Watermarking Schemes for High Definition Videos

Thesis submitted to the
Indian Institute of Technology Guwahati
for the award of the degree

of

Doctor of Philosophy
in
Computer Science and Engineering

Submitted by
Sibaji Gaj

Under the guidance of
Dr. Arijit Sur and **Prof. Prabin Kumar Bora**

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
June, 2017

Abstract

With the increased popularity of *High Definition* (HD) and beyond-HD videos ($4K \times 2K, 8K \times 4K$) and the emergence of the *High Efficiency Video Coding* (H.265/HEVC) compression standard for HD and beyond-HD video formats, security issues like copyright protection, content authentication etc. of HD videos have become an important research field. In this research work, video watermarking is considered as the tool for ensuring secure HD video transmission and a few robust watermarking schemes have been designed for addressing different problems regarding HD video watermarking.

In the first contributory chapter of this dissertation, a compressed domain robust watermarking scheme is proposed using one of the side information (intra prediction mode) of H.265/HEVC compressed video. Since the probability of the mode change during re-compression has been increased in H.265/HEVC in comparison with the previous standards, the existing mode-based watermarking schemes (for previous standards like H.264/AVC) are not robust enough if they are applied on HEVC videos. As a countermeasure, first the 4×4 luma *Prediction Block* (PB)s of intra frame are chosen for embedding based on the sustaining probability after re-compression. Further, the intra prediction modes of H.265/HEVC are grouped in a fashion such that the embedded mode-change due to re-compression can be closed within a group. Experimental results on various test sequences show that the proposed scheme is robust to re-compression while maintains decent visual quality of the watermarked video.

In the second chapter, the robustness of the above scheme is enhanced by embedding the watermark in the transform coefficients of 4×4

transform blocks of the HEVC video sequence. In HEVC, the coefficient distribution of the residual blocks becomes more sparse due to more accurate prediction. It is experimentally observed that perturbing such sparse coefficients leads to substantial visual artifacts. To embed the watermark with less visual degradation, temporally homogeneous blocks in the consecutive Intra frame are first found out using only the compress domain parameters. As these blocks within short temporal neighborhood have a similar texture which leads to similar Number of Nonzero Transform Coefficients (NNZ). Thus, a robust watermarking scheme is proposed by the minimal perturbing of the transform coefficients in a pair of temporally homogeneous blocks.

Both of the proposed schemes are well suited for the re-compression attack but not resilient to the drift errors. So, a compressed domain watermarking scheme is proposed in the third contributory chapter of this dissertation for H.265/HEVC video which can handle drift error propagation both for the intra and the inter prediction processes and improves visual quality of the watermarked video. The intra drift error prevention is achieved by excluding those residual coefficients which are used in the intra prediction process. The inter frame drift error has been prevented by predicting current block from a spatial homogeneous block which has high spatial correlation with the watermarked block. The proposed scheme shows adequate robustness against the re-compression attack as well as common image processing attacks while maintaining the decent visual quality.

Recently, camcorder based video copy attack has become a serious threat to film industries where the movies are distributed using HD and beyond-HD format. In the final contributory chapter, a watermarking scheme is presented which is robust against video the camcording attack. During camcording attack, the embedded watermark suffers from several geometric distortions and temporal desynchronizations. In the proposed scheme, the required temporal synchronization is done by segmenting the cover video into shot seg-

ments using the geometric invariant *Scale-Invariant Feature Transform* (SIFT) features. Further, the 3D-*Discrete wavelet transformation* (DWT) and SIFT are used to find watermark embedding regions which are robust against geometric distortions and the frame blending attack. Finally, the dissertation concludes by briefly summarizing the work presented in dissertation and explaining the future research directions.

Declaration

I certify that:

a. The work contained in this thesis is original and has been done by me under the guidance of my supervisor.

b. The work has not been submitted to any other Institute for any degree or diploma.

c. I have followed the guidelines provided by the Institute in preparing the thesis.

d. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

e. Whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Sibaji Gaj

Copyright

Certificate

This is to certify that this thesis entitled **"Watermarking Schemes for High Definition Videos"** being submitted by **Sibaji Gaj**, to Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, for partial fulfillment of the award of the degree of Doctor of Philosophy, is a bonafide work carried out by him under our supervision and guidance. The thesis, in our opinion, is worthy of consideration for award of the degree of Doctor of Philosophy in accordance with the regulation of the institute. To the best of our knowledge, it has not been submitted elsewhere for the award of the degree.

.....

Dr. Arijit Sur
Associate Professor
Department of Computer Science and Engineering
IIT Guwahati

.....

Prof. Prabin Kumar Bora
Professor
Department of Electronics and Electrical Engineering
IIT Guwahati

Dedicated to

Baba, Ma, Mimi, Thirtha

Whose blessing, love and inspiration paved my path of success

Acknowledgments

A great many people have contributed to the production of this dissertation. I owe my gratitude to all those people who have made this possible.

I wish to express my deepest gratitude to my supervisors, Dr. Arijit Sur and Prof. Prabin Kumar Bora for their valuable guidance, inspiration, and advice. I feel very privileged to have had the opportunity to learn from, and work with them. Their constant guidance and support not only paved the way for my development as a research scientist also changed my personality, ability, and nature in many ways. I have been fortunate to have such advisors who gave me the freedom to explore on my own and at the same time the guidance to recover when my steps faltered. Besides my advisor, I would like to thank the rest of my thesis committee: Prof. Shivashankar B. Nair, Dr. Pinaki Mitra, and Dr. Prithwijit Guha, for their insightful comments and encouragement. Their comments and suggestions helped me to widen my research from various perspectives.

I also like to express my heartful gratitude to the director, the deans and other managements of IIT Guwahati whose collective efforts has made this institute a place for world-class studies and eduaction. I am thankful to all faculty and staff of Dept. of Computer Science and Engineering for extending their co-operation in terms of technical and official support for the successful completion of my research work.

I am thankful to my friends Satish, Sangeetda and Rana for supporting and motivating to overcome any problems either in work and otherwise. The countless discussions, sharing ideas has improved our research. I am also grateful to all my seniors, friends and juniors

especially Senko da, Mamata di, Shishendu da, Shilpa di, Nil da, Mayank da, Anirban, Sathish, Brijesh, Rahul, Rajesh and many others for their unconditional help and support. You made my life at IIT Guwahati a memorable.

Most importantly, none of this would have been possible without the love and patience of my family. I want to thank my parents, Mimi and Thirtha for being a constant source of love, concern, support, and strength all these years.

Contents

List of Figures

List of Algorithms

List of Tables

List of Acronyms

AGC *Automatic Gain Control*

AMP *Asymmetric Motion Partitioning*

AWB *Automatic White Balance*

AVC *Advanced Video Coding*

BIR *Bit Increase Rate*

CTU *Coding Tree Unit*

CU *Coding Unit*

CTB *Coding Tree Block*

CB *Coding Block*

DCT *Discrete Cosine Transformation*

DoG *Difference of Gaussian*

DRM *Digital Right Management*

DST *Discrete Sine Transform*

DWT *Discrete wavelet transformation*

GOP *Group of picture*

HD *High Definition*

H.265/HEVC *High Efficiency Video Coding*

HVS *Human Visual System*

IDST *Inverse Discrete Sine Transform*

ISO/IEC *International Organization for Standardization / International Electrotechnical Commission*

ITU *International Telecommunication Union*

JCT-VC *Joint Collaborative Team on Video Coding*

LSB *Least Significant Bit*

LPB 4×4 *Luma Prediction Block*

LTB 4×4 *Luma Transform Block*

Mode *Intra Prediction Mode*

MC *Motion Compensation*

ME *Motion Estimation*

MV *Motion Vector*

NNZ Number of Nonzero Transform Coefficients

PB *Prediction Block*

Prediction Mode Intra Prediction Mode

PSNR *Peak Signal to Noise Ratio*

QP *Quantization Parameter*

SIFT *Scale-Invariant Feature Transform*

SPS *Sequence Parameter Set*

SSIM *Structural Similarity*

TB *Transform Block*

VIFp *Visual Information Fidelity in pixel domain*

List of Symbols

α	Watermark Strength
$\mathbb{B}$	The subset of $\mathbb{LB}$ selected for mode alteration
$\mathbb{BR}$	The subset of $\mathbb{B}$ sustained after re-compression
$\mathcal{C}$	The minimum Lagrangian cost for PB partitioning
$\mathcal{D}$	The rate distortion due to compression
f_I	Number of Intra Frames in V
f_P	Number of Inter Frames in V
f_{total}	Total Number of Frames in V
f_{GOP}	Number of Frames in a GOP
$\mathbb{LB}$	The set of all LPBs before re-compression
$\mathbb{LR}$	The set of all LPBs after re-compression
$\mathbb{M}$	The set of intra prediction modes
$\mathcal{N}$	Number of Non-zero Transformed Coefficients
$\mathcal{N}_{th}$	NNZ threshold
$\mathcal{N}_{low}$	Lowest NNZ threshold

$\mathcal{N}_{high}$	Highest NNZ threshold
n_{LTB}	Number of LTBs in a Frame
n_{TB}	Number of TBs in a Frame
n_{LPB}	Number of LPBs in a Frame
n_{PB}	Number of PBs in a Frame
n_B	Number of 4×4 Blocks in a Frame
$\mathcal{R}$	The bit requirement during compression
V	Input video/ Original Video
VW	Watermarked Video
WS	Watermark Sequence
WE	Extracted Watermark Sequence
WL	Watermarked Location Map

Chapter 1

Introduction

The digital media has been swiftly adopted over the analog media in the last few decades because of the ease of storing and seamless communication through the internet. The very nature of digital media makes copying and distribution easy and this in turns makes the hacking and piracy as a potential threat to the efficient media communication. In recent past, digital watermarking has become a popular and a promising solution for the ownership as well as content media authentication. In digital watermarking, a digital signal is embedded in the media. The embedded signal can be extracted or detected at the receiver end for different forensics applications such as content authentication against media forgery, ownership authentication against piracy, traitor tracing, broadcast monitoring etc. Moreover, the embedded watermark does not prevent the user to view, listen, examine or manipulate the media.

Digital video is one of the most widely used and distributed digital media because of digital television, streaming services over internet, Blu-ray discs etc. For ensuring content and copyright authentication, watermarking has been regarded as an efficient *Digital Right Management* (DRM) tool for digital videos in the last decade [1]. In video watermarking, the digital signature (watermark) is embedded in the selected regions of a video frame without affecting the visual quality of the watermarked signal.

Substantial improvement in the communication technology in recent years and the availability of the cheaper and wider varieties of video playing devices have made *High Definition* (HD) and beyond-HD video ($4K \times 2K, 8K \times 4K$) transmission more popular. In this regards, the latest addition is *High Efficiency Video Coding* (H.265/HEVC) compression standard [2] which supports HD and beyond-HD video formats and achieves a remarking improvement over the current versions of video streams such as H.264/AVC. With the emergence of such HD video, security issues like copyright protection, content authentication etc. of HD videos have become an important research field. In this research work, video watermarking is considered as the tool for ensuring secure HD video transmission. In next section, an overview of digital video watermarking is briefly described.

1.1 Digital Video Watermarking

Digital video watermarking is a technique which inserts a digital signature(number sequence, text, logo etc.) into the video stream which can be extracted or computed to authenticate the ownership of the media or the media itself. A fundamental video watermarking system is described in Figure 1.1 [1]. The system combines of two algorithms, one for embedding the watermark and the other for extraction. A key is used to embed and extract the watermark.

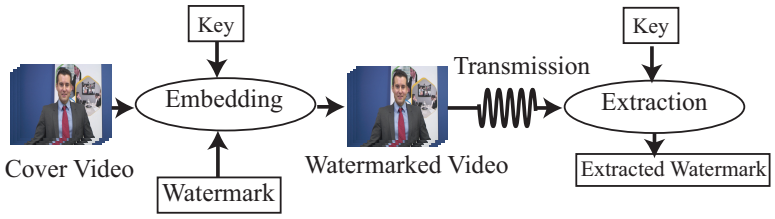


Figure 1.1: *Watermarking in digital video*

1.1.1 Evaluation Parameters

There are a number of parameters to check the efficiency of a watermarking scheme or some times used to categorize them. Some time they are conflicting to each other. Few important parameters are described below [1, 3, 4].

Robustness: The robustness of a watermarking scheme is defined as how efficiently the watermark withstands against intentional and unintentional attacks. Depending on that, a watermark is called robust, fragile or semi-fragile.

Imperceptibility: Imperceptibility implies that the watermark should not be perceptually noticeable in the watermarked video.

Payload: Payload measures the number of bits or the size of the watermark embedded.

Blindness: A watermarking scheme is called blind if the original video is not needed at extraction.

Bit Increase Rate: Often bit rate gets increased due to embedding. A good watermark embedding scheme does not increase the *Bit Increase Rate* (BIR) significantly.

1.1.2 Classification of Video Watermarking Techniques

Video watermarking techniques can be classified into different categories [3–6]. Some of the categories are as follows,

Robust, Fragile, Semi-Fragile : In robust watermarking, the embedded watermark can resist different attacks such as lossy compression, image processing, common editing etc. Robust watermarking is mainly used for embedding the copyright information to carry ownership authentication and access control. Fragile watermarking is mainly used for integrity protection

of digital content such as legal, military, medical applications where the watermark is not detectable after slightest modification of watermarked video. Semi-fragile watermarking is capable of tolerating some degree of the intentional modification to a watermarked video. Semi-fragile watermarking is mostly used in content dependent watermarking where the content needs to be strictly protected, but the exact representation during communication and storage is not guaranteed.

Blind, Non-Blind : Based on detection requirement, the watermarking technique can be classified into blind or non-blind. In non-blind watermarking, the original video is required during extraction or detection of the watermark. On the other hand, the blind algorithms do not require the original video at the time of extraction.

Compressed Domain, Uncompressed Domain : In uncompressed domain watermarking, the raw video is used to embed the watermark. In compressed domain watermarking, the compressed domain features such as the motion vectors, the quantized transformed coefficients of block residuals, prediction information are modified to embed the watermark. In compressed domain watermarking, full decoding and re-encoding of the compressed bit stream is not required.

1.1.3 Applications

Though, the digital watermarking is used mainly for copyright protection and authentication, there are several other applications for digital watermarking [1, 3–6]. Some of the popular applications of digital watermarking are listed below,

Ownership Authentication / Copyright protection [7]: The copyright issues of digital media can be resolved by using the watermark information as the copyright data.

Video Authentication [8]**:** The digital video can be altered easily and the alteration is difficult to detect. A fragile watermark can be embedded in the digital video so that the embedded watermark gets destroyed whenever any sort of modification is made to the content. Thus, the content authentication can be done.

Traitor Tracing [9]**:** Unique watermarking in digital media can trace the source of leak, which helps to stop unauthorized distributions.

Broadcast Monitoring [10]**:** Unique watermarking of each video clip is used to keep track of when and where an advertisement is played in an automated broadcast monitoring system.

Medical Application [11]**:** To avoid mix up of the X-ray reports and MRI scans, the watermarking technique can be used to identify the patient accurately.

1.2 HD Video Representation

In this PhD dissertation, the watermarking schemes have been proposed for both compressed and uncompressed HD videos. In case of uncompressed domain watermarking, the HD videos are in YCbCr color space with 4:2:0 sampling. Each of Y, Cb, Cr is represented using eight bits. In case of compressed domain, H.265/HEVC compressed video [2] [12] [13] which is the newly emerged compression standard for HD video, is used for embedding. In next section, a brief introduction to H.265/HEVC and the modifications compared to the previous standard H.264/AVC is presented.

1.2.1 HEVC Coding Design and Feature Highlights

Video coding standards have been evolved primarily by *International Telecommunication Union* (ITU) and *International Organization for Standardization / International Electrotechnical Commission* (ISO/IEC). Previous standards such

as H.261, H.263, MPEG-1, MPEG-4 go through changes to maximize the compression capability and minimize the data loss during compression while they consider computational complexity and computation resource utilization. In 2013 [2], H.265/HEVC standard is released by *Joint Collaborative Team on Video Coding* (JCT-VC) [14] for HD video compression. For the same visual quality, the compression by H.265/HEVC is around twofold higher than the past standard H.264/AVC [2] [15]. Because of the expanding prevalence of the HD and beyond-HD videos and the increase in processing power of the end user devices like the cellular device, PDA etc., H.265/HEVC is relied upon to supplant the past standard H.264/AVC as it has more compression efficiency.

In video compression, the video data are partially de-correlated by spatial and temporal prediction to reduce the spatial and temporal redundancies. This partially de-correlated data are transformed and quantized to get the compressed bit stream. In case of H.265/HEVC, the video coding layer follows the same hybrid coding methods (intra, inter prediction, 2D transform coding, entropy coding) as used in earlier standards since H.261 [2]. The block diagram of HEVC architecture is shown in Figure 1.2. The overview of the encoder algorithm is as follows,

1. Each frame is split into variable size block shaped regions. The exact block partitioning information is conveyed to the decoder for de-compression.

2. The first frame a *Group of picture* (GOP)s [17, 18] is encoded using spatial prediction where the prediction of data is done spatially from region-to-region within the same frame, but has no dependency on the other frames. This procedure is called intra prediction and the frames encoded using intra prediction are called I frames.

3. The remaining frames are temporally predicted where a block in the current frame is compared with all of the possible regions in the reference frame to find the best match regions. This process of finding the best match is

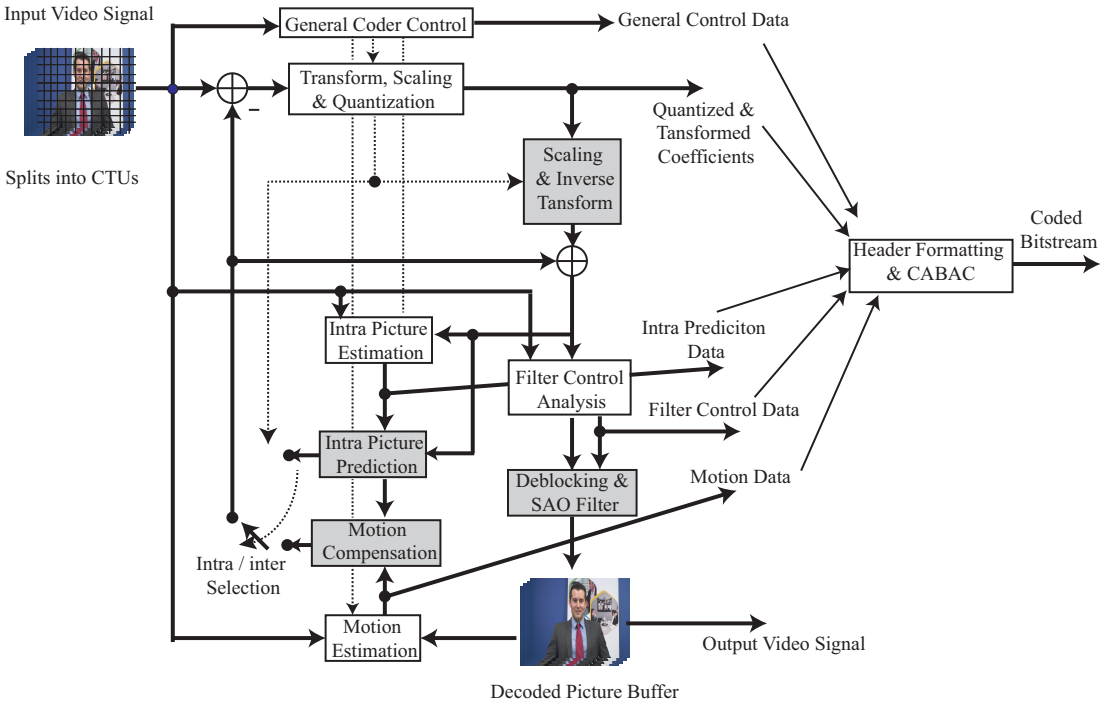


Figure 1.2: *HEVC encoder [2, 16]*

known as *Motion Estimation* (ME) and the search area is called search space. After finding the best match, the difference between the current block and the best match block is encoded with the refer frame number and the deviation in position of the block from current frame to reference (it is called *Motion Vector* (MV)). If the prediction procedure uses only one frame to encode then the current frame is called a unidirectionally predicted frame or P frame. Some frames can be predicted using two reference frames and are called Bidirectionally predicted frames or B frames. The encoder and decoder generate identical inter frame prediction signals by applying *Motion Compensation* (MC) using the MV and mode decision data, which are transmitted as side information.

