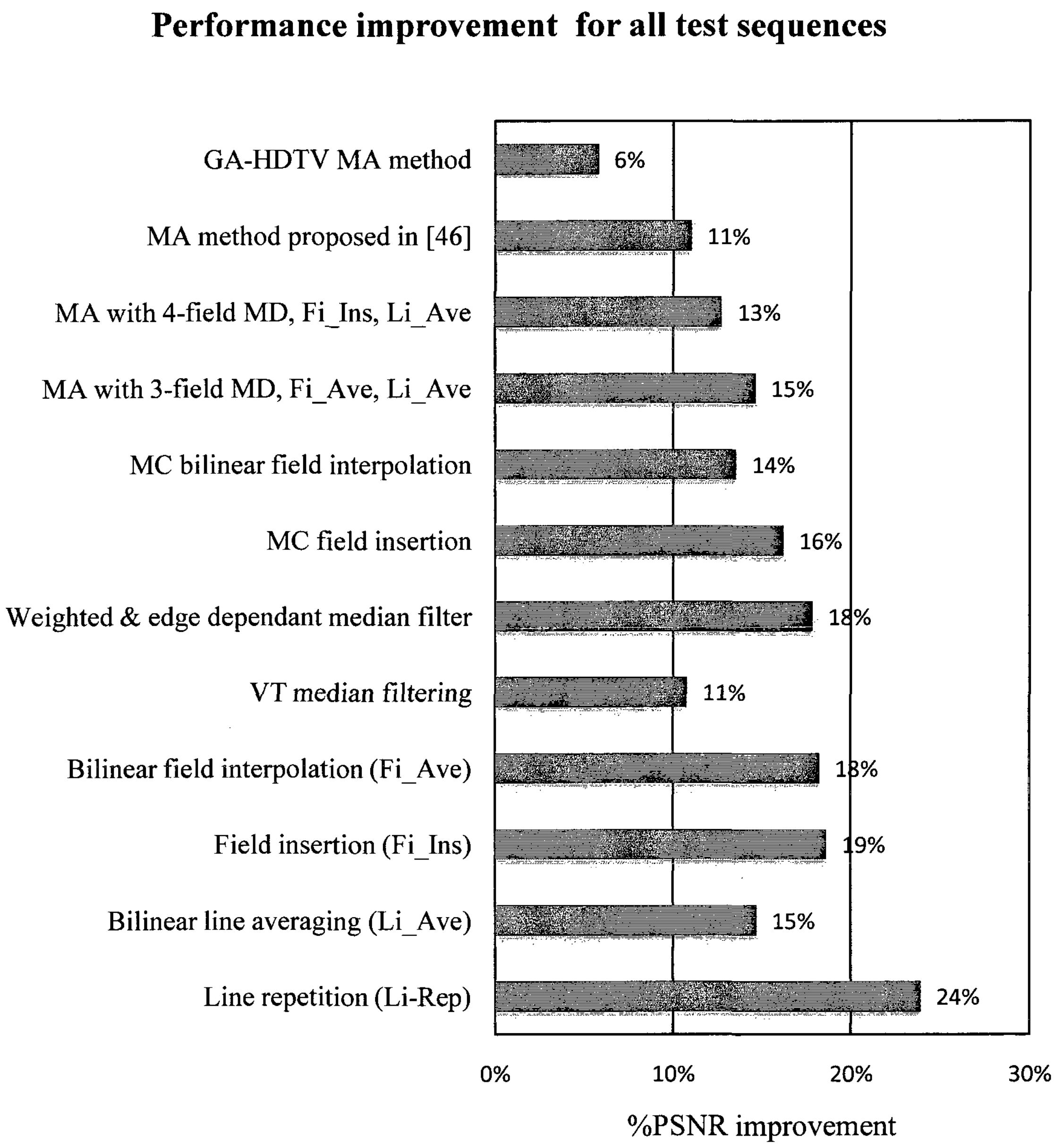


Figure 8-22 Percentage PSNR improvement compared to the proposed deinterlacing method over all test sequences.

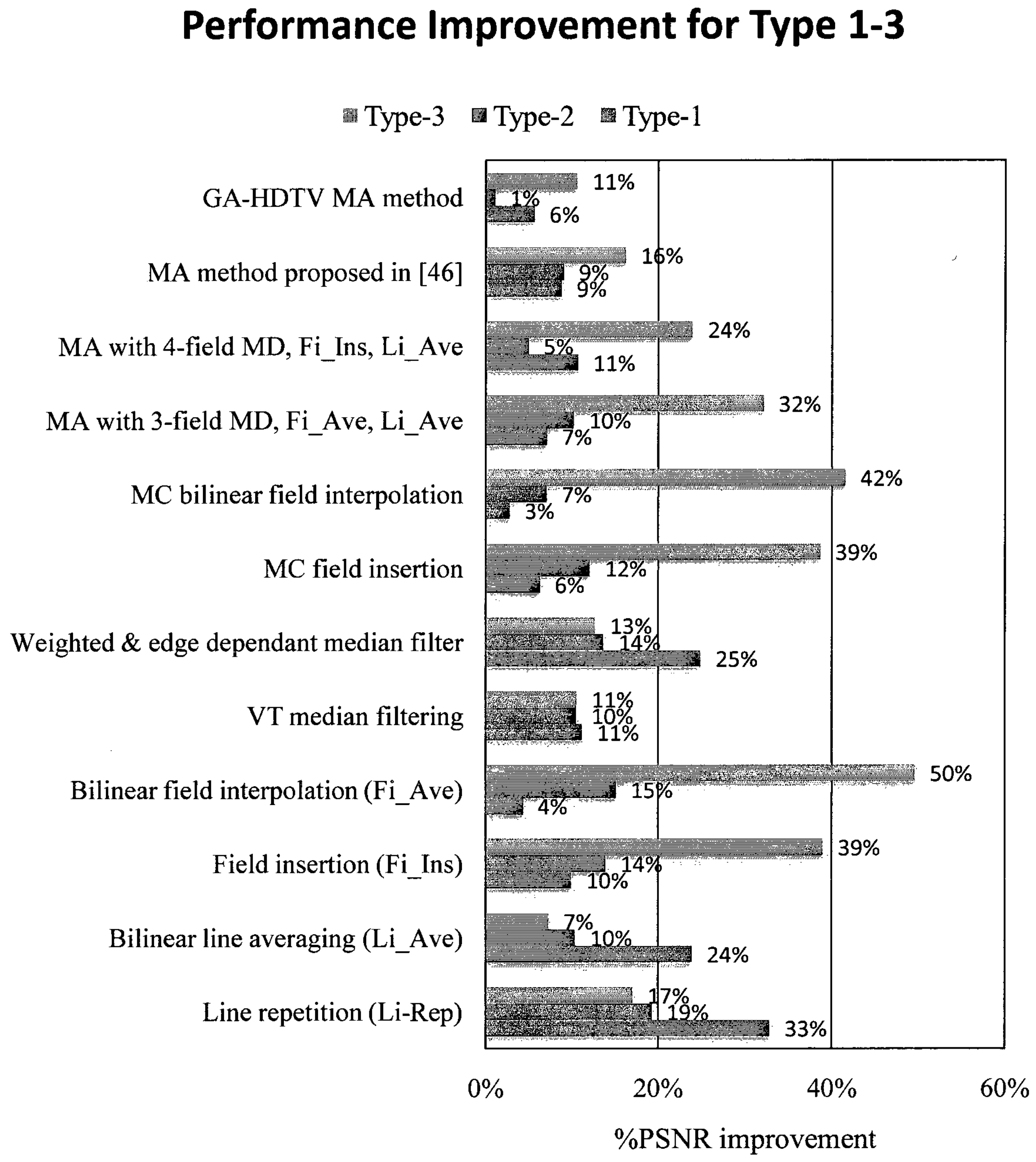


Figure 8-23 Percentage PSNR improvement compared to the proposed deinterlacing method for each video type.

8.3.1.2 Performance Robustness

The performance fluctuation of deinterlacing methods versus frame rate has been studied in this section. Stennis video sequence which has fine details with a wide range of motion speed has been chosen for this objective.

As the original Stennis video sequence is 60 frames per second (fps), it has been down-sampled by two to generate a video sequence with 30 fps. It has also been down-sampled by four to make its corresponding 15fps video sequence. The average PSNR values of Stennis 60fps, Stennis 30fps, and Stennis 15fps for deinterlacing methods listed in Table 8-5 has been calculated. Figure 8-24 shows these values. Figure 8-25. shows the percentage of performance decrease versus frame rate for each deinterlacing method.

As shown in Figure 8-25, the performance of spatial methods does not change with frame rate, which is an expected result. As spatial methods do not use any temporal data for deinterlacing, their performance is independent of frame rate.

The performance of temporal methods drops sharply with frame rate. This result is justified by the fact that they only use temporal data for interpolation; time correlation between consecutive frames drop by time.

The performance of hybrid methods is less sensitive to frame rate. However different methods have different level of sensitivity. Figure 8-25 shows that the proposed deinterlacing method has the minimum performance fluctuation versus frame rate. Thus it is more robust than other implemented methods.

