R.YAZGAN M.S Thesis 2008

ARTIFACT REDUCTION IN OCCLUDED REGIONS FOR MOTION COMPENSATED FRAME INTERPOLATION

RIFAT YAZGAN

IŞIK UNIVERSITY
2008

.

ARTIFACT REDUCTION IN OCCLUDED REGIONS FOR MOTION COMPENSATED FRAME INTERPOLATION

RIFAT YAZGAN

B.S., Electronics Engineering, Işık University, 2004

Submitted to the Graduate School of Science and Engineering
in partial fulfillment of the requirements for the degree of
Master of Science
in
Electronics Engineering

.

IŞIK UNIVERSITY
2008

ISIK UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

ARTIFACT REDUCTION IN OCCLUDED REGIONS FOR MOTION COMPENSATED FRAME INTERPOLATION

RIFAT YAZGAN

.

APPROVED BY:

Assist. Prof. Hasan Fehmi Ateş, (Thesis Supervisor)	(Işık University)	__________________
Assist.Prof. Onur Kaya,	(Işık University)	__________________
Assist. Prof. Taner Eskil,	(Işık University)	__________________

APPROVAL DATE:

ARTIFACT REDUCTION IN OCCLUDED
REGIONS FOR MOTION COMPENSATED FRAME
INTERPOLATION

Abstract

In this paper, we propose a new method to be used in occluded regions for motion compensated frame interpolation. Frame rate up conversion is becoming much more important when usage of digital video systems increase. Standard conversion techniques are not effective in increasing the temporal resolution of video without any degradation. Therefore, motion compensated post processing techniques are developed to obtain the best possible image at the receiver. Specifying the correct or reliable motion vectors is the most important issue to be able to prevent the artifacts and deformed structures in interpolated frames. There are many methods in literature developed for that purpose. However, these methods generally create annoying artifacts in interpolated frames. Significantly, they can not provide acceptable results for covered and uncovered regions. In our work, we build upon the successful multi-stage post processing method which is used to obtain the correct vector for motion compensated frame interpolation. According to our analysis, all these post processing stages are effective to specify the reliable motion vectors, but they are not sufficient to solve the problem in covered and uncovered regions. In thesis, we suggest methods to be able to solve the problem occurring in these regions. In most of the adaptive methods, either past or future frames are used for interpolation in covered/uncovered regions, respectively, but in our proposals we interpolate the new frame by using a weighted combination of the four neighboring frames to minimize the sharp blocking artifacts that occur because of uni-directional interpolation in occluded regions. Experimental results show that our proposed method is robust in occluded areas, and reduces the sharp artifacts in the interpolated frame for covered and uncovered regions, and provides better picture quality.

DEVİNİM DENKLEŞTİRMELİ ÇERÇEVE
ARADEĞERLEMESİ İÇİN ÖRTÜLMÜŞ BÖLGELERDEKI
YAPAY HATA AZALTILMASI

Özet

Bu makalede, devinim denkleştirmeli çerçeve ara değerlemesinin örtülmüş bölgelerinde uygulanması için yeni bir metod öneriyoruz. Dijital video sistemlerinin kullanımı arttıkça çerçeve hızının yukarı çevriminin önemi artıyor. Standart çevrim teknikleri çözünürlüğün bozulma olmadan arttırılması için yeterli değil. Bu nedenle alıcıların en iyi görüntüyü alabilmesi için devinim denkleştirmeli art işlemci teknikleri geliştiriliyor. Doğru yada güvenilir devinim vektörlerinin belirlenmesi aradeğerlenmiş çerçevedeki yapay hataların ve biçim bozulmalarının önlenmesi için en önemli unsurdur. Literatürde bu amaç için geliştirilmiş birçok metod bulunmaktadır. Fakat bu metodlar çerçeve aradeğerlemesinde göze batan yapay hatalara sebep olmaktadır. Özellikle bu metodlar kapatılan ve açığa çıkarılan bölgeler için kabul edilebilir sonuç vermemektedir. Çalışmamızda devinim denkleştirmeli çerçeve ara değerlemesinde kullanılacak doğru vektörün bulunması için geliştirilmiş çok aşamalı başarılı bir metoda güvenerek hareket ediyoruz. Analizlerimize göre bütün bu art işlemci aşamaları güvenilir devinim vektörlerinin bulunması için etkili olmakla beraber kapatılan ve açığa çıkarılan bölgelerdeki problemin çözülmesiyle ilgili yetersiz kalmaktadır. Bu tezde, bu bölgelerdeki sorunun çözülmesi için yeni metodlar önermekteyiz. Birçok uyarlanmış metotta sadece geçmiş veya gelecek çerçevelerden elde edilen bilgiler örtülmüş bölgelerdeki çerçeve aradeğerlemesi için kullanılırken, bizim önerimizde örtülmüş bölgelerde gruplanmış yapay hatayı önlemek amacıyla dört komşu çerçeveden elde edilen bilgilerin ağırlıklı birleşimi kullanılıyor. Deneyler sonuçları gösteriyor ki, önerdiğimiz methodlar örtülen alanlarda çerçeve aradeğerlenmesi yaparken oluşabilecek yapay hata probleminin çözümü için etkili olmakta, keskin yapay bozulmaları azaltmakta ve daha iyi görüntü kalitesi sağlamaktadır.

Acknowledgements

There are many people who helped to make my years at the master school most valuable. First, I would like to thank Assist. Prof. Hasan Fehmi Ateş , my thesis supervisor. Having regular meetings with him and his intellectual & technical support provided me an important opportunity for improving my work. I thank him for his suggestions and expertise.

I would also like to thank to my master student colleagues who helped me all through the years full of class work and exams.

The last words of thanks go to my family. I thank my parents Muzaffer and Feyal Yazgan and my brother Serhat Yazgan for their patience and encouragement.

Table of Contents

List of Figures

List of Symbols

d	: Distance
$\vec{D}_{\tau}(\vec{x}, n)$	: Displacement vector
f	: Video sequence
f_t	: Video frame
$f_n(x, y)$	: Pixel intensity at (x, y) coordinate
n	: Time domain index
v_x, v_y	: Motion vectors
α_p	: Weight of backward prediction
α_n	: Weight of forward prediction
$\in$	: Error
ω^k	: Weight of motion compensated prediction
Ψ	: Smoothness measure

List of Abbreviations

ABPD : Absolute Bidirectional Prediction Difference
BMA : Block Matching Algorithm
BPD : Bidirectional Prediction Difference
FRC : Frame Rate Conversion
MB : Macro Block
MCFI : Motion Compensated Frame Interpolation
MV : Motion Vector
MVF : Motion Vector Field
OBMC : Overlapped Block Motion Compensation
PSNR : Signal to Noise Ratio
SAD : Sum of Absolute Difference
SSIM : Structure Similarity Index Value

Chapter 1
Introduction

The technology development in television and multimedia systems caused the necessity of using different kind of display formats and video sources. Digital video systems – digital high definition TV, videophone, etc- have various spatial and temporal resolution requirements that provide the different formats and standards to store, transmit and display video signals. Frame/field rate up/down conversion and interlacing/de-interlacing are the methods to make changes on formats of digital videos.

The frequency or display rate of TV may be 50, 60 or 100 Hz, and PC monitor rate may be anything between 60 and 120 Hz. This is why human eye does not perceive flicker when the display is more than 50 frames per second. On the other hand, the standard wireless transmission systems have limited bandwidth. Digital video standards (CCIR601) limits 24 frame per second per channel. The frame rate is doubled at the decoder to prevent the human eye to perceive flickers. Frame rate up conversion (FRUC) may be used in different kinds of applications. You can see commonly used video standards that conversion application integrated in Figure 1.1.

FRUC is very important method and applied in most of the up to date digital video systems likewise LCD driver, mobile broadcast and movie DVD. Figure 1.2 represents the application of conversion in these areas. FRUC is used to improve the video quality in a low bit rate coded video like video conferencing. Additionally, FRUC can be used in slow-motion playback application by interpolating frames for smoothing the slow-motion.

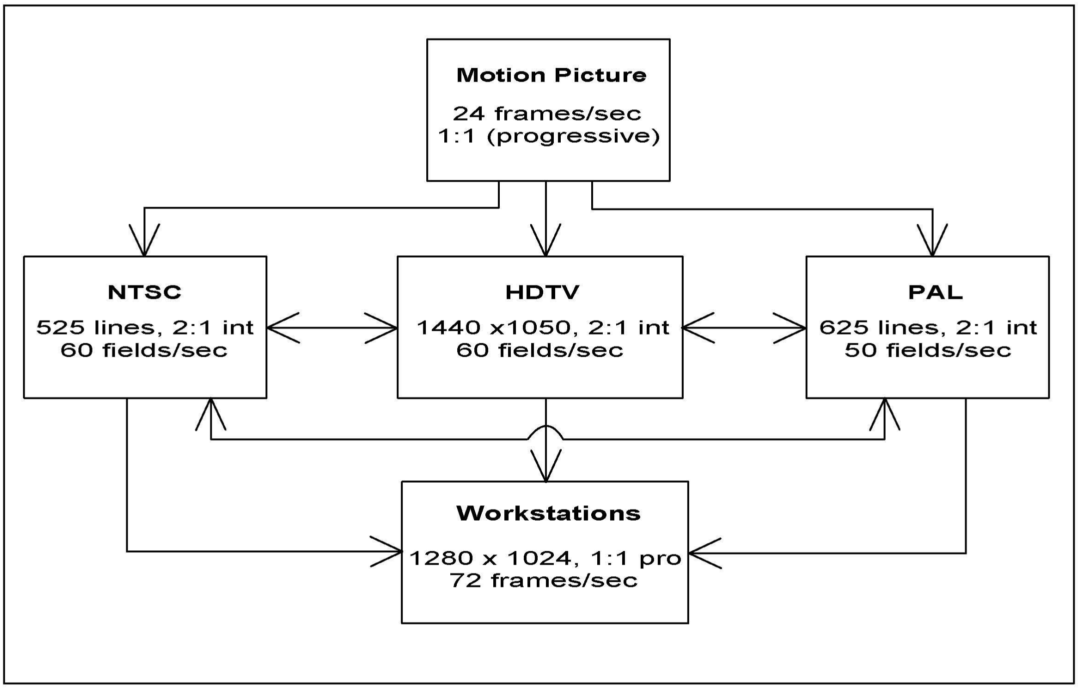

Figure 1.1: Conversion applications for typical standards

Another application area of FRUC is related with the structure of LCD TV. LCD has disadvantage in motion quality because of its long lighting effect. This causes blurring (not to be able to focus) in high motion picture. On the other hand in CRT we do not face this problem because of its short lighting period. If the lighting period is shortened in LCD, then there will be luminance flickering problem in video. FRUC can be applied for the aim of solving this kind of problem, too.

Frame interpolation can be applied in several ways. The conventional FRUC methods are repetition and linear frame interpolation. In these methods, we do not take into account the motion, so they may cause artifacts likewise motion jerkiness (because of repetition) and image blurring (because of linear frame interpolation).

As explained above, the rate 24 frames/sec has to be increased to 50 or 100 frames/sec to prevent human eye to perceive flickers. For that purpose, the same frames are interpolated between the existing frames. Judder problem is an example of standard FRUC methods, and occurs because of repetition of frames to catch the target frame rate. You can see judder problem more clearly in larger, high brightness and contrast displays.

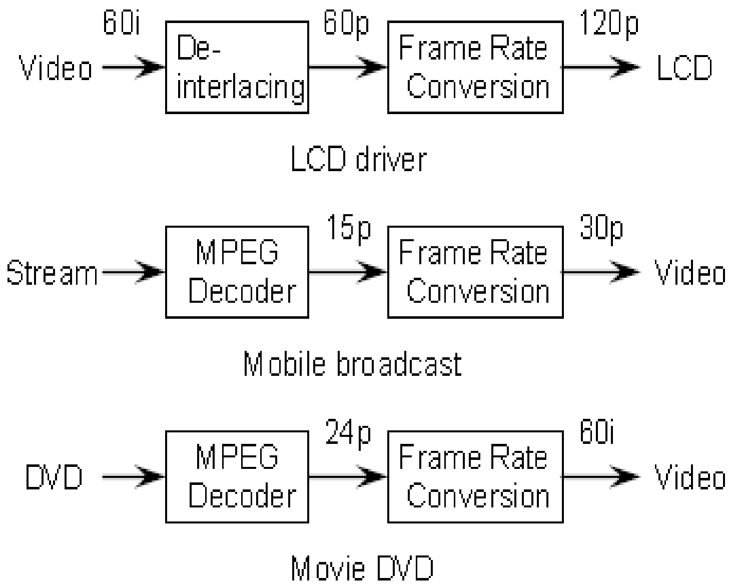

Figure 1.2: Application of Frame Rate Conversion

The most efficient way of frame interpolation is motion compensated frame interpolation (MCFI). This method increases the quality of video by making the interpolation of new frame by observing block motions. In MCFI, the motion between previous and current frames is decided, and the new frame is constructed between these two frames by using the motion information. Making the interpolation of the frames where information was not transmitted, by using motion information removes the artifacts caused by standard interpolation techniques

In Chapter 2, MCFI and the related subjects are described, the motion estimation techniques that are used in the multi-stage method [1] is provided and also previously applied interpolation methods are discussed. In Chapter 3, the successful multi-stage motion vector processing method [1] for MCFI is explained together with all its steps and results, the several additions to this method and weakness of this method are discussed. In Chapter 4, motion estimation problem in occluded regions is considered, the literature survey of interpolation methods for occluded areas is included, and effective methods to be used in occluded regions is described. Also in this chapter, our new proposal which is developed to solve the problem in covered and uncovered regions is described in detail, and different versions of new proposal are also included in this chapter. In Chapter 5, simulations of different methods are presented, and analysis of the experimental results obtained by using these methods is discussed. In Chapter 6 summary of our work is specified as a conclusion.