4. The residual data of inter and intra predictions are the difference between original and predicted block data. This residual is transformed using the

linear spatial transform. The transform coefficient are scaled, quantized and transmitted to decoder side.

5. The encoder duplicates the decoder processing loop such that both will generate identical predictions for subsequent data. First, the quantized transform coefficients are constructed by inverse scaling and then inverse transformed to duplicate the decoded approximation of the residual signal. The residual is then added to the prediction, and the result of that addition may then be fed into one or two loop filters to smooth out artifacts induced by block-wise processing and quantization. The final frame representation (that is a duplicate of the output of the decoder) is stored in a decoded picture buffer to be used for the prediction of subsequent frames.

H.265/HEVC improves almost every step of the hybrid coding architecture, while introducing a new hierarchical syntax representation and *Discrete Sine Transform* (DST) for the residual transformation [2, 16]. These features are highlighted as follows,

1.2.1.1 Structural Improvements

Quad Tree Structure: In case of H.264/AVC the whole frame is divided into non-overlapping square regions that are called macro blocks. It is a fixed size 16×16 for all types of video. In HEVC, the resolution is much higher than previous standard. So HEVC allows larger block sizes i.e.$64 \times 64, 32 \times 32, 16 \times 16$. Depending on the limitations of encoder and decoder such as delay constraint and memory requirements and architectural characteristics, the block size can vary. The comparison of block sizes is shown in Figure 1.3 [15]. This block is called a *Coding Tree Unit* (CTU) or *Coding Unit* (CU).

In case of 4:2:0 color sampling, one CTU unit has one luma *Coding Tree Block* (CTB) and two Chroma CTBs. In case of H.264/AVC, a macro block is divided into smaller size blocks ($16 \times 8, 8 \times 16, 8 \times 8, 8 \times 4, 4 \times 8, 4 \times 4$) and used for prediction, and transformation. But in case of HEVC, separate quad

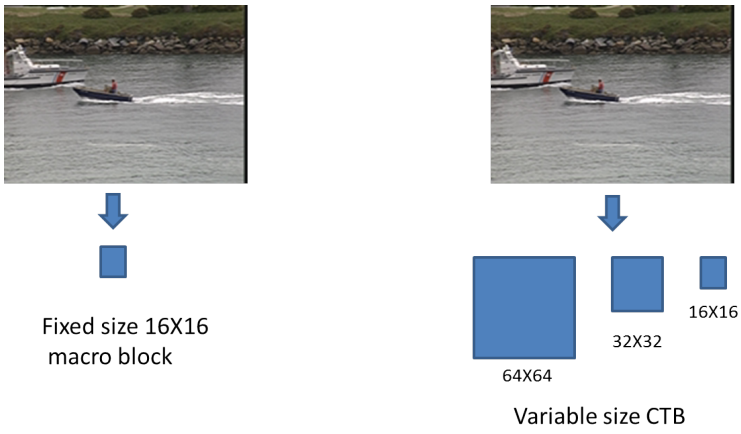


Figure 1.3: *AVC and HEVC block division*

tree structure is used for division of frame, prediction and transform [2].

Coding Block (CB): A CTB can be divided into one *Coding Block* (CB) or multiple CBs depending upon the signal characteristics (more busy area CB can divided into further small sizes for better prediction). This iterating process goes through until the CB size reaches at the minimum allowed size specified into the *Sequence Parameter Set* (SPS). The minimum CB size allowed is 8×8 in HEVC. In case of 4:2:0 color sampling, the chroma CTB size corresponding $L \times L$ luma CTB, is $L/2 \times L/2$ as shown in Figure 1.4 [16].

Prediction Block(PB): For each CU, the prediction mode is signaled as intra

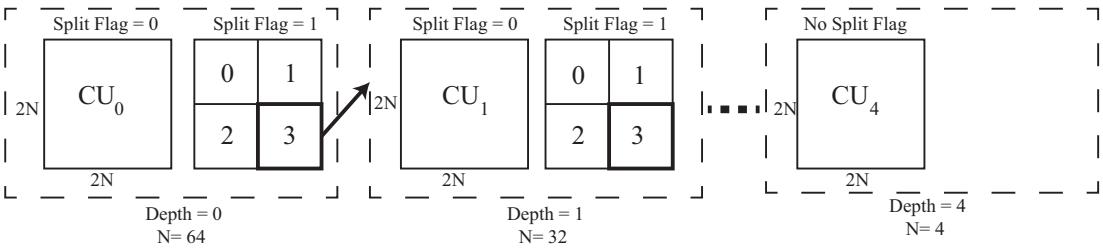


Figure 1.4: *HEVC CTB division [16]*

or inter. When the mode is signaled as intra, the PB size is same as CB, except the CB is the smallest size allowed in SPS. In the latter case, the PB is divided into 4 PB quadrants and each of them has its own prediction mode. The reason

for allowing the split, is to enable distinct intra picture prediction mode selections for blocks as small as 4×4 in size [2]. When CU is signaled for inter prediction, the CB can be split into two or four PBs. Splitting into four PBs is allowed only in case of the minimum allowed CB size in the SPS.

Transform Block (TB): For residual coding, a CB can be recursively parti-

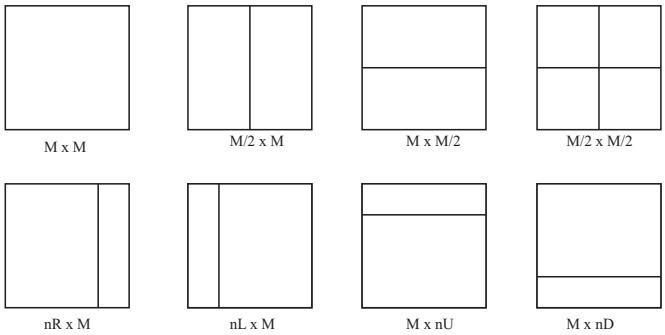


Figure 1.5: *HEVC PB division*

tioned into *Transform Block* (TB)s. The partitioning is signaled by a residual quad tree [2]. The maximum depth of the residual quad tree is signaled by the SPS. The encoder indicates the maximum and the minimum TB that will be used in the codec. If CB size is greater than the maximum TB size, the splitting is implicit. In inter prediction TB can span over multiple PBs within the same CB to maximize the benefits of the prediction coding efficiency.

Slice and Tiles: Slices are self-contained independently decodable parts of a

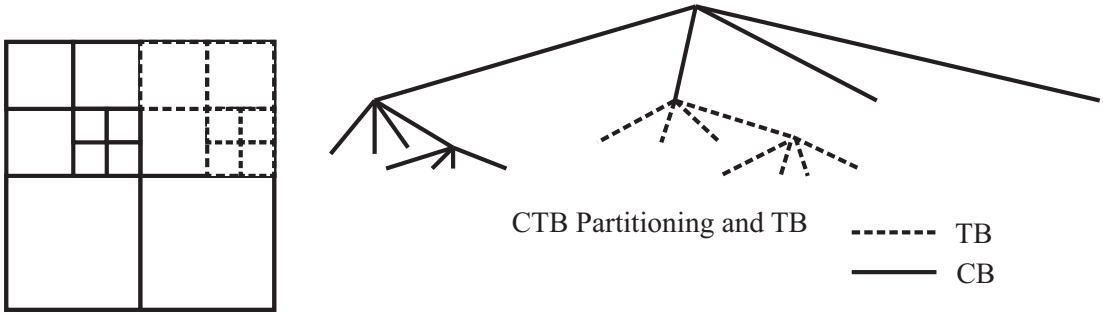


Figure 1.6: *HEVC TB division [2]*

frame, i.e. these pixels can be independently decodable . There are three types

of slices:

I slice: All CUs in this slice are coded using intra prediction

P slice: All CUs in this slice are unidirectionally predicted using one reference picture list only.

B slice: All CUs in this slice are bidirectionally predicted using two reference picture lists.

Slices can have varying number of CTUs. For parallel processing, HEVC introduces a tile. This is an independently decodable square rectangular region of frame. Tiles and slices are shown in Figure 1.7 [2].

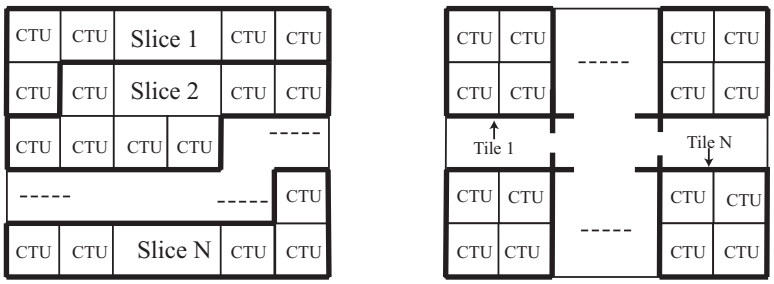


Figure 1.7: *Slice and Tiles [2]*

1.2.1.2 Improvements in Intra Prediction

For intra prediction of the current block by utilizing the previously decoded blocks, H.264/AVC supports nine intra prediction modes. This number of modes is insufficient to predict accurately the directional structures present in a typical video for larger block sizes of H.265/HEVC. Therefore, It supports a total of 35 prediction modes as depicted in Figure 1.8. *In this thesis, the prediction mode and the mode are used interchangeably to represent the intra prediction mode.* Mode 0 refers to the planar prediction, Mode 1 represents the DC prediction and 2-34 are different directional modes with 10 as the horizontal and 26 as the vertical prediction mode. The planar mode is used in H.264/AVC to encode homogeneous regions. However, this mode does not guarantee the continuity at block boundaries which results in block artefacts. In H.265/HEVC, this problem is solved by

using adaptive smoothing of the predicted boundary samples for Mode 1, Mode 10 and Mode 26 [19, 20]. Based on observation, 33 different angular modes are defined to optimize the prediction accuracy. Each block has one reference row and one reference column for prediction as shown in Figure 1.9. In the prediction process, each predicted sample $p_{i,j}$ is projected to its reference row or column $\Re$ depending on the prediction direction. Then the value is linearly interpolated using two nearest reference pixels up to 1/32 pixel accuracy. The interpolation equation is given as follows [19],

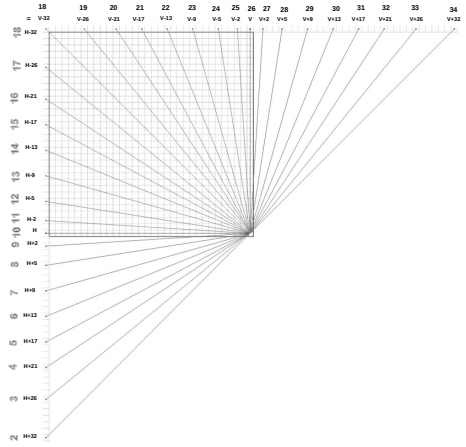


Figure 1.8: *HEVC intra prediction modes with their displacement parameters. H and V are horizontal and vertical directionalities*

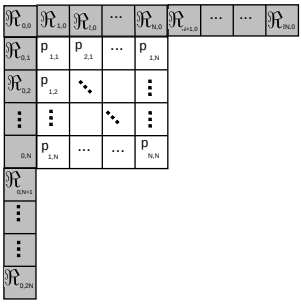


Figure 1.9: *Reference samples $\Re_{x,y}$ used in prediction to obtain predicted samples $p_{i,j}$*

$$p_{i,j} = ((32 - w_j).\Re_{x,0} + w_j.\Re_{x+1,0} + 16) >> 5 \tag{1.1}$$

where w_j and $(32 - w_j)$ are the weighting factor for the two reference samples $\Re_{x,0}$ and $\Re_{x+1,0}$. In Eq.(1.1), $>>$ represents binary right shift operation. The weighting parameter w_j and x are calculated based on the projection displacement d associated with the selected prediction direction and given by,

$$c_j = (j.d) >> 5,\ w_j = (j.d)\&31\ and\ x = i + c_j \tag{1.2}$$

where the value of d ranges from -32 to 32, c_j is the intermediate parameter depend on the coordinate j, displacement d and "&" represents the bitwise AND

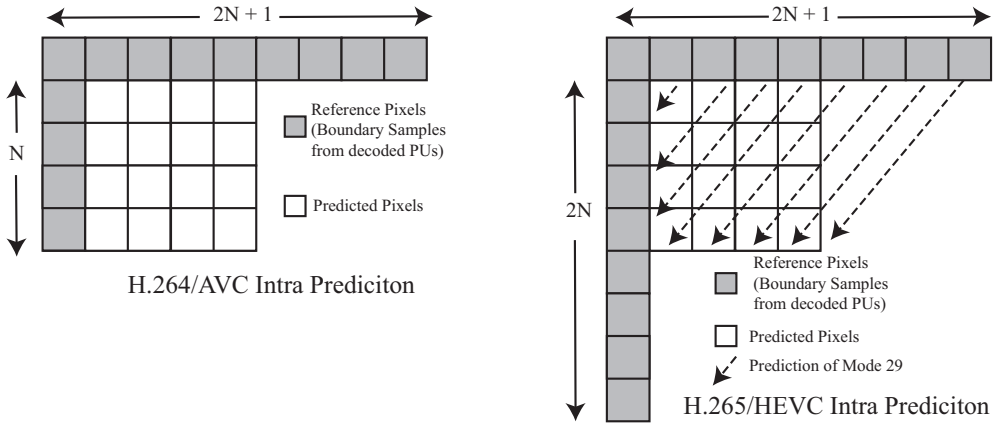


Figure 1.10: *Intra prediction references in AVC and HEVC [2] [21]*

operation. In H.264/AVC, the reference samples obtained from above, right above and left block are used for intra prediction as shown in Figure 1.10. So the total $3N + 1$ reference samples are used for prediction in H.264/AVC, compared to $4N + 1$ spatial neighbor reference samples of H.265/HEVC. In Figure 1.10, the prediction process of mode 29 is shown for HEVC.

1.2.1.3 Inter Prediction Mode Improvements

For inter prediction in H.264/AVC, a macro block can be divided into two symmetric sub macro blocks or four equal size sub macro blocks depending upon heterogeneous motion of block or busyness of that area. The block can be divided up to 8×4 or 4×8 size. In Figure 1.11, H.264/AVC and H.265/HEVC block partitioning has been shown. Compared to H.264/AVC, inter partitioning in HEVC allows *Asymmetric Motion Partitioning* (AMP). As the transform quad tree is separate from PB, and the TB can operate on several PBs, AMP is incorporated in H.265/HEVC. To minimize worst-case memory bandwidth, PBs of luma size4×4 are not allowed for inter frame, and PBs of luma sizes 4×8 and 8×4 are restricted to unipredictive coding [2]. Quarter-sample precision is used for the MVs, and 7-tap or 8-tap filters are used for interpolation of fractional-sample positions (compared to six-tap filtering of half-sample positions followed by linear interpolation for quarter-sample positions in H.264/AVC).

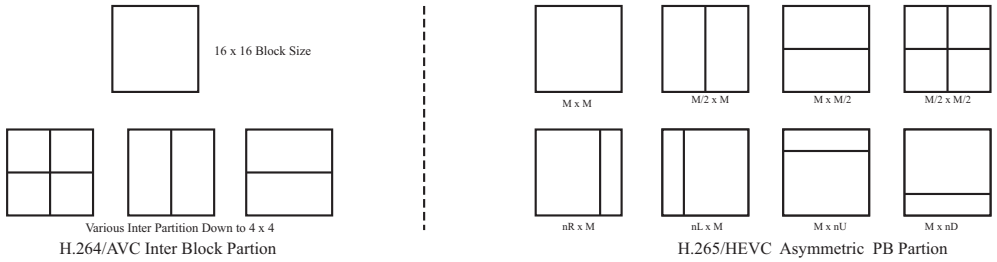


Figure 1.11: *Inter predicition block partitioning in AVC and HEVC*

1.2.1.4 Improvements in Transform

Another major improvement in H.265/HEVC is that it includes *Discrete Sine Transform* (DST) for 4×4 TBs in intra frame where as the *Discrete Cosine Transformation* (DCT) is used for all blocks in H.264/AVC. The DST-VII kernel used in 4×4 intra TB is as follows [22]:

$$H = \begin{bmatrix} 29 & 55 & 74 & 84 \\ 74 & 74 & 0 & -74 \\ 84 & -29 & -74 & 55 \\ 55 & -84 & 74 & -29 \end{bmatrix} \sim 128 \times \left[\frac{2}{\sqrt{2N+1}} \sin\left(\frac{\pi(2i-1)(j)}{2N+1} \right) \right] \tag{1.3}$$

where i,j denotes the position in matrix H and i,j=1,...,N and N=4

As *Inverse Discrete Sine Transform* (IDST) of H.265/HEVC is used in drift error calculation in Chapter 5, a brief implementation details of IDST in H.265/HEVC is described here. In H.265/HEVC, the 4×4 IDST has been implemented by applying the 1-D transforms in the horizontal and vertical direction. Further, the efficiency has been improved by minimizing the number of required multiplications. Given an input vector of length 4 as $Y_{4\times1} = [y_0,\ y_1,\ y_2,\ y_3]^T$, the output vector $X_{4\times1} = [x_0,\ x_1,\ x_2,\ x_3]^T$ of 1D-DST is obtained in H.265/HEVC in two steps. First, a set of intermediate values $b_0,\ b_1,\ b_2$ and b_3 are calculated as follows:

$$b_0 = (y_0 + y_2),\ b_1 = (y_2 + y_3),\ b_2 = (y_0 - y_3),\ b_3 = 74 \times y_1. \tag{1.4}$$

Then, the output values are calculated as follows:

$$\begin{aligned} x_0 &= S((29 \times b_0 + 55 \times b_1 + b_3 + rd) >> sh) \\ x_1 &= S((55 \times b_1 - 29 \times b_2 + b_3 + rd) >> sh) \\ x_2 &= S((74 \times (y_0 - y_2 + y_3) + rd) >> sh) \\ x_3 &= S((55 \times b_0 - 29 \times b_2 + b_3 + rd) >> sh) \end{aligned} \tag{1.5}$$

where $S()$ is a saturating function for the output values to keep the values within a boundary, $>>$ represents the right shift operation, rd and sh are used for the scaling of the output values. When the 1-D IDST is applied on a column, rd and sh values are taken as 64 and 7 respectively. In a second step, when the 1-D IDST applied on a row, the rd and sh values are taken as 2048 and 12 respectively.

1.3 Literature Survey

It is discussed in the previous section that the HD video has become popular and security issues like copyright protection, content authentication etc. of HD videos have emerged as an important field of research in recent times. But, the literature reveals that relatively less attention has been paid to the HD video watermarking in comparison with the general video watermarking. Though there exists many image watermarking schemes [23–30] which can be extended for frame by frame video watermarking, many limitations exist for such extensions. Since relatively fewer works on HD watermarking have been reported in the literature, a brief description of existing digital video watermarking schemes is presented here. Existing schemes are divided into two major groups as compressed and uncompressed domain watermarking and an analysis of their merits and demerits are given below.