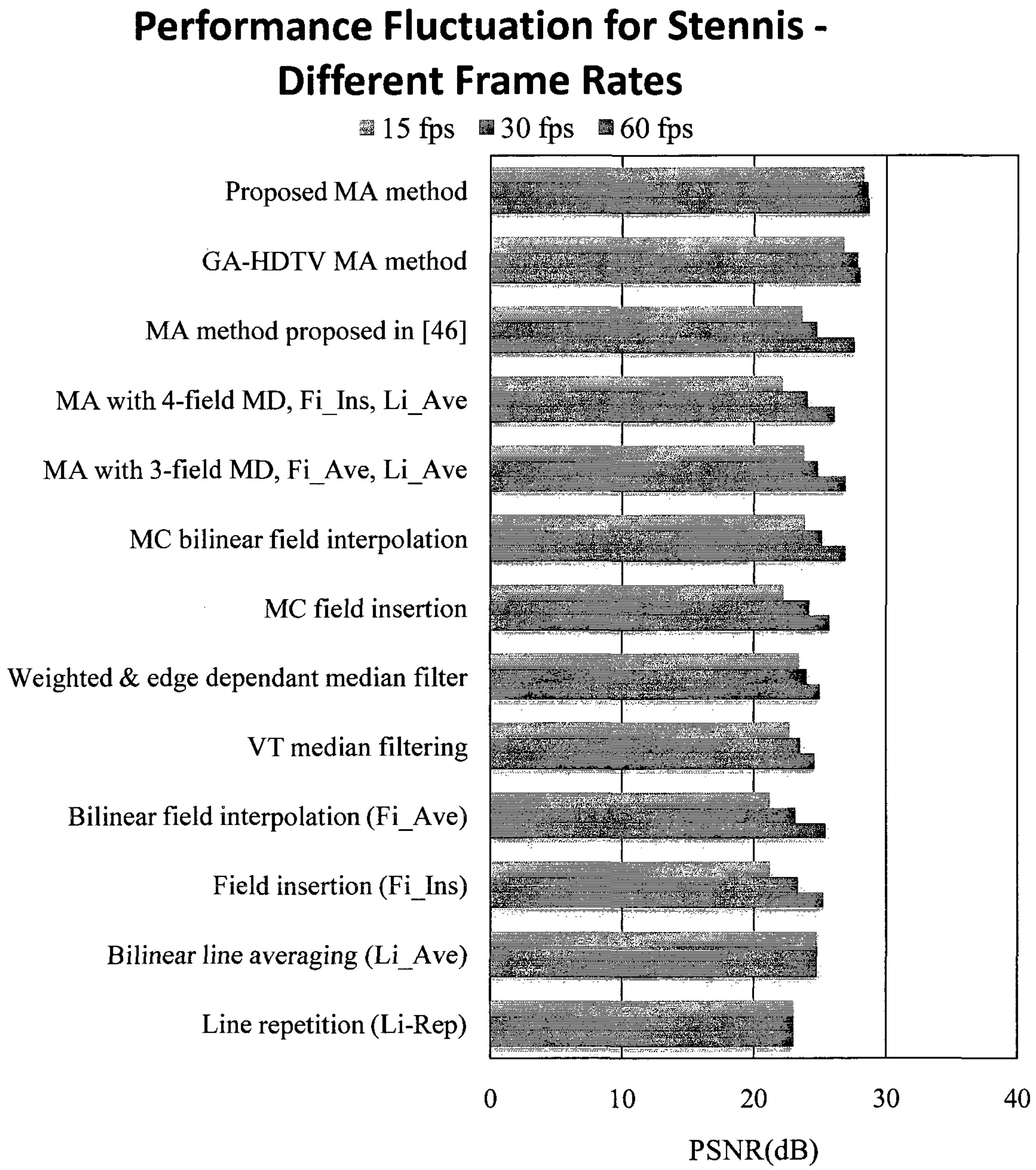


Figure 8-24 Comparing average PSNR versus frame rate. Average PSNR of each method has been calculated for Stennis video sequence on 60fps, 30fps, and 15fps.