Chapter 2
Overview of Motion Compensated Frame Interpolation

Basically, we can divide the interpolation techniques into three main parts: frame repetition, linear frame interpolation and motion compensated frame interpolation.
Standard up conversion methods can be classified as frame/field rate up conversion and de interlacing. Frame/field rate up conversion depends on the repetition of existing information. De interlacing method basically depends on line repetition, line averaging and non linear spatial interpolation methods. De interlacing can decrease line flickers and improves the vertical resolution of displayed images. Both of these methods depend on similar concepts and they are used in improved-definition TV to increase video quality. On the other hand, they can cause artifacts in video, because these methods do not use motion information. Motion compensated frame interpolation is preferred to be able get better image quality at the receiver.

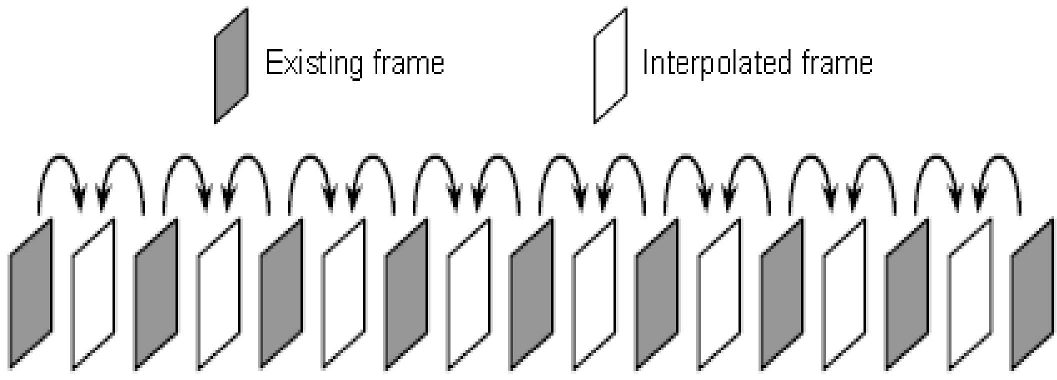

Figure 2.1: Frame Rate up Conversion

MCFI is optimal FRUC method to prevent loss of temporal resolution when video data is transmitted over bandwidth limited channels. For a good MCFI, 3 steps should be followed:

i) Motion estimation which is used to find the motion vector (MV) between two frames

ii) Post processing of motion estimation which can be applied to improve the accuracy of the motion estimation

iii) Making an artifact free frame interpolation by using the correct MVs obtained from post processing motion estimation stages.

In MCFI, the target frame f_t is interpolated between two consecutive frames f_{t-1} and f_{t+1} by considering bidirectional prediction difference. As you see in Figure 2.2, the bidirectional interpolation makes a synthesis between frames, this means pixels of the interpolated frame is determined by considering pixels of the forward and backward frames.

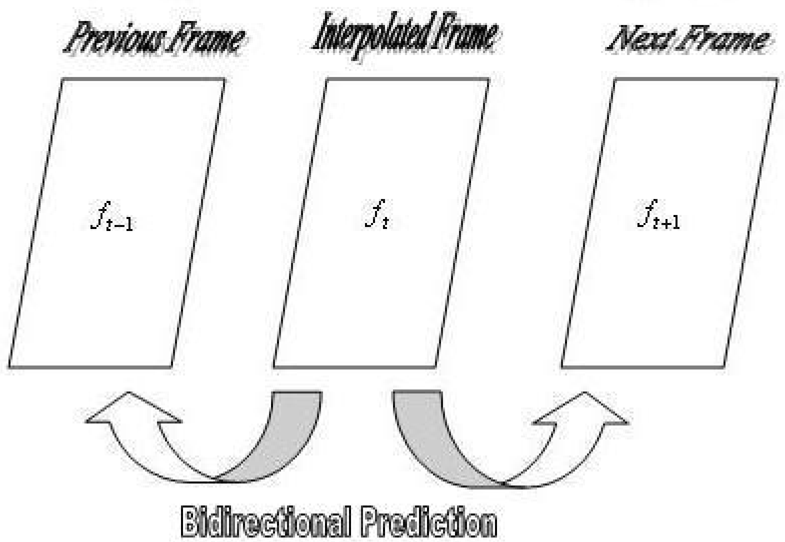

Figure 2.2: Representation of bidirectional prediction

$$f_t(x, y) = \omega_f \cdot f_{t-1}(x + \frac{1}{2}v_x, y + \frac{1}{2}v_y) + \omega_b \cdot f_{t+1}(x - \frac{1}{2}v_x, y - \frac{1}{2}v_y), \qquad (2.1)$$

The motion vector field $V = (V_{x,} V_y)$ is applied to get the f_t. Forward motion compensation means prediction of current frame from a previous frame; the backward motion compensation means that prediction of current frame from a future frame. We use ω_f for weight of forward compensation, and we use ω_b for weight of backward compensation. For the moment, they are taken as 0.5 by considering the same amount effect of forward and backward compensation.

2.1 Literature Survey for Motion Compensated Interpolation Methods

MCFI methods synthesize new frame between two consecutive frames by using the motion information and pixel intensity of these two frames. The most important issue related with MCFI is MV processing technique. There are many kinds of different MV processing techniques used to remove or improve unreliable MVs.

The basic method among all MCFI methods is direct interpolation. In this method bidirectional interpolation is done by using forward and backward compensation with the same weights 0.5. Also, in direct interpolation the all the MVs are assumed to be reliable, so there is no MV post processing applied. As a result of direct interpolation artifact occurrence is unavoidable.

Among the MCFI techniques that focus on to correct irregular MVs, one of the most popular techniques is MV smoothing. This is applied in the areas that MV is incompatible with its adjacent vectors. This incompatible situation causes artifacts in the video sequence. For that reason the incompatible or unreliable MV is replaced by another MV that can keep the smoothly changing structure of MVF. The method can be applied only when the adjacent MVs are compatible with each other. However, there may be some estimation accuracy loss in the MV; MV smoothing is effective to prevent discontinuities for interpolated frames. As you see in Figure 2.3, B vector is incompatible with N vectors, so it may cause artifact in the interpolated frame. Therefore, B is made compatible with its' neighbors by smoothing process.

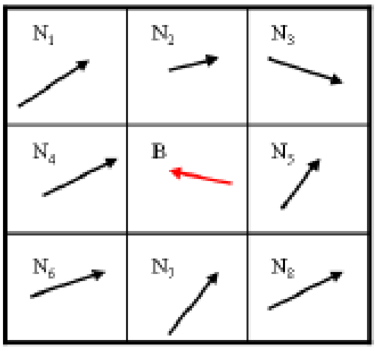

Figure 2.3: Application of MV smoothing for incompatible MV

Vector median filter is another method which is applied to correct or remove unreliable MVs. This is a nonlinear edge adaptive method. Median operation does not accept the unreliable MVs, so that it prevents the blurring across the edge. On the other hand, median filtering can cause artifact because of lack of some vectors after median process. There is also weighted median filter which the weights are assigned to the samples in median filter.

Weighted vector median filter (WVM) can be defined as;

$$Arg\min(\vec{v}_{WVM}) = \sum_{i=1}^{N} \omega_i \left\| \vec{v}_{WVM} - \vec{v}_i \right\|_p \le \sum_{i=1}^{N} \omega_i \left\| \vec{v}_j - \vec{v}_i \right\|_p$$

j=1,2,....,N (2.2)

Where V is set of vectors, W is set of weights. This filter chooses the MV that minimizes the right hand side of equality.

In addition to these methods, we describe robust MV multi-stage vector processing method [1] in Chapter 3. To make a comparison between these basic methods and the multi-stage method in terms of their interpolation performance, we represent the example in Figure 2.4. The figure includes interpolation methods of direct interpolation, vector median filter, MV smoothing, MV selection [1] and multi-stage MV processing [1] methods (MV selection is the sub stage of this method). You can easily see difference in the interpolated frame when the proposed MV selection and proposed multi-stage MV processing method applied. In Figure 2.4, you also have a chance make comparison between signal to noise ratio (PSNR) and structure similarity index value (SSIM) performance of these methods. SSIM examines degration of structure information and less senstive to pixel shift effect. SSIM value carries the information about similarity of object structures. PSNR value gives us information about the relation of the interpolated picture with the original one.

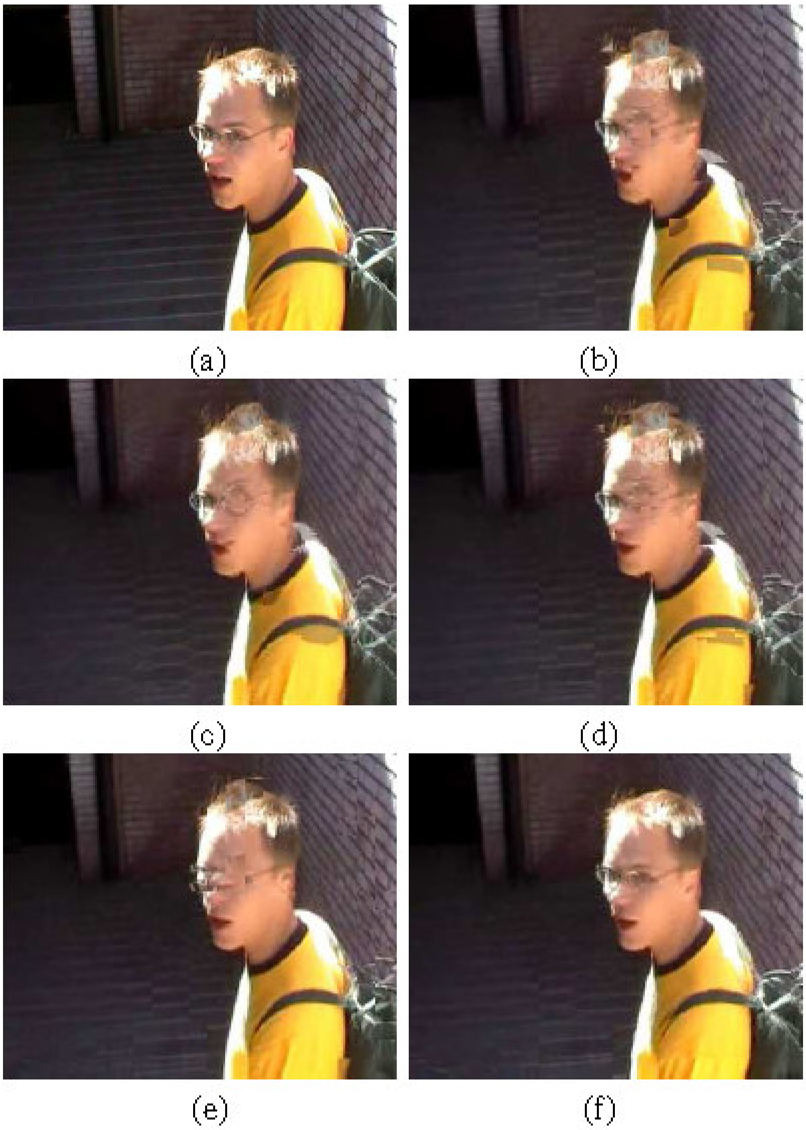

Figure 2.4: The interpolated results of 288 frame of WALK video sequence

(a) Original frame

(b) Direct interpolation (PSNR: 21.92dB, SSIM: 0.7183)

(c) MV smoothing (PSNR: 22.35dB, SSIM: 0.7407)

(d) Vector median filtering (PSNR: 22.04dB, SSIM: 0.7224)

(e) Proposed MV selection described in [1]
(PSNR: 22.22dB, SSIM: 0.7388)

(f) Proposed multi-stage vector processing described in [1]
(PSNR: 22.39dB, SSIM: 0.7606)

In Section 2.1, some other significant proposed methods for frame interpolation are presented.

2.1.1 Block Segmentation

Block segmentation is based on dividing the blocks which have motion discontinuities or motion failure, into N different blocks, and specify new MVs for these new blocks. As described in the method [3], the block is divided two arbitrary shaped blocks which must be contiguous and must be adjacent with the block which MV it takes. Directional filters are used to find the motion discontinuities and to select the blocks for segmentation.

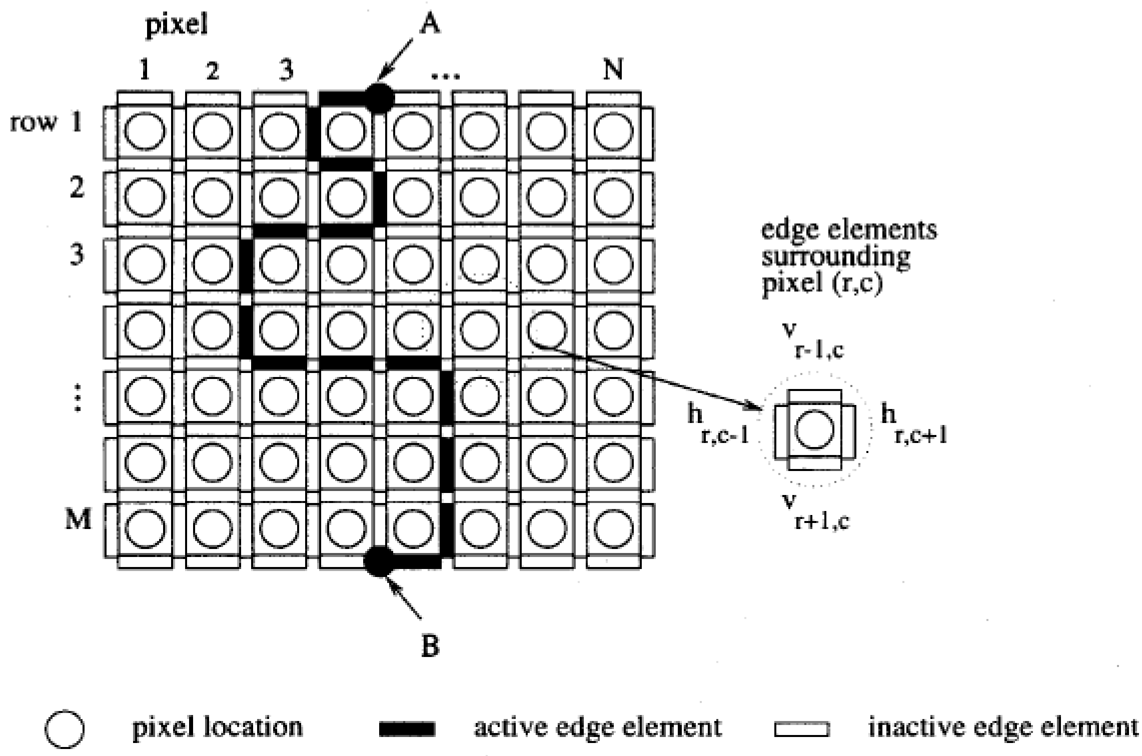

Figure 2.5: Block segmentation boundary

As you see in Figure 2.5, there is an edge between each two pixels. According to results we obtain from the filters, these edges are decided as active or inactive. If there is an activity it means there is a motion discontinuity in that edge, so it is convenience for motion segmentation. The sum of these edges as specified in Figure 2.5 (between A-B) is called as segmentation of the block. Minimizing the sum between A and B maximizes the probability of segmentation. This minimization can be done by using Viterbi Algorithm.