1.3.1 Compressed Domain Watermarking Schemes

Video files are generally stored and distributed in lossy compression format to spare the memory space and cut down the bandwidth requirement. So, many watermarking techniques have been developed to embed the watermark directly in the compressed video by exploiting some parameters of the compression standards such as H.265/HEVC, H.264/AVC, MPEG 2 etc. In compressed domain watermarking, the compressed bit-stream is first entropy decoded and parsed for finding the suitable syntax elements like coefficients, prediction modes, motion vectors etc.. Then, the watermark is embedded in the syntax domain by mod-

ifying transform coefficients [31] [32], intra prediction modes [33, 34] or motion vectors [35]. The modified syntax is then entropy re-encoded. The state-of-the-art literature on watermarking in H.265/HEVC encoded video is still in its rudimentary phase and a very limited number of schemes for H.265/HEVC watermarking can be found in the literature. So, a brief survey of watermarking for H.264/AVC is first described and then the applicability of these existing schemes to H.265/HEVC is analyzed.

1.3.1.1 Watermarking Schemes Using Transformed Coefficients

Noorkami and Mersereau [31] have proposed a compressed domain low complexity algorithm by altering the *Least Significant Bit* (LSB) of quantized AC coefficient of 4×4 macro-blocks. However, this scheme is not robust against common watermarking attacks. In [36], the authors have proposed an watermarking scheme by modifying the last three bit planes of 8-bit coefficients in the luminance and chrominance channels. Xu et al. [37] have proposed a semi-fragile watermarking scheme for H.264/AVC to detect both spatial and temporal tampering by changing parity of the sign of the transformed coefficients. In [38], authors have embedded the watermark in H.265/HEVC bit stream by altering least significant bit during encoding of the video. In [39], Noorkami et al. has improved their scheme [31] by imposing robustness and proposed another non-blind watermarking scheme where visual quality of watermarked video is improved by using Watson's human visual model [40]. In these schemes, a bipolar watermark is embedded by altering DCT coefficients of 4×4 macro-blocks. After any simple attack residual value changes but due to linear property of DCT transform, the watermark still retains in watermarked video and it can be detected above a level of distortion. Zhang et al. [41] proposed a grayscale watermark pre-processing technique to increase the robustness and the capacity of the watermarking algorithm. In this scheme, the watermark is embedded by altering the sign of a quantized AC coefficient based on the watermark bit. In [32], authors have implemented another compressed domain robust watermarking scheme for H.264/AVC

video streams by considering the minimization of spatial noise and temporal flickering. In this work, the robustness is improved by analyzing the mode change probability of 4×4 blocks depending on the number of nonzero DCT coefficients (NNZ) and estimating the motion activity within a GOP. The watermark is embedded by varying the NNZ values of selected 4×4 blocks. Noorkami and Mersereau [42] have extend their I frame watermarking scheme [43] to P frames. A bipolar watermark has been embedded in non-zero AC coefficients of blocks which are selected by calculating the motion intensity.

1.3.1.2 Watermarking Schemes Using Prediction Syntax

In [44], Zou and Bloom have proposed an algorithm for embedding the watermark in the entropy encoded bit stream. Through off-line analysis, this algorithms first finds suitable blocks which remain compliant and valid bit stream even after embedding. Then, these blocks are filtered by removing the blocks whose change could violate the fidelity and degrade the quality of the imagery. The watermark is embedded by altering the intra prediction mode of 16×16 macro-block of H.264/AVC, that is, by changing mb_type field [17] of the compressed stream. Subsequently, Hu et al. [33] have given an algorithm that hides the information in the intra 4×4 luma blocks by modifying its intra prediction mode. For each suitable 4×4 embedding block, the optimal intra prediction mode is modified to the next sub-optimal intra prediction mode based on the watermark bit to maintain watermarked video quality. In a similar direction, Song et al. [45] have recently proposed another scheme where intra prediction modes of 8×8 macro-blocks of H.264/AVC are altered to the next sub-optimal intra prediction modes causing least visual distortion. In another work [46], Wang et al. have extended the sub-optimal intra prediction mode grouping for data hiding in H.265/HEVC. Yang et al. [34] have enhanced Hu's [33] work where the embeddable 4×4 blocks are selected based on intra mode distribution of 4×4 micro-blocks within a 16×16 macro-block. The nine prediction modes are divided into even and odd gatherings. To embed the watermark, the mode of the I4-block is changed from even

mode to odd mode or vice-versa depending on the mapping rule of the watermarking bits. In another scheme [47], a fragile watermarking scheme is proposed for authentication of H.264/AVC video where the watermark is embedded in the 4×4 blocks in the I-frames during encoding. The watermark is generated using the energy comparison of luma macro-blocks in all previous encoded frames within a GOP. For embedding the watermark, the prediction mode of the selected block is compelled to be odd or even depending on the watermark is 0 or 1. During authentication, the two watermarks, one generated from non-zero quantized coefficients and the other extracted from 4×4 intra prediction modes are compared to validate the authenticity of the video stream. Although, the *Peak Signal to Noise Ratio* (PSNR) of the watermarked video is quite high for this approach, it results in high BIR.

Watermarking can be also achieved by using the motion vector attributes such as the phase angle, the horizontal, and the vertical magnitudes. Feng [35]and Swaraja [48] have proposed watermarking in P frame by changing the motion vector. Guo et al. [49] proposed a method to embed the watermark in the motion vectors between two P frames. In this scheme, the horizontal and vertical offsets (i.e., odd or even) in motion vectors are modified to embed information. But these method are fragile to different attacks. Tewet al. [50] have proposed a watermarking scheme for H.265/HEVC by altering the coding block size of each coding tree.

1.3.1.3 Drift Compensated Watermarking Schemes

Due to the presence of the intra as well as the inter prediction process in the video encoding, the watermark embedding noise may propagate to the predicted blocks. The process in which this noise accumulates and propagates to the subsequent blocks in the prediction process until the next I-frame, is known as *drift error propagation*. Several watermarking schemes have been proposed for H.264/AVC where drift error is compensated after watermark embedding. Gong and Lu [51] have proposed a scheme by modifying the drift compensated quantized DC coef-

ficients in the luminance residual blocks. Zhang et al. [52] have proposed a watermarking scheme in which the drift error is eliminated by subtracting it from all of the affected blocks. Huo et al. [53] have improved the Zhang et al. scheme [52] by reducing the complexity of the drift compensation process. Different drift compensations are used for different groups of watermarked AC coefficients in this scheme. Ma et al. [54] have proposed a drift compensated watermarking scheme in which pairs of AC coefficients are selected for compensating the drift error in the selected intra direction. In this scheme [54], embedding blocks are selected depending upon directions of neighboring intra predicted blocks. In [55], authors have proposed a drift compensated reversible watermarking method by selecting embeddable 4×4 blocks depending on the intra prediction mode of the neighboring blocks. If the neighboring blocks do not use the current block for intra prediction, then the current block is selected for watermark embedding. Chen et al. [56] have proposed a drift compensated watermarking scheme for H.264/AVC by altering a group of DCT coefficients which are selected such that the reference pixels do not get modified due to embedding. Recently, Chang et al. [57] have proposed an intra frame data hiding scheme for H.265/HEVC by altering a triplet of DST coefficients in 4×4 blocks and a group of DCT coefficients in larger size of blocks.

1.3.1.4 Challenges of HEVC Watermarking

The purpose of H.265/HEVC standard is to compress the high resolution or high definition videos more efficiently than any existing standards. Since watermarking provides a plausible solution for ownership as well as content authentication, watermarking for HEVC stream can pose an interesting research problem. The very few watermarking scheme [38, 50, 57], available for HEVC encoded videos, are not robust against common video processing attacks such as re-compression, transcoding, noise addition etc. There exists several H.264/AVC watermarking schemes [32, 39, 41, 58] which embed robust watermark by altering quantized transformed coefficients. But these are not directly applicable to H.265/HEVC

because of structural differences such as the use of DST instead of DCT, more number of prediction modes, and different quad tree structures for prediction and transformation etc. The spatial correlation between the prediction planes generated by two consecutive prediction modes in HEVC has become higher due to presence of a large number of prediction modes. This increased correlation raises the probability of the mode change during re-compression. If the modes based watermarking schemes of H.264/AVC [33, 34, 44, 47] are directly applied in H.265/HEVC, the watermark is likely to be lost during re-compression. Finally, the coefficient distribution of the residual blocks becomes more sparse due to more accurate prediction in the case of the HEVC. Embedding the watermark by perturbing such sparse coefficients with a decent visual quality and bit rate becomes challenge in case of HEVC watermarking.

1.3.2 Uncompressed Domain Watermarking Schemes

Over the last decades, a number of watermarking methods [3, 7, 59–62] have been proposed to fulfill different requirements of digital video watermarking where spatial alteration [63–65] or transform domain alteration [66–68] is done to embed the signature. Some of these methods [69–83] are robust to geometric attacks where the watermarked video is distorted due to rotation, scaling or translation. These methods can not be directly applied to HD videos because of the texture difference in HD videos. Specially, it is observed that the inter-pixel correlation is generally high for the natural image or video frame in a local spatial neighborhood. This neighborhood location (with high inter pixel correlation) is increased for a higher resolution image / video frame. Thus, a new zone selection algorithm can be incorporated for HD videos. Further, camcorder theft is one of the biggest problems, that the film industries are facing and it is one of the largest source for video piracy. According to a recent study by the Motion Picture Association of America (MPAA), such thefts are responsible for 90 percent of the pirated copies of newly released films available in the internet or on DVDs [84]. Moreover, about

65% of camcorded copies emerge within a week of theatrical release [85, 86]. In a poorly supervised movie theatre, a high quality camcorder records a HD copy of the movie from the large screen and hence is able to maintain the original resolution and quality of the movie. However, due to improper placement (intentionally or unintentionally) of the camcorder, geometric attacks, such as upscaling, rotation, and cropping usually occur during camcording. Further addition of noise can occur due to frame rate conversion, inbuilt image processing functions of camcoder and the re-compression to make the video compatible with portable devices, such as smart phones, portable multimedia players (PMPs), personal digital assistants (PDAs) and tablets. These transforms de-synchronizes the embedded watermark. Thus, the watermarking schemes against camcording attacks are also focused in this dissertation.

1.3.2.1 Watermarking Schemes against Camcording Attack

Several schemes are proposed in the literature to handle the attacks during the camcording. Leest et al. [87] proposed a watermarking scheme where a watermark bit is embedded by altering the frame's mean luminance value in a group of consecutive frames. As the mean luminance value is robust to the geometric distortion, the scheme is inherently robust to geometrical distortions. However, changing the luminance values causes the flickering artifacts and the degradation of the visual quality of the watermarked video. Do et al. [88] altered the frame luminance by adding/subtracting a watermark pattern based on the pixel-value histogram. The pattern is subtracted/added in a group of consecutive frames (named as Watermarking Group of Pictures or WGP) to embed one bit of watermark information. In similar direction, Li et al. [89] recently proposed another watermarking scheme against camcording attack by altering the luminance relationship of the consecutive frames. In this scheme, each frame is divided into 8×8 blocks and the DC coefficients of those blocks are altered to embed the watermark. The Watson model [40] is used to lower the flickering effect due to embedding. Since, a large number of pixels are altered to embed the watermark, the visual quality of

watermarked video may degrade in these schemes [87–89]. Delannay et al. [90] restored the geometrically distorted comcorded videos using different geometric transforms such as affine, bilinear, curved transforms etc. prior to the extraction of the watermark. In this work, the minimum mean square error estimation is used to estimate the transformation parameters from time-synchronized cam-recorded video. The scheme is complex and non-blind as the time-synchronization process requires the original video. Lee et al. [91] proposed another watermarking scheme in which the local auto-correlation function (LACF) is used to estimate the projective distortion in a cam-recorded video. Then, the watermark pattern is restored by applying the inverse projective transform on distorted videos and the watermark is detected from those restored videos. In [86], the authors have extended the scheme in [91] for the composite projective distortion. However, it is difficult to estimate the watermark as the peaks of the LACF are not clear after several signal-processing attacks. In [92], the authors have implemented a compressed domain watermarking scheme where low frequency DCT coefficients are altered in a WGP according to the watermark information bit for MPEG-2 videos. This scheme assumes that the mean of the low frequency DCT coefficients of consecutive frames remain constant in a particular video. To avoid the watermark drift error, only B frames are selected for embedding. Stankowski et al. [93] have given an algorithm where the watermark is embedded in H.264/AVC videos. Depending on the watermarking bit values, the quantization matrix in the Picture Parameter Set is altered to change the energy of vertical or horizontal coefficients of the 4×4 or 8×8 blocks. Thus, the transform coefficients in selected P and B frames are indirectly modified for embedding. Siast et al. [94] improve the previous scheme by embedding in selected blocks which do not participated in intra prediction. In both the schemes, the watermark is detected using energy comparison between temporally aligned original video and watermarked video. As a result, temporal de-synchronization due to camcording may destroy the watermark at the time of detection.

1.3.2.2 Challenges of HD Watermarking against Camcording Attack

The process of camcording of a digital video may cause different distortions in recorded video [95]. These distortions may be due to the digital to analog/ analog to digital conversion, the pose of camera with respect to display, the camera sensor sub-sampling, additional image processing inside the camera logic etc. These distortions can be broadly classified into two groups, namely spatial distortion and temporal distortions. Spatial distortions can occur due to several reasons such as the zooming in / zooming out functionality of the camcorder, the position of the camcorder with respect to the screen and non-linear geometric transforms of the video because of the lens distortions. Further, the inbuilt image processing functionalities of camcording devices (such as Automatic Gain Control (AGC), Automatic White Balance (AWB) etc.) also cause spatial distortion. Because of the frame rate difference between display device and capturing device, temporal distortions also occur in the recorded video. Embedding a watermark by the new zone selection algorithm which is robust against spatial and temporal de-synchronizations and has decent visual quality, becomes a challenge in case of HD watermarking.

1.4 Motivation and Objectives

From the above discussions, it can be observed that H.265/HEVC is designed to handle high resolution or high definition videos more efficiently than any existing standards. Since watermarking provides a plausible solution for ownership as well as content authentication, compressed domain watermarking for H.265/HEVC stream can be an interesting research problem. In this regard, re-compression is one of the major problems in achieving robust compressed domain watermarking. One of the primary motivations of this dissertation is to resist re-compression attack for HD video sequence. Further, it is also observed that H.265/HEVC compression has some major differences with previous compression standards (H.261, H.264/AVC etc.), that may play a critical role in the performance of the

watermarking schemes. For example, the DST is included in H.265/HEVC for transformation of 4×4 TBs in place of DCT which has been used transform in case of H.264/AVC. The number of prediction modes as well as the heterogeneity among the macro-block sizes are increased in HEVC. Moreover, the coefficient distribution of the residual blocks becomes more sparse due to more accurate prediction in the case of the HEVC. It is experimentally observed that perturbing such sparse coefficients leads to substantial visual artifacts. Therefore, a direct extension of existing H.264/AVC based watermarking algorithms to the HEVC stream may not always give the optimal results.

It is observed additionally that HD and beyond-HD videos can be copied with very minimal distortion using advanced camcording devices. In the camcorded videos, the embedded watermark suffers from geometric, projective and temporal modifications due to digital to analog and then analog to digital conversions [95]. Additionally, different noise elimination filters are applied to these recorded videos for improving quality. These attacks cause de-synchronization of the watermark in cam-corded videos. Thus, the camcorder based video copying becomes a serious threat to a digital watermarking based authentication system.

Motivated by these issues, the main objective of this dissertation is to enhance the robustness of the watermarking schemes for the HD and beyond-HD videos against different attacks like re-compression, transcoding, common image and video processing attacks etc. while maintaining the decent visual quality of the watermarked video. This has been carried out by,

- Proposing robust watermarking schemes for H.265/HEVC compressed videos against the re-compression, image and video processing attacks using only H.265/HEVC compressed domain parameters such as prediction modes, transformed coefficients, motion vectors etc.

- Maintaining a decent visual quality of the watermarked video by proposing suitable zone selection methods for watermarking specifically designed for HEVC compression and

- Using of spatial and temporal invariant regions of uncompressed HD videos to propose robust watermarking against camcording attack.

1.5 Contribution of the thesis

The major contributions of the thesis are as follows,

1.5.1 Prediction Mode Based H.265/HEVC Video Watermarking Resisting Re-compression Attack

In the first contributory chapter, a compressed domain blind watermarking scheme is proposed. It embeds watermark by altering the intra prediction modes of selected 4×4 intra prediction blocks of the H.265/HEVC video. Though, several watermarking schemes exists for H.264/AVC using the intra prediction mode, they are all fragile against the re-compression attacks. Moreover, the spatial correlation between prediction planes generated by two consecutive prediction modes in HEVC becomes higher due to the presence of a large number of prediction modes. This increased correlation increases the probability of the mode change during re-compression. If mode based watermarking schemes of H.264/AVC are directly applied to H.265/HEVC, the watermark is likely to be lost during re-compression. To achieve robustness, first the 4×4 luma PBs of intra frame are chosen for embedding based on the sustaining probability after re-compression. Then, the intra prediction modes of H.265/HEVC are grouped in a fashion so that each group represents two watermark bits. The groups ensure that the mode change due to re-compression can be closed within a group. Experimental results on various test sequences show that the scheme is robust to re-compression and has very low effect on the bitrate and the visual quality of the watermarked video.

1.5.2 Robust Watermarking of H.265/HEVC Videos using Temporally Homogeneous Blocks

In this chapter, the robustness and visual quality of watermark video are improved by exploiting temporal homogeneity of textures. In this work, the watermark is embedded by altering the number of nonzero transform co-efficients (NNZ) of 4×4 transform blocks of the HEVC video sequence. To embed the watermark, temporally homogeneous blocks in the consecutive intra frame are first found out using motion characteristics of the intra frame which is determined by the motion information of the inter (P or B) predicted frames of its close neighborhood. As consecutive frames within a small temporal neighborhood have very high correlation when no scene change is detected, the many blocks can be found within short temporal neighborhood having similar textures. Therefore, if a pair of temporally homogeneous blocks is selected from consecutive I-frames, they have similar NNZ values. Due to very less NNZ difference in the selected pair of motion coherent block, perturbing of the transform coefficients due to watermark embedding is very minimal. This leads to very less degradation in the visual quality. In this scheme, two groups of such temporally homogeneous TBs from consecutive intra frames are used to embed the watermark by altering the NNZ values. Further, the block selection is done in such a way that the watermark can also be extracted when the watermarked 4×4 TBs are merged into a 8×8 TB. It is experimentally shown that the proposed scheme performs well against re-compression and image processing attacks while maintaining a decent visual quality (PSNR) and BIR of the watermarked video.

1.5.3 Drift Compensated Robust Watermarking for H.265/HEVC Videos

It has been observed in the recent literature that the drift error due to watermarking degrades the visual quality of the embedded video. The existing drift error handling strategies for recent video standards such as H.264/AVC may not

be directly applicable to H.265/HEVC videos due to different compression architectures. Specially, the use of the DST transformation for 4×4 TBs in intra frame instead of the DCT restrict the use of H.264/AVC based drift compensated algorithm in H.265/HEVC. Therefore, a compressed domain watermarking scheme is proposed for H.265/HEVC bit stream which can handle drift error propagation both for intra and inter prediction processes. The prevention of intra drift errors is achieved by excluding those residual coefficients which are used in the intra prediction process. The inter frame drift error has been prevented by excluding the watermarked blocks from the inter prediction process. Instead, a block having high spatial correlation with the watermarked block is used for the inter prediction process. So, the embedding noise can not propagate to the predicted blocks. Additionally, the proposed scheme shows adequate robustness against re-compression attack as well as common image processing attacks while maintaining a decent visual quality. A comprehensive set of experiments has been carried out to justify the efficacy of the proposed scheme over the existing literature.