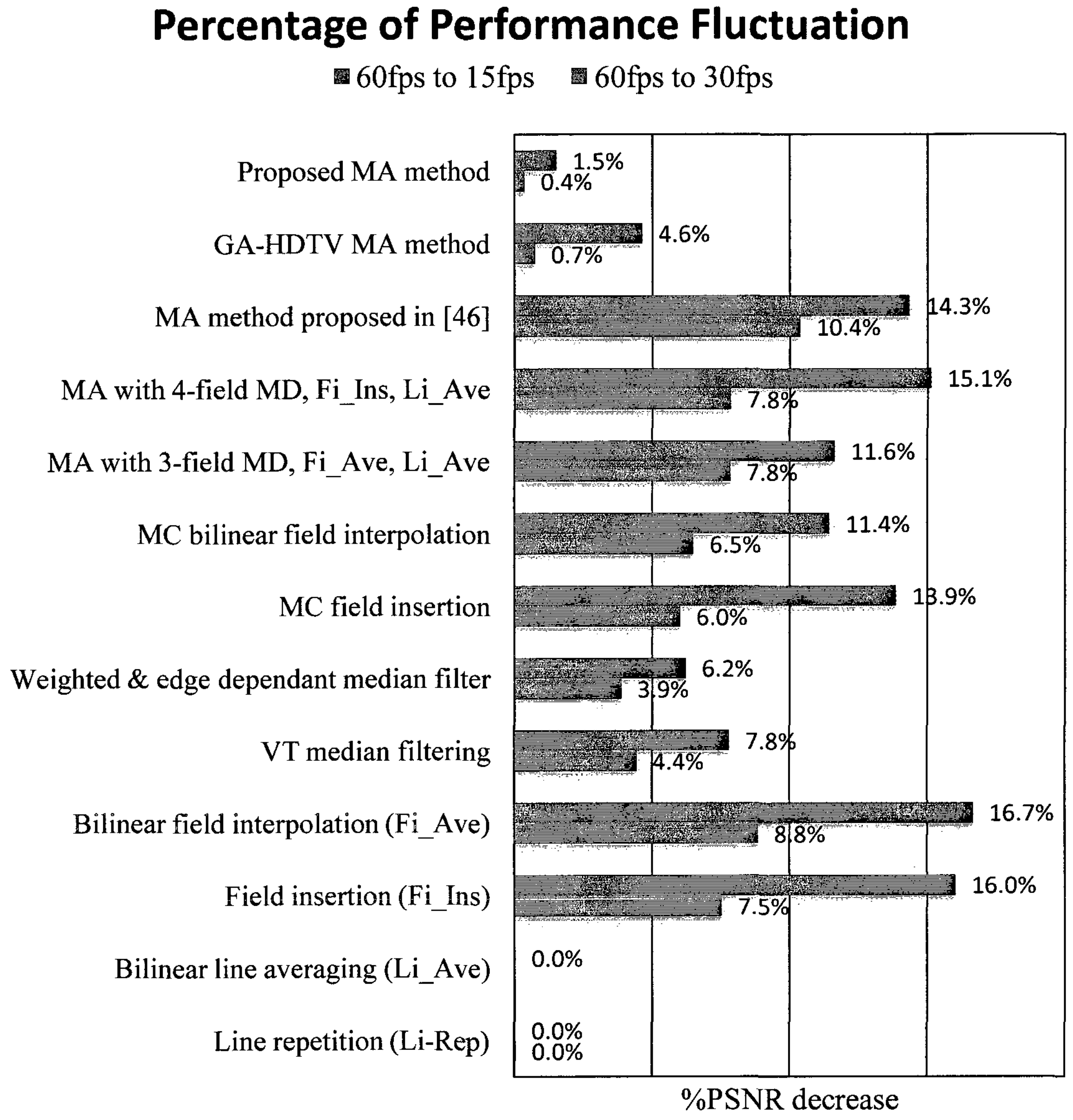


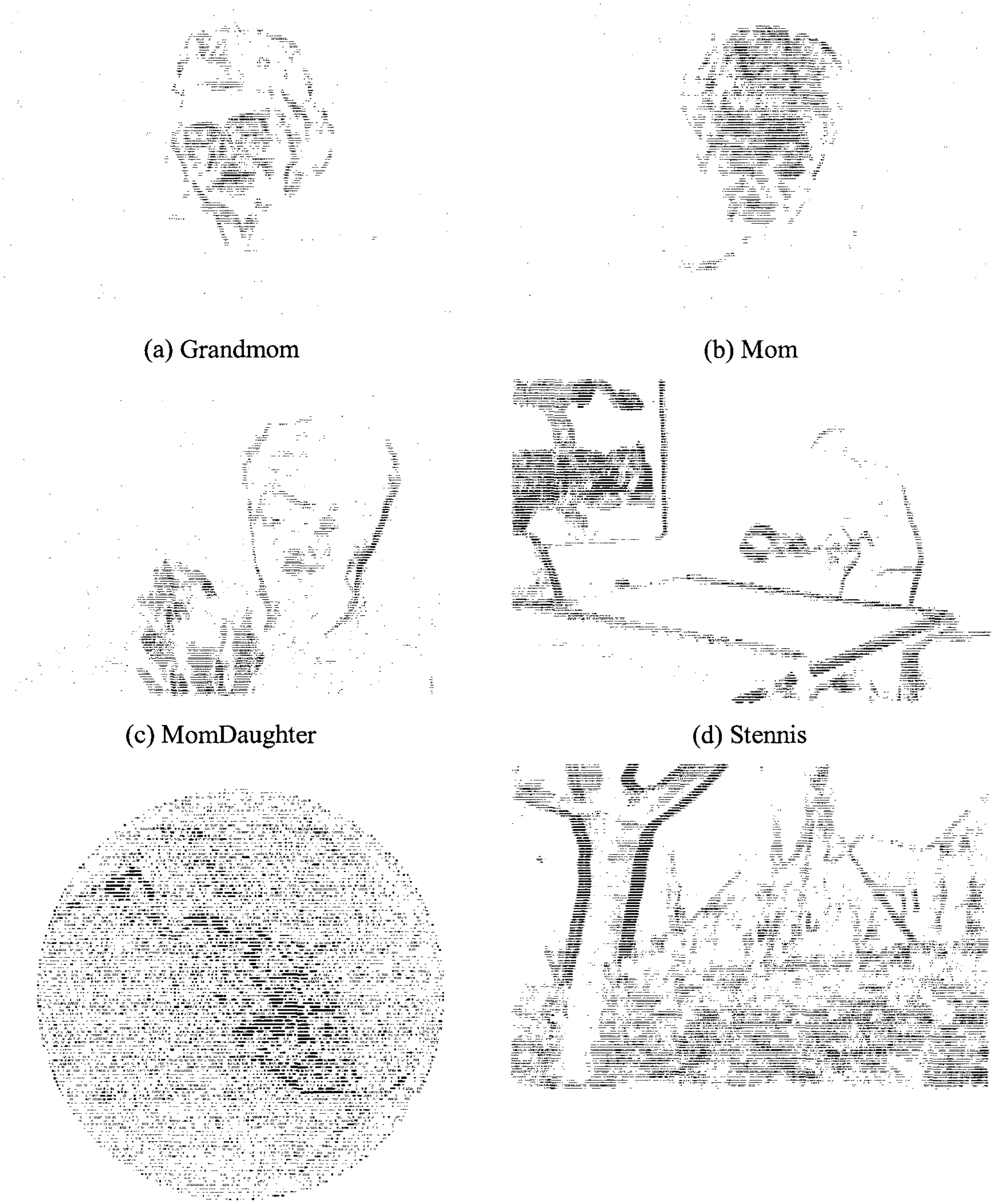
Figure 8-25 Comparing percentage of performance fluctuation versus frame rate. Percentage of PSNR fluctuation has been calculated for Stennis video sequence on 60fps, 30fps, and 15fps

8.3.1.3 Evaluation of Proposed Motion Detection

Figure 8-26 shows a sample motion detection results for each of the test sequences. It shows that the proposed motion detection result is accurate. Furthermore, it shows that the proposed motion detection method is capable of detecting a wide range of motions; from slow motion present in Grandmom video sequence to very fast motion available in Stennis and Movi sequences.

Contribution of the proposed motion detection method in improving the overall deinterlacing performance is analyzed in this section. For this purpose, the interpolation method proposed in this thesis has been combined with other motion detection methods studied in Chapter 5.

Figure 8-27 shows the average PSNR of proposed interpolation method using 2-field motion detection, 3-field motion detection, 4-field motion detection and GA-HDTV proposed motion detection method compared with average PSNR using proposed motion detection method. Figure 8-28 compares the percentage of performance improvement gained using proposed motion detection method over other motion detection methods.



(a) Grandmom

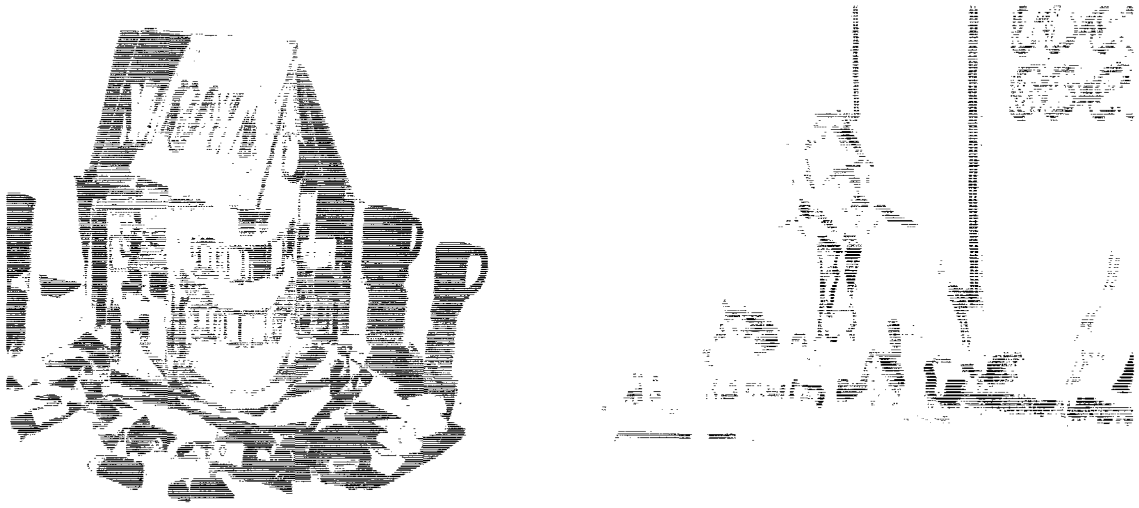
(b) Mom

(c) MomDaughter

(d) Stennis

(e) Heart

(f) Sflowg



(g) Movi

(h) Disku

Figure 8-26 Proposed motion detection results for a single field of each test sequence.

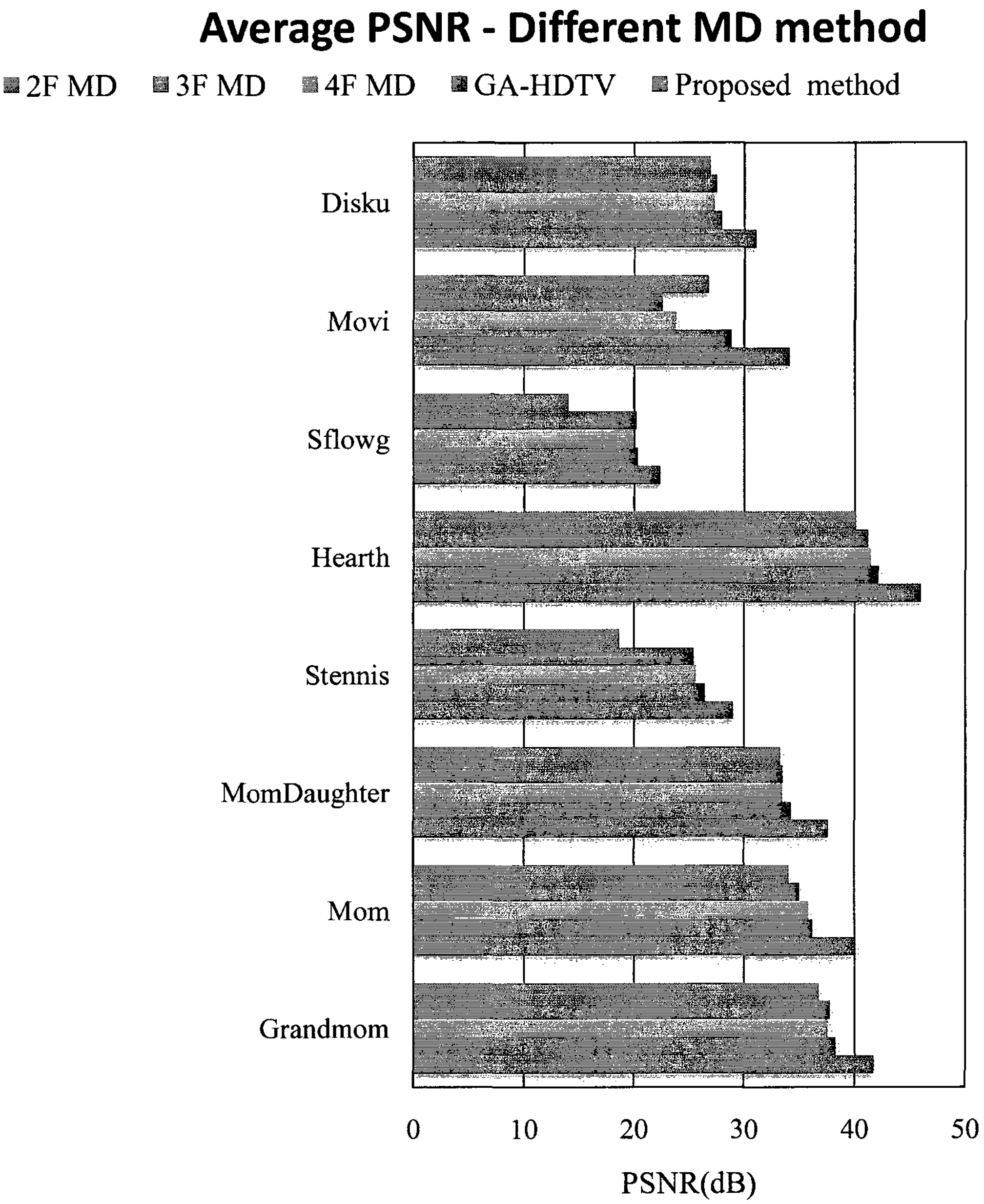


Figure 8-27 Comparing average PSNR of proposed interpolation method by different motion detection methods.