2.1.2 Adaptive Overlapped Block Motion Compensated Interpolation

Overlapped block motion compensation (OBMC) is used in video coding applications to avoid blocking artifacts. It’s an applicable method solving discontinuities that occur because of unreliable MVs or insufficient information in residual area. As described in [4], OBMC is used to solve over-blurred problem [5], and also proposed OBMC in [4] aimed to reduce the complexity of OBMC. Decision making about the block motion along block boundaries is important issue in OBMC. For that purpose, the difference between the current blocks MV and 4 neighbor blocks MVs is calculated. If the difference is big, then the boundary between the blocks will be labeled as blocky and current block will be overlapped with 4x8 and 8x4 by neighbor block predictors, as this situation is represented in Figure 2.6. Current block pixels are evaluated by taking the weighted sum of maximum 3 prediction values that is obtained by MV of current block, MV of left or right neighbor block, MV of above or below neighbor block.

The weighting matrices represented in Figure 2.6(b)-(d) show the prediction of motions generated by current, left or right, current or top blocks. In current block the weights decrease when come close the block boundary, in the other blocks shown in Figure 2.6(b)-(d) weights increase towards block boundary. This provides smooth transition in block boundaries.

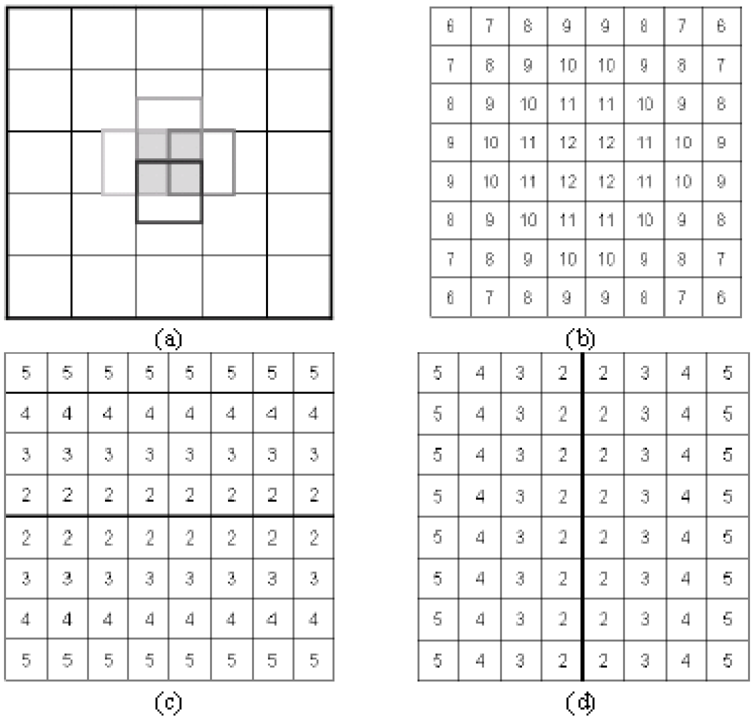

Figure 2.6: Overlapped block motion compensation and weighting matrices of current, left, right, top, bottom blocks

2.1.3 Motion Estimation Using Normalized SAD

Sum of absolute difference (SAD) calculation is effective and important method to analyze the difference of the blocks in current and previous frame. The result we obtain from this analysis provides us to classify the motion vectors according to their reliability levels. SAD calculation way described in [6] depends on normalization of SAD considering the block activity. Figure 2.7 shows the possible activities inside block. This normalization affects the SAD calculation, so that it directly affects the reliability of MV when we compare the SAD of MV with a threshold value.

The normalized SAD calculation is done as follow;

$$SAD_{NORM}(x,y) = SAD(x,y)/ACT(x,y)$$

$$ACT(x, y) = \sqrt{\sum_{i=0}^{N}\sum_{j=0}^{M}(P(x+i, y+j) - AVE(x,y))^2}$$

$$\text{AVE}(x,y) = \left\{ \sum_{i=0}^{N} \sum_{j=0}^{M} P(x+i,\ y+j) \right\} / NM \qquad (2.3)$$

where ACT(x,y) is deviation value and AVE is average value of pixels.

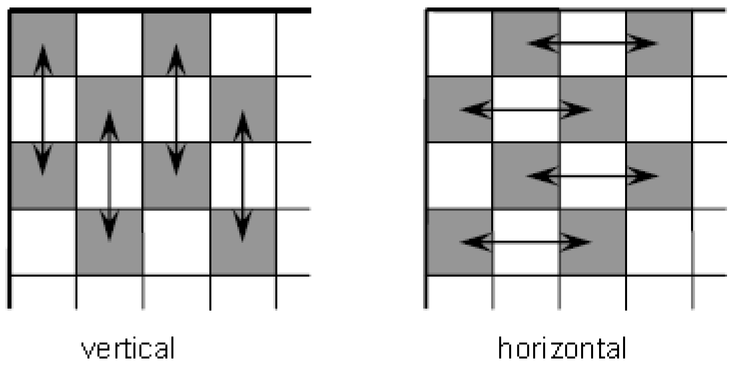

Figure 2.7: Pixel activity which is used in normalized SAD calculation

2.2 Block Matching Motion Estimation

The most important issue in MCFI is to select the correct motion vector. If the interpolation is made by using correct MVs, then we obtain the true motion. Otherwise, different kind of artifacts (blurring, ghost effect, sharp discontinuities, structural deformation, etc) may be included in the video.

In motion compensation methods, usually block based motion estimation (BBME) is used to decrease temporal redundancy. Block-based motion compensation has also been adopted in the international standards of H.261 and MPEG 1-2 for the video compression.

Block matching algorithm that has low hardware complexity is used as the first step to find the reliable MVs. In this manner, the reliability of motion vectors must be decided correctly by using block matching algorithm (BMA), and additionally true motion vectors must be estimated in the areas that motion information is not available or insufficient. However, BMA is not sufficcient to find the correct MVs, so the post processing techniques that are also block based are used to correct unreliable MVs.

In BMA, we divide the image into non-overlapping small regions (blocks), we characterize the motion in each block by a parametric model, e.g., constant, affine, or bilinear. If block is small enough, then this model can be quite accurate. The block shape could be arbitrary, but square shape is used exclusively. The motion in each block is taken as constant considering the simplest case, and the complete motion is assumed as pure translation, this is called block-wise transitional model. Motion estimation aims to find the MV for each block. This algorithm is called bock matching algorithm.

In BBME, the sizes of the blocks for which MVs are identified are selectable. When the size of the block is small, we'll be able to minimize residual energy which is represented as prediction difference between anchor and tracked block. On the other hand, if we select bigger block size, we will have more possibility to have true motion vectors. Video coding standards allow different block sizes; MPEG-4 supports 8x8 and 16x16 block sizes, H.264/AVC allows 4x4 and 8x8 block sizes. In our work we choose 8x8 block size, so the prediction residues are constructed according to 8x8 block size.

The block matching algorithm finds the best matching block in the tracked frame for each block in the anchor frame, as shown in Figure 2.8.

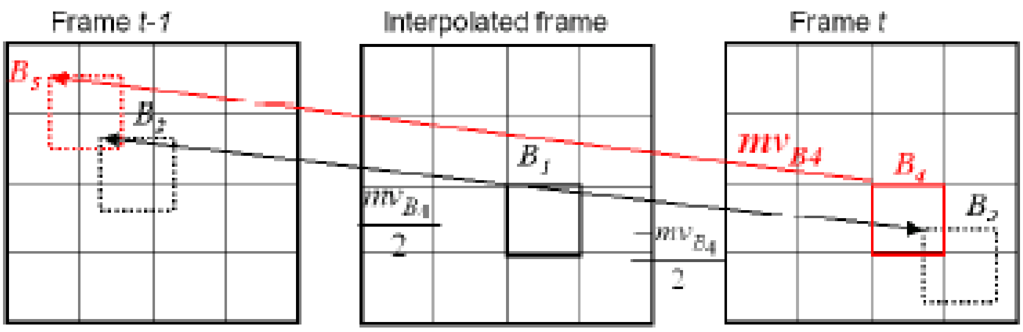

Figure 2.8: Generalized block matching algorithm

In block motion estimation, different kind of search procedures can be applied to find the MV. Computational complexity for search procedures is very important that it takes big amount of computational force. Conventional block matching algorithms such as FSBMA(Full Search Block Matching Algorithm) need very big amount of computational power at receiver. Therefore, we focus on fast search algorithms by accepting some loss in estimation accuracy. In our work, we use hexagon-based search (HEXBS) algorithm which has an advantage on speed, distortion minimization and computational power. The number of search points is the basic concept to understand about the search procedure's computational complexity. The low number of search points mean fast search algorithm. Hexagonal search start with 7 point hexagon at center of block(0,0). As a first the search is done for these 7 points. If the minimum block distortion(MBD) is found at the center, then the search is done within the inner hexagon to get the best candidate search point. If we can not obtain the best search point in center of the big hexagon, we draw another hexagon pattern around the search point which we obtain the MBD value from the first search. This process continues until we obtain the best candidate search point in the center of the hexagon. After we obtain it, we finalize the process with the search within the inner hexagon. The MBD is calculated related to mean absolute difference. In the example shown below, the MV is located at the (+4,-4), and the number of the search points in this process is 20.

After we obtain the MVs, the analysis should be done to understand about which of these vectors are correct. The vectors that we obtain by hexagonal search may not be correct. The block motion can be affected by different kind of factors such as noise, motion blurring, complex 3-D motion that we can not take these factors in consideration when we use hexagon search. Therefore, we should classify the vectors according to their reliability, and then the post processing techniques should be applied to improve the vectors that we do not trust its reliability.

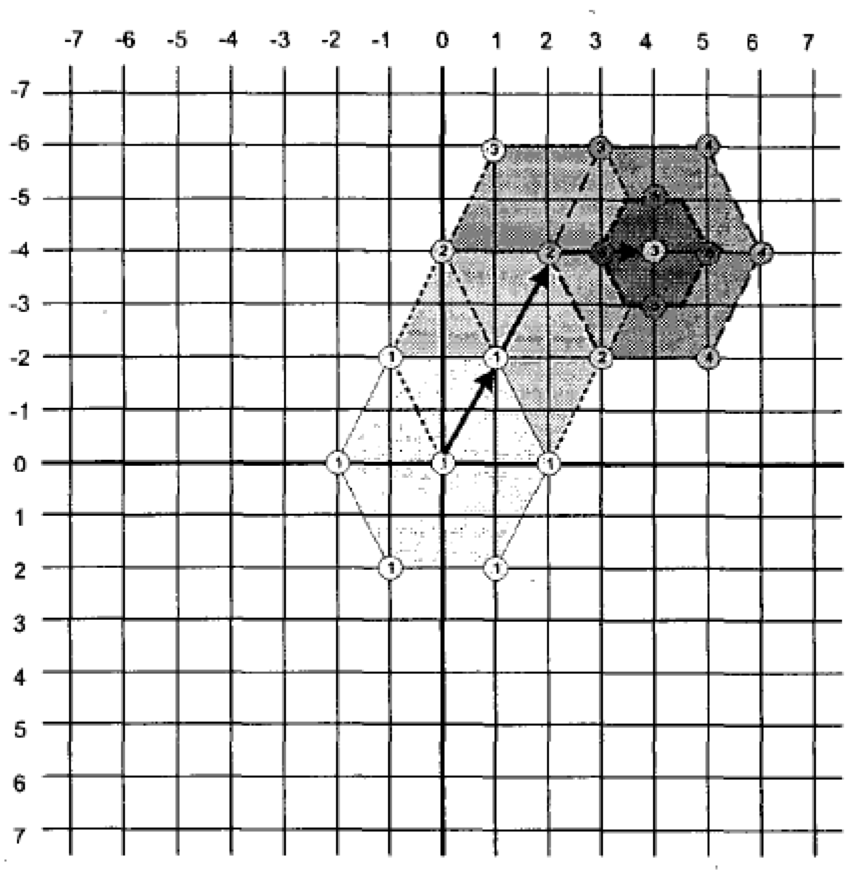

Figure 2.9: Process of hexagon-based search (HEXBS) algorithm

2.3 Motion Vector Classification Based on SAD and BAD Values

The most important issue for an interpolation of a video sequence is to have the correct motion information. The conventional algorithms like block matching algorithm may end up with wrong motion information.