1.5.4 A Robust Watermarking Scheme for HD Videos against Camcording Attack

In this final contributory chapter, a watermarking scheme is presented, which is robust against video camcording attack. The camcording video is mostly affected by the frame blending and projection attack which cause both spatial and temporal de-synchronization of the embedded watermark. The required temporal synchronization to resist these attacks, is achieved in two steps. First, the use of shot segments which are robust against geometric distortion and frame blending, maintains the synchronization of a cam-corded video with the original video. Secondly, the shot segment is divided into embedding segments and a bit of watermark is embedded into the low frequency sub-band of an embedding segment. Thus the use of low frequency in temporal direction makes the scheme more robust against frame blending. Further, the regions around robust *Scale-Invariant*

Feature Transform (SIFT) points in the low frequency sub-band is used to make the scheme robust against geometric distortion. The effectiveness of the scheme is experimentally verified against the camcording attacks.

1.6 Organization of the Thesis:

This PhD dissertation consists of seven chapters. The first chapter consists of a brief introduction of image and video watermarking, Compression of HD Video, a brief literature survey, research motivation and objectives of the thesis, contribution of the thesis. The rest of the thesis is organized as follows,

- Chapter 2 describes the background of the research which includes some preliminary concepts like the 3D-*Discrete wavelet transformation* (DWT), the *Scale-Invariant Feature Transform* (SIFT) etc., evaluation metrics, experimental data set etc. which are to be used in later chapters.
- In Chapter 3, a robust watermarking scheme is proposed for H.265/HEVC video sequences by altering a compressed domain parameter, namely, the intra prediction mode of 4×4 prediction blocks. The robustness of the proposed scheme is achieved by using sustainable mode grouping.
- Chapter 4 first proposes a the procedure of obtaining temporally homogeneous blocks in H.265/HEVC videos using only compressed domain parameters. Then, a robust watermarking algorithm is proposed by altering the number of nonzero transform co-efficients of these temporally homogeneous blocks.
- Chapter 5 presents a compressed domain robust watermarking scheme with improved visual quality by preventing drift propagation error both for intra and inter prediction processes. First, the problems with the extension of existing drift compensation algorithm to H.265/HEVC have been discussed. Then the drift compensated watermarking process for H.265/HEVC is presented.

- Chapter 6 describes a watermarking algorithm for HD videos against camcording attack. During camcording, both spatial and temporal distortion synchronize the watermark in the captured video. In this chapter, first those distortions are discussed briefly. Then a novel blind watermarking scheme is proposed which are robust against camcording attack.
- In Chapter 7, a brief conclusion of the Ph.D. dissertation is presented with the future scope of research directions.

1.7 Summary

In this introductory chapter, a brief introduction is presented over the HD video research domain to formulate the scope of research in this field. First, the representation of the HD video sequences is explained for both compressed and uncompressed video sequence. Then, a brief literature of HD video watermarking with the limitations is described. Based on the limitations, motivation and the objective of the research work are formulated. Finally, a brief description of contribution and the organization of the thesis is presented.

Chapter 2

Research Background

In this chapter, a brief overview of mathematical preliminaries and theoretical foundations relevant to the topics of interests are presented. This includes a discussion on the SIFT and the 3D-DWT. In addition, the different evaluation parameters for the proposed algorithms and corresponding data set used for experimentations are also described.

2.1 Scale Invariant Feature Transform (SIFT)

The SIFT [96] is used to find the distinct rotation and scale invariant features from images. These features (called the SIFT features) are also robust against the luminance change and invariant to changing viewpoints up to a certain point. The SIFT features can be used to perform reliable matching between different views of an object or scene as these are highly distinctive and are matched with a high probability against large image distortions. The steps of detecting SIFT features of an image are as follows:

2.1.1 Detection of Scale-space Extrema

To make the SIFT feature points invariant to the scale change, stable locations across all possible scales are identified using a continuous function of scale known as the scale space. The scale space of an image $I(x, y)$ is obtained by convolving

it with variable-scale a Gaussian kernel $G(x, y, \sigma)$ where σ is standard deviation and represents the scale. The scale space of an image $I(x, y)$ can be represented as,

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.1}$$

where $*$ is the convolution operator in x and y and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} exp^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.2}$$

The maxima and minima of the scale-normalized Laplacian of the Gaussian produce the most stable image features compared to other possible image functions such as the gradient, the Hessian, or the Harris corner function. The *Difference of Gaussian* (DoG) function provides a close approximation to the scale-normalized Laplacian of Gaussian [97, 98]. So the DoG function, $D(x, y, \sigma)$ is obtained by subtracting two nearby scales separated by a constant multiplicative factor k,

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y, \sigma) \tag{2.3}$$

A local maximum or minimum of the DoG is computed based on the comparison of the sample point and its eight neighbors in the current image as well as the nine neighbors in the scale above and below. These extrema are selected as the initial candidates of feature points.

2.1.2 Keypoint Localization

In the next stage, more accurate feature points are selected by discarding the low contrast points (sensitive to noise) and the poorly localized points along the edges (unstable). The low contrast points are detected based on $\mid D(x, y, \sigma) \mid$. If $\mid D(x, y, \sigma) \mid$ is below the threshold value (the *peak threshold*), the point is discarded. The further outliers in those feature points can be removed by discarding the poorly localized points along the edges. It has been observed that a poorly defined peak in the DoG function has a very low principal curvature along the edge and a very high principal curvature across the edge. Based on the

ratio of principal curvatures of each candidate point, the unstable key points are discarded by a threshold ratio (the *edge threshold*).

2.1.3 Orientation Assignment

After selecting the robust feature points, each of them is assigned with one or more orientations based on the local image gradient direction to make it rotation invariant. For each image sample $L(x, y)$ at the key-point scale σ, the gradient magnitude $mag(x, y)$ and orientation $ori(x, y)$ are calculated using pixel differences as follow,

$$\begin{aligned} mag(x,y) &= \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \\ ori(x,y) &= \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y))) \end{aligned} \tag{2.4}$$

Then an orientation histogram with 36 bins is formed from the gradient orientations of sample points within a region around the keypoint, and the peak of this histogram is selected as the direction of that feature point.

2.1.4 Keypoint Descriptors

Finally, a highly distinctive descriptor (typically 128 dimensional) is computed for each feature point using the gradient magnitude and the orientation at each image pixels within the 16×16 neighborhood. This region is sub-divided into 16 4×4 regions and an 8 bin orientation histogram is created for each of them. Finally, a total of 128 bin values is available, which is represented as a vector to form the keypoint descriptor.

2.1.5 Keypoint Matching

The best candidate match for each keypoint is found by identifying the descriptor from another set of keypoints which have the minimum Euclidean distance from the candidate keypoint descriptor. To reduce the false matches, only those matches are selected for which the ratio between distances to the nearest and the second nearest point is less than a certain threshold.

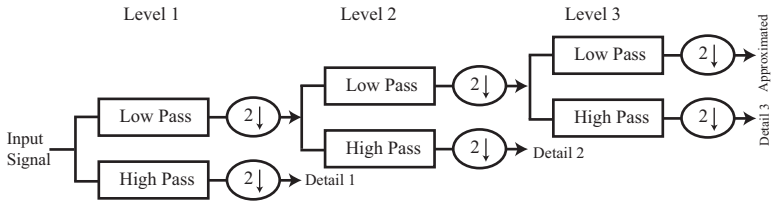


Figure 2.1: *DWT Multi-scale representation [99]*

2.2 Discrete Wavelet Transform (DWT)

The discrete wavelet transform (DWT) provides a multiresolution decomposition of a given signal and is widely applied in many image processing fields such as noise attenuation, image enhancement, and motion detection. The DWT can be described mathematically as a set of inner products between a finite-length sequence $f(n)$ and a discretized wavelet basis $\psi_{j,k}(n)$ as follows [100],

$$WF(j.k) = \sum_{n=0}^{N-1} f(n).\psi^{c}_{j,k}(n) \tag{2.5}$$

where $WF(j.k)$ are the DWT coefficients, N is the length of the sequence and c denotes a complex conjugate. The discretized wavelet basis can be expressed as

$$\psi_{j,k}(n) = \frac{1}{\sqrt{s^{j}_{0}}}\psi(\frac{n - s^{j}_{0}.k}{s^{j}_{0}}) \tag{2.6}$$

where s^{j}_{0} and $s^{j}_{0}.k$ are the discretized versions of the scale and translation parameters respectively and ψ is the mother wavelet. For a dyadic DWT, $s_0 = 2$. In practice, the dyadic DWT can be implemented using a set of high-pass and low-pass filters in a filter bank. As illustrated in Figure 2.1, the DWT of a signal is calculated by recursively applying the lowpass/highpass frequency decomposition to the lowpass output [99]. In case of 2D signal such as an image, it can be decomposed into four subbands LL, HL, LH, and HH using the DWT where LL sub-band represents the approximation of the image. Most of the energy is concentrated in low frequency sub-band LL and the remaining sub bands HL, LH, and HH represent the finest scale wavelet coefficients, i.e., the detailed image.

These high frequency sub-bands include the edge and the texture of the image. The DWT is also used in processing 3D data such as video and medial image sequences. The 3D-DWT is a separable dyadic tree structured transform where a video segment V of size $n_1 \times n_2 \times n_3$ is decomposed into multiple sub-bands based on variations both in the spatial and the temporal dimension [101]. Given a video segment $V[n_1, n_2, n_3]$, its one level 3D DWT is obtained by performing a subband decomposition by appling 1-D analysis filter bank, first on the rows, then on the columns, and finally on the temporal dimension, thus obtaining the eight subbands $V^{(1),a,b,c}[\frac{n_1}{2}, \frac{n_2}{2}, \frac{n_3}{2}]$ with $(a, b, c) \in \{L, H\}$, where a, b, and c refer to the lowpass (L) and to the high pass (H) representation of the given signal along the rows, columns, and the temporal dimension respectively. Thus, after l level decomposition of $V[n_1, n_2, n_3]$, the subbands can be written as $V^{(l),a,b,c}[\frac{n_1}{2^l}, \frac{n_2}{2^l}, \frac{n_3}{2^l}]$ and the lowest frequency subband can be presented as $V^{(l),L,L,L}[\frac{n_1}{2^l}, \frac{n_2}{2^l}, \frac{n_3}{2^l}]$. The block diagram of one level decomposition of a video is illustrated in the Figure 2.2.

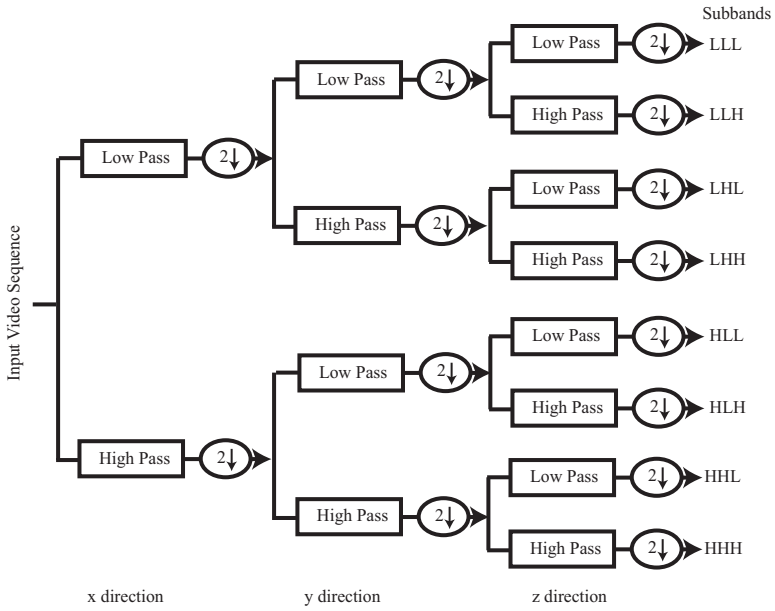


Figure 2.2: *Block diagram of a single level decomposition for the 3D DWT [101]*

2.3 Drift Error Propagation

Due to the presence of the intra as well as the inter prediction processes in the video encoding, the watermark embedding noise may propagate to the predicted blocks. The process in which this noise accumulates and propagates to the subsequent blocks in the prediction process until the next I-frame, is known as *drift error propagation* [54]. In Fig 2.3, the drift error propagation for an intra frame is shown. In this case, the reference pixel is altered due to watermark embedding. During intra prediction of the block, the embedding induced distortion of the reference block will propagate to the predicted block. This visual distortion would accumulate from the upper left to the lower right in an intra frame. During inter prediction, the noise propagates to the predicted blocks in inter frames. In this fashion, the noise propagates until next intra predicted frame.

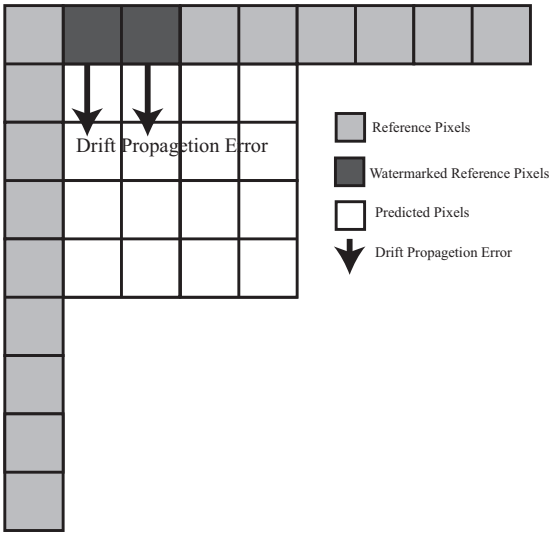


Figure 2.3: *Intra drift error propagation*

2.4 Video Watermarking Attacks

The robustness of the watermarking algorithms can be evaluated against different intentional and unintentional attacks. Some of them are described as follows,

- **Synchronization Error:** Synchronization error [32] mainly occurs in the case of highly compressed video watermarking such as H.265/HEVC, H.264/AVC. When the watermark is embedded in such compressed video, a subset of coefficients should be selected to embed the watermark for maintaining visual quality. However at the time of watermark detection or extraction, the detector/decoder may not find the same subset due to changes after re-compression with different QP values. Even if the re-compression is done with the same compression parameters, the error occurs due to changes in the compressed domain parameters during watermarking. Apart from the re-compression attack, synchronization error can also occur in the case of transcoding attack where the video format is modified in order to access the video content in the end user's devices.

- **Image Processing Attacks:** Some common image processing attacks are noise addition, filtering [32] [43] etc. In noise addition, the video signal is degraded by adding noise (Salt & Pepper noise, Gaussian noise etc.) with different intensity. Image filtering (Gaussian filter, average filter etc.) with different parameters can also be applied to degrade the watermark by changing the pixel values. Also, different geometric distortions can be done by applying scaling, rotation the each frames of video to de-synchronize the watermark.

- **Temporal Attacks:** Temporal attacks can be performed by altering the video in the temporal direction [102]. One such attack is frame dropping where some of the frames are removed from the video sequences. In frame averaging attack, each frame in the video sequence is replaced by another frame which is generated by averaging the current frame with the neighbor frames.

- **Camcording Attacks:** Because of the recent advancement in the display devices and the camcording devices, digital movies can be copied with min-

imal distortions using a camcording device [95]. During this process, the embedded watermark suffers from different distortions in recorded video due to the digital to analog/ analog to digital conversions, the pose of the camera respect to the display and the camera sensor sub-sampling, additional image processing inside the camera etc. [89].

2.5 Evaluation Parameters

The watermarking algorithms can be evaluated with respect to different parameters such as visual quality, watermark strength, bit increase rate. These parameters are outlined below.

2.5.1 Visual Quality

In this work, the average *Peak Signal to Noise Ratio* (PSNR), the average *Structural Similarity* (SSIM) index and the average *Visual Information Fidelity in pixel domain* (VIFp) [103–105] are measured to evaluate the visual quality of the watermarked video. These metrics of the watermarked video are calculated with respect to original YUV video sequence using the VQM Tool [103].

- **PSNR:** The PSNR is the ratio between the maximum power of the signal and the power of distorting noise that affects the quality of the signal. The PSNR of a watermarked image I' is calculated as follows,

$$PSNR = 10 \log_{10} \frac{Peak^2 \times N_1 \times N_2}{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (I(i,j) - I'(i,j))^2} \tag{2.7}$$

  where $Peak$ is maximum possible intensity, N_1 and N_2 are the width and the height of the frame/image respectively and I is the original image.

- **SSIM:** The SSIM index is an image quality assessment metric based on the degradation of structural information. The SSIM index between original image I and watermarked image I' can be measured using luminance

similarity, contrast similarity and structural similarity [104]. The SSIM index is first computed within local patches which are obtained using a sliding window that moves pixel-by-pixel across the image, resulting in a SSIM map. Then the SSIM score of the entire image is then computed by averaging the SSIM values from the SSIM map across the image. Suppose that x and y are local image patches taken from the same location of two images that are being compared. The local SSIM index is measured using luminance similarity $l(x, y)$ of the local patch, contrast similarity $c(x, y)$ of the local patch and structural similarity $s(x, y)$ of the local patch [106].

$$\begin{aligned} SSIM &= l(x,y)c(x,y)s(x,y) \\ &= \left(\frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}\right)\left(\frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}\right)\left(\frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}\right) \end{aligned} \tag{2.8}$$

where $\mu_x$and μ_y are (respectively) the local sample means of x and y,σ_x and σ_y are (respectively) the local sample standard deviations of x and y, and σ_{xy} is the sample cross correlation of x and y after removing their means. The items C_1,C_2, and C_3 are small positive constants that stabilize each term, so that near-zero sample means, variances, or correlations do not lead to numerical instability.

- **VIFp:** The VIFp is a measures that quantifies the information fidelity for the entire image in a statistical model of the *Human Visual System* (HVS) [105]. The VIFp considers two data variables. The first one is the statistics between the initial and the final stage of the visual channel when there is no distortion. The second variable is mutual data between the input of distortion block and the output of visual system block. The VIFp is computed for a collection of $N \times M$ wavelet coefficients from each sub-band using the following equationVIFp [105],

$$VIFp = \frac{\sum_{j \in subbands} I(C^{N,j}; F^{N,j} | s^{N,j})}{\sum_{j \in subbands} I(C^{N,j}; E^{N,j} | s^{N,j})} \tag{2.9}$$

where,$I(C^{N,j}; F^{N,j}|s^{N,j})$ and I $(C^{N,j}; E^{N,j}|s^{N,j})$ represents the information that could ideally be extracted by the brain from a particular sub-band in the reference and the test images, respectively.

2.5.2 Robustness Parameter

To measure the robustness of the watermarking scheme, the normalized correlation coefficient ($corr$) between the extracted watermark vector (WE) and the original watermark vector (WS) is calculated as follows [107],

$$corr = \frac{WS.WE}{\sqrt{WE.WE} \times \sqrt{WS.WS}} \tag{2.10}$$

where $WS.WE$ represents the scalar product between WS and WE.

2.5.3 Bit Increase Rate (BIR)

The BIR is defined as the percentage of the increase in the bit-rate per embedded bit [32]. It is given by,

$$BIR = \frac{BR(Watermarked) - BR(Original)}{embeddedCapacity \times BR(Original)} \times 100 \tag{2.11}$$

where $BR(Watermarked)$ and $BR(Original)$ are the number of bits utilized for coding the watermarked and the original sequences respectively and $embeddedCapacity$ defines total number of bits embedded in the image or video sequence.

2.6 Experimental Dataset

For experimentation, a wide range of videos sequences with different resolutions, textures, and motions are used. For example, *BQMall* and *People on street* video sequences have high texture regions and motion, and *Jonny* and *Kristen & Sara* sequences have lower texture variations and very less motion. The proposed algorithms are implemented using the H.265/HEVC reference software HM.10 [13]. Details of test sequences are tabulated in Table 2.1.