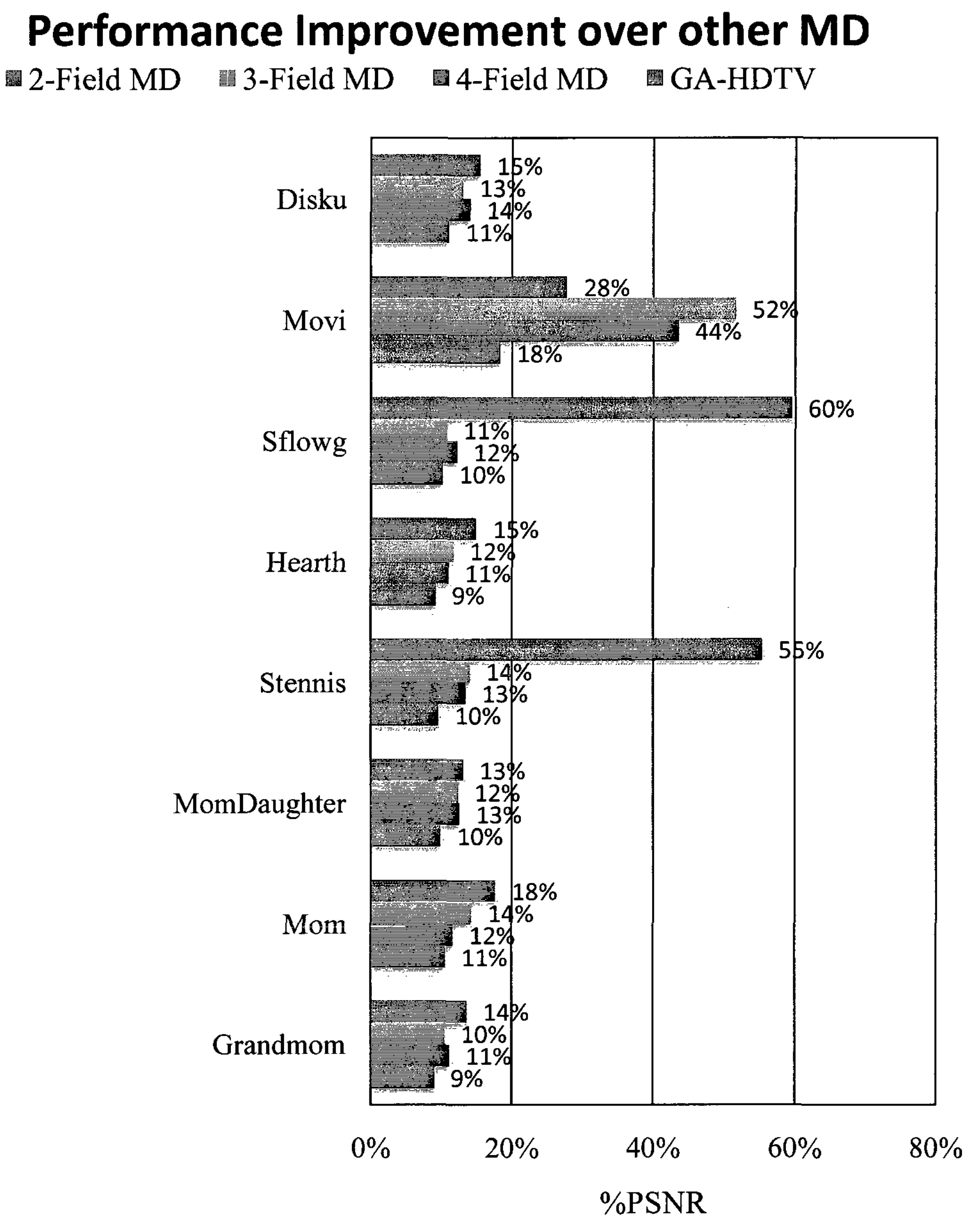
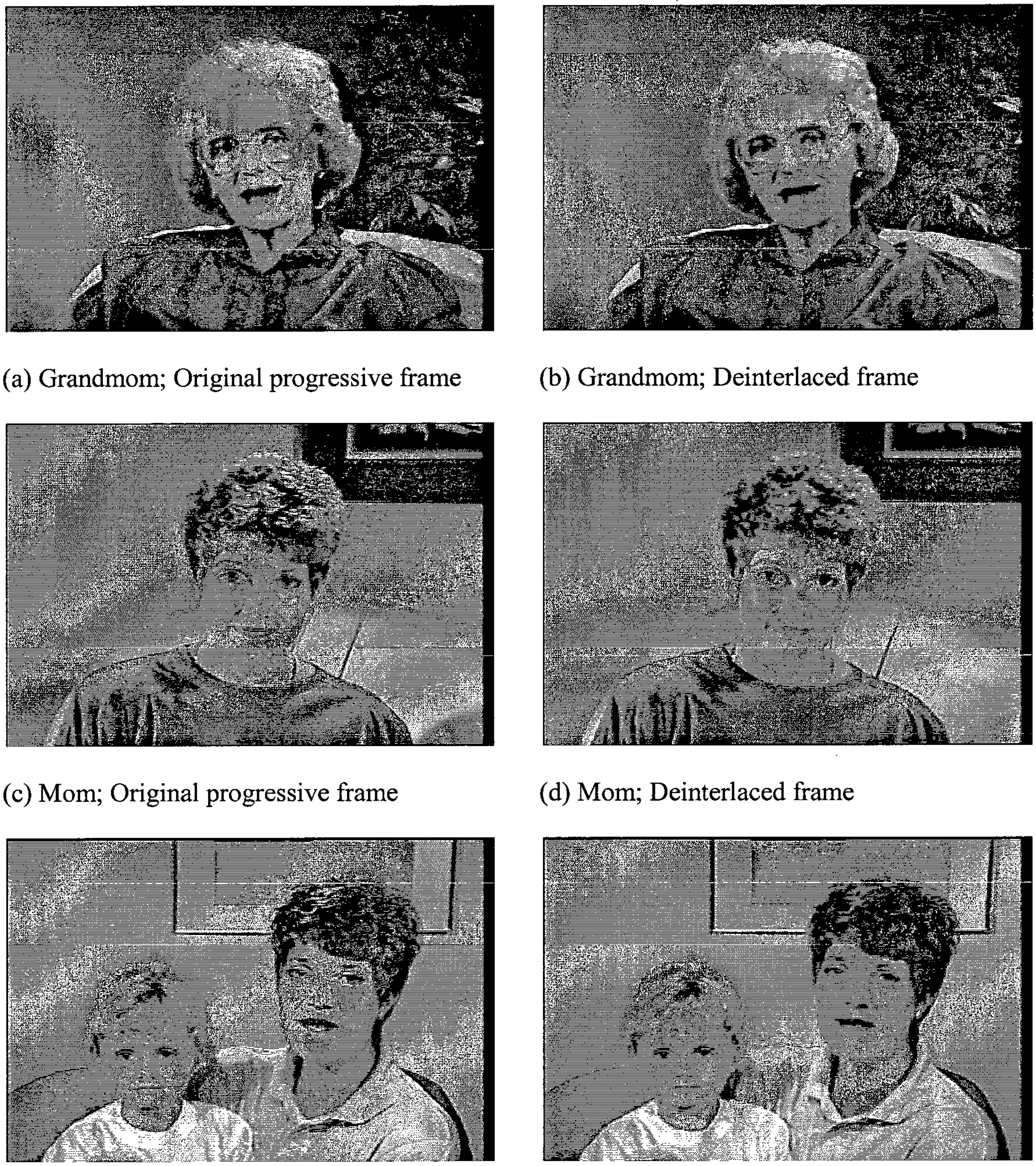


Figure 8-28 Percentage of performance improvement achieved by proposed motion detection method.

8.3.2 Subjective Evaluation Results

Subjective quality test, for our proposed motion adaptive deinterlacing algorithm, has been done according to ITU-R BT.500-11 explained in section 8.2.2. Twenty five observers have evaluated deinterlaced video sequences and scored each of them. They have been both male and female non-expert observers between ages 15 and 65.