For the aim of understanding about the reliability of motion vectors, we should check the difference of a specific block located in the previous and current frames. The most effective way of representing the difference of the blocks is calculating sum of absolute difference (SAD) between the current frame and previous frame. This method is applied by taking the sum of differences of each pixel in the block. We can also call this as sum of prediction error. The other application is boundary absolute difference (BAD) which measures the smoothness between the edges of the interior and exterior of the block. We can formulize the SAD and BAD between the blocks by the defined equations below;

$$SAD_B = \sum_{i=0}^{7} \sum_{j=0}^{7} | f_{n-1}(m_{B_1} + i, n_{B_1} + j) - f_n(m_{B_2} + i, n_{B_2} + j\ ) | \qquad (2.4\text{ -a})$$

$$BAD_B = \begin{pmatrix} \sum_{j=0}^{7} (| f_{n-1}(m_{B_1}\ , n_{B_1} + j) - f_n(m_{B_2} - 1\ , n_{B_2} + j) | + \\ | f_{n-1}(m_{B_1} + 7\ , n_{B_1} + j\ ) - f_n(m_{B_2} + 8\ , n_{B_2} + j) |) \end{pmatrix} +$$

$$\begin{pmatrix} \sum_{i=0}^{7} (| f_{n-1}(m_{B_1} + i, n_{B_1}) - f_n(m_{B_2} + i\ \ , n_{B_2} - 1) | + \\ + f_{n-1}(m_{B_2} + i\ \ , n_{B_2} + 7) - f_n(m_{B_2} + i\ , n_{B_2} + 8) |) \end{pmatrix}, \qquad (2.4\text{-b})$$

$f_n(x, y)$: Pixel intensity at (x, y) coordinate in the time instance n

f_n : Current frame

f_{n-1} : Previous frame

B : Block

n : The time domain indexing

(i,j) : Spatial domain indexing of the pixels in the block

(n_b, m_b): Coordinate for 8x8 block B

These SAD and BAD values are compared with some specific threshold values.

If SAD > Th or BAD > Th then MV is classified as unreliable

If SAD < Th or BAD < Th then MV is classified as reliable

Chapter 3

The Multi-Stage Motion Vector Processing Method

As explained in above sections, there are lots of methods developed for MCFI. All of them have advantages over direct interpolation, but they're not sufficient to reach the image quality we aim to achieve. Motion vector processing that is applied to correct unreliable vector is essential and very important to minimize the artifacts in the interpolated frame

In this chapter, we explain a successful method [1] developed for motion vector improvement. The method includes several steps, and in each step MVs reliabilities increase in some amount. The MV processing steps are described and formalized in the subsections. The results of each step are also discussed in subsections. The algorithm is based on the successful method described in [1] and is implemented by Burak Çizmeci and Assist.Prof, Hasan Fehmi Ates. Also, an additional step is applied for better frame synthesis at the end. The description of this step is done in 3.5. At the end of the section, we discuss the results that we obtain after all steps are applied, and we explain the insufficient ways of the method, especially for covered and uncovered areas.

In this method, as a first we classify the vectors according to their reliability level by analyzing the image consisting of residual energies. Residual energy is similar to SAD and represent us the difference between the previous frame and current frame for the same block. We can find the residual energies by taking the sum of absolute differences for prediction errors of each pixel. Even though, the motion estimation usually is done in luminance domain; we consider both luminance and chrominance residues to obtain more accurate information regarding the MVs reliability. Sometimes MV is classified as reliable because of luminance difference is minimum, but colors are mismatched. Using also the chrominance information provides us identification of these unreliable MVs. You can see the effect of chrominance and luminance on residual energy in the equation below;

$$E_{m,n} = \sum_{(i,j)\in \mathbf{b}^{Y}_{m,n}} | r_Y(i,j) | + \alpha \cdot \left(\sum_{(i,j)\in \mathbf{b}^{Cb}_{m,n}} | r_{Cb}(i,j) | + \sum_{(i,j)\in \mathbf{b}^{Cr}_{m,n}} | r_{Cr}(i,j) | \right) \qquad (2.5)$$

These residual energies are used to classify the motion vectors of each block as reliable, possibly reliable or unreliable. We compare the residual energy of the block with a specified threshold value. High residual energies are usually seen at motion boundaries and at corners of moving object, so that the vector whose residual energy is higher than threshold is classified as unreliable. These are the areas where artifacts usually occur in the interpolated frame. This unreliable vector is supposed to be in the motion boundary, so that the adjacent block vectors are also classified as possibly unreliable. These adjacent vectors are not classified as reliable even if their residual energies are below threshold. Only the vectors which are not classified yet and the ones that have residual energy lower than threshold are specified as reliable. You can see the details about classifying vectors in Equation 2.6. These vector regions that are constructed according to vectors reliability give us information about in which part of the interpolated frame we can face artifacts

$$MVRM(m,n) = \begin{cases} L_1, & \text{if } E_{m,n} \geq \varepsilon_1, \\ L_2, & \text{if any MV in the same MB or} \\ & \text{in the adjacent MBs} \in L_1, \\ L_3, & \text{otherwise.} \end{cases} \qquad (2.6)$$

In this equation; L_1 is defined as unreliable vector set

L_2 is defined as possibly reliable vector set

L_3 is defined as reliable vector set

After finding the residual areas, we apply merging process for the blocks to understand and to use residual energy connection of these areas. In merging process, we analyze all the motion blocks that have unreliable motion vectors. Firstly, we observe two adjacent blocks. If they‘ve unreliable motion vectors that are connected to each other in vertical, horizontal or diagonal directions, then we merge these blocks. Otherwise, the block that its unreliable vector has not any connection with the adjacent blocks' unreliable vectors is hold as single 16x16 blocks.

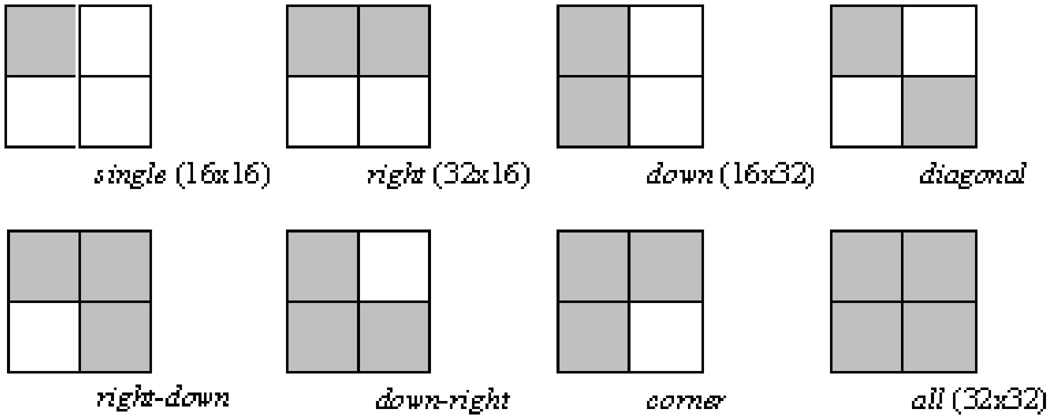

Figure 3.1: Possible merging shapes

The possible options for the merged blocks of 16x16 can be seen in Figure 3.1. As you see, it's not preferable to increase the block size much more than 32x32. Increasing the block size so much may prevent to get a quality interpolated frame, because it becomes harder to represent all the motion in block with a single MV when its' size increases any higher (In our case we select 32x32 as maximum). Therefore, correction of these unreliable vectors specified for bigger block size is always much more difficult. The motion that is represented in smaller block size can always carry more accurate information.

Motion block merging map represents us these merged blocks, so that we can specify one motion vector for these combined blocks.

At the end of all these MVs classification process, all the unreliable vectors are defined. For these unreliable MVs the post-processing techniques will be applied to be able to improve them. The possible artifacts that we can obtain at the final interpolated image because of these unreliable MVs are to be eliminated by using post processing techniques,.

In this mute-stage method that you can see the block diagram in Figure 3.2, we start the process by selecting the best MV of merged group from its own or neighboring reliable and possibly reliable vectors. Then, by taking the bidirectional prediction difference of MVs, we perform MV reliability reclassification. After that, we carry our refinement of the unreliable MVs that we obtain from reclassification stage. Then, MV smoothing is applied to decrease blocking artifacts. After smoothing process, the improved interpolation stage is implemented. This additional stage aims to improve the synthesized block by taking the weighted average of prediction value of center macro block and neighboring MBs. After each process, we obtain a new vector field and the block size reduces from 32x32 to 8x8, and finally to 4x4.

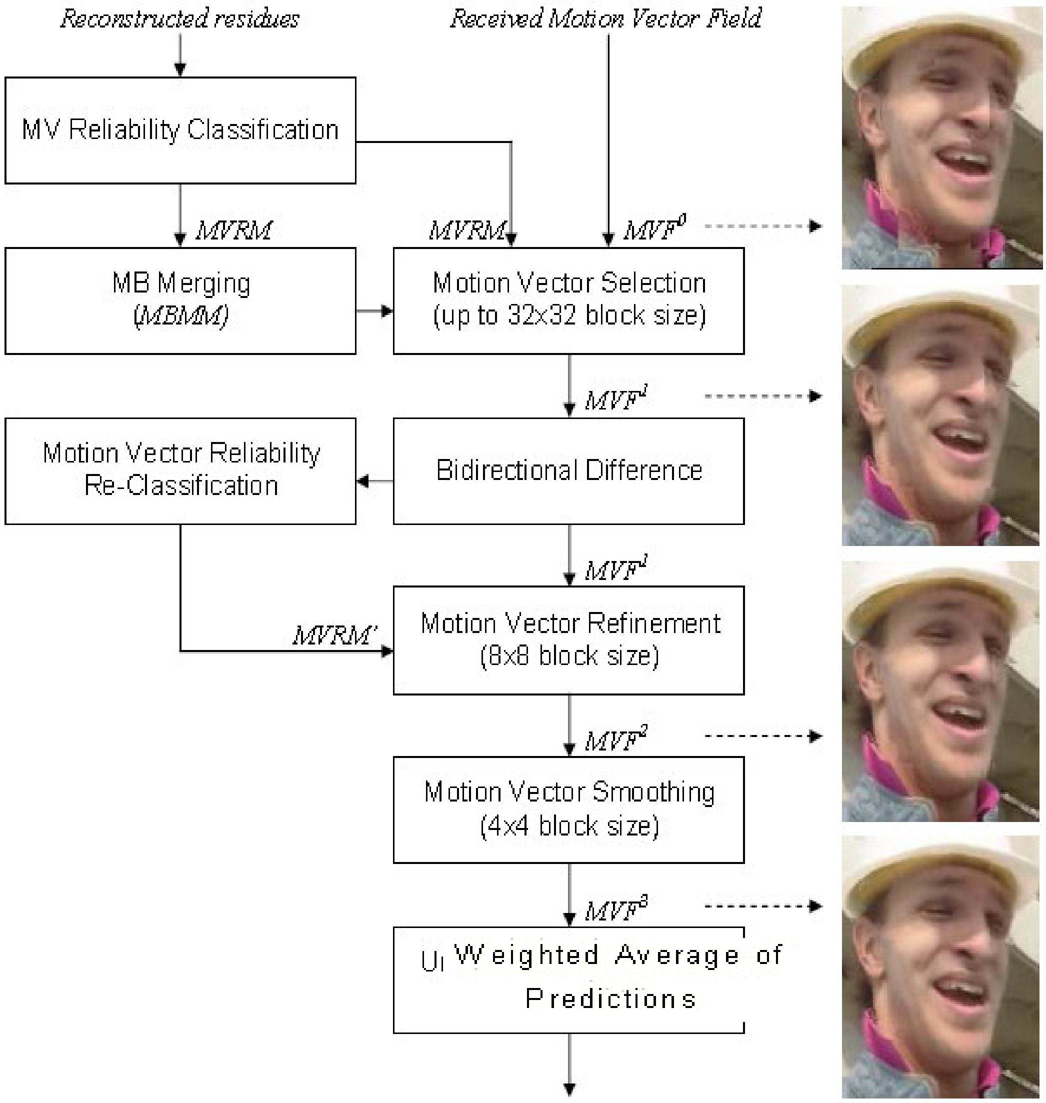

Figure 3.2: Block diagram of multi-stage vector processing method

MVF^k represents updated motion vector field after each process

3.1 Motion Vector Selection

In merging process, we aim to keep boundary information and the entirety of moving object structure by specifying single MV for them. By observing the MB merging map, we consider merged block MV and adjacent MVs from neighboring merged blocks as MV candidates for the current block. Then, we choose the best MV among all of them which minimizes the average absolute bidirectional prediction difference (ABPD) between forward and backward predictions.

$$ABPD(v) \;=\; \frac{1}{N_G}\sum_{x,y\in G} | f_{t-1}(x+\frac{1}{2}v_x, y+\frac{1}{2}v_y) - f_{t+1}(x-\frac{1}{2}v_x, y-\frac{1}{2}v_y) | . \qquad (3.1)$$

$v = (v_x; v_y)$ is set of vectors

Note: Both luminance and chrominance information are considered in the equation

After finding the best MV by minimizing ABPD, we compare it with a threshold value E1. If the value is below E1 then we replace the merged block old MV with the new one. If the value is equal or higher than E1, then we skip this merged block. We wait for the adjacent MVs of neighbor merged blocks to be updated with the better ones. This will help to specify MV for that merged block. If still there is no solution, we specify the best MV in motion vector refinement process. MV selection continues until a new MV is specified for all merged blocks or until number of iterations reached. If we increase the E1 value in the next iterations, then obviously all MVs of merged blocks will be assigned with new MV, because ABPD will be below threshold

By the way, these new MVs are not efficient to show the local details in the image as they are block-based. We apply the other stages MV reclassification, MV refinement, MV smoothing to be able find best MV for the block.

3.2 Motion Vector Re-classification Based on BPD

Motion selection process provides us to specify a new MV for blocks that have high residual energy. The MV that we obtain from the MV selection gives us information about the main motion. On the other hand, it’s highly possible that this block may include multiple motions. It can be understood by evaluating the big difference between forward and backward predictions. If the distance is sufficiently big, then there should be multiple motions in the block. In this step, we find the BPD of MVs that are classified as possibly reliable and unreliable in first classification. We do not use this step for the reliable ones.