Table 2.1: ***Details of Test Sequences***

Parameters for Test Sequences	**Values Taken**
Video Sequence Used	BasketBall, BlowingBubbles, BQTerrace, BQMall, Cactus, Chipmunks, Craving, FlowerVase, HoneyBee, Johnny, Jocky, Kristen & Sara, Komono, ParkScene, People on street, ReadySteadyGo, shields, Tennis, Yachtride
Frame Rate	30, 50, 60, 120 Frame per Sec.
Video Resolutions	416×240, 832×240, 1280×720, 1920×1080, 2560×1600
Bit Depth	8 bit/pixel

2.7 Summary

In this chapter, background concepts on the SIFT, the DWT and the Drift error propagation are presented. Also, few evaluation parameters are explained. These concepts are used in the watermarking schemes which are proposed in the later chapters. The evaluation parameters described are used to measure the efficiency of those schemes. The dataset used for experimentation is also summarized in this chapter.

Chapter 3

Prediction Mode Based H.265/HEVC Video Watermarking Resisting Re-compression Attack

As mentioned in Chapter 1, the H.265/HEVC [2] is the most recent joint video project of the ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG). It is specially designed for compression of HD videos. For the same visual quality, the compression by H.265/HEVC is around twofold higher than that by the past standard H.264/AVC [15] [21] [17]. Because of the expanding prevalence of the HD and beyond HD videos and the increase in processing power of the end user devices like the cellular device, PDA etc., H.265/HEVC is relied upon to supplant the past standard H.264/AVC as it has more compression efficiency. Since the distribution of duplicated digital video can be performed easily, the copyright protection and the content authentication of the H.265/HEVC video become an important aspect. In this Chapter, a robust watermarking scheme is proposed to protect the ownership of the H.265/HEVC video.

The proposed watermarking scheme is designed considering only the compressed domain parameters as this compressed domain embedding leads to lower time complexity than the uncompressed domain embedding [43] and most of the

videos are stored and transmitted in a compressed format. In compressed domain watermarking, the watermark is embedded in the syntax domain by modifying transform coefficients [43] [32], intra prediction modes [33, 34] or motion vectors [35]. As discussed in Chapter 1, quite a few papers have been reported in the recent literature discussing methods for watermarking of H.264/AVC videos. Though, the majority of them cannot be directly applied for the watermarking of the H.265/HEVC videos due to the complex coding structure [2] [21] of H.265/HEVC. Also, the complex architecture leads to fewer transform coefficients to represent a video for same visual quality than the previous standards and more side information for encoding, such as the increase in the number of intra prediction modes, quad tree structure etc.. In this chapter, such an increased side information, the intra prediction mode is used to insert the watermark. As very few works have been done for intra prediction mode based H.265/HEVC watermarking, a comprehensive survey on prediction mode based watermarking for H.264/AVC is first described and then the applicability of these existing schemes to H.265/HEVC is analyzed.

In 2008, Zou and Bloom [44] have proposed an algorithm for embedding the watermark by changing the mb_type field [17] of 16×16 intra predicted macro-blocks of the H.264/AVC encoded bit stream. Subsequently, Hu et al. [33] have given an algorithm that hides the information in the intra 4×4 luma blocks by modifying its intra prediction mode. For each suitable 4×4 embedding block, the optimal intra prediction mode is modified to the next sub-optimal intra prediction mode based on the watermarking bit to maintain watermarked video quality. In a similar direction, recently Song et al. [45] proposed another scheme where intra prediction modes of 8×8 macro-blocks of H.264/AVC are altered to the next sub-optimal intra prediction modes causing least visual distortion. In [46], Wang et al. have extended sub-optimal intra prediction mode grouping for data hiding in H.265/HEVC. Due to the diversity of video textures, the joint probability of mode being optimal and another mode being sub-optimal mode is very low. These make proposed schemes in [33], [45] and [46] complex and ineffective for

H.265/HEVC. Additionally, the BIR for these schemes is also very high. Yang et al. [34] have enhanced Hu's [33] work where the embeddable 4×4 block is selected based on intra mode distribution of 4×4 micro-blocks within a 16×16 macro-block. The nine prediction modes are divided into even and odd gatherings. To embed the watermark, the mode of the I4-block is changed from even mode to odd mode or vice-versa depending on the mapping rule of the watermarking bits. In another scheme [47], fragile watermarking is proposed for H.264/AVC content authentication where the watermark is embedded in the 4×4 blocks in the I-frames during encoding. The watermark is generated using the energy comparison of luma macro-blocks in all previously encoded frames within a GOP. For embedding the watermark, the prediction mode of the selected block is compelled to be odd or even according to the watermark is 0 or 1. During authentication, the two watermarks, one generated from non-zero quantized coefficients and the other extracted from 4×4 intra prediction modes are compared to validate the authenticity of the video stream. Although the PSNR of the watermarked video is quite high for this approach, it results in high BIR.

The main disadvantage of the above mentioned schemes is that they are all fragile against the re-compression attacks. The spatial correlation between prediction planes generated by two consecutive prediction modes in HEVC has become higher due to the presence of a large number of prediction modes. This increased correlation raises the probability of the mode change during re-compression. If modes based watermarking schemes of H.264/AVC are directly applied in H.265/HEVC, the watermark is likely to be lost during re-compression. On the other hand, the inter-correlation between prediction planes generated by this large number of modes can be analyzed to propose a robust intra prediction mode based watermarking scheme. The main objective of this work is to cluster the prediction modes in such a way that they are sustained within the same group after re-compression.

3.1 Proposed Scheme

From the aforementioned literature, it has been observed that the watermarking schemes which embed the watermark by modifying the structural information (such as macro block size, intra prediction mode, motion vector etc.) are not robust against the re-compression attack. In the Section1.2.1.2, it is illustrated that for the efficient compression of HD and beyond HD Videos, H.265/HEVC has adopted more intra prediction modes. The goal of this chapter is to propose a robust watermarking scheme by exploiting this increased structural information for H.265/HEVC compressed videos.

3.1.1 Embedding Condition

In the proposed scheme, a watermark is embedded by altering the mode information of the selected luma PBs. Intuitively, a PB having a small size and with a larger number of non zero transform coefficients belongs to a higher texture region of the video frame. So, the intra predicted 4×4 luma PBs are selected for embedding to minimize the visual artifacts. *Henceforth, LPB denotes* 4×4 *Luma PB and* $\mathcal{N}$ *denotes the number of non-zero transformed coefficients in a block.* In H.265/HEVC, the prediction block structure is different from the transformation block structure. So, the $\mathcal{N}$ value for a PB can be derived from its co-located TB. Similarly, the $\mathcal{N}$ value for a larger PB (say 8×8) may be derived by accumulating the $\mathcal{N}$ values of the corresponding TBs (i.e. accumulating the $\mathcal{N}$ values of all 4 TBs each having size 4×4). In this section, spatial texture analysis is done based on the $\mathcal{N}$ of the LPB to reduce the desynchronization error due to the re-compression of H.265/HEVC videos. Achieving the robustness against re-compression attacks, it is important to analyze the sustainability of an LPB (denoted as *LPB sustainability*) as well as the prediction mode across the re-compression process. In the next subsection, it is shown how the LPBs are sustained against the re-compression process when no watermark has been embedded. Subsequently, the prediction mode sustainability has been analyzed for

both the cases when a watermark is not embedded (say *unmarked mode sustainability*) and when a watermark is embedded (say *marked mode sustainability*).

3.1.1.1 Relation Between $\mathcal{N}$ and LPB sustainability

In this subsection, a relation between the $\mathcal{N}$ and LPB sustainability is analyzed for H.265/HEVC videos. H.265/HEVC selects the block structure in such a way that minimizes the Lagrangian cost with respect to the distortion and the bit requirements for encoding a block. The choice of the PB and its subsequent partitioning into the TBs are determined from a set Υ of coding decisions as illustrated in Figure 3.1. Let ρ be the prediction block chosen from Υ by the

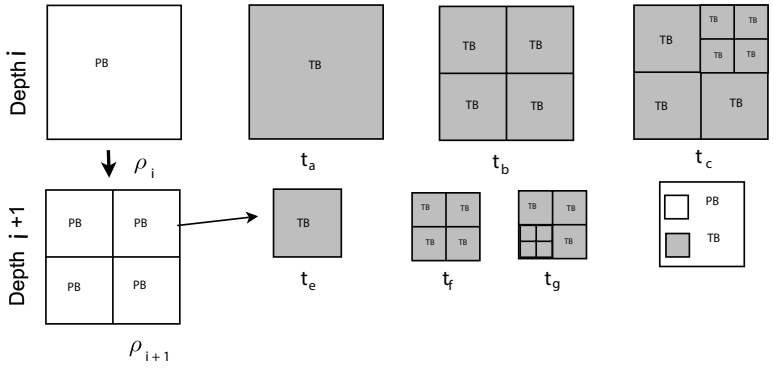


Figure 3.1: *Different coding decision for a block of original pixels. In figure, the PB ρ_i can be encoded using different set of TU t_a, t_b, t_c PB and different intra modes. ρ_i can also be partitioned into lower size PBs say ρ_{i+1} and each ρ_{i+1} can be encoded with different modes and size of TBs.*

Lagrangian cost minimization function. The associated minimum Lagrangian cost $\mathcal{C}$ for the PB ρ can be obtained by the following rate-distortion function

$$\mathcal{C} = \min_{\rho \in \Upsilon} \mathcal{D}(\rho) + \lambda.\mathcal{R}(\rho) \tag{3.1}$$

where $\lambda \geq 0$ is the Lagrangian multiplier, $\mathcal{D}(\rho)$ is the rate distortion due to compression with PB ρ and $\mathcal{R}(\rho)$ includes bit requirement for signaling the coding mode, side information (intra modes, motion vector and reference index and coding modes of sub blocks) and transform coefficients. If ρ is further subdivided

into i parts, then the total cost of subdivision ($\mathcal{C}'$) can be calculated as,

$$\mathcal{C}' = \sum_{i} \min_{\rho_i \in \Upsilon_{k,i}} \mathcal{D}(\rho_i) + \lambda.\mathcal{R}(\rho_i) \tag{3.2}$$

where, $\mathcal{D}(\rho_i)$ and $\mathcal{R}(\rho_i)$ are respectively the rate distortion and the bit requirement for i^{th} partition of ρ i.e. ρ_i. Intuitively, if $\mathcal{C}'$ is lower than $\mathcal{C}$, ρ is divided into smaller parts. The bit requirements to encode a block depend on two factors: the $\mathcal{N}$ value of the block and the associated side information.

If the *Quantization Parameter* (QP) has been increased during re-compression, the $\mathcal{N}$ value for a block decreases. So the bit requirement may be more depending on the side information for smaller blocks (where side information is relatively higher). In this scenario, the subdivisions of a PB may result in a higher Lagrangian cost. Particularly, in the case of re-compression with a higher QP, smaller PBs having lower $\mathcal{N}$ values tend to be clustered with neighbor's PBs to make a larger PB and thus shows lower *LPB sustainability*. This fact is evident from the Figure 3.2 where the probability of LPB sustaining P_{BN} for different $\mathcal{N}$ has been shown. These results are obtained using different standard HD

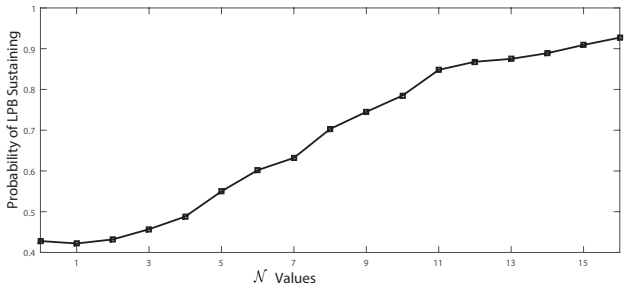


Figure 3.2: *Probability of LPB sustaining for different $\mathcal{N}$ values due to re-compression of QP 16 to 28*

videos (*PeopleOnStreet, KristenAndSara, Kimono, Johnny, BQMall* etc.) with re-compression from QP=16 to QP=28. Thus, to achieve efficient watermarking, the selected Luma PB should have higher $\mathcal{N}$.

3.1.1.2 Relation Between $\mathcal{N}$ and Frequency of LPB

The main goal of the H.265/HEVC intra prediction is to reduce the bit rate by predicting the most correlated block for each PB. It has been observed for HEVC, that the residual error for intra prediction is very small for most of the PBs. Therefore, a very few PBs having high $\mathcal{N}$ value are found. Let $\mathbb{I}$ be the set of all intra frames of a video sequences and N=$|\mathbb{I}|$, where $|\ |$ represents cardinality of the set. Let the set of LPBs of frame i be denoted by $\mathbb{L}_i$ and the subset of $\mathbb{L}_i$ having $\mathcal{N}$=n be denoted by $\mathbb{L}_i^n$. The conditional probability of finding an LPB in a intra frame for given $\mathcal{N} = n$, denoted by P_{FN}, can be obtained as,

$$P_{FN} = \frac{1}{N} \times \left(\sum_{\forall i \in I} \frac{|\mathbb{L}_i^n|}{|\mathbb{L}_i|}\right) \tag{3.3}$$

In Figure 3.3, the plot of P_{FN} for different $\mathcal{N}$ is shown. It can be observed that one has to choose the blocks having relatively lower $\mathcal{N}$ to increase the embedding capacity.

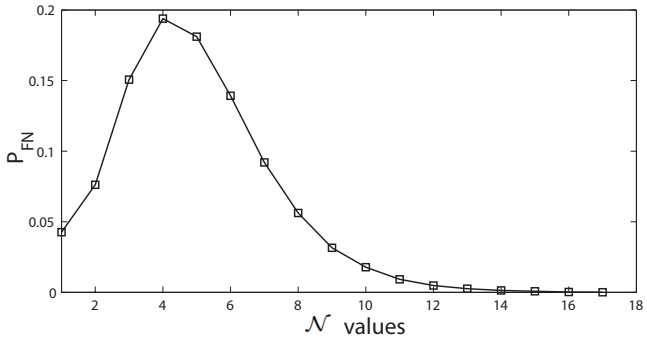


Figure 3.3: *P_{FN} for different $\mathcal{N}$ and for different videos with compression QP=16*

3.1.1.3 Relation Between $\mathcal{N}$ and the Unmarked Mode Sustainability

As discussed in Sec. 3.1.1.1, the prediction modes of H.265/HEVC for the intra predicted blocks are selected by minimizing the Lagrangian cost function of spatial distortion and the encoding bit requirement. During re-compression, the intra prediction mode of a PB may change due to this minimization. In this subsection,

the relation between $\mathcal{N}$ and *Unmarked Mode Sustainability* (i.e. sustainability of intra prediction mode of a PB when no watermark signal has been inserted) is analyzed. Consider the following,

$\mathbb{M}$: The set of intra prediction modes. Here, $|\mathbb{M}|$=35.

m_b : The intra prediction mode of an LPB before re-compression.

m_r : The intra prediction mode of an LPB after re-compression.

$\mathbb{LB}$: The set of all LPBs before re-compression.

$\mathbb{LB}^{m_b,n}$: The subset of $\mathbb{LB}$ having mode m_b and $\mathcal{N} = n$.

$\mathbb{LB}^{m_b}$: The subset of $\mathbb{LB}$ having mode m_b. $\mathbb{LB}^{m_b} = \bigcup_{n=0}^{16} \mathbb{LB}^{m_b,n}$

$\mathbb{LR}$: The set of all LPBs after re-compression.

$\mathbb{LR}^{m_r}$: The subset of $\mathbb{LR}$ having mode m_r after re-compression.

The set LPBs sustained after re-compression can be represented by $\mathbb{LB} \cap \mathbb{LR}$.

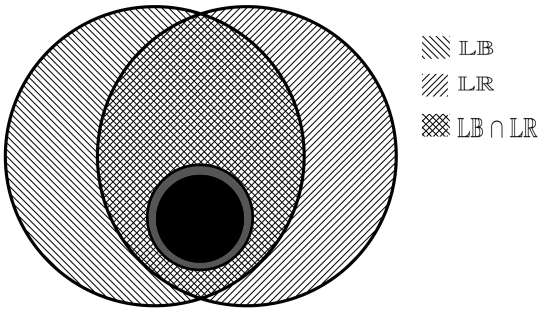


Figure 3.4: *The Set of* 4×4 *Luma PBs before and after re-compression. The intersection* $\mathbb{LB} \cap \mathbb{LR}$ *represents the sustained LPBs after re-compression. The grey inner circle represents sustained LPBs which have* $\mathcal{N} = n$ *before re-compression. The black inner circle represents sustained LPBs which have* $\mathcal{N} = n$ *before re-compression and* $m_r = m_b$.

In Figure 3.4, $\mathbb{LB}$ and $\mathbb{LR}$ are represented by a Venn-diagram and their intersection represents the sustained LPBs. $\mathbb{LB} \setminus \mathbb{LR}$ is the set of all lost LPBs during re-compression. To analyze the relation between the $\mathcal{N}$ and unmarked mode sustainability, those lost LPBs are eliminated from analysis. The relation between the $\mathcal{N}$ and unmarked mode sustainability can be found by *Unmarked Mode Sustaining Probability* (denoted as P_{UN}) which is the conditional probability of

unmarked mode sustaining for a given $\mathcal{N}$. If an LPB has same intra prediction mode before and after re-compression (i.e. $m_b = m_r$ for this LPB), then the mode is sustained for that LPB. The subset of sustained LPBs having $\mathcal{N} = n$ before re-compression, is $(\bigcup_{m_b \in \mathbb{M}} \mathbb{LB}^{m_b,n}) \cap \mathbb{LR}$. In Figure 3.4, it is represented by grey inner circle. A subset of these sustained LPBs which have same intra prediction mode m_b after re-compression, is $\mathbb{LB}^{m_b,n} \cap \mathbb{LR}^{m_b}$. So, the set of LPBs which sustain their respective intra prediction mode before and after re-compression for $\mathcal{N}{=}n$ is $\bigcup_{m_b \in \mathbb{M}} (\mathbb{LB}^{m_b,n} \cap \mathbb{LR}^{m_b})$. In Figure 3.4, it is represented by black inner circle. The *Retained Mode Sustaining Probability* for a given $\mathcal{N}{=}n$ is calculated as,

$$P_{UN} = P((m_r = m_b) \mid \mathcal{N} = n) = \frac{\left|\bigcup_{m_b \in \mathbb{M}} (\mathbb{LB}^{m_b,n} \cap \mathbb{LR}^{m_b})\right|}{\left|(\bigcup_{m_b \in \mathbb{M}} \mathbb{LB}^{m_b,n}) \cap \mathbb{LR}\right|} \tag{3.4}$$

A plot of P_{UN} has been shown in Figure 3.5 for different values of $\mathcal{N}$. The figure depicts that P_{UN} is very low for smaller $\mathcal{N}$ and mostly similar for $\mathcal{N}$ values greater than 3.