A single deinterlaced frame of each video sequence is shown in Figure 8-29. The mean score of all observers over all test sequences is equal to 4.74 and its 95% confidence interval is [4.6662 4.8138] . Figure 8-30 shows the mean score value and 95% confidence interval of each video sequence.



(a) Grandmom; Original progressive frame

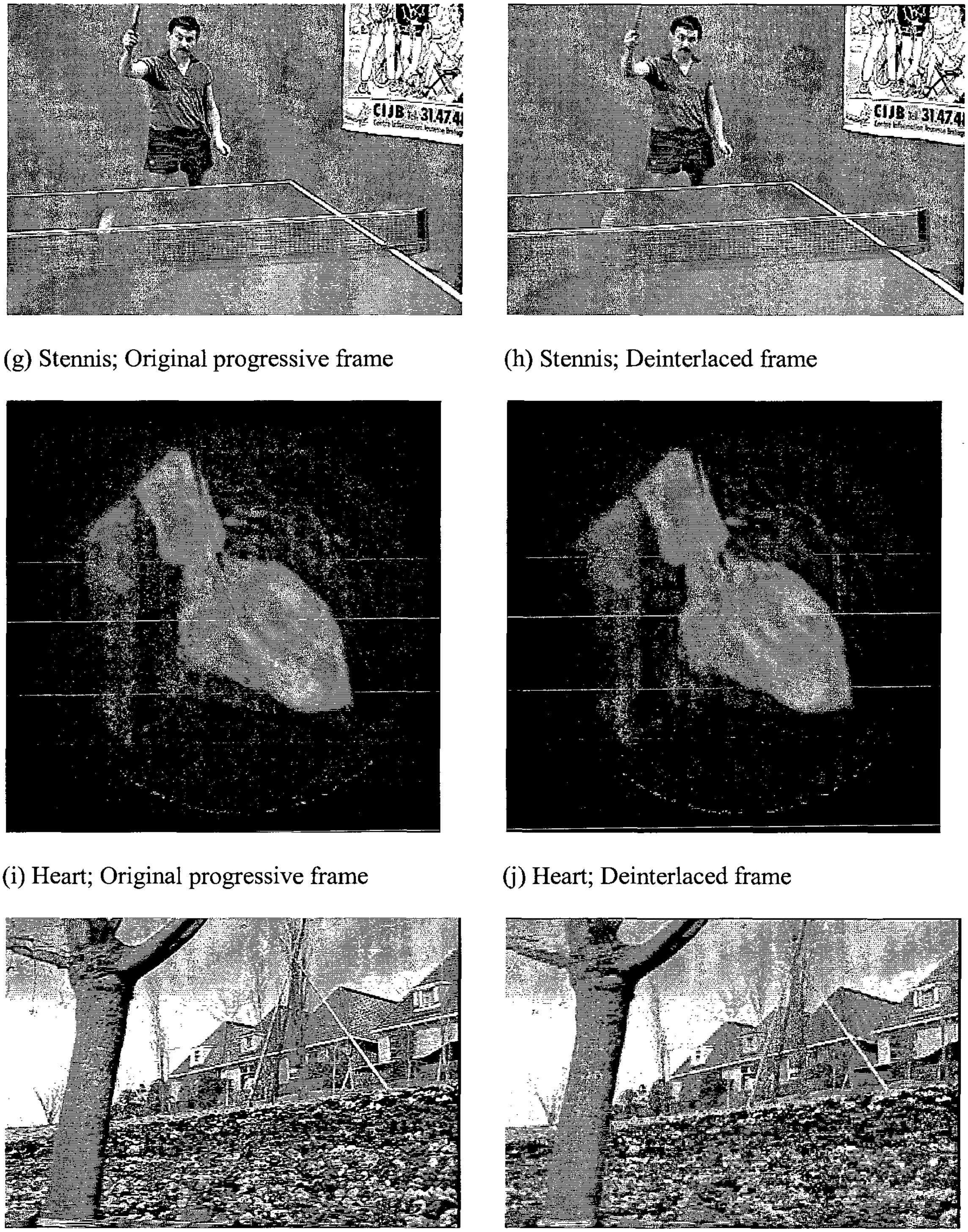
(b) Grandmom; Deinterlaced frame

(c) Mom; Original progressive frame

(d) Mom; Deinterlaced frame

(e) MomDaughter; Original progressive frame

(f) MomDaughter; Deinterlaced frame



(g) Stennis; Original progressive frame

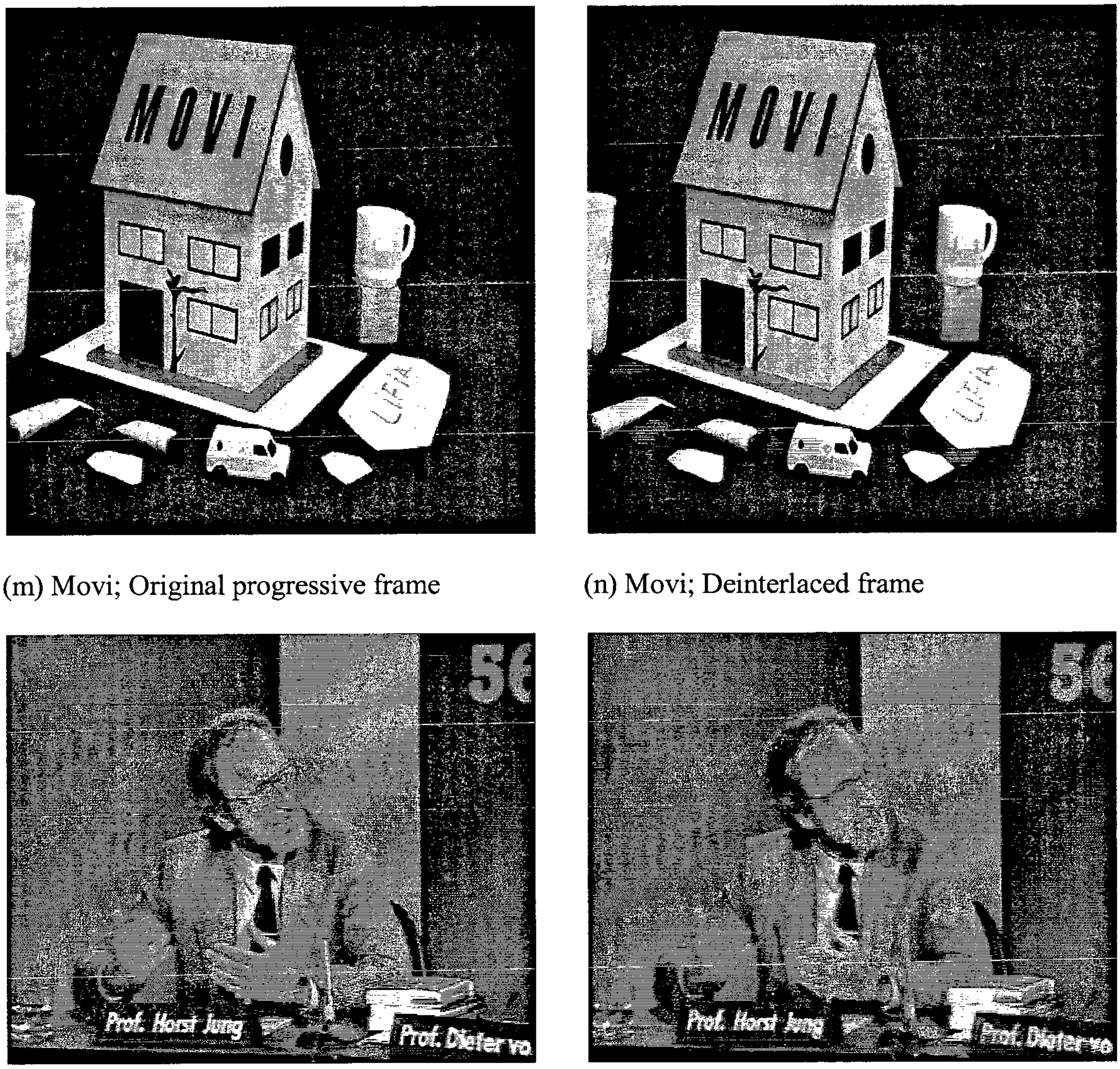
(h) Stennis; Deinterlaced frame

(i) Heart; Original progressive frame

(j) Heart; Deinterlaced frame

(k) Sflowg; Original progressive frame

(l) Sflowg; Deinterlaced frame



(m) Movi; Original progressive frame

(n) Movi; Deinterlaced frame

(o) Disku; Original progressive frame

(p) Disku; Deinterlaced frame

Figure 8-29 Proposed motion adaptive deinterlacing method results. In part (a) to (p) a single frame of each deinterlaced test sequence is shown on the left while their corresponding original frame is shown on the rigth.