When we make the re-classification of MVs by using new energy analysis based on BPD applied for 8x8 block size, we consider both the luminance and chrominance characteristics of image together with their changeable weights

$$BPD_{m,n} = BPD^{Y}_{m,n} + \alpha \cdot (BPD^{Cb}_{m,n} + BPD^{Cr}_{m,n}), \tag{3.2}$$

After finding the BPD of MVs, we compare these values with a certain threshold E3, and we decide whether these MVs are reliable or not. The possibly reliable list is not used in this section

$$MVRM(m,n) = \begin{cases} L_1, & if\ BPD(m,n) \geq \varepsilon_3, \\ L_3 & if\ BPD(m,n) < \varepsilon_3, \end{cases} \tag{3.3}$$

The difference between the forward and backward prediction is generally higher than initial prediction error, so we specify bigger threshold value to find the MVs that may cause significant blocking artifacts. The new unreliable MVs in the new classification map will be used for MV refinement process.

3.3 Motion Vector Refinement

The unreliable vectors in new classification map are corrected by using a similarity and reliability constrained vector median filter;

$$v^{*}_{m,n} = \arg\min_{v \in S} \sum_{i=m-1}^{m+1} \sum_{j=n-1}^{n+1} \omega_{i,j} \|v - v_{i,j}\|_2,$$

where

$$\omega_{i,j} = \begin{cases} 0, & if\ MV\ RM(i,j) = L_1, \\ 1, & if\ MV\ RM(i,j) = L_3\ and\ d_{i,j} > \varepsilon_4. \end{cases}$$

$$d_{i,j} = 1 - \frac{v_{m,n} \cdot v_{i,j}}{|v_{m,n}||v_{i,j}|} = 1 - \cos\theta, \qquad (3.4)$$

We find the distance between the original MV and its candidate MV, and then we compare it with a threshold value E4 to decide the similarity between each other. On the other hand, we do not prefer the MV to be same or similar with the adjacent one. The reason of specifying this MV as unreliable is because of the possibility of this MB to have dissimilar motion or belong to another object. Considering this situation, the vector median filter chooses the most possible MV between the MVs that passes the similarity check

Motion refinement is not applicable if MB has many subblocks with high prediction error difference. There should be low percentage amount of MVs in the block, to be able to perform refinement. Motion refinement aims to correct the MVs that could not be corrected during vector selection, and vector re-classification. These steps provide the major MV for the MB; motion refinement can correct only some of MVs that are located at motion boundary. If the block has big amount of unreliable MVs, the reason may be luminance and chrominance change; and to apply motion refinement in such situation may break the original structure.

In this manner, the BPD of the candidate MV should be lower than the original one, otherwise different candidate MV should be waited during the next iterations. If the MV can not be corrected to the main motion until the end of the iterations, then this unreliable MV is modified in 4x4 block size by smoothing process. If these unreliable MVs are not updated in smoothing process too, then this will cause important artifacts in the interpolated frames.

3.4 Motion Vector Smoothing

In this step, we aim to make the motion field smoother that means it will be closer to the physical motion field. The vectors that reach to this stage as unreliable cause blocking artifacts that originates from discontinuities in the motion field. To be able to reduce the blocking artifact we perform vector smoothing. When we perform this method, we use center MV and its adjacent MVs. Some of the vectors do not have any relation with its neighbors so that considerable blocking artifacts occur and the image quality reduces. In this method, each 8x8 block is divided into four 4x4 sub blocks to have motion field in a smoother scale. Then; we obtain the motion vectors of these blocks by minimizing smoothness measure Ψ. This smoothness measure is calculated by the sum of smoothness measures between the center MV and its adjacent MVs in north, south, east, west, diagonal and center directions.

$$\Psi = \Psi_N + \Psi_S + \Psi_E + \Psi_W + \Psi_D + \Psi_C . \quad (3.5)$$

$$\Psi_N = \| v^1_{m,n} - v^3_{m,n-1} \|^2 + \| v^2_{m,n} - v^4_{m,n-1} \|^2 + \| v^3_{m,n} - v^1_{m,n} \|^2 + \| v^4_{m,n} - v^2_{m,n} \|^2$$

We process the smoothing for the unreliable and possibly unreliable vectors which are the main reason of blocking artifacts. We consider reliability of the vectors according to their initial situation, so that we also use MVs corrected until this step, too.

As a result, although smoothing may cause some loss of estimation accuracy, on the other hand providing proper motion vectors with its neighbors decrease blocking artifact in considerable amount for interpolated frame.

3.5 Prediction Decision for a Macro Block by Weighted Average Method

This stage is not included in method [1]. This step has been developed and added into algorithm implemented by Burak Çizmeci and Assist.Prof, Hasan Fehmi Ates. The proposed stage aims to decrease the blocking artifact and to have smoother image in the interpolated video sequence. The method [9p] is some way similar to the MV smoothing and it aims to increase the prediction similarity between center block and its adjacent blocks. We apply this method for the blocks whose SAD value between previous and next frames is higher than specified threshold. In this method, we find the final prediction of the center block by taking the weighted average of prediction provided by the center MV and predictions provided by the 8 adjacent MVs as shown Figure 3.3. Weights are inversely proportional to SAD of each MV. The block which has higher SAD will have lower effect, while the block with lower SAD value will have more weight in the algorithm.

1	**2**	**3**
4	**Center MB**	**5**
6	**7**	**8**

Figure 3.3: Taking the weighted average of prediction values of center and its' neighbor vectors

We can define taking the weighted average of prediction as follows;

If $\text{SAD}_i > \text{Th}$

$$\hat{f}_t = \sum_{k=0}^{8} w_k \hat{f}_k \qquad \text{where} \qquad \omega^k = \frac{1/SAD_k}{\sum_{k=0}^{8} 1/SAD_k} \qquad (3.6)$$

$\hat{f}_k$ is the motion compensated prediction as defined in equation 2.1 using the motion vector of k^{th} neighboring MB.

After we apply this method, the resulting block will be more proper for the motion field, so we can obtain smoother regions with reduced blocking artifact.

3.6 Results of the Multi-Stage Method

This hierarchical MV processing algorithm has a result of getting quality improved video at the interpolated frame. Also, it provides reduction in computational complexity by the efficient methods used to find and improve the unreliable MVs that cause blocking artifact in video sequence. The method explained in paper [1] has been modified by adding extra stage that improves the smoothness of the interpolated frame. When the effects of this multi-stage method on some examples are observed, it can easily be seen that proposed method provides better visual quality than the other MCFI methods. Also, it has better PSNR and SSIM values. The method is successful to be used in complex textures, to hold the structural integrity and to eliminate the visual artifacts. As a result, this robust and low complexity method has lots of advantages compared to the conventional methods.

By the way, it should be still developed further to have better image quality in the interpolated frame. Even though, the applied multi-stage method is low-complexity due to its specification of achieving object based frame interpolation without actual edge detection or motion segmentation; it's especially insufficient to prevent the visual artifacts that occur in boundaries of overlapping objects (or background). This problem is common in most of the MCFI methods, and new approaches are needed to perform for the solution of the problem. The problem can be called as "occlusion problem" where the problematic area of the frame is called as "covered/uncovered region" or "occluded region". In Chapter 4, we analyze the occlusion problem in detail, and we suggest a new algorithm for artifact reduction purpose in these regions.

Chapter 4
New Proposal for Covered and Uncovered Regions

For the purpose of increasing the image quality in digital video systems, we should focus on covered and uncovered problem in MCFI. Occlusion problem is the main reason of the blocking artifacts likewise motion discontinuities, high frequency artifacts...etc, so that solution of the occlusion problem should have priority for the development of interpolation process.

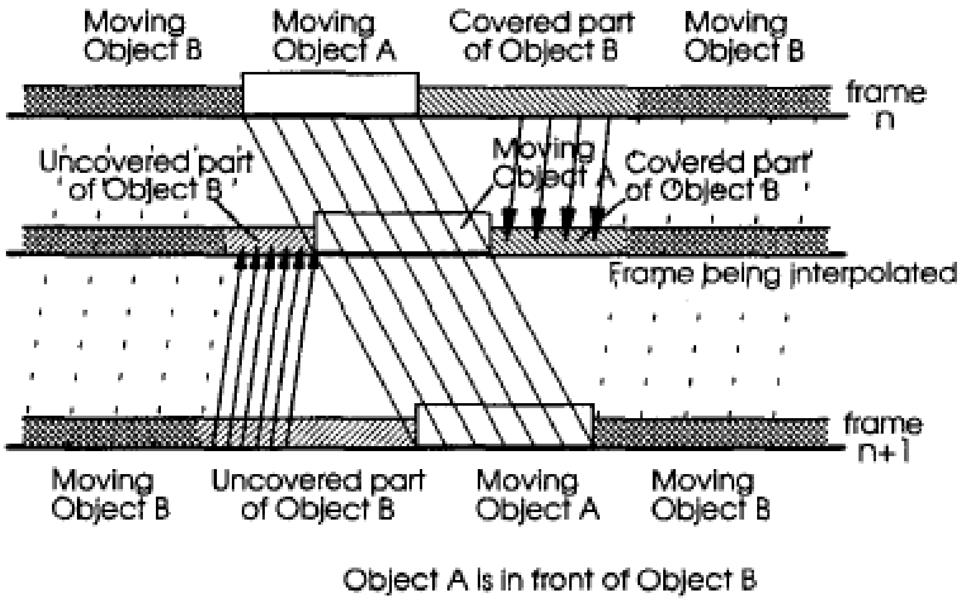

Figure 4.1: Representation of occlusion problem

Occlusion areas are located at the boundary of moving object, and these are the areas that the main errors emerge in the interpolated frame. The reason of this common problem is it's not easy to find the correct motion vectors for occluded regions. When we skip a frame, we use bidirectional prediction difference, and it's assumed that the pixels of the interpolated frame can be found at both forward and backward frames. If there is a motion in the picture, it will be highly possible that the object/background may be covered or uncovered in one of the two neighboring frames. You can see problem more clearly in Figure 4.1. Therefore, the motion vector that we aim to specify for the occluded region will not be correct. These incorrect motion vectors specified for occluded regions will cause an artifact in the interpolated video sequence. For the ability of specifying correct vectors for the occluded regions, many different approaches have been suggested and still new methods are being developed for the artifact reduction in these regions.

4.1 Overview of Methods Developed for Occluded Regions

Before specifying MV for the occluded regions, as a first we should decide about the area whether it is covered or uncovered. The moving objects boundaries are the places that you can face the occlusion problem. We can determine the object boundaries according to the residual energies of the blocks as described in Chapter 3. High residual energies are presented at object boundaries and at edges of moving object. If there is extremely high residual energy, this area can be considered as covered or uncovered. The occluded and uncovered area can also be decided by SAD of forward and backward prediction of the blocks as described in [4]. If SAD is high value, then it can be determined that there is such a big difference between previous and next frame, so that occlusion situation may be the reason of this high SAD value.

The most important result of the occlusion problem is hardness of specifying correct motion vector for a block by using bidirectional prediction. Bidirectional prediction depends on the averaging the previous and next frame. In this situation, the motion information will only be correct in either the previous or the next frame. Bidirectional interpolation will normally use the unreliable MV which will cause a blocking artifact or ghost effect in the interpolated frame. Considering occluded regions, the reliable information to select MV is usually derived from past frames for covered areas, and it's in the future frames for uncovered areas. If information obtained from both the past and future frames is used, then the interpolated frame will represent wrong information. In this case, the new picture will be the mixture of background and foreground objects. Therefore, in most of the methods developed to solve occlusion problem, it's usually preferred to use uni-directional prediction (forward or backward) for covered or uncovered regions.

In paper [4], it is explained that after checking the SAD and deciding for occluded region, then BAD is calculated for both previous and next frames. According to the minimum BAD value the next or previous frame is chosen to be used for interpolation of new frame.

Different kind of motion compensated frame interpolation algorithms in covered and uncovered areas have been developed to prevent the artifact occuring because of this problem. The artifacts are called as “Halo” and may be in several ways such as high frequency artifacts, object ghosts and blurring.

4.1.1 Applying Median Filters for Selecting Minimum Luminance Value

In paper [7], a non-linear method using median filters is described. As a main idea of this method, there are several vectors for every position. Every vectors candidate pixels will be classified according to their luminance values and minimum one is selected to represent all the pixels luminance value.

$$a_0 = \text{med}\left[a_{-1}, Av_1, a_1\right]$$

with

$$A_v = \frac{1}{2}(F(\vec{x},n) + F(\vec{x},n+1))$$

and

$$\text{Med}(a,b,c) = \begin{cases} a, \textit{if}\,(b \le a \le c)\ or\ (c \le a \le b) \\ b, \textit{if}\,(a \le b \le c)\ or\ (c \le b \le a) \\ \quad c, \textit{otherwise} \end{cases} \tag{4.1}$$

If one of the MV is correct, then the output will also be correct, because all the candidate vectors will have same luminance value. If the vectors are not correct, then the average of the luminance between previous and next frame will be taken.
The method explained in [7] is called as “graceful degradation”, but especially if none of the MV is correct, then sharp artifacts will be seen at the motion boundaries. The advantage of the method is that, it can be more successful than standard interpolation in stationary parts such as stationary text of subtitles. The reason of this case is one of MVs (null vector) will be correct for this types of stationary parts.

4.1.2 Motion Compensated Center Weight Medians Application

In paper [8], motion compensated center weight medians are described to be applied in artifact reduction in occluded areas. In this method, there is a tolerance for incorrect MVs. The aperture of the median filter includes number of pixels, and in the vicinity of the pixels addressed by the motion vector (such as v_1, v_{-1}), in order to compensate in case the given motion vector wrong. Then, this filter is applied on incorrect MV for which the error of MV must be smaller than filter aperture. If the error in the motion vector is larger than the filter, then the filter will act like a noise filter. It will cause "halo" artifacts and breaking down the details, and these artifacts will ve visible in the interpolated frame.

4.1.3 Uni-Directional Motion Compensation

The other commonly used method is using four neighbor frames of interpolated frame in order to find correct MV and correct luminance value in occluded regions. 2 of 4 are chosen from the past frames, and 2 of 4 are chosen from the future frames. According to the basic idea of this method, at least 2 of the frames will have a good match, so correct interpolation will be possible for the occluded area.

In paper [9], the method by using 4 neighbor frames is described. The basic idea is as follows;

Three motion vectors are defined according to the three match errors.