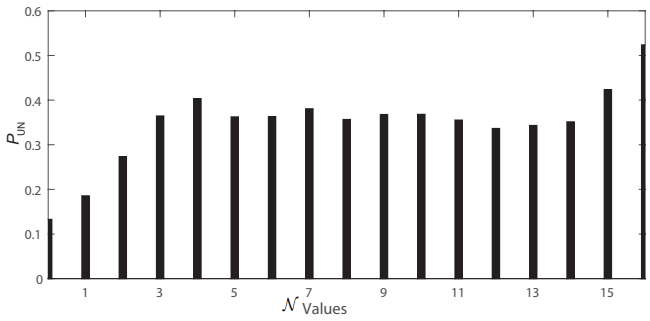


Figure 3.5: *P_{UN} for different $\mathcal{N}$ values. In this case QP is altered from 16 to 28 during re-compression*

$\mathbb{LB}^{m_b} \cap \mathbb{LR}^{m_r}$ is the set of LPBs which have modes m_b and m_r before and after re-compression respectively. $\mathbb{LB}^{m_b} \cap \mathbb{LR}$ represents the set of LPBs sustained after re-compression and had mode m_b before re-compression. The conditional mode transform probability $Pt(m_r \mid m_b)$ of a given intra mode m_b transform to

another intra mode m_r during re-compression is,

$$Pt(m_r \mid m_b) = \frac{|\mathbb{LB}^{m_b} \cap \mathbb{LR}^{m_r}|}{|\mathbb{LB}^{m_b} \cap \mathbb{LR}|} \tag{3.5}$$

Here, the lost blocks due to merging $\mathbb{LB} \setminus \mathbb{LR}$ are eliminated from analysis. From the experimentation, it is observed that during re-compression with QP 16 to 28, the sustaining probability of intra mode is around 0.4 and in most of the cases, the intra mode is changed to its neighboring modes. In Figure 3.6, $Pt(m_r \mid m_b)$ is plotted for selected intra modes (i.e. m_b is $0, 16, 27, 34$). From this figure, it can observe that the modes m_b are likely to be changed to the neighboring modes.

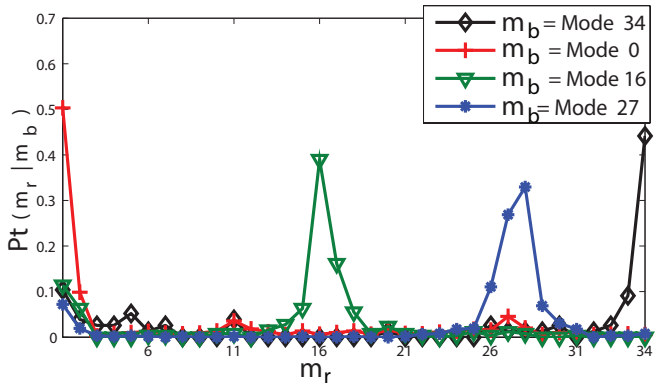


Figure 3.6: *$Pt(m_r \mid m_b)$ for re-compression with QP 16 from 28 when unchanged intra modes m_b is 0, 16, 27, 34.*

3.1.1.4 Relation Between $\mathcal{N}$ and Marked Mode Sustainability

In this subsection, the mode perturbation due to watermarking is considered. The mode sustainability under this perturbation is termed as *Marked Mode Sustainability*. The relation between $\mathcal{N}$ and marked mode sustainability is analyzed in terms of *Marked Mode Sustaining Probability* (denoted as P_{WN}) which is the conditional probability of altered mode sustaining for a given $\mathcal{N}$. To find marked mode sustaining probability, a subset of $\mathbb{LB}$s is chosen and their intra prediction modes are altered for embedding. After re-compression, the sustaining probability of the altered mode is measured for different values of $\mathcal{N}$. To avoid the drift error

during re-encoding, neighboring LPBs are not selected for this experimentations. Consider the following notations,

$\mathbb{B}$: The subset of $\mathbb{LB}$ selected for mode alteration.

$\mathbb{B}_k$: The k^{th} LPB of $\mathbb{B}$.

$\mathbb{B}^{m_b,n}$: The subset of $\mathbb{B}$ having mode m_b and $\mathcal{N} = n$.

$\mathbb{BR}$: The subset of $\mathbb{B}$ sustained after re-compression.

$\mathbb{BR}^{m_r}$: The subset of $\mathbb{BR}$ having intra prediction mode m_r.

m_a : The modes of LPBs belonging to $\mathbb{B}$ are altered to a mode m_a.

In Figure 3.7, $\mathbb{B}$ and $\mathbb{BR}$ are represented by a Venn diagram. To measure the

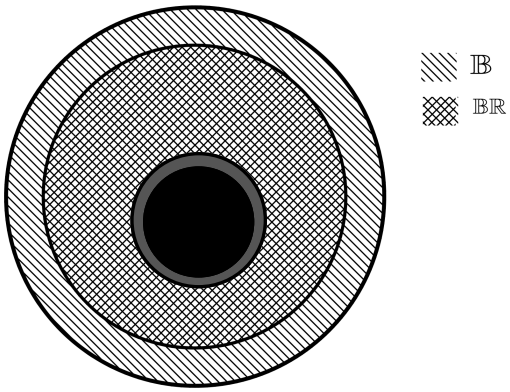


Figure 3.7: *The set of LPBs before and after re-compression in case of mode alteration. The grey inner region represents* $(\mathbb{BR} \cap \mathbb{B}^n)$ *and the black region represents* $(\mathbb{BR}^{m_a} \cap \mathbb{B}^n)$.

probability of marked mode sustainability, the intra modes of LPBs $\in \mathbb{B}$ are altered to an intra mode m_a. This altered mode is sustained for an LPB if after re-compression the mode remains m_a (i.e. $m_r = m_a$ for an LPB with sustained altered mode). Let the subset of such LPBs (sustained with altered mode) be $\mathbb{BR}^{m_a}$. The subset of $\mathbb{BR}^{m_a}$ having $\mathcal{N} = n$ before re-compression is $(\mathbb{BR}^{m_a} \cap \mathbb{B}^n)$. Again, suppose the subset of $\mathbb{B}$ having $\mathcal{N}=n$ and sustained after re-compression is $(\mathbb{BR} \cap \mathbb{B}^n)$. Then conditional probability of LPB sustaining with altered mode (denoted as P_{WN,m_a}) when intra modes are altered to m_a for selected LBPs with

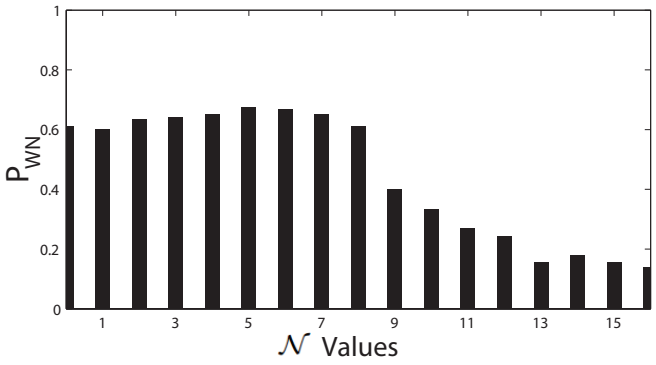


Figure 3.8: *P_{WN} for different $\mathcal{N}$ values. In this case QP is changed from 16 to 28 during re-compression.*

$\mathcal{N} = n$ is

$$P_{WN,m_a} = P((m_r = m_a) \mid \mathcal{N} = n, altered\ mode = m_a) = \frac{\left|(\mathbb{BR}^{m_a} \cap \mathbb{B}^n)\right|}{\left|(\mathbb{B}^n \cap \mathbb{BR})\right|} \tag{3.6}$$

The total probability is used to find the *marked mode sustaining probability* using P_{WN,m_a}. First, P_{WN,m_a} is calculated for all possible values of m_a. As m_a can be any of the intra prediction modes with equal probability, P_{WN} can be calculated as follows,

$$\begin{aligned} &P_{WN} = P((m_r = m_a) \mid \mathcal{N} = n) = \textstyle\sum_{m=0}^{34}(P_{WN,m} \times P(m_a = m)) \\ &= 1/35(\textstyle\sum_{m=0}^{34} P_{WN,m}) \end{aligned} \tag{3.7}$$

For different $\mathcal{N}$ values, P_{WN} is plotted in Figure 3.8 when QP is changed from 16 to 28 during re-compression. From the figure, it is observed that P_{WN} is almost constant for lower values of $\mathcal{N}$ and it has lower values at higher values of $\mathcal{N}$. This can be explained as follows. Consider an LPB $\mathbb{B}_k$ which is watermarked. Suppose,

$o_{i,j}$: The original pixel intensities of $\mathbb{B}_k$ at position i, j. *In this thesis, the pixel intensity and pixel are interchangeably used to represent pixel intensity.*

$o'_{i,j}$, $o''_{i,j}$: The pixels of $\mathbb{B}_k$ at position i, j after decompression and re-compression respectively.

$p_{m,i,j}$, $p'_{m,i,j}$, $p''_{m,i,j}$: The predicted pixels of $\mathbb{B}_k$ at position i, j, due to a mode m during compression, decompression and re-compression respectively.

$r_{m,i,j}$, $r'_{m,i,j}$, $r''_{m,i,j}$: The residuals of $\mathbb{B}_k$ at position i, j, due to a mode m during compression, decompression and re-compression respectively.
$R_{m,i,j}$, $R'_{m,i,j}$, $R''_{m,i,j}$: The transform coefficients of $\mathbb{B}_k$ at position i, j, due to a mode m during compression, decompression and re-compression respectively.
$\mathbb{B}_k$ has mode m_b before watermark embedding. The original pixels can be written as the sum of the prediction and the residual.

$$o_{i,j} = p_{m_b,i,j} + r_{m_b,i,j} \tag{3.8}$$

Suppose, the m_b is altered to a mode m_a due to watermark embedding. This alteration of mode does not change the $R_{m_b,i,j}$s. Hence, the corresponding spatial values of residuals $r_{m_b,i,j}$ remain unchanged. During decompression of the video, the prediction pixels $p'_{m_a,i,j}$s are generated by the altered mode m_a. Then, the changed pixels $o'_{i,j}$ of $\mathbb{B}_k$ are given by,

$$o'_{i,j} = p'_{m_a,i,j} + r_{m_b,i,j} \tag{3.9}$$

During the re-compression of video with different QPs, the mode m_r is chosen as the best mode for $\mathbb{B}_k$. Similarly, the residual and transform coefficients generated by mode m_r are $r''_{m_r,i,j}$ and $R''_{m_r,i,j}$ respectively. The residual and transform coefficients generated by mode m_a are $r''_{m_a,i,j}$ and $R''_{m_a,i,j}$ respectively. Now, m_r and m_a are different, if the residual error generated by mode m_a is greater than the residual error generated by mode m_r. So, the condition for changing m_a to m_r during re-compression can be written as,

$$\sum_{i=1}^{4}\sum_{j=1}^{4}(R''_{m_a,i,j})^2 > \sum_{i=1}^{4}\sum_{j=1}^{4}(R''_{m_r,i,j})^2 \tag{3.10}$$

Using the property of linear transformation, the above inequality can be written for the spatial domain as follows:

$$\begin{aligned} &\textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(r''_{m_a,i,j})^2 > \sum_{i=1}^{4}\sum_{j=1}^{4}(r''_{m_r,i,j})^2 \\ &\Rightarrow \textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(o'_{i,j} - p''_{m_a,i,j})^2 > \sum_{i=1}^{4}\sum_{j=1}^{4}(o'_{i,j} - p''_{m_r,i,j})^2 \end{aligned} \tag{3.11}$$

The set of reference pixels used by $\mathbb{B}_k$ remains approximately same during re-compression. Therefore, the predicted pixels which are generated by mode m_a

at the time of decoding and re-compression, are approximately same. Thus Eq. (3.11) can be rewritten as,

$$\begin{aligned}
&\textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(o'_{i,j} - p'_{m_a,i,j})^2 > \sum_{i=1}^{4}\sum_{j=1}^{4}(o'_{i,j} - p''_{m_r,i,j})^2 \\
&\Rightarrow \textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(p'_{m_a,i,j} + r_{m_b,i,j} - p'_{m_a,i,j})^2 > \\
&\textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(p'_{m_a,i,j} + r_{m_b,i,j} - p''_{m_r,i,j})^2 \ \{using\ Eq.\ (3.9)\} \\
&\Rightarrow \textstyle\sum_{i=1}^{4}\sum_{j=1}^{4}(r_{m_b,i,j})^2 > \sum_{i=1}^{4}\sum_{j=1}^{4}(p'_{m_a,i,j} - p''_{m_r,i,j} + r_{m_b,i,j})^2
\end{aligned} \tag{3.12}$$

The above condition will not be satisfied if most of the residuals $r_{m_b,i,j}$ have zero value or very small values. In other words, lower $\mathcal{N}$ implies the higher probability of altered mode sustainability i.e. higher P_{WN}. As $\mathcal{N}$ increases, the sum of residuals $\sum_{i=1}^{4}\sum_{j=1}^{4}(r_{m_b,i,j})^2$ increases. So the decompressed video pixels plane o' becomes closer to the prediction plane p''_{m_r} (generated by any other mode m_r) in comparison to prediction plane p''_{m_a} (generated by altered mode m_a). In Figure 3.8, experimental results validate the same hypothesis. For lower value of $\mathcal{N}$, the chances of sustaining the altered mode m_a are high. But, for very low values of $\mathcal{N}$ (e.g. $\mathcal{N}$ =0,1), P_{WN} decreases as LPBs having lower $\mathcal{N}$ values have a general tendency for merging into higher size PB during re-compression.

3.1.1.5 Range of $\mathcal{N}$ for selection of embeddable LPBs

The range of $\mathcal{N}$ should be chosen in such a way that different evaluation parameters of the watermarking scheme like robustness, visual quality, payload, bit increase rate are optimized. These parameters are often mutually conflicting. So the choice of $\mathcal{N}$ is an important task which has been analyzed in this subsection based on above analysis. The observations regarding the choice of $\mathcal{N}$ for different watermarking requirements are summarized below:

1. The choice of $\mathcal{N}$ for robustness depends on the following embedding parameters:

   (a) *LPB Sustainability*: As observed in Sec. 3.1.1.1, LPB sustainability increases with $\mathcal{N}$ values.

(b) *Unmarked Mode Sustainability*: The unmarked mode sustainability probability (P_{UN}) remains approximately constant for $\mathcal{N}$ greater than 3 [Please refer to Sec. 3.1.1.3].

(c) *Marked Mode Sustainability*: The marked mode sustaining probability (P_{WN}) becomes lower for higher $\mathcal{N}$ values [Please refer to Sec. 3.1.1.4].

So, selecting the middle range $\mathcal{N}$ values for embedding may increase the robustness of the watermark.

2. Since the LPBs with a higher number of $\mathcal{N}$ values are very less in number [Please refer to Sec. 3.1.1.2], the $\mathcal{N}$ values are chosen in such a way that a considerable number of blocks can be used for embedding. In other words, lower $\mathcal{N}$ values should be selected for increasing the payload.

3. To strengthen the security, a number of LPBs are randomly selected from the set of embeddable LPBs. Again, to get a large set of embeddable LPBs, chosen $\mathcal{N}$ values should not be high.

4. LPBs having very low $\mathcal{N}$ belong to the smooth regions of the video. Embedding in those LPBs causes visual artifact.

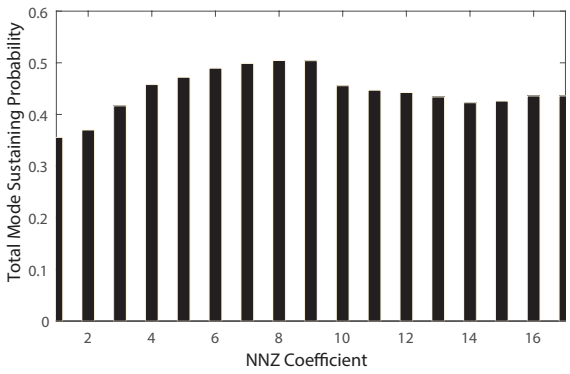


Figure 3.9: *P_{TN} for different $\mathcal{N}$ values.*

Based on these observations, a *total mode sustaining probability* (P_{TN}) for a given $\mathcal{N}$ can be written as,

$$P_{TN} = (P_{BN} + P_{FN} + P_{UN} + P_{WN})/4 \tag{3.13}$$

In Figure 3.9, P_{TN} has been depicted for different NNZ values. It shows that LPBs having $\mathcal{N}$ between 6 to 9 are having higher sustainable probability and can be chosen for watermark embedding.

3.1.2 Grouping of Modes

In the proposed scheme, the intra mode of the selected LPB is altered for embedding. From the Figure 3.9, it is observed that the LPB have mode sustaining probability less than 0.5. Thus the embedded mode (m_a) may be changed to some different mode (m_r) during re-compression. Thus the watermark may be destroyed during re-compression. To resolve this problem, H.265/HEVC modes are grouped in the proposed scheme, depending on the changing probability between each other due to re-compression. The relation between embedded mode (m_a) and post re-compression mode (m_r) can be derived from the intra prediction process as follows.

The interpolation method for prediction is explained in Eq. (1.1). The right shift operation in Eq. (1.1) is replaced by integer division operation and can be written as.

$$p_{i,j} = ((32 - w_j).\Re_{x,0} + w_j.\Re_{x+1,0} + 16)/32 \tag{3.14}$$

where $\Re$ is the set of reference pixels [Please refer Figure 1.9]. During decoding, $\Re$ remains same. Therefore, the predicted pixel $p'_{a,i,j}$ for mode m_a generated during decoding process can be written as,

$$p'_{a,i,j} = ((32 - w_{aj}).\Re_{x_a,0} + w_{aj}.\Re_{x_a+1,0} + 16)/32 \tag{3.15}$$

where x_a is the reference pixel position and w_{aj} is weighting parameter for mode m_a and are calculated similar to Eq. (1.2). Thus,

$$w_{aj} = (j.d_a)\&31,\ x_a = i + c_{aj},\ and\ c_{aj} = (j.d_a)/32$$

where d_a is the displacement parameter associated with m_a and c_{aj} is an intermediate value depending on co-ordinate j and d_a. During re-compression, the reference set of pixels $\Re$ remains the same. So, the prediction plane p'_a is regenerated by mode m_a at the time of re-compression. If m_a is changed to mode m_r at re-compression, then the predicted pixel $p''_{r,i,j}$ generated for m_r can be written as

$$p''_{r,i,j} = ((32 - w_{rj}).\Re_{x_r,0} + w_{rj}.\Re_{x_r+1,0} + 16)/32 \tag{3.16}$$

The displacement parameter d_r associated with m_r can be written as,

$$d_r = d_a + \Delta d \tag{3.17}$$

where Δd represents the difference between the displacement parameters associated with m_a and m_r. Then the weighting factors can be written as,

$$\begin{aligned} w_{rj} = (j.d_r)\&31 = (j.(d_a + \Delta d))\&31 = ((j.d_a)\&31 + (j.\Delta d)\&31)\&31 \\ \approx w_{aj} + (j.\Delta d)\&31 = w_{aj} + \Delta w_{aj} \end{aligned} \tag{3.18}$$

The position of reference pixels for mode m_r is now given as,

$$x_r = i + c_{rj}, \text{ where, } c_{rj} = (j.d_r)/32 = (j.(d_a + \Delta d))/32 \tag{3.19}$$

From Eq. (3.15) and Eq.(3.16), the difference between predicted pixels is given by,

$$\begin{aligned} & p''_{r,i,j} - p'_{a,i,j} = \{((32 - w_{rj}).\Re_{x_r,0} + w_{rj}.\Re_{x_r+1,0} + 16)/32\} - \\ & \{((32 - w_{aj}).\Re_{x_a,0} + w_{aj}.\Re_{x_a+1,0} + 16)/32\} \\ & = (32.(\Re_{x_r,0} - \Re_{x_a,0}) - (w_{aj} + \Delta w_{aj}).\Re_{x_r,0} + \\ & (w_{aj} + \Delta w_{aj}).\Re_{x_r+1,0} + w_{aj}.\Re_{x_a,0} - w_{aj}.\Re_{x_a+1,0})/32 \\ & \{using\ Eq.\ (3.18)\} \\ & = (32.(\Re_{x_r,0} - \Re_{x_a,0}) + w_{aj}.(\Re_{x_r+1,0} - \Re_{x_r,0}) + \\ & w_{aj}.(\Re_{x_a,0} - \Re_{x_a+1,0}) + \Delta w_{aj}.(\Re_{x_r+1,0} - \Re_{x_r,0}))/32 \\ & \approx (32.(\Re_{x_r,0} - \Re_{x_a,0}) + w_{aj}.(\Re_{x_r+1,0} - \Re_{x_r,0}) + \\ & w_{aj}.(\Re_{x_a,0} - \Re_{x_a+1,0}) + ((j.\Delta d)\&31).(\Re_{x_r+1,0} - \Re_{x_r,0}))/32 \end{aligned} \tag{3.20}$$