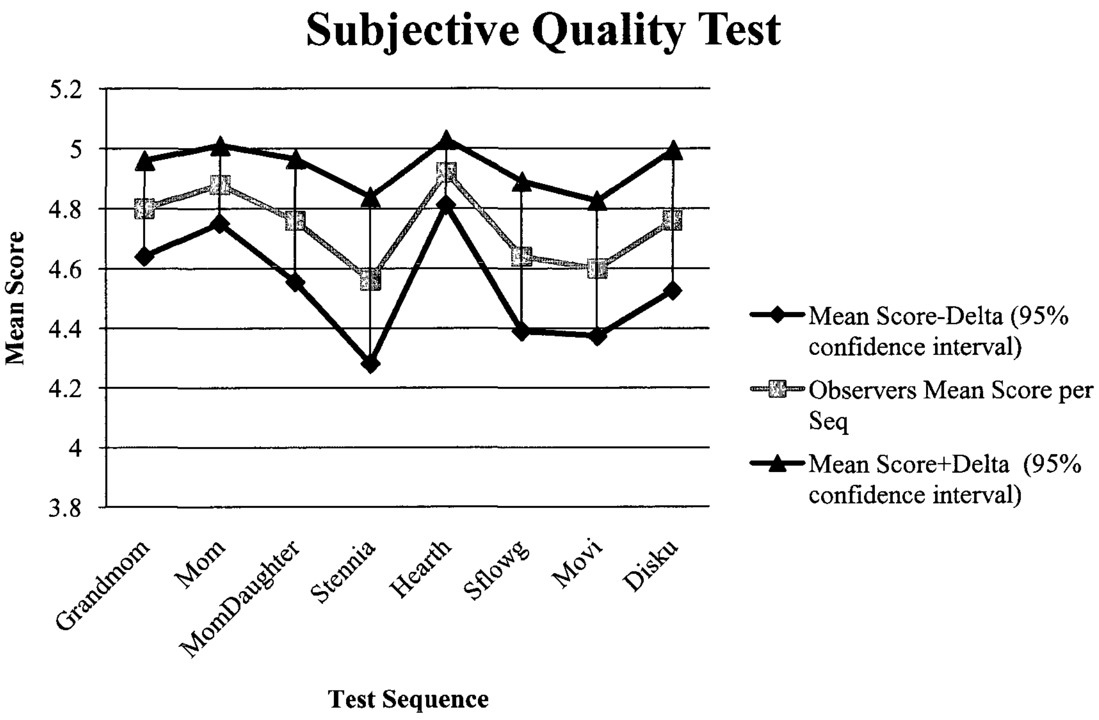


Figure 8-30 Subjective quality test results.

8.4 Conclusion

The experimental setup for objective and subjective evaluation of proposed deinterlacing method was explained in this chapter.

The proposed deinterlacing method was compared with several deinterlacing methods and it was shown that it outperforms all of them in performance and robustness to frame rate.

Chapter 9

Conclusion

9.1 Summary

In this thesis, we investigated various deinterlacing methods. We briefly surveyed the latest developments in deinterlacing methods. Our survey showed that the available methods suffer from low picture quality. Some of them have high quality for special type of video sequences but their performance deteriorates in some other video sequences. We proposed a motion adaptive deinterlacing method which provides high quality deinterlacing results for all tested video sequences.

The proposed deinterlacing method has a powerful motion detection algorithm which determines the motion activity level of the video by a hierarchical structure. The proposed motion detection algorithm captures a wide range of motions from slow to fast. It uses five consecutive interlaced video fields for motion detection. Utilizing a hierarchal structure, the algorithm starts with detecting motion in large partitions of a given field. Depending on the detected motion activity level for that partition, the motion detection algorithm might recursively be applied to sub-blocks of the original partition. The hierarchical structure uses some threshold values for decision making in each stage. The default values of these thresholds are found through some error minimization techniques. However they might be optimized for a specific application by knowing the type of the video in advance.

Two different low pass filters are used during the motion detection to increase the algorithm accuracy. In our default setting the first low pass filter is a two dimensional spatial averaging filter and the second low pass filter is a two dimensional spatial median filter. In order to improve motion detection accuracy, the motion detection algorithm is separately applied to all color components of a video sequence. These results are then compared to each other to reach to a unique motion detection result.

The motion detection result which is a measure of motion activity level in video sequence is then transformed to motion possibility value using a non-linear transformation. The calculated motion possibility value specifies the possibility of motion for each pixel in a video sequence. Motion possibility value is then used in the proposed motion adaptive interpolator.

The proposed motion adaptive interpolator combines a temporal and a spatial interpolation method to estimate missing pixels. The temporal deinterlacing method used in our implementation is a vertical temporal median filter. The spatial deinterlacing method is a bilinear averaging method. Motion possibility values are used to obtain best weighting factor for combining the spatial and temporal results.

Experimental results prove that proposed deinterlacing algorithm has outperformed other available deinterlacing methods. Objective evaluation results showed that the proposed deinterlacing method provides higher quality deinterlacing results. It is also more robust to frame rate changes. While the performance of other deinterlacing methods drops sharply with frame rate the proposed deinterlacing has minimum fluctuation.

9.2 Recommendations for Future Work

Deinterlacing is not a straightforward sampling rate up-conversion. Two diverse causes raise this problem. One cause is that the frequency response of the retina of an observer may change with scene content or with an observer eye tracking an object. The other cause is that the TV signals do not fulfill the requirements of sampling theorem.

One possible future work is the study of how various frequencies due to object motion map to the frequencies at the retina either if an observer is tracking an object or not. The result of this study may help to design more effective interpolation filters.

The fact that TV signals does not obey the Nyquist criterion is a major obstacle for deinterlacing. It makes perfect interlacing a theoretically impossible problem. Consequently it is always possible to improve the performance of a deinterlacing method by adding some more considerations or constraints.

Improvements to motion compensated methods could be obtained by designing more reliable motion estimation methods and utilizing more efficient temporal methods.

Improvements to motion adaptive methods could be obtain by adding object detection algorithms and utilizing object motion information in interpolation step. Adding pattern recognition methods and using their results in interpolation steps may improve the overall performance as well. In overall, for further improving the performance of motion adaptive deinterlacing methods, the effect of adding anti aliasing filters in interpolation step, the effect of adding object information, and the effect of adding pattern information in interpolation step, could be studied.

Considering a proper combination of motion adaptive deinterlacing with motion compensated deinterlacing may improve the performance as well. For example a motion compensated method might be used for temporal deinterlacing and combined with a spatial deinterlacing method to obtain an accurate motion adaptive deinterlacing.

References

[1]Canadian Radio-television and Telecommunications Commission, “Broadcasting Public Notice CRTC 2007-53,” http://www.crtc.gc.ca/eng/archive/2007/pb2007-53.htm (last accessed April 27, 2009)

[2]Federal Communications Commission’s Digital TV website: http://www.dtv.gov/

[3] Advanced Television System Committee, “A/53: ATSC Digital Television Standard,” http://www.atsc.org/standards/a_53-Part-1-6-2007.pdf (last accessed April 27, 2009)

[4] Engstrom, E.W., "A Study of Television Image Characteristics: Part Two: Determination of Frame Frequency for Television in Terms of Flicker Characteristics," *Proceedings of the IRE* , vol.23, no.4, pp. 295-310, April 1935

[5] de Haan, G.; Bellers, E.B., “De-interlacing: A Key Technology for Scan Rate Conversion,” Elsevier, ISBN 0444505946, 2000

[6] US Patent #5621470, Maher A. Sid-Ahmed “Interpixel and interframe interpolation of television pictures with conversion from interlaced,” Apr 15, 1997

[7] Chung, C.-H. Fan, Y.-C.; “A De-Interlacing Algorithm Using Spatial-Temporal Correlation Assisted Motion Estimation,” Circuits and Systems for Video Technology, IEEE Transactions on, Accepted for future publication, vol.19,2009

[8]Xinbo Gao; Juxia Gu; Jie Li, "De-interlacing algorithms based on motion compensation," *Consumer Electronics, IEEE Transactions on* , vol.51, no.2, pp. 589-599, May 2005

[9]Kalevo, O.; Haavisto, P., "Deinterlacing of video signals using nonlinear interpolation with simple motion compensation," *Nonlinear Digital Signal Processing, 1993. IEEE Winter Workshop on* , vol., no., pp.P_4.1-P_4.6, 1993

[10]Renxiang Li; Bing Zheng; Liou, M.L., "Reliable motion detection/compensation for interlaced sequences and its applications to deinterlacing," *Circuits and Systems for Video Technology, IEEE Transactions on* , vol.10, no.1, pp.23-29, Feb 2000

[11] Clarke, R.J., "Basic principles of motion estimation and compensation," *Applications of Motion Compensation, IEE Colloquium on* , vol., no., pp.1/1-1/7, 8 Oct 1990

[12] Zaccarin, A.; Liu, B., "Adaptive deinterlacing applied to HDTV coding," *Circuits and Systems, 1992. ISCAS '92. Proceedings., 1992 IEEE International Symposium on* , vol.1, no., pp.455-458 vol.1, 10-13 May 1992

[13] Shyh-Feng Lin; Yu-Ling Chang; Liang-Gee Chen, "Motion adaptive interpolation with horizontal motion detection for deinterlacing," *Consumer Electronics, IEEE Transactions on* , vol.49, no.4, pp. 1256-1265, Nov. 2003