These match errors are;

$$\in_{pre} (\vec{D}, \vec{x}, n) = | a_{-2} - a_{-1} |$$

$$\in_{cur} (\vec{D}, \vec{x}, n+1) = | a_{-1} - a_{1} |$$

$$\in_{next} (\vec{D}, \vec{x}, n+2) = | a_{1} - a_{2} | \tag{4.2}$$

Where $a_2 = F(\vec{x} + (2-\tau)\vec{D}_\tau, n+2)$, $\qquad a_{-2} = F(\vec{x} - (1+\tau)\vec{D}_\tau, n-1)$

Then, the motion vector that gives the least error value among three of them is selected as proper MV. If there is an occlusion in this area, the $\in_c$ must not have the lowest value. The area can be defined as covered if $\in_{pre}$ is the lowest, and it can be defined as uncovered if $\in_{next}$ is the lowest value. By the way, this method does not provide accuracy in specifying the correct MV and luminance value for the covered or uncovered area. It is low possibility finding low match errors for these kinds of regions. The reason is for providing low match error, the motion vectors for the picture pairs must be correct or the luminance value in the pictures must not change or there must not be any kinds of change in pixels of consecutive frames. If one of these specifications is not included, then all three match errors will be very high, so that we will possibly not specify correct MV for that area. The situation of having three high match errors usually occurs when only one pixel of the picture is considered for match error calculation.

As it's has been understood that uni-directional motion compensation (forward or backward) is commonly preferred for specifying the correct motion vector for covered and uncovered regions. It is also applied in the method explained in paper [6]. Adaptive interpolation using four frames can be seen in Figure 4.2. The frame interpolation between 1 and 2 can be done by using the compensations between 0-1 and 2-3.

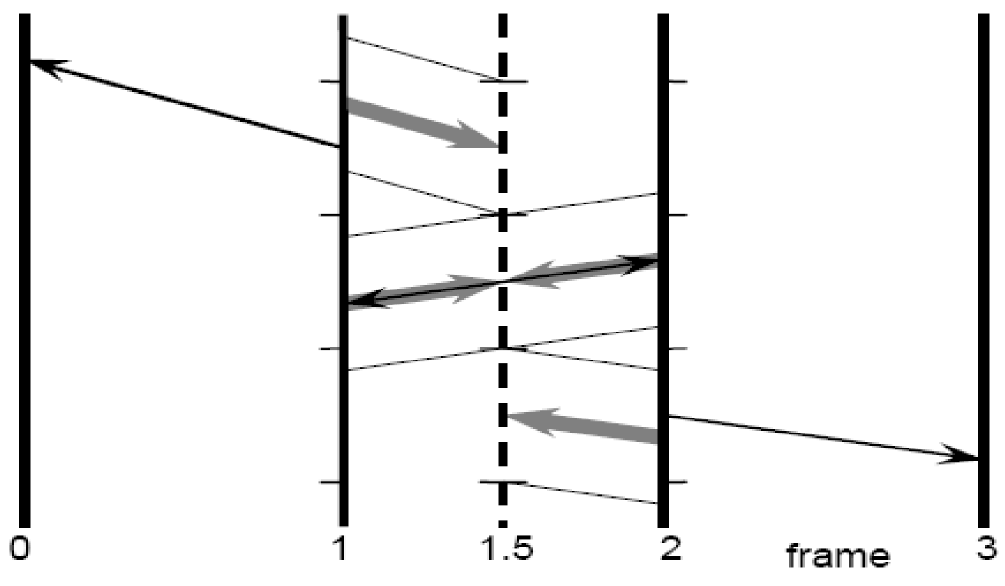

Figure 4.2: Adaptive interpolation for occluded regions by using four frames

4.1.4 Block Segmentation Application

Another approach for covering and uncovering is applied in block segmentation to be able to find best segmentation when there is uncovering. According to the method described in [3], the block is segmented into three regions; left, right and uncovered regions. The uncovered region is bounded between by left and right regions. As it's represented in Figure 4.3, the part of block B that goes in left or right direction is uncovered region, and we can consider these parts as the complementary of $B^{/}$ and $B^{//}$. For the reason of this complementary situation, the same technique is applied for block segmentation in both uncovering or without uncovering case, described in Chapter 2.

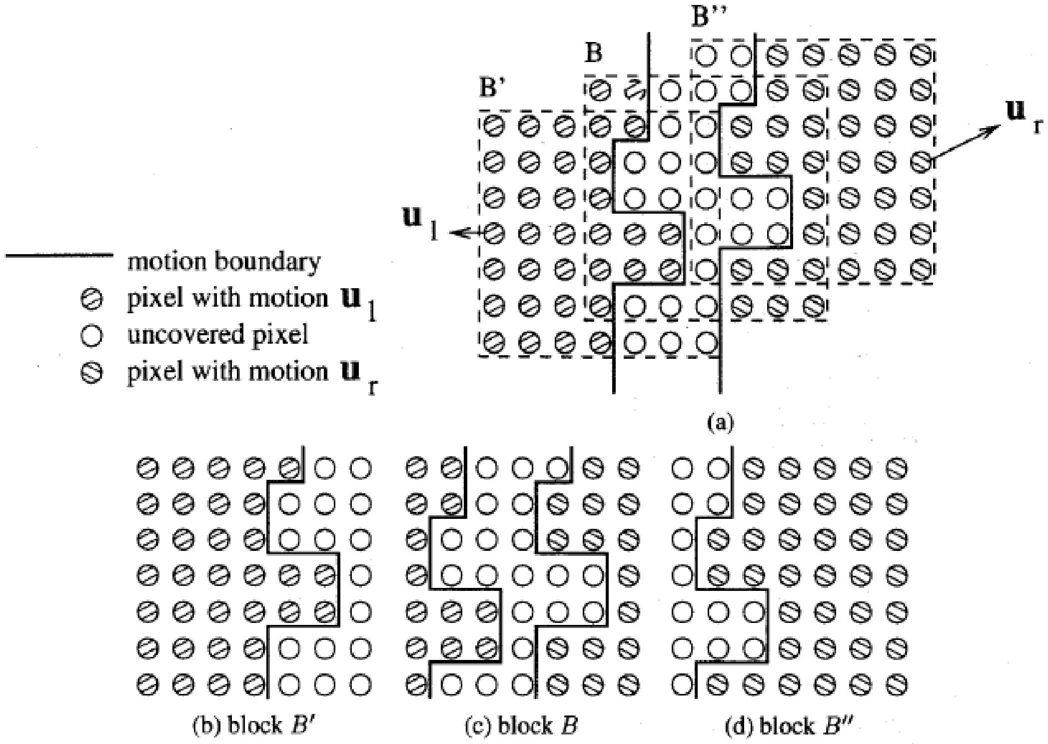

Figure 4.3: Block segmentation in uncovered regions

a) Uncovering region occurrence (b)-(d) Blocks B, $B^{/}$ and $B^{//}$

4.1.5 Occ-3F Method for Interpolation of Covered and Uncovered Regions

One of the most successful methods proposed for occlusion problem is Occ-3F method which is described in [2]. In this method, as a first step of the process, the areas that possibly have occlusion problem are decided. The motion discontinuities represent us the motion boundaries of the objects that occlusion problem occur. We use diplacement vectors to decide about the movement of the object. The difference of displacement vectors are calculated and compared with threshold value to be able to determine the motion discontinuities. The collation results also inform us about the problem type is either covered or uncovered.

The motion vectors that are used to calculate displacement difference are assigned for left and right hand side of block in the current field n;

$$\vec{D}_{\tau 1} = \vec{D}_{\tau 1}(\vec{x} - \vec{K}, n)$$

$$\vec{D}_{\tau r}(\vec{x} = \vec{D}_{\tau}(\vec{x} + \vec{K}, n) \qquad \vec{K} = (k,0)^T$$ Where k is constant

The horizontal and vertical displacement difference comparison with a specified threshold value can be defined as follows;

$$|\; x_{\vec{D}_{\tau 1}} - x_{\vec{D}_{\tau r}} \;| > \text{thre or } |\; y_{\vec{D}_{\tau 1}} - y_{\vec{D}_{\tau r}} \;| > \text{thre} \qquad (4.3)$$

If the differences or in common case one of the displacement difference is bigger than threshold, then the area is considered as covered or uncovered. The option of being covered or uncovered is selected according to sign value of equation.

$$\vec{D}_{\tau 1}(\vec{x}, n) - \vec{D}_{\tau r}(\vec{x}, n)$$

The area is specifed as uncovering if the result positive, and it s being accepted as covering if the result is negative.

When the picture interpolation is done, the two neighbor pictures and earlier picture are considered in covering situation. $\in_{central}$ and $\in_{covered}$ match errors are calculated for covering situation. On the other hand, for uncovering situation the two neighbor pictures and next picture are considered, and$\in_{central}$, $\in_{uncovered}$ match errors are calculated.

$$\in_{central}(\vec{D}_\tau, \vec{x}, n+\tau) = \sum_{\vec{x} \in b'(\vec{x})} | a_{(-1)} - a_{(1)} |$$

$$\in_{covered}(\vec{D}_\tau, \vec{x}, n+\tau) = \sum_{\vec{x} \in b'(\vec{x})} | a_{(-2)} - a_{(-1)} | \quad (4.4)$$

$$\in_{uncovered}(\vec{D}_\tau, \vec{x}, n+\tau) = \sum_{\vec{x} \in b'(\vec{x})} | a_{(1)} - a_{(2)} |$$ where b' is block with 2x2 pixel.

According to proposal, same match errors for the same pictures are also calculated with a different vector $\vec{D}_2$, too. $\vec{D}_2$ is taken from neighborhood of the position to be interpolated. $\vec{D}$ and $\vec{D}_2$ are supposed to describe foreground and background velocities. By using the match errors obtained by using these vectors, b' blocks' pixels are decided whether it's belong to foreground or background object. If the minimum error is found between the two central pictures, then b' is supposed to be belonging by foreground object. Otherwise, it's supposed to be belonging by background object. The one of these two vectors that gives the minimum error is assigned for pixels in block b'. After specifying the MV $\vec{D}_b$ for the block, then interpolation can be made according to the classification information of the area that may be one of among covered, uncovered and foreground. As you see in the equation below, when the interpolation of new frame is made, for the covering situation only previous pictures are used, for the uncovering situation only next pictures are used. In all other cases, the neighbor frames are used for MCFI.

$$F(\vec{x}, n+\tau) = \begin{cases} \frac{1}{2} F(\vec{x} - (1+\tau)\vec{D}_b, n-1) + F(\vec{x} - \tau)\vec{D}_b, n)) \\ (\text{covering}) \\ \frac{1}{2} F(\vec{x} + (1-\tau)\vec{D}_b, n+1) + F(\vec{x} + (2-\tau)\vec{D}_b\, n+2)), \\ (\text{uncovering}) \\ \frac{1}{2} F(\vec{x} - \tau)\vec{D}_b, n) + F(\vec{x} + (1-\tau)\vec{D}_b\, n+1)), \\ (\text{otherwise}) \end{cases} \quad (4.5)$$

4.2 Weighted Synthesis of Previous and Next Frames in Occluded Regions by Using Errors Obtained From Current, Previous and Next Frames

While occlusion problem is still most important reason of artifacts in MCFI, new methods are needed to be developed for solution of MCFI in occluded regions. As described in the previous section, most of the methods developed for the solution of the problem prefer to use uni-directional interpolation, meaning that only two past or only two future frames are used for interpolation of new frame in occluded areas. These methods do not use both forward and backward compensation at the same time for interpolation in occluded regions. However, there are disadvantages of uni-directional interpolation, so that we can not have accuracy in the results that we obtain by applying these methods in covered and uncovered regions. The main reason of the bad results that are obtained by using these uni-directional methods is the high possibility of not being able to find the good matches in previous or next frames. If the motion vector is wrong, then we can not find the correct matches of the interpolated block in previous or next frames. Therefore, we make the interpolation for that block by synthesizing two wrong blocks obtained from just past 2 frames or just future 2 frames. At the end of this interpolation, the interpolated block will be completely wrong. For that reason, sharp discontinuities, high frequency artifacts...etc that will be easily noticeable in video sequence will occur in the interpolated frame.

In our main proposal (Version I), we focus on method that prevents sharp discontinuities in video sequence by the way of improving the interpolation for occluded regions. In our method, we use both forward and backward compensation like its being done for standard interpolation. The main difference between standard interpolation and our method is that the weights we choose for forward and backward compensations are not equal in our method unlike the same value they take(0.5) in the standard interpolation.

Similar to the common methods which use uni-directional interpolation for occlusion problem; in our method we also use information of four neighbor frames for MCFI in covered and uncovered regions. Different from uni-directional interpolation methods, we do not directly use these frames for interpolation of new frame. We use these frames for the estimation of SAD values that will be used for calculating the weights of forward and backward compensation. You can see the detail related to the SAD calculation in Figure 4.4.