In natural images and videos, inter pixel correlation is in general very high within a small neighborhood. Thus, the difference between $\Re$ of an LPB is very small if

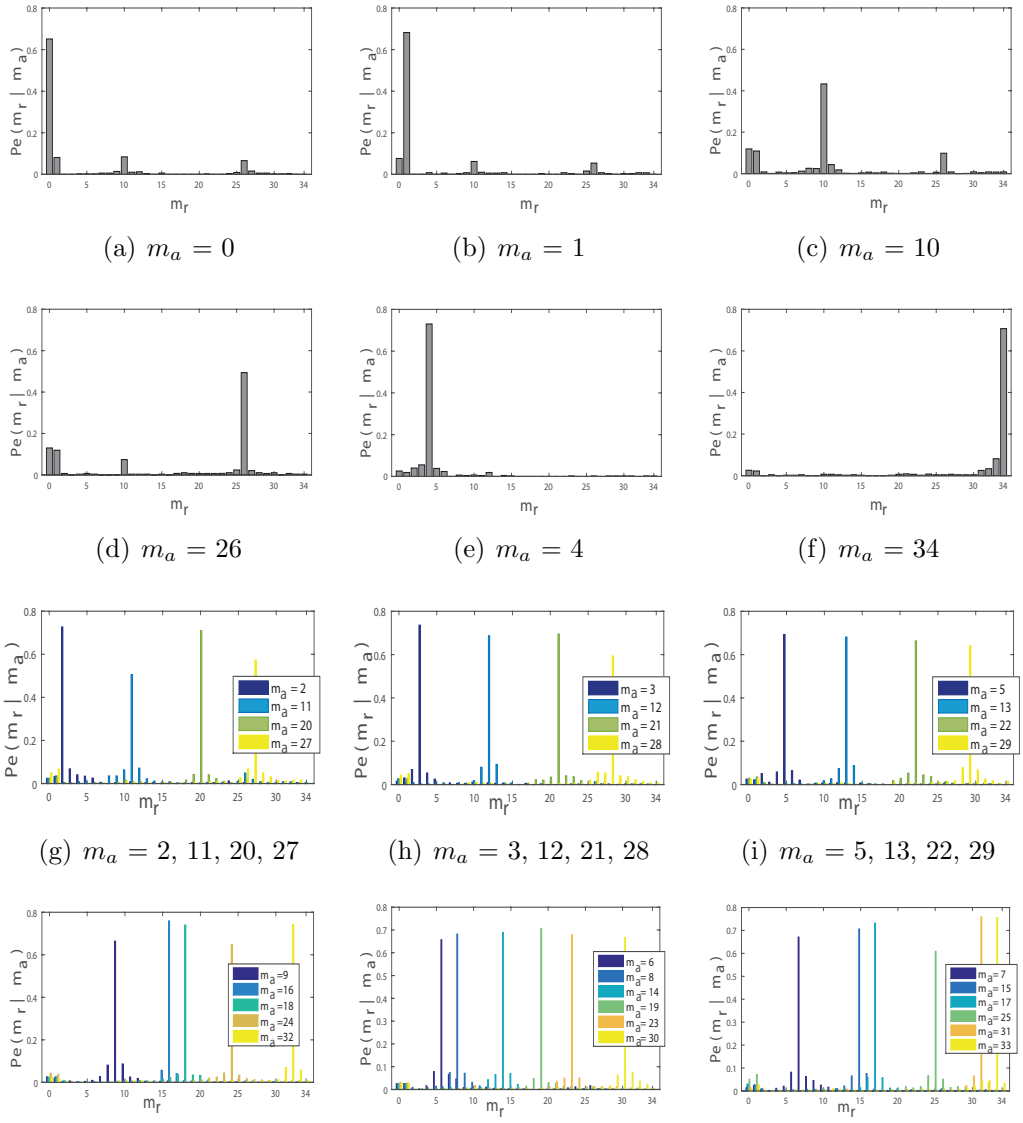
they are not at any edges. So the variation in $\mid p''_r - p'_a \mid$ is dominated by $\mid \Delta d \mid$ which increases with the increase of difference between the prediction modes. Thus, the value of $\mid p''_r - p'_a \mid$ is relatively smaller for the neighboring modes than the distant modes. For example, $\mid p''_r - p'_a \mid$ for mode 2 and 3 is lower than that of mode 2 and 7. During decompression, the predicted pixel value is added to the residual error to generate the decompressed pixel. Thus, it can be inferred that the mode changing probability for an LPB during re-compression depends on the residual error as well as $\mid p''_r - p'_a \mid$.

Thus, if the absolute sum of the residual error coefficients is low, then the post re-compression mode (m_r) should be close to the embedded mode (m_a) after re-compression. This can be shown using the conditional probability $Pe(m_r \mid m_a)$ of mode changing of given embedded mode m_a to a mode m_r during re-compression. $Pe(m_r \mid m_a)$ for the proposed scheme is written as,

$$Pe(m_r \mid m_a) = P(changed\ mode = m_r \mid embedded\ mode = m_a) = \frac{|\mathbb{B}\mathbb{R}^{m_r}|}{|\mathbb{B}|} \tag{3.21}$$

where, $\mathbb{B}$ represents the set of sustained embedded LPBs which were embedded with mode m_a and $\mathbb{B}\mathbb{R}^{m_r}$ is the subset of $\mathbb{B}$ with intra prediction mode m_r after re-compression. In Figure 3.10(e), $Pe(m_r \mid m_a)$ for $m_a = 4$ is plotted. It can be observed from the Figure 3.10(e) that changed modes due to re-compression (m_r) are always in very close proximity with the embedding mode (here m_a=4). Similarly, other $Pe(m_r \mid m_a)$ for different m_a have been shown in Figure 3.10. These results show that chances of embedded mode changing to neighboring intra modes during re-compression are higher for the proposed scheme.

In the proposed work, one of the main goals is to group the prediction modes in such a way that an embedded prediction mode (m_a) and prediction mode after re-compression (m_r) should belong to the same group. For such grouping of modes, the $Pe(m_r \mid m_a)$ s for different embedded modes are used in the proposed scheme. In Figure 3.10, $Pe(m_r \mid m_a)$s for selected values of m_as are plotted. From the figure two observations can be made.



(a) $m_a = 0$ (b) $m_a = 1$ (c) $m_a = 10$

(d) $m_a = 26$ (e) $m_a = 4$ (f) $m_a = 34$

(g) $m_a = 2, 11, 20, 27$ (h) $m_a = 3, 12, 21, 28$ (i) $m_a = 5, 13, 22, 29$

(j) $m_a = 9, 16, 18, 24, 32$ (k) $m_a = 6, 8, 14, 19, 23, 30$ (l) $m_a = 7, 15, 17, 25, 31, 33$

Figure 3.10: *$Pe(m_r \mid m_a)$ values for different embedded modes m_a*

1. The values of $Pe(m_r \mid m_a)$ for $m_a = 0$ are higher where m_r is $0, 1, 10$ and 26. Also, the values of $Pe(m_r \mid m_a)$ for $m_a = 1, 10, 26$ are higher where m_r is $0, 1, 10, 26$. This implies that the mode alterations between mode DC (mode 1), planar (mode 0) and directly horizontal (mode 10) and directly vertical (mode 26) are higher.

2. For the rest of the cases, the embedded modes m_a are likely to be changed to the modes which generate more similar texture. As the neighboring modes use the same set of reference pixels and have less difference in displacement parameter d, the texture generated by the neighbor modes are more similar. So, $Pe(m_r \mid m_a)$ for neighbor modes are higher.

Moreover in HEVC, the intra prediction process does not always include the whole set of reference pixels. Generally each octant of eight intra prediction modes is associated with a different set of reference pixels [19] (Please refer to Figure 1.9). In Table 3.1, the groups of prediction modes which utilize the same reference set of pixels are given. In this work, the grouping for sustainable modes has been performed based on the above observations. The modes are clustered into four different groups as given in Table 3.2. For embedding the watermark, the modes

Table 3.1: ***Reference Pixel Sets Use by Different Intra Modes***

Intra Prediction mode	Corresponding Set of Reference Pixels Use in Prediction
lower octant of horizontal modes (Intra mode 2,3,..,9)	$\Re_{0,0}$, $\Re_{0,1}$,.. ,$\Re_{0,2N}$
upper octant of horizontal modes (Intra mode 10,11,12,..,17)	$\Re_{0,0}$, $\Re_{0,1}$,.. , $\Re_{0,N}$ and $\Re_{1,0}$, $\Re_{2,0}$, ...,$\Re_{N,0}$,
left octant of vertical modes (Intra mode 18,19,20,..,26)	$\Re_{0,0}$, $\Re_{0,1}$,.. , $\Re_{0,N}$ and $\Re_{1,0}$, $\Re_{2,0}$, ...,$\Re_{N,0}$,
right octant of vertical modes (Intra mode 27,28,29,..,34)	$\Re_{0,0}$, $\Re_{1,0}$, ...,$\Re_{2N,0}$
planar mode (intra mode 1) and DC mode (intra mode 0)	all reference pixels $\Re$

of one group are modified to the modes of the other groups, according to the embedded watermark bits. For robust watermarking, the altered mode should sustain within the group after re-compression. In order to do so, one mode is chosen to represent each group. Let the mode m_p of group G_i, has least $\sum Pe(m_r \mid m_p)$ where $m_r \in G_j$ and $i \neq j$. Then the mode m_p is chosen to represent the group G_i. So, the *Group Sustaining Probability* (GSP) between two groups G_i and G_j, is measured by altering all modes of group G_i to representing mode m_p

Table 3.2: ***Sustainable Grouping of Intra Prediction Modes***

Group Name	Set of Intra Prediction Modes
G_0	Horizontal Mode(Mode 10), Vertical Mode(Mode 26) DC Mode(Mode 1), Planar Mode(Mode 0)
G_1	Lower Octant of Horizontal Modes (Mode 2, Mode 3, ... , Mode 9)
G_2	Upper Octant of Horizontal modes and Left Octant of Vertical Modes (i.e. Mode 11, Mode 12, ... , Mode 17 and Mode 18, .. ,Mode 25)
G_3	Right Octant of Vertical Modes (Mode 27, Mode 28, ... , Mode 34)

of group G_j. Let $\mathbb{B}^{m_b}$ represents the embedded LPB set, having mode m_b. Then $\cup_{m_b \in G_i} \mathbb{B}^{m_b}$ represents set of embeddable LPB of group G_i. The modes of these LPBs are altered to the representing mode of Group G_j. Note that $\cup_{m_r \in G_j} \mathbb{BR}^{m_r}$ represents subset of embedded LPBs which are sustained after re-compression with the mode belonging to the embedded group G_j. The GSP can now be obtained by the following equation:

$$Pg_{G_i,G_j} = \frac{|\cup_{m_r \in G_j} \mathbb{BR}^{m_r}|}{|\cup_{m_b \in G_i} \mathbb{B}^{m_b}|} \tag{3.22}$$

Figure 3.10 shows that some modes of each group have relatively higher $Pe(m_r \mid m_a)$ values where m_r belongs to the other group. For example, the embedded *mode* 8 and *mode* 9 have a greater likelihood for changing respectively to *mode* 10 and *mode* 11 which belongs to other groups. For improving the GSP, these modes are also altered to the representing mode of the same group. In case of G_1, to improve the GSP, *mode* 8 and *mode* 9 are altered to the representing mode of G_1. The GSPs for different groups are given in Table 3.3. For enhancing the security, a set of modes can be used for representing a group. This number of modes for representing a group, can vary depending on the robustness and the security of watermark. A key can be used to choose the representing mode from the set of alternatives.

Table 3.3: ***Group Sustaining Probability***(Pg_{G_i,G_j})

Original Group(G_i)	Altered Group(G_j)			
	G_0	G_1	G_2	G_3
G_0	0.9548	0.93	0.9019	0.915
G_1	0.867	0.945	0.898	0.904
G_2	0.874	0.916	0.932	0.916
G_3	0.892	0.90	0.89	0.947

3.1.3 Watermark Embedding and Extraction

3.1.3.1 Embedding Scheme:

In HEVC, a set $\mathbb{M}_3$ of three most probable prediction modes is used to encode the prediction mode information for an LPB. These three most probable modes are selected from the prediction modes of the neighboring LPBs (for example, immediate top or immediate left LPB). The modes of both neighboring PBs are included in the set. The third mode may be any of the *mode* 0, *mode* 1, *mode* 26. The H.265/HEVC encoder sets one bit flag, if the mode of an LPB belongs to $\mathbb{M}_3$. In that case, only two bits are required to encode the most probable mode. Otherwise, a five bit code is required for encoding the intra prediction mode. If this flag is set, then altering the intra prediction mode for watermarking will increase the bit rate as the altered mode may not belong to $\mathbb{M}_3$. So, the LPBs having modes in $\mathbb{M}_3$, are not chosen for watermarking.

In the proposed scheme, the number of intra mode groups for embedding the watermark is four. Therefore, the mode group $\mathbb{G}$=$\{G_0, G_1, G_2, G_3\}$ is mapped to binary bit-pair set $\mathbb{BP}$=$\{00, 01, 10, 11\}$. This one to one mapping function from $\mathbb{G}$ to $\mathbb{BP}$, is defined as $f_{\mathbb{G}\rightarrow\mathbb{BP}}$ and the corresponding inverse mapping is denoted as $f_{\mathbb{BP}\rightarrow\mathbb{G}}$. Let G_0, G_1, G_2 and G_3 are mapped to 00, 01, 10 and 11 respectively. A range $[\mathcal{N}_{low},.., \mathcal{N}_{high}]$ of $\mathcal{N}$ is considered for embeddable LPB selection based on observations in Sec. 3.1.1.5. Let an LPB (PB_k) has $\mathcal{N}$=N_k, the intra prediction mode m_k and the most probable mode set $\mathbb{M}_{3k}$. Using Table 3.2, the group G_k for

ALGORITHM 1: Watermark Embedding(V, WS)

Input: V:Video to be watermarked, WS= Watermark Bit Stream
Output: VW: Watermarked Video Sequence, WL:Watermark Location Map

```
begin
    for each intra frame F_i ∈ V where i represents frame index in V do
        for each LPB PB_k ∈ F_i where k represents LPB index in F_i do
            N_k = 𝒩 value of PB_k;
            if 𝒩_low ≤ N_k ≤ 𝒩_high then
                δ = the euclidean distance between previously watermarked
                LPB and PB_k;
                Find the intra prediction mode m_k of PB_k ;
                Find the most probable modes set 𝕄_3k for PB_k based on
                neighbor PBs modes;
                b is a pseudo-random binary bit for choosing LPB;
                if m_k ∉ 𝕄_3k and δ ≥ δ_th and b == 1 then
                    select group G_k from 𝔾, if m_k ∈ G_k;
                    /*using Table 3.2*/
                    Let the watermarking bit pair w_1w_2 ∈ WS;
                    map w_1w_2 to corresponding intra mode group G_w using
                    f_{𝔹ℙ→𝔾}(w_1w_2);
                    if G_w ≠ G_k then
                        Select representing mode m_p of G_w;
                        change m_k to m_p;
                    else
                        if m_k is edge mode of group G_k then
                            Change m_k to representing mode of Group G_k;
                    save the location of PB_k in location map WL;
```

the prediction mode m_k can be found from $\mathbb{G}$ ($m_k \in G_k$). In the proposed scheme, PB_k is considered for embedding if $N_k \in [\mathcal{N}_{low}, .., \mathcal{N}_{high}]$ and $m_k \notin \mathbb{M}_{3k}$. Let w_1w_2 be the watermark bit pair for embedding which is mapped to group G_w (where $G_w \in \mathbb{G}$) using $f_{\mathbb{BP}\to\mathbb{G}}$. Depending upon w_1w_2, m_k is altered to the representing mode of group G_w. The location of watermark embedded LPB (i.e. PB_k) is saved in the location map (WL) which is used at the time of watermark extraction.

If the selected LPBs are neighbors of each other, then altering one LPB's intra mode will affect the prediction process of selected neighbor LPBs. This may alter the embedded mode of a neighbor LPB. Thus, the extracted watermark will be erroneous. To overcome this, in the embedding process a minimum distance (δ_{th}) between the positions of two embeddable LPBs is maintained. The embedding process is described in Algorithm 1.

3.1.3.2 Extraction Scheme

The watermark extraction is performed after the entropy decoding. The process is quite simple. Using the location map (WL) which is generated at the time of embedding, the intra mode m_k of the watermarked LPB PB_k is found. Then, the group which contains m_k, is found using Table 3.2. After that, the mapping function $f_{\mathbb{G}\to\mathbb{BP}}$ is used to find corresponding watermark bit pair. The extraction process is given in Algorithm 2.

ALGORITHM 2: Watermark Extraction(VW, WL)

Input: VW : Watermarked Video, WL : Watermark Location Map
Output: WE= Extracted Watermark Bit Stream
begin
  Find watermarked Prediction Block PB_k using location map WL;
  m_k=intra prediction mode of PB_k;
  select group G_k from G, if $m_k \in G_k$;
  /*w_1w_2 are the extracted watermark bit*/
  $w_1w_2 = f_{\mathbb{G}\to\mathbb{B}}(m_k)$;
  /*using inverse mapping function find the extracted watermark bit*/
  add w_1w_2 at the trail of WE;

3.2 Experimental Results

The proposed algorithm is implemented using the H.265/HEVC reference software HM.10 [13]. To check the effectiveness of the scheme, the algorithm is tested on 15 standard video sequences. Details of test sequences along with experimental

setup are tabulated in Table 3.4. The test sequences have different resolutions, textures and motions. For example, *BQMall* and *People on street* video sequences have high texture regions and motion while *Jonny* and *Kristen & Sara* sequences have lower texture variations and very less motion. The $\mathcal{N}_{low}$ and $\mathcal{N}_{high}$ are chosen based on the analysis given in Sec. 3.1.1.5. In proposed scheme, a text file

Table 3.4: ***Details of Experimental Set Up***

Parameters for Watermarking	**Values Taken**
H.265/HEVC version	HM-10.0
GOP structure	I-P-P-P-P-P-P-P
Intra Period	8
Quantization Parameter(QP)	16
Video Sequence Used	1. BasketBall, 2. BQTerrace, 3. BQMall, 4. Cactus, 5. Chipmunks, 6. Craving, 7. FlowerVase, 8. HoneyBee, 9. Johnny, 10. Kristen & Sara, 11. Komono, 12. ParkScene, 13. People on street, 14. Tennis, 15. Yachtride
video resolutions :	416×240, 832×240, 1280×720, 1920×1080, 2560×1600
Frame Rate	Original Frame Rate of Videos
Bit Depth	8 bit/pixel
Lower NNZ Threshold ($\mathcal{N}_{low}$)	6
Upper NNZ Threshold ($\mathcal{N}_{high}$)	9

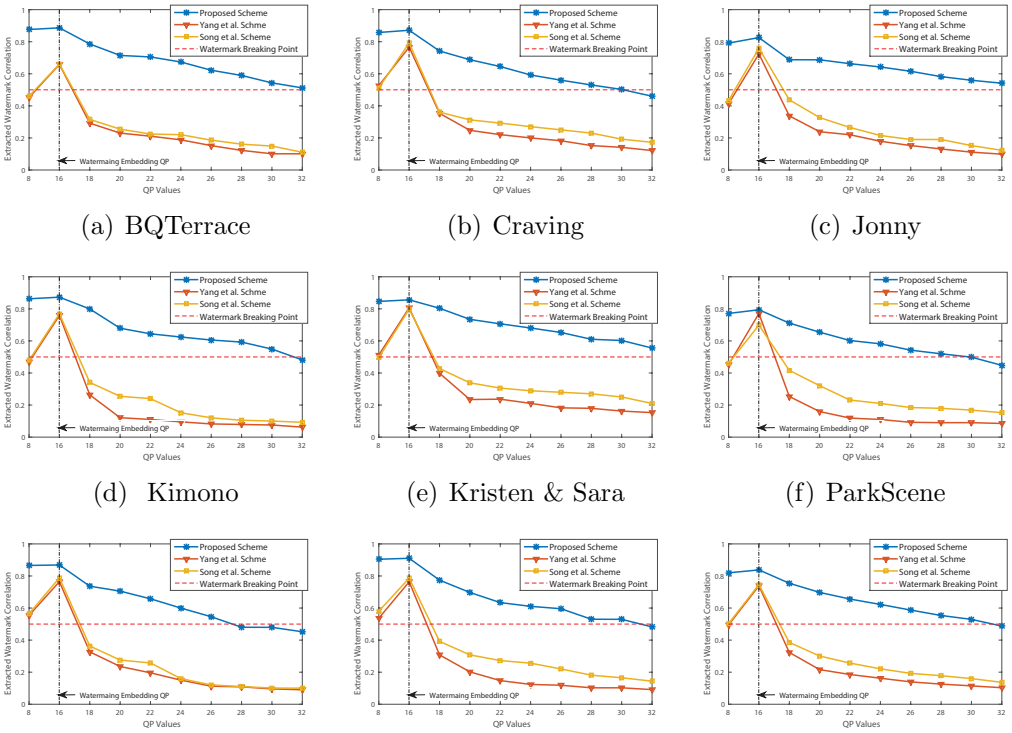
for location map which is around 300 bytes/GOP size is needed at the time of extraction. The size of location map is negligible science it is observed as 0.1% of the compressed bit stream. The performance of the watermarking scheme is evaluated in terms of three metrics: robustness, visual quality and bit increase rate.