[14] Gwo Giun Lee; Hsin-Te Li; Ming-Jiun Wang; He-Yuan Lin, "Motion Adaptive Deinterlacing via Edge Pattern Recognition," *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on* , vol., no., pp.2662-2665, 27-30 May 2007

[15] Chung-chi Lin; Ming-hwa Sheu; Huann-keng Chiang; Chih-Jen Wei, "The VLSI Design of Motion Adaptive De-interlacing with Horizontal and Vertical Motions Detection," *Circuits and Systems, 2006. APCCAS 2006. IEEE Asia Pacific Conference on* , vol., no., pp.1587-1590, 4-7 Dec. 2006

[16] Lejun Yu; Jintao Li; Yongdong Zhang; Yanfei Shen, "Motion adaptive deinterlacing with accurate motion detection and anti-aliasing interpolation filter," *Consumer Electronics, IEEE Transactions on* , vol.52, no.2, pp. 712-717, May 2006

[17]Evan I. Schwartz; "Who Really Invented Television?," Technology Review, published by MIT; Sept 2000 http://www.technologyreview.com/business/12186/?a=f (last accessed April 26, 2009)

[18]Wikipedia contributors, "History of television," Wikipedia, The Free Encyclopedia, http://en.wikipedia.org/w/index.php?title=History_of_television&oldid=286060467 (last accessed April 26, 2009).

[19]Wikipedia contributors, "Nipkow disk," Wikipedia, The Free Encyclopedia, http://en.wikipedia.org/w/index.php?title=Nipkow_disk&oldid=282677930 (last accessed April 26, 2009)

[20] Claudy, L., "Countdown to the end [digital TV]," *Spectrum, IEEE* , vol.42, no.10, pp. 30-31, Oct. 2005

[21] Wiley, R.E., "Digital television: the end of the beginning," *Communications Magazine, IEEE* , vol.33, no.12, pp.56-57, Dec 1995

[22] Bhatt, B.; Birks, D.; Hermreck, D., "Digital television: making it work," *Spectrum, IEEE* , vol.34, no.10, pp.19-28, Oct 1997

[23] Fox, B., "Digital TV comes down to earth," *Spectrum, IEEE* , vol.35, no.10, pp.23-29, Oct 1998

[24] Federal Communications Commission "DTV transition delay act," http://www.fcc.gov/cgb/consumerfacts/dtvdelay.pdf (last accessed April 27, 2009)

[25] Bellifemine, F.; Chimienti, A.; Picco, R., "Evolution and trends of HDTV," *CompEuro '91. Advanced Computer Technology, Reliable Systems and Applications. 5th Annual European Computer Conference. Proceedings.* , vol., no., pp.155-163, 13-16 May 1991

[26] Ninomiya, Y., "High definition television systems," *Proceedings of the IEEE* , vol.83, no.7, pp.1086-1093, Jul 1995

[27] Davies, K.P., "HDTV evolves for the digital era," *Communications Magazine, IEEE* , vol.34, no.6, pp.110-112, Jun 1996

[28] Rast, R.M., "The dawn of digital TV," *Spectrum, IEEE* , vol.42, no.10, pp. 26-31, Oct. 2005

[29] Wolfe, A., "HDTV - ready for the long drive?," *Spectrum, IEEE* , vol.40, no.6, pp. 13-15, June 2003

[30] Petajan, E., "The HDTV Grand Alliance System," *Communications Magazine, IEEE* , vol.34, no.6, pp.126-132, Jun 1996

[31] Challapali, K.; Lebegue, X.; Lim, J.S.; Paik, W.H.; Snopko, P.A., "The Grand Alliance system for US HDTV," *Proceedings of the IEEE* , vol.83, no.2, pp.158-174, Feb 1995

[32] Hopkins, R., "Digital terrestrial HDTV for North America: the Grand Alliance HDTV system," *Consumer Electronics, IEEE Transactions on* , vol.40, no.3, pp.185-198, Aug 1994

[33] Haskell, B., "High Definition Television (HDTV)--Compatibility and Distribution," *Communications, IEEE Transactions on* , vol.31, no.12, pp. 1308-1317, Dec 1983

[34] Fujio, T., "High-definition television systems," *Proceedings of the IEEE* , vol.73, no.4, pp. 646-655, April 1985

[35] Martin, E., "HDTV--A DBS Perspective," *Selected Areas in Communications, IEEE Journal on* , vol.3, no.1, pp. 76-86, Jan 1985

[36] Nakashima, K.; Ono, Y., "HDTV broadcast, from studio to home," *Consumer Electronics, IEEE Transactions on* , vol.38, no.3, pp.296-299, Aug 1992

[37] Charik, S.I., "HDTV via satellite-an overview," *Transmission of HDTV by Satellite, IEE Colloquium on* , vol., no., pp.1/1-1/3, 29 May 1990

[38] Groenenboom, P., "16:9 format of the future [HDTV]," *Broadcasting Convention, 1992. IBC., International* , vol., no., pp.553-555, 3-7 Jul 1992

[39] Pearlman, J.K., "HDTV: the American Agenda," *Consumer Electronics, IEEE Transactions on* , vol.36, no.3, pp.XII-XXV, Aug 1990

[40] Jain, P.C., "Digital HDTV system," *TENCON '93. Proceedings. Computer, Communication, Control and Power Engineering.1993 IEEE Region 10 Conference on* , vol., no.0, pp.422-425 vol.3, 19-21 Oct 1993

[41] Stare, E., "Development of a prototype system for digital terrestrial HDTV," *Applications of Video Compression in Broadcasting, IEE Colloquium on* , vol., no., pp.12/1-12/6, 20 Oct 1992

[42] Jurgen, R.K., "Chasing Japan in the HDTV race," *Spectrum, IEEE* , vol.26, no.10, pp.26-30, Oct 1989

[43] Baaren, E.; van de Wijngaert, L.; Huizer, E., ""Who's Afraid of High Def?' Institutional Factors Influencing HDTV Diffusion in the Netherlands," *Digital Society,*

2008 Second International Conference on the , vol., no., pp.42-48, 10-15 Feb. 2008

[44] Hatori, M.; Nakamura, Y., "1125/60 HDTV studio standard intended to be a worldwide unified HDTV standard," *Broadcasting, IEEE Transactions on* , vol.35, no.3, pp.270-278, Sep 1989

[45] Braham, R., "Consumer electronics," *Spectrum, IEEE* , vol.33, no.1, pp.46-50, Jan 1996

[46] Hearn, C., "Tutorial on HD - "variations on a theme"," *IT to HD: Visions of Broadcasting in the 21st Century, The IEE 2-Day Seminar on (Ref. No. 2004/10760)* , vol., no., pp. 179-190, 30 Nov.-1 Dec. 2004

[47] Sadiku, M.N.O.; Nelatury, S.R., "High definition television in detail," *Potentials, IEEE* , vol.26, no.1, pp.31-35, Jan.-Feb. 2007

[48] Bell, T.E., "The HDTV `test kitchens'," *Spectrum, IEEE* , vol.32, no.4, pp.46-49, Apr 1995

[49] Clark, D., "PC and TV makers battle over convergence," *Computer* , vol.30, no.6, pp.14-16, Jun 1997

[50] Basile, C.; Cavallerano, A.P.; Deiss, M.S.; Keeler, R.; Lim, J.S.; Luplow, W.C.; Paik, W.H.; Petajan, E.; Rast, R.; Reitmeier, G.; Smith, T.R.; Todd, C., "The US HDTV standard," *Spectrum, IEEE* , vol.32, no.4, pp.36-45, Apr 1995

[51] Bhatt, B.; Templin, F.; Cavallerano, A.; Gornstein, V.; Hogstrom, B.; Derovanessian, H.; Lamadrid, S.; Mailhot, J., "Grand Alliance HDTV multi-format scan converter," *Consumer Electronics, IEEE Transactions on* , vol.41, no.4, pp.1020-1031, Nov 1995

[52] Advance Television System Committee Inc, "ATSC Digital Television Standard Part 4 – MPEG-2 Video System Characteristics," http://www.atsc.org/standards/a_53-Part-4-2007-w-Amend-1.pdf (accessed April 26 2009)

[53] Advance Television System Committee Inc, "Digital Audio Compression Standard (AC-3, E-AC-3)," http://www.atsc.org/standards/a_52b.pdf (accessed April 26 2009)

[54] Advance Television System Committee Inc, "ATSC Standard: Program and System Information Protocol for Terrestrial Broadcast and Cable (Revision C)," (accessed April 26 2009)

[55] De Haan, G.; Bellers, E.B., "Deinterlacing-an overview," *Proceedings of the IEEE* , vol.86, no.9, pp.1839-1857, Sep 1998

[56] Oppenheim, Schafer with Buck, "Discrete-Time Signal Processing," 2nd ed, Upper Saddle River, NJ: Prentice-Hall, 1999, ISBN: 0137549202