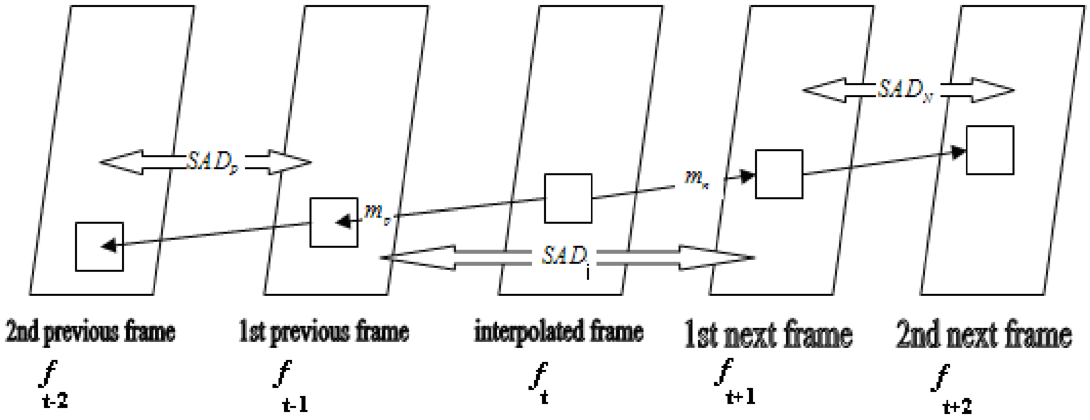

Figure 4.4: Calculation of SAD values that are used for weighted compensation of forward and backward predictions

In our work, we take the method [9p] which is described in Section 3.5 as reference and we develop our proposal by integrating it to this method. In method [9p], the weighted average of prediction values of center block and its' 8 neighbor blocks are calculated to determine the final prediction value of center block. In our proposal, taking the weighted average of backward and forward compensations is added to the method [9p] to provide better results in covered and uncovered regions

As a first step, the SAD initial (SAD_i) value for each block is calculated considering the BPD between previous and next frames as described in Equation 4.6. Same as the method [9p], if we obtain higher SAD value between previous and next frames than the specified threshold, we specify these regions as problematic that it's a high possibility of being an occlusion problem in that area. In our proposal, we call this problematic area as occluded region, and we develop our proposal for the improvement of interpolation in these occluded regions.

$$SAD_i = \frac{1}{N_G} \sum_{x,y \in G} | f_{t-1}(i + \frac{1}{2}v_x, j + \frac{1}{2}v_y) - f_{t+1}(i - \frac{1}{2}v_x, j - \frac{1}{2}v_y) | . \qquad (4.6)$$

$v = (v_x ; v_y)$ is set of vectors obtained from motion vector field between previous and next frames

G is block size

If SAD_i > Th, then our proposal is applied for the interpolation of new frame for the areas which satisfies this condition,

After we decide for the occluded regions, we calculate SAD between previous two frames (SAD_P), and SAD between next two frames (SAD_N) for each occluded center block and its' 8 neighbor blocks like explained in 9p method

$$SAD_P = \frac{1}{N_G} \sum_{x,y \in G} | f_{t-2}(i+v_x, j+v_y) - f_{t-1}(i+\frac{1}{2}v_x, j+\frac{1}{2}v_y) |. \quad (4.7)$$

$$SAD_N = \frac{1}{N_G} \sum_{x,y \in G} | f_{t+2}(i-v_x, j-v_y) - f_{t+1}(i-\frac{1}{2}v_x, j-\frac{1}{2}v_y) |.$$

When the new frame is interpolated, we sum the backward frame with the weight of α_p and forward frame with the weight of α_n as shown in the Equation 4.8.

$$f_t(i,j) = \alpha_p f_{t-1}(i+\frac{1}{2}v_x, j+\frac{1}{2}v_y) + \alpha_n f_{t+1}(i-\frac{1}{2}v_x, j-\frac{1}{2}v_y) |. \quad (4.8)$$

In standard interpolation process, the previous and next frames are taken with same weights;

$\alpha_p = \alpha_n = 0.5$

In our work, we determine different α_p and α_n values for these occluded blocks by using SAD_P, SAD_N and SAD_c values. At the moment, SAD current (SAD_c) is calculated same as SAD_i, but considering the parameter changes in calculation of SAD_i and SAD_c for occluded areas in other versions of our proposal; we use two different parameters from now on.

We take weighted average of the SAD values to determine new weights α_p and α_n as shown in below;

$$\alpha_p = \frac{\left(\frac{1}{SAD_N}\right)+\left(\frac{1}{SAD_C}\right)}{\left(\frac{1}{SAD_N}\right)+\left(\frac{1}{SAD_P}\right)+\left(\frac{2}{SAD_C}\right)}$$

$$\alpha_n = \frac{\left(\frac{1}{SAD_P}\right)+\left(\frac{1}{SAD_C}\right)}{\left(\frac{1}{SAD_P}\right)+\left(\frac{1}{SAD_N}\right)+\left(\frac{2}{SAD_C}\right)} \qquad (4.9)$$

These α values should be inversely proportional with their SAD values, so that if $SAD_P > SAD_N$ then the relation between α values will be $\alpha_p < \alpha_n$

$SAD_P < SAD_N$ then the relation between α values will be $\alpha_p > \alpha_n$

We also add the SAD_c value into the formula, so that α values dependences to SAD_P and SAD_N decreases. α values will be closer to each other, so that smoothness may increase in the interpolation. On the other hand, some loss of accuracy in the interpolated frame may be possible in some cases.

After calculation of α_p and α_n values for the center occluded block and it's 8 neighbors, the final interpolation of a new frame for occluded block is done as shown below;

$$f_t(i,j) = \sum_{k=0}^{9} \omega^k \ \alpha_p f_{t-1}(i+\frac{1}{2}v^k{}_x, j+\frac{1}{2}v^k{}_y) + \omega^k \alpha_n f_{t+1}(i-\frac{1}{2}v^k{}_x, j-\frac{1}{2}v^k{}_y)). \quad (4.10)$$

$$v = (v^k{}_x; v^k{}_y)$$

Where estimation of the "weighted average of SAD values" (ω^k) described in Section 3.6

When different weights apart from 0.5 are specified for f_{t-1} and f_{t+1}, it's aimed to use one of the f_{t-1} and f_{t+1} with higher weight for that occluded region. In construction of new frame, f_{t-1} is used with higher weight than f_{t+1} for covered areas, and f_{t+1} is used with higher weight than f_{t-1} for uncovered areas. The method assumes that for covered areas the true information is usually founded in previous frames, and for uncovered areas the true information is usually founded in next frames. The difference between our method and most of the other methods developed for occlusion problem is that those methods just use f_{t-1}, f_{t-2} information for covered regions and f_{t+1}, f_{t+2} information for uncovered regions; on the other hand our method still uses both f_{t-1}, f_{t+1} information for interpolation in covered and uncovered regions. We apply the covered and uncovered situation into our method by specifiying different α_p and α_n values for each occluded block. The advantage of our method is when just using f_{t-1}, f_{t-2} or just using f_{t+1}, f_{t+2} for the interpolation of f_t, probably you can not find true information in those frames because of wrong MVs. These methods make hard decision, and when making hard decision neighbor blocks can be determined differently as either covered/uncovered. This situation provides inconsistency in the interpolated frame. Therefore, applying this kind of hard decision for interpolated frame may finalize with sharp discontinuties and high frequency artifacts. Our method prevents these kinds of artifacts, and gives us more smooth and reliable results.

4.3 Modified Versions of New Proposal

In our work, we've tested different modifications of our proposal described in Section 4.2. Several versions of original proposal are described in this section. In each version, we focus on different ideas for the solution of occlusion problem. In Chapter 5, we'll observe the improvement in the constructed image when different methods applied.

4.3.1 Using Increased Block Size and Different Vectors for SAD Calculation (Version II)

As it's been described in version I, SAD_P and SAD_N are calculated for determining the weights α_p and α_n. In version II, we increase the block size to obtain more reliable SAD values in occluded regions, because the SAD values calculated by using bigger block size increase consistency in the interpolated frame. Also, different from version I, we use MVs of distant blocks in SAD calculation because of the high possibility of having unreliable vectors in closer vicinity of these problematic regions.

The new SAD_P, SAD_N and SAD_c values will be like as follows;

$$SAD_P = \frac{1}{N_{G_c}} \sum_{x,y \in G_c} | f_{t-2}(i + v_{x\_f}, j + v_{y\_f}) - f_{t-1}(i + \frac{1}{2} v_{x\_f}, y + \frac{1}{2} v_{y\_f}) | . \qquad (4.11)$$

$$SAD_N = \frac{1}{N_{G_c}} \sum_{x,y \in G_c} | f_{t+2}(i - v_{x\_f}, j - v_{y\_f}) - f_{t+1}(i -, \frac{1}{2} v_{x\_f}, y - \frac{1}{2} v_{y\_f}) | .$$

$v = (v_{x\_f}; v_{y\_f})$ is set of vectors further from the center macro block vector

G_c is the size of current macro block and $G_c > G$

SAD_C is also calculated between f_{t-1}, f_{t+1} by using new vectors and new block size.

$$SAD_C = \frac{1}{N_{G_c}} \sum_{x,y \in G_c} | f_{t-1}(i + \frac{1}{2} v_{x\_f}, j + \frac{1}{2} v_{y\_f}) - f_{t+1}(i - \frac{1}{2} v_{x\_f}, y - \frac{1}{2} v_{y\_f}) | .$$

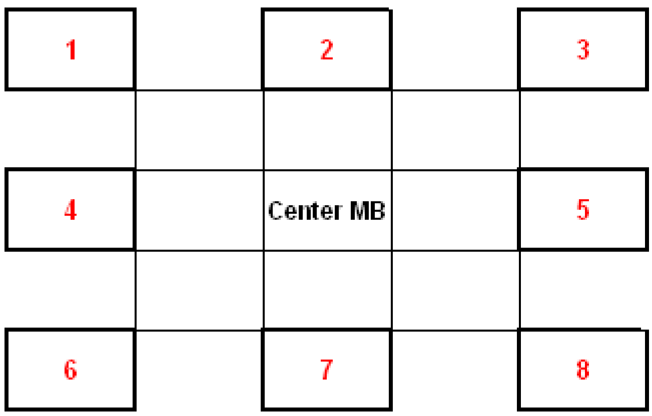

Figure 4.5: Example of searching further macro blocks' vectors

After the calculation of SAD_C, SAD_P and SAD_N values the new α_p and α_n values are determined by using the Equation 4.9. Then, we construct the new frame for the occluded regions by using these weights. You can see formulation of new frames' construction in Equation 4.10.

4.3.2 Searching Further Blocks' Vectors Progressively for Minimum SAD Calculation (Version III)

In version II, we use bigger block size for SAD calculation. The main idea of the method is we obtain more reliable SAD values by using bigger block size and MVs from different MBs which are further from the original block. In version III, we extend the method described in version II by adding progressive vector selection algorithm. We aim to increase the reliability of α values by adding this modification to our proposal.

After we obtain SAD_P, SAD_N and SAD_C values in version II, we make a comparison between SAD_P, SAD_N and SAD_C values.

If $SAD_P < SAD_c$ or $SAD_N < SAD_c$;

Then α values are calculated by using SAD_P, SAD_N and SAD_C values, as in Equation 4.11.

Otherwise, we make progressive vector search around each of the 9 blocks that does not satisfy the condition above.

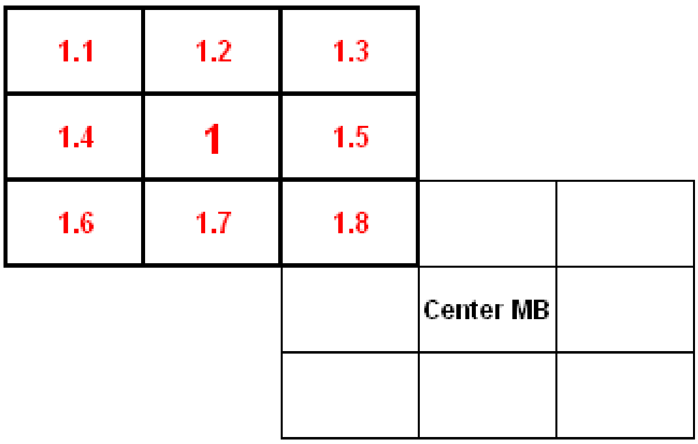

Figure 4.6: Progressive search of blocks to obtain minimum SAD_P or SAD_N values

During the progressive search algorithm, we try to find the block whose MV minimizes one of SAD_P or SAD_N values. For that purpose, we make searches around the each further blocks' vectors which are shown in Figure 4.5 (In our example, one of these further blocks vector is Block 1). We calculate SAD_P and SAD_N value for block 1 and its' neighbor blocks' vectors as shown in Figure 4.6 Among these 9 vectors, we select the vector which has minimum SAD_P or SAD_N.

v_b =argmin ($SAD_P(v)$ or $SAD_N(v)$)

v is vector of the block that satisfies this condition

Then, selected blocks' vector is used for calculation of SAD_P, SAD_N and SAD_C instead of MV of the initial MB (Center MB). These SAD values are used to determine α_p and α_n values as described in Equation 4.9. The construction of the new frame for these areas is done like as shown in Equation 4.10.

4.3.3 Integration of Vectors Obtained from Previous and Next Vector Field into the Weighted Average Prediction (Version IV)

In the method described in 3.5, the prediction decision of a macro block is made by weighted average of 9 predictions. These 9 predictions include prediction of center macro block and predicitions of its' 8 neighbors. All these predictions are estimated by using the vectors in current vector field. In our proposal version IV, we enlarge the number of blocks that will be used in weighted average prediction method by adding the predictions obtained from next and previous vector fields.

MVF (Previous)

10	11	12
13	14	15
16	17	18

MVF (Current)

1	2	3
4	Center MB	5
6	7	8

MVF (Next)

19	20	21
22	23	24
25	26	27

Figure 4.7: Using the vectors obtained from previous, next and current fields

In the algorithm of version IV, the hexagonal search is applied between f_{t-1}, f_{t-2} to obtain MVs for previous MVF and applied between f_{t+1}, f_{t+2} to obtain MVs for next

MVF. Then, post processing steps described in Chapter 3 applied for the all these vectors obtained by hexagonal search.

After obtaining the vectors from current, previous and next vector fields, same as the version I, SAD_P, SAD_N and SAD_c values are calculated for each of 27 vectors as shown in the Equation 4.7. Then, α_p and α_n are estimated for each vector by Equation 4.9.

The Equation 3.5 is updated for version IV as shown below;

$$\hat{f}_t = \sum_{k=0}^{27} w_k \hat{f}_k \qquad \text{where} \qquad \omega^k = \frac{1/SAD_k}{\sum_{k=1}^{27} 1/SAD_k} \qquad (4.12)$$

$v_c = (v_x; v_y)$ set of vectors for blocks 1-9

$v_p = (v_x; v_y)$ set of vectors for blocks 10-18

$v_n = (v_x; v_y)$ set of vectors for blocks 19-27

The construction of the new frame is made for that occluded block as follows;

$$f_t(i,j) = \sum_{k=0}^{27} \omega^k \ \alpha_p f_{t-1}(i + \frac{1}{2}v^k{}_x, j + \frac{1}{2}v^k{}_y) + \omega^k \alpha_n f_{t+1}(i - \frac{1}{2}v^k{}_x, j - \frac{1}{2}v^k{}_y)).$$

$v = (v^k{}_x; v^k{}_y)$ is set of 27 vectors

Version IV is based on the idea of unreliability of all 9 vectors in current vector field, so that more reliable vectors for that center block possibly can be found at neighboring vector fields. Therefore, the vectors and their prediction values obtained from neighbor vector fields are added to weighted synthesis. On the other hand, weighted synthesis of this many vectors prediction may finalize with artifacts like blurring and corruption of image when none of the vectors are reliable.