3.2.1 Robustness

The main goal of this work is to resist re-compression attack by embedding a robust watermark using the intra prediction modes. The robustness of the proposed scheme is analyzed against the re-compression attack and common image processing attacks. In literature, several watermarking schemes are found which embed the watermark using the intra prediction modes. Recently, Song et al. [45] have proposed a low complexity intra prediction mode based watermarking scheme which is relatively more robust in comparison with existing schemes as it uses

8×8 blocks for embedding. Yang et al. [34] proposed another intra prediction mode based watermark embedding scheme which has the least degradation of the visual quality as it alters the minimum number of modes for embedding. Also, this scheme [34] alters mode to the next optimum mode to maintain the visual quality. As, these schemes are showing relatively better visual quality and having higher robustness, they are chosen for the comparison with the proposed scheme. In Figure 3.11, the robustness comparison among the proposed scheme and these existing schemes [34] [45] against re-compression attack is presented. The robustness is measured in terms of the correlation coefficient, which is an objective metric for similarity between the original watermark and the extracted watermark. The re-compression has been performed for QP=[8, 18, 20, 22, 24, 26, 28, 30, 32]. It can be observed from the Figure 3.11 that the proposed scheme has outperformed the existing ones [34] [45] in terms of robustness. From these figures, it can be observed that these existing schemes [34] [45] fail during re-compression with higher QPs. On the other hand, the proposed scheme shows sustainability up to re-compression with QP=32.

The performance evaluation of the proposed scheme has also been done against different types of image processing attacks such as the addition of Gaussian Noise and Salt-&-Pepper Noise and Gaussian filtering. The correlation between the original watermark and the extracted watermark from the attacked video sequence has been tabulated in Table 3.5. It can be observed that the proposed scheme is more resistant against these common image processing attacks than the existing schemes [34] [45]. Video transcoding is another popular attack that modifies video formats to access the content in the user end devices. As H.264/AVC is one of the most popular formats, the transcoding attack is performed using it. From the Table 3.6, it can be observed that the proposed scheme is quite robust against the transcoding attack.



(a) BQTerrace (b) Craving (c) Jonny

(d) Kimono (e) Kristen & Sara (f) ParkScene

(g) Peopleonstreet (h) Tennis (i) Averaged

Figure 3.11: *Comparison of robustness of the proposed approach with the existing schemes [34] [45] against re-compression attack in case of different video sequences*

Table 3.5: ***Watermark Correlation against Image Processing Attacks of Proposed Scheme with Existing Schemes [34] [45]***

Video Sequences	Attack Type								
	Different Schemes	salt & pepper Noise		Gaussian Noise		Gaussian Filter			
		Noise Strength		Mean = 0 Varience =0.0001	Mean = 0 Varience =0.0002	Filter Size =3 × 3 Sigma = 0.1	Filter Size =3 × 3 Sigma = 0.5	Filter Size =5 × 5 Sigma = 0.1	Filter Size =5 × 5 Sigma = 0.5
		0.01	0.02						
BQTerrace	Proposed	0.8467	0.8256	0.7867	0.7652	0.8898	0.7771	0.8896	0.7574
	Yang et al.	0.4742	0.3708	0.2973	0.2513	0.6052	0.2753	0.5852	0.2227
	Song et al.	0.4757	0.3892	0.3194	0.2603	0.6248	0.3147	0.5948	0.2817
Cactus	Proposed	0.7983	0.7835	0.7672	0.7218	0.8933	0.7064	0.8933	0.6977
	Yang et al.	0.4913	0.4022	0.3495	0.3245	0.7002	0.2504	0.6703	0.2413
	Song et al.	0.5287	0.4172	0.3657	0.3452	0.7197	0.2595	0.6807	0.2586
Carving	Proposed	0.7874	0.7298	0.7235	0.6851	0.846	0.7188	0.846	0.7162
	Yang et al.	0.5175	0.4079	0.2116	0.2386	0.7124	0.2493	0.6946	0.2631
	Song et al.	0.5275	0.4143	0.2152	0.2814	0.7261	0.2517	0.7034	0.2632
HoneyBee	Proposed	0.788	0.7556	0.7645	0.7388	0.8273	0.7731	0.8247	0.7529
	Yang et al.	0.4306	0.3425	0.2506	0.2035	0.5912	0.2742	0.5512	0.2578
	Song et al.	0.4393	0.3675	0.2594	0.2164	0.6079	0.2857	0.5979	0.267
Average	Proposed	0.8051	0.7636	0.7415	0.7125	0.8541	0.7338	0.8234	0.7311
	Yang et al.	0.4884	0.3985	0.2854	0.2675	0.6425	0.2523	0.6225	0.2375
	Song et al.	0.4928	0.4174	0.3029	0.2725	0.6796	0.2879	0.6342	0.2632

Table 3.6: ***Robustness of the Proposed Scheme against Transcoding Attack***

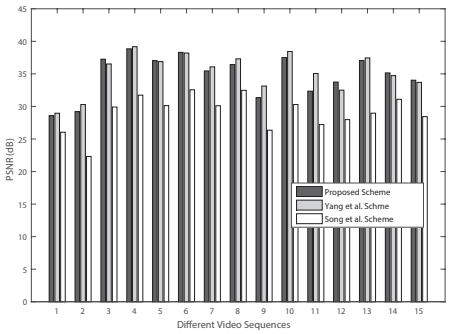
H.264/AVC Recompression QP	Exacted Watermark Correlation after Transcoding Attack										
	BasketBall	BQMall	BQTerrace	Cactus	Carving	Chipmunks	Flowervase	Jonny	Kristen & Sara	Kimono	Average
16	0.7583	0.7871	0.8321	0.8743	0.8036	0.7811	0.7616	0.8393	0.7981	0.7951	0.8127
20	0.7338	0.7568	0.7859	0.8229	0.7510	0.7508	0.7315	0.8063	0.7325	0.6828	0.7745
24	0.7128	0.7422	0.7537	0.7604	0.6751	0.7183	0.6801	0.7353	0.6954	0.6765	0.7357
28	0.6821	0.6503	0.6909	0.7424	0.6451	0.6384	0.6500	0.6881	0.6227	0.6420	0.6804

3.2.2 Visual Quality

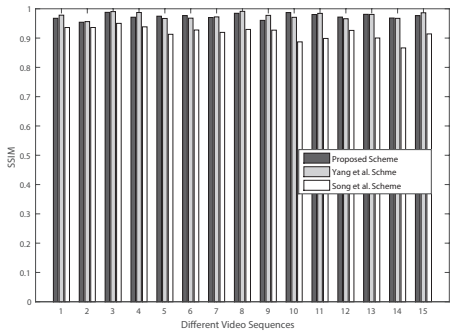
An original intra frame and the corresponding watermarked frame of the *BQMall* and the *Kimono* video sequences are shown in Figure 3.13 and 3.14 respectively. It can be observed that there exist hardly any visual traces of watermarking. The average Peak Signal to Noise Ratio (PSNR), average Structural Similarity (SSIM) index and Visual Information Fidelity (VIFp) [103] [104] [105] over all embedded frames are measured to evaluate the visual quality of the watermarked video. These metrics of the watermarked video are calculated with respect to original YUV video sequence using the VQM Tool [103]. The experiment is performed with fixed QP=16. 100 bits are embedded in each frame for all video sequences. The visual quality of the proposed approach and the existing schemes [34] [45] is shown in Figure 3.12. From these figures, it can be observed that the visual quality of proposed scheme is better than Song et al. scheme [45] as the number of pixels changes is higher in this scheme. In some cases, proposed approach has poorer visual quality than that of the Yang et al. scheme [34]. Because, the scheme [34] always embeds the watermark using two adjacent prediction modes. However, this may make the scheme more susceptible to re-compression attack.

3.2.3 Bit Increase Rate

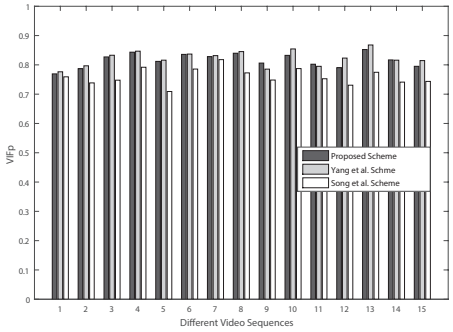
In the proposed scheme, the watermark is embedded in the LPB whose intra prediction mode $m \notin \mathbb{M}_3$ (as discussed in Sec.3.1.3). So, BIR due to the embedding is very less for the proposed scheme. In Figure 3.15 the BIR is compared with the existing schemes [34] [45]. The embedding is done for different video sequences with QP=16 and embedding rate is 100 bits/I Frame. From Figure 3.15, it can be observed that BIR is negligible and similar to the existing scheme.



(a) PSNR



(b) SSIM



(c) VIFp

Figure 3.12: *Comparison of visual quality of the proposed approach and the existing schemes [34] [45] in case of different video sequences. The video sequence index is same as given in Table 3.4*

3.3 Time Complexity Analysis

The proposed scheme embeds the watermark by modifying the intra prediction modes of 4×4 PBs. The selection of LPBs depends only on the compressed domain parameters. Thus, full decoding and re-compression of the bit stream are not required which results in low complexity embedding process in case of high resolution H.265/HEVC video streams. The proposed scheme can be implemented for real-time watermarking. Let the total number of LPBs be n_{LPB} for an intra frame and the total number of intra frame in a sequence be f_I. In the embedding process, all LPBs in a frame are processed one time for selection

(a) Original Frame (b) Watermarked Frame

Figure 3.13: *Cover frame of BQMALL video sequence*

(a) Original Frame (b) Watermarked Frame

Figure 3.14: *Cover frame of Kimono video sequence*

as an embeddable LPB. After selecting an LPB, embedding is done in constant time. Therefore, using Algorithm 1, the time complexity for embedding can be derived as $\mathcal{O}(f_I \times n_{LPB})$.

3.4 Summary

In this chapter, a real time, low complexity and robust watermarking scheme is proposed to authenticate the H.265/HEVC compressed HD and Beyond-HD videos. As the watermark is embedded using only the compressed domain parameters, full encoding and decoding of video are not required. This makes the scheme suitable for real time applications. To make the watermark robust against re-compression, the sustainability of the PBs and intra prediction modes have been analyzed with respect to another compressed domain feature i.e. nonzero transformed coefficients. Then, the intra prediction modes of H.265/HEVC stream

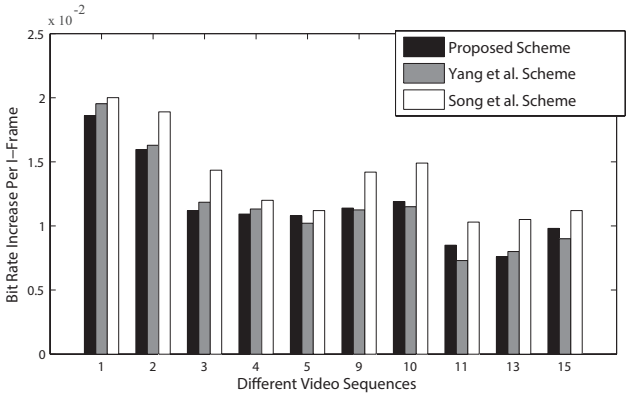


Figure 3.15: *Comparison of BIR for the proposed approach with the existing schemes [34] [45]. The video sequence index is same as depicted in Table 3.4*

are grouped in a fashion such that each group represents two watermark bits. These groups also ensure that the mode change due to re-compression can be closed within a group. Finally, the robustness and visual quality are compared to recent existing schemes to justify the applicability of the proposed scheme. This scheme can be further improved by selecting the PBs optimizing the visual quality and the robustness of the watermark. In the next chapter, the robustness of the watermark has been improved by considering the temporal homogeneity between PBs.

3. PREDICTION MODE BASED H.265/HEVC VIDEO WATERMARKING RESISTING RE-COMPRESSION ATTACK

Chapter 4

Robust Watermarking of H.265/HEVC Videos using Temporally Homogeneous Blocks

As discussed in Sec.1.2.1.2, H.265/HEVC [2] [15] has improved the coding structure to achieve higher compression efficiency. The use of different quad tree structures in the prediction and the residual transform, the increase in the number of intra prediction modes and the use of both DST and the DCT in prediction error transform have reduced spatial redundancy in case of intra prediction. Due to this improved prediction accuracy, the blocks have fewer nonzero transform coefficients. Thus, the main issue of watermark embedding by altering these sparse transform coefficients is that it leads to degradation in visual quality for watermarked video. In this chapter, a watermarking scheme for H.265/HEVC videos is proposed by altering transform coefficients so that the visual quality can be maintained by exploiting the temporal homogeneity of texture. Further, the robustness of the watermark is improved in comparison with mode based watermarking proposed in the previous chapter as the difference of the number of nonzero transform coefficients between temporal homogeneous blocks is more robust against re-compression.

Similar to the mode based scheme proposed in the previous chapter, this

watermarking scheme is also designed considering only compressed domain parameters as this compressed domain embedding leads to lower time complexity than the uncompressed domain embedding [43] and the most of the videos are stored and transmitted in a compressed format. As discussed in Chapter 1, there exist watermarking schemes which have altered the transformed coefficients of compressed video to embed the watermark. But very few of these schemes are proposed for H.265/HEVC videos. The direct extension of watermarking of H.264/AVC videos to the H.265/HEVC videos is not always gives optimal results as H.265/HEVC have different encoding structure. So the existing watermarking schemes and their robustness against re-compression attack have been analyzed in this chapter. Ma et al. [54] proposed an effective watermarking scheme by altering a pair of quantized DCT coefficients of the 4×4 luma blocks for H.264/AVC video sequences. The H.265/HEVC encoder utilizes the DST instead of the DCT for transforming the prediction errors of 4×4 intra predicted blocks. So, the transform domain data hiding algorithms which are developed based on 4×4 DCT blocks of H.264/AVC, cannot be applied to H.265/HEVC. Recently, Chang et al. [57] have extended Ma's work for H.265/HEVC videos where the watermark is embedded in selected 4×4 blocks of I-frames of H.265/HEVC videos by altering triplet of quantized DST coefficients. The problem with this existing H.265/HEVC watermarking scheme based on altering transform coefficient is that it is fragile against the re-compression attack. As, H.265/HEVC has different codec structures such as more number of prediction modes, different quad tree structure for prediction and transformation etc., the direct extension of H.264/AVC watermarking schemes to H.265/HEVC videos may not give suitable results. So, the goal of this work is to propose a robust watermarking scheme for H.265/HEVC videos by altering the sparse transform coefficients. Further, the coefficients should be altered such that the visual quality of watermarked video is maintained.

4.1 Proposed Scheme

Video contents are highly redundant in nature. Most of the frames within a short temporal neighborhood are perceptually similar with small differences. So, the blocks with a similar texture can be detected from the consecutive intra frames having a small temporal distances if no scene change has occurred. In this scheme, first such visually coherent TBs are found out from H.265/HEVC compressed video using compressed domain parameters only. Then the watermark is embedded altering the NNZ difference of those TBs. As these blocks within short temporal neighborhood have a similar texture, they have similar NNZ values. Due to a very less NNZ difference in the selected pair of TBs, perturbing of transform coefficients due to watermark embedding is very minimal. This leads to very less degradation in the visual quality. In next subsections, the detection of those temporally homogeneous blocks and the watermark embedding process are discussed.

4.1.1 Detection of Temporally Homogeneous TBs

To find temporally homogeneous TBs in consecutive intra frames, first the motion coherent blocks need to be identified. In [58], authors have embedded a robust non-blind watermark in motion coherent objects of H.264/AVC video sequences. In [58], first pseudo motion is calculated for all blocks of intra frame as intra blocks do not have motion vector. These blocks are clustered using motion and residual error into motion similar regions which are used to detect objects in compressed video sequences. Then, the watermark is embedded in 4×4 blocks of motion coherent objects by altering quantized DCT coefficients. Though the scheme is robust for H.264/AVC, the watermark extraction is not blind. In this section, first these motion coherent blocks detection is extended for H.265/HEVC videos to find the temporal homogeneous TBs. Then, a robust blind watermarking scheme is proposed using these TBs for H.265/HEVC sequences.

In the case of H.264/AVC, the prediction process and transformation of resid-

uals are done using a single quad-tree structure. On the other hand, H.265/HEVC uses the Coding Block (CB) which is divided into two different quad-tree structures for the prediction of a block and the transformation of residuals. Further, the asymmetric block partitioning is introduced in the case of inter frames compression. So, a PB can span over several TBs or a TB can be span over several PBs. Thus, a direct extension of the scheme proposed in [58] is not possible. As in the embedding scheme only 4×4 luma TBs are used for embedding (Sec. 4.1.2), the temporally homogeneous luma blocks of size 4×4 are only detected as follows:

Step 1: Divide the P-frames into non-overlapping 4×4 blocks. Assign motion vector to each of these blocks depending upon the PB to which it belongs. If the frame is predicted using multiple reference frames, normalize these assigned motion vectors to ensure that it points directly to the location in the immediately previous frame. If any of these blocks belongs within intra-coded PBs in P frames, estimate motion vector for these blocks using neighboring blocks.

Step 2: Assign a pseudo NNZ value for each of these blocks depending upon the TB to which it belongs.

Step 3: Assign the pseudo motion vector to each 4×4 TBs in I-frames by interpolating motion vectors at the same location in the nearest P-frames.

Step 4: Using these motion information and NNZ values, motion coherent 4×4 luma TBs are obtained in I frames.

Step 5: If selected motion coherent TBs