[57] Enden A. W. M. van den; Verhoeckx N. A. M.; "Discrete-time Signal Processing: An Introduction," Prentice-Hall, 1989, ISBN 0132167557

[58] Tekalp, A. Murat, "Digital Video Processing," Prentice Hall PTR, 1995, ISBN 0131900757

[59] Sid-Ahmed, Maher A. "Image Processing: Theory, Algorithms, and Architectures/Book and Disk" McGraw-Hill, 19951 ISBN 9780070572409

[60] Doyle, T. Looymans, M. "Progressive scan conversion using edge information," Proc. 3rd Int. Workshop on HDTV, pp. 711-721, Torino,. Italy, Aug. 1989

[61]Ryu, C.; Kim, S.P., "Deinterlacing using motion compensated local spectra," *Signals, Systems and Computers, 1995. 1995 Conference Record of the Twenty-Ninth Asilomar*

Conference on , vol.2, no., pp.1394-1397 vol.2, 30 Oct-1 Nov 1995

[62]Mohammadi, H.M.; Langlois, P.; Savaria, Y., "A Five-Field Motion Compensated Deinterlacing Method Based on Vertical Motion," *Consumer Electronics, IEEE Transactions on* , vol.53, no.3, pp.1117-1124, Aug. 2007

[63]Dong-Ho Lee, "A New Edge-based Intra-field Interpolation Method for Deinterlacing Using Locally Adaptive-thresholded Binary Image," *Consumer Electronics, IEEE Transactions on* , vol.54, no.1, pp.110-115, February 2008

[64]Gwo Giun Lee; He-Yuan Lin; Ming-Jiun Wang; Rong-Lai Lai; Chih Wen Jhuo, "A high-quality spatial-temporal content-adaptive deinterlacing algorithm," *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on* , vol., no., pp.2594-2597, 18-21 May 2008

[65]Beric, A.; de Haan, G.; Sethuraman, R.; van Meerbergen, J., "Algorithm/architecture co-design of the generalized sampling theorem based de-interlacer [video signal processing]," *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on* , vol., no., pp. 2943-2946 Vol. 3, 23-26 May 2005

[66]Zhao, M.; de Haan, G., "Intra-field de-interlacing with advanced up-scaling methods," *Consumer Electronics, 2004 IEEE International Symposium on* , vol., no., pp. 315-319, Sept. 1-3, 2004

[67]de Haan, G.; Bellers, E.B., "De-interlacing of video data," *Consumer Electronics, IEEE Transactions on* , vol.43, no.3, pp.819-825, Aug 1997

[68]Zhao, M.; de Haan, G., "Content adaptive vertical temporal filtering for de-interlacing," *Consumer Electronics, 2005. (ISCE 2005). Proceedings of the Ninth*

International Symposium on , vol., no., pp. 69-73, 14-16 June 2005

[69]de Haan, G.; Lodder, R., "De-interlacing of video data using motion vectors and edge information," *Consumer Electronics, 2002. ICCE. 2002 Digest of Technical Papers. International Conference on* , vol., no., pp. 70-71, 2002

[70]Stiller, C.; Konrad, J., "Estimating motion in image sequences," *Signal Processing Magazine, IEEE* , vol.16, no.4, pp.70-91, Jul 1999

[71]Myeong-Hwan Lee; Jeong-Hoon Kim; Jeong-Sang Lee; Kyeong-Keol Ryu; Dong-Il Song, "A new algorithm for interlaced to progressive scan conversion based on directional correlations and its IC design," *Consumer Electronics, IEEE Transactions on* , vol.40, no.2, pp.119-129, May 1994

[72]Salo, J.; Neuvo, Y.; Hameenaho, V., "Improving TV picture quality with linear-median type operations," *Consumer Electronics, IEEE Transactions on* , vol.34, no.3, pp. 373-379, Aug 1988

[73]Wang, Y.; Mitra, S.K., "Motion/pattern adaptive interpolation of interlaced video sequences," *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on* , vol., no., pp.2829-2832 vol.4, 14-17 Apr 1991

[74]Feng Fan; Simon, S.; Bruyland, I.; Weile Zhu; De Canne, B.; Van Bladel, M., "A method for hierarchical subband HDTV splitting," *Circuits and Systems for Video Technology, IEEE Transactions on* , vol.5, no.3, pp.225-230, Jun 1995

[75]Yong Ding; Sheng-li Lu; Long-xing Shi, "A NOVEL DE-INTERLACE BASED ON SPATIO-TEMPORAL WEIGHT ADAPTIVE AND EDGE-DIRECTED INTERPOLATION," *Signal Processing, 2006 8th International Conference on* , vol.1,

no., pp.-, 16-20 2006

[76]Latzel, M.; Tsotsos, J.K., "A robust motion detection and estimation filter for video signals ," *Image Processing, 2001. Proceedings. 2001 International Conference on* , vol.1, no., pp.381-384 vol.1, 2001

[77]Huang, Q.; Gao, W.; Zhao, D.; Sun, H., "An efficient and robust adaptive deinterlacing technique," *Consumer Electronics, IEEE Transactions on* , vol.52, no.3, pp.888-895, Aug. 2006

[78]Kai-Yao Chen; Sau-Gee Chen, "Efficient algorithms for interlaced to progressive scan conversion ," *Speech, Image Processing and Neural Networks, 1994. Proceedings, ISSIPNN '94., 1994 International Symposium on* , vol., no., pp.288-291 vol.1, 13-16 Apr 1994

[79]Yeong-Taeg Kim; Shin-Haeng Kim; Se-Woong Park, "Motion decision feedback deinterlacing algorithms," *Image Processing. 2002. Proceedings. 2002 International Conference on* , vol.3, no., pp. III-397-III-400 vol.3, 2002

[80]Charng Lang Lee; Shifan Chang; Chein-Wei Jen, "Motion detection and motion adaptive pro-scan conversion," *Circuits and Systems, 1991., IEEE International Sympoisum on* , vol., no., pp.666-669 vol.1, 11-14 Jun 1991

[81]Myeong-Hwan Lee; Jeong-Hoon Kim; Jeong-Sang Lee; Kyeong-Keol Ryu; Dong-Il Song, "A new algorithm for interlaced to progressive scan conversion based on directional correlations and its IC design," *Consumer Electronics, IEEE Transactions on* , vol.40, no.2, pp.119-129, May 1994

[82]Dongil Han; Chang-Yong Shin; Seung-Jong Choi; Jong-Seok Park, "A motion

adaptive 3-D de-interlacing algorithm based on the brightness profile pattern difference," *Consumer Electronics, IEEE Transactions on* , vol.45, no.3, pp.690-697, Aug 1999

[83]Kojima, A.; Sakurai, N.; Kishigami, J.I., "Motion detection using 3D-FFT spectrum," *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on* , vol.5, no., pp.213-216 vol.5, 27-30 Apr 1993

[84]Koivunen, T., "Motion detection of an interlaced video signal," *Consumer Electronics, IEEE Transactions on* , vol.40, no.3, pp.753-760, Aug 1994

[85] Shahinfard, E.; Sid- Ahmed, M.A.; Ahmadi, M., "A motion adaptive deinterlacing method with hierarchical motion detection algorithm," *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on* , vol., no., pp.889-892, 12-15 Oct. 2008

[86] Shahinfard, E.; Sid-Ahmed, M.; Ahmadi, M., "An Improved Motion Adaptive Deinterlacing Method Using Variable Block-Size Motion Detection," *Signal Processing and Information Technology, 2007 IEEE International Symposium on* , vol., no., pp.122-126, 15-18 Dec. 2007

[87] International Telecommunication Union/ITU Radiocommunication Sector; ITU-R BT.500-11: “Methodology for the subjective assessment of the quality of television pictures,” Jan 1 2002

[88] Amin, M.B., "Subjective assessments of the quality of HDTV pictures impaired by noise," *Broadcasting Convention, 1990. IBC 1990., International* , vol., no., pp.30-38, 21-25 Sep 1990

[89] Hoffmann, H.; Itagaki, T.; Wood, D.; Hinz, T.; Wiegand, T., "A Novel Method for Subjective Picture Quality Assessment and Further Studies of HDTV Formats,"

Broadcasting, IEEE Transactions on , vol.54, no.1, pp.1-13, March 2008

[90] Hoffmann, H.; Wood, D.; Itagaki, T., "Psycho-physical method of television picture quality evaluation (EBU-II)," *Electronics Letters* , vol.43, no.4, pp.212-213, February 15 2007

[91]Huynh-Thu, Q.; Ghanbari, M., "Scope of validity of PSNR in image/video quality assessment," *Electronics Letters* , vol.44, no.13, pp.800-801, June 19 2008

Vita Auctoris

Name:	Elham Shahinfard
Place of Birth:	Tehran, Iran
Year of Birth:	1976
Education	Iran University of Science and Technology, Tehran Iran
	1994-1999 B.Sc.
	Sharif University of Technology, Tehran, Iran
	1999-2001 M.Sc.
	University of Windsor, Windsor, Ontario, Canada
	2004-2009 Ph.D.