4.3.4 Weighted Synthesis of Previous and Next Frames in Occluded Regions by Using Vectors Obtained From Current, Previous and Next Fields (Version V)

In our proposal version V, the construction of new frame is similar to our proposal version III. The only difference is related α_p and α_n calculation. In our proposal version V, the α_p and α_n are calculated independent from SAD_c value as shownas follows;

$$\alpha_p = \frac{\left(\frac{1}{SAD_N}\right)}{\left(\frac{1}{SAD_N}\right)+\left(\frac{1}{SAD_P}\right)} \qquad \alpha_n = \frac{\left(\frac{1}{SAD_P}\right)}{\left(\frac{1}{SAD_P}\right)+\left(\frac{1}{SAD_N}\right)} \qquad (4.13)$$

Then, the construction of new fame is made by using these new α_p and α_n values as shown in the Equation 4.10. This type of calculation provides α values to be much more dependent to the SAD_P, SAD_N. This may give us better results in some cases that construction of image for these occluded areas becomes much more dependent on just previous or just next frames.

Chaper 5

Experimental Results

For a goal of understanding more clearly about the results of the proposed methods, a detailed comparison should be done. We apply our proposals as an addition to the main algorithm that is based on multi stages motion vector processing method [1], and that algorithm was developed and updated with extra synthesis stage by Burak Çizmeci and Assist.Prof, Hasan Fehmi Ates. We compare the results obtained by using 5 different versions of our proposal, result obtained by using multi-stages vector processing algorithm (9p method) [1], and result with the Occ-3F method [2]. We also give an example for interpolation of frame when best constant α_p and α_n values are specified.

In these simulations, we use video sequence “Running Girl 2” which is with high velocity motion, and use video sequences “Running Girl 1” and “Running Girl 11” which are with detailed background and high velocity motion. These video sequences are proper examples to experiment with occlusion problem, and we can see the visual differences after different methods are applied on the video sequences

In our proposals version I and version IV, we select block size G=4 for SAD calculation in covered uncovered regions, while block size G_c=8 is selected in version II, version III and version V for SAD calculaion in occluded regions. Also, in version II, version III and version V, we choose further vectors of MBs which are one MB further than neighbors of center MB.

We apply our algorithms for the blocks which satisfy the condition below;
$\text{SAD}_i > 150$

The simulations represent the interpolated frames after different methods applied on the occluded regions. In the simulations, interpolated results of frame 18 of "Runnning Girl 2" (Figure 5.1), frame 30 of "Running Girl 1" (Figure 5.2) and frame 111 of "Running Girl 11" (Figure 5.3) are included.

In Figure 5.1, we can see the improvement in the interpolated frame when our proposal applied on the occluded regions. As you see easily, the part of the girl's hair disappear in Figure 5.1 (a), by the way, the methods Figure 5.1 (b), (c), (d), (e) are better in preserving the hair structure. Also, Figure 5.1 (b), (c), (d), (e) are giving more accurate results in showing distinction between the legs. In Figure 5.1 (d), there is a mixture in foot part; this is the result of taking the weighted average of so many vectors which are unreliable. In Figure 5.1 (f), when the method version V is insufficient in preserving hair structure and providing distinction between the legs, on the other hand it achieves to fulfill one of the legs' bottom parts. The comparison between version III and version V shows us calculating the α values like it's explained in version I give better results. Selecting bigger block size and further vectors in version II provides more consistent background between the legs; however it's not as good as result obtained by using version I in preserving hair structure and distinction between the legs. Figure 5.1 (g) represents the interpolated frame when we choose best constant α values for forward and backward compensation. This result shows us the importance of choosing the correct α values. In Figure 5.1 (h), Occ-3F method as a hard decision method, finalizes with blockiness and high frequency artifacts. Even, some part of the one of the legs is completely lost in the interpolated frame. This is because of wrong decision of covered/uncovered regions for the neighbor blocks.

Another example is given in Figure 5.2 that shows the differences among the interpolated of the frame by using different methods. The girl is running through the path that foreground and background of the path are full of detailed objects. Figure 5.2 (b) provides more clear and smooth face, body and leg structure compared to Figure 5.2 (a). This means choosing the α values just dependent to SAD_P, SAD_N (Version V) prevents the background covering on girl structure and gives us the best result. The Figure 5.2 (c), (d) also provides improvement on girl's structure compared to Figure 5.2 (a). As a conclusion of another comparison, Version II gives more smooth results compared to version I. By choosing best constant α_p and α_n values in 5.2 (e), we obtain very clear girl structure. When we observe Figure 5.2 (f), we again see the sharp blockiness artifacts as a result of hard decision of Occ-3F.

The interpolation results of frame of “Running Girl 3” are represented in Figure 5.3 In Figure 5.3 (b), (c), (e) the wooden stick which is uncovering in the backside of the girl is clearer than reference result Figure 5.3 (a). Also, hand of the girl is more visible in the results obtained by using our methods. In this example, the Figure 5.3 (d) shows that the method version V is not that much successful in specifying these details, so that for this interpolation it’s better to prefer to calculate α values like it’s explained in version I. Figure 5.3 (e) represents choosing of vectors from 3 vector fields give us good results in the interpolated frame. Figure 5.3 (f) shows that choosing the best constant α_p and α_n is giving the best result in interpolation of wooden stick; on the other hand it’s also insufficient in preserving the girl’s right hand structure.

In our work, we also test the usefulness of the basic idea of determining different weights apart from 0.5 for the past and future frames, in construction of new frame for the occluded regions. For that purpose, we down sample the video sequence and choose the frames for construction purpose as follows;

Previous Frame 2 = f_t, Next Frame = f_{t+4}

Previous Frame = f_{t+2} Next Frame 2 = f_{t+6}

In this case, the constructed frame will be in the same location with original frame which is f_{t+3}. For this experimental analysis, we calculate α_p and α_n values that minimize the SAD between original frame and constructed frame.

$$(\alpha_p, \alpha_n) = \text{argmin}\,(SAD\ (f_{t+3} - f_{constrcuted}(\alpha_p, \alpha_n)) \qquad (5.1)$$

Our hyphothesis is that, the α values obtained by Equation 5.1 have to be the most appropriate ones for this region. We test truthness of this idea by making comparison between the visual results of $f_{constrcuted}$ and f_t which f_t is obtained from one of our proposals, as the results can be seen in Figure 5.4. This experiment shows us how much improvement is possible by using appropriate α values, and also shows us how successful our α selection approach is.

When we observe the visual results of our proposals and the other methods in occluded regions, we see that by using our proposals we obtain smoother interpolated frames than the ones obtained by using other methods. Especially, we generally obtain better results by using our methods in occluded regions compared to reference frame obtained by using 9p method. Significantly, our proposals are expected to prevent sharp discontinuities that are usually observed during interpolation of occluded regions. As it's understandable that the proposed methods can not improve the deformed structures in the interpolated frame, because these are result of wrong vectors which can not be corrected during the synthesis step of post-processing. These vectors should be corrected in previous stages of post-processing algorithm. By using our proposals, it is aimed to make a soft decision to suppress sharp discontinuities or high frequency artifacts that may occur in occluded regions, and at the end this target is achieved.

Figure 5.1(a) 9p method

Figure 5.1(b) Version I

Figure 5.1(c) Version II

Figure 5.1(d) Version IV

Figure 5.1(e) Version III

Figure 5.1(f) Version V

Figure 5.1(g) α_p=0.9; α_n=0.1

Figure 5.1 (h) Occ-3F method

Figure 5.2(a) 9p method

Figure 5.2(b) Version V

Figure 5.2(c) Version I

Figure 5.2(d) Version II

Figure 5.2(e) α_p =0.1; α_n =0.9

Figure 5.2(f) Occ-3F method

Figure 5.3(a) 9p method

Figure 5.3(b) Version I

Figure 5.3(c) Version III

Figure 5.3(d) Version V

Figure 5.3(e) Version IV

Figure 5.3(f) α_p=0.1; α_n=0.9

(a) Result obtained by using optimal α_p and α_n

(b) Result obtained by using our proposal Version III

Figure 5.4: Correctness of α selection approach

Chapter 6
Conclusion

MCFI is the most effective way among all the other FRUC methods, and provides us better image quality by using the motion information for the interpolation of new frame. MCFI can remove possible artifacts in the interpolated video sequence. In this manner, as a first the most possible motion vectors for the blocks in the image are searched. Then, these vectors are classified according to their reliability levels that are decided by observing the difference between previous and next frames. After this classification, MV processing steps are applied to improve the incorrect motion vectors that cause the blocking artifacts. There are different kinds methods developed for MV processing. In our work, we focus on multi-stage MV processing method that improves MVs obtained after each stage. The results we obtain from the examples by using this method show this method is more effective than the other applied methods in eliminating artifacts and in preserving structure information. The PSNR and SSIM values show that multi-stage method has better performance with the reduced computational complexity.

However, the state-of-art methods developed for MCFI are not efficient to eliminate all the artifacts. Significantly, the problems that are faced when specifying the correct MVs for occluded regions motivate us to focus on solution of this problem. Occluded regions occur on the boundary of moving object. The covered and uncovered regions should be determined properly to be able to specify the correct MV for these areas. In our work, the different kinds of approaches in literature developed for occlusion problem have been discussed in details. These methods usually use uni-directional interpolation, and as it's a hard decision type, so that may cause sharp discontinuties at final frame. Our new proposal has been developed to make a soft decision for these regions and does not use uni directional interpolation. The main method and its' different versions allow frame interpolation with increased correctness and smoothness in occluded areas. The new methods can not remove the artifacts that occur because of wrong vector information obtained as a result of post processing stages in occluded regions, but they can achieve more consistent and smoother synthesis in the interpolated frame. The comparison between our proposals and the other developed methods show that our proposals give better results in making true interpolation and avoiding sharp artifacts. In conclusion, our new proposal is a robust method with smoother results and better artifact reduction property for occluded regions.

Curriculum Vitae

Rıfat YAZGAN was born on 7 November 1980, in Yalova. He studied in Isık University Electronics Engineering department with a scholarship, and he received his BS degree in Electronics Engineering in 2003 from Işık University. His research interests include digital communication, digital video processing, and hardware applications of these subjects. Since 2005 he has been working as a project manager in a private company related with security systems.

R. Yazgan*, Utilization Digital Signal Applications on an Evaluation Board Which ADSP2189M Microprocessor Integrated on Board*, B.S. Thesis, Isık University, 2008.

References

[1] A-M. Huang, T. Nguyen, "A Multi-Stage Motion Vector Processing Method for Motion Compensated Frame İnterpolation", *IEEE Trans. on Image Processing*, May 2008.

[2] A. Pelagotti and G. de Haan, "A New Algorithm for High Quality Video Format Conversion", *Proceeding of International Conference on Image Processing*, Oct. 2001.

[3] D P Elias, N G Kingsbury, "An Efficient Block Segmentation Algorithm for True Motion Estimation", *Sixth International Conference on Image Processing and Its Applications,* Jul. 1997.

[4] Y.-T. Yang, Y.-S. Tung, J.-L. Wu, "Quality Enhancement of Frame Rate Up-Converted Video by Adaptive Frame Skip and Reliable Motion Extraction" , *IEEE Trans. on Circuits and Systems for Video Technology,* Dec. 2007.

[5] S.-H. Lee, O. Kwon, and R.-H. Park, "Weighted-Adaptive Motion-Compensated Frame Rate-Up Conversion", *IEEE Trans. on Consumer Electronics*, Aug. 2003.

[6] K Sugiyama, T. Aoki, S. Hangai, "Motion Compensated Frame Rate Conversion Using Normalized Motion Estimation", *IEEE Workshop on Signal Processing Systems Design and Implementation,* Nov. 2005.

[7] G. de Haan, P.W.A.C. Biezen, H. Huijgen, and O.A. Ojo, *Graceful Degradation in Motion Compensated Field-Rate Conversion,* Signal Processing of HDTV, L. Stenger, L. Chiariglione and M. Akgun (Eds.), 1994.

[8] H. Blume and H. Schroder, "Image Format Conversion - Algorithms, Architectures, Applications", *in Proc. of the ProRISC/IEEE Workshop on Circuits, Systems and Signal Processing,* Nov. 1996

[9] G. A. Thomas and M. Burl, *Video İmage Processing*, PCT Application No: WO 92105662, Sept. 1991.

[10] A. M. Tekalp, *Digital Video Processing*. Prentice hall, NJ, 1995.

[11] Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Video Processing and Communications,* Prentice Hall, 2002.

[12] L. Alparone, M. Barni, F. Bartolini, V. Cappellini, "Adaptively Weighted Vector-Median Filters for Motion-Fields Smoothing", *IEEE International Conference on Acoustics, Speech, and Signal Processing,* May 1996.

[13] W. R. Sung, E. K. Kang, J. S. Choi, "Adaptive Motion Estimation Technique for Motion Compensated İnterframe İnterpolation", *IEEE Transactions on Consumer Electronics*, Aug. 1999.

[14] Csillag, P., Boroczky, L., "Enhancement of Video Data Using Motion-Compensated Postprocessing Techniques", *IEEE International Conference on Acoustics, Speech, and Signal Processing,* April 1997.

[15] C. Zhu, X. Lin, L.-P. Chau, K.-P. Lim, H.-A. Ang, C.-Y. Ong, "A Novel Hexagon-Based Search Algorithm for Fast Block Motion Estimation", *IEEE International Conference on Acoustics, Speech, and Signal Processing,* May 2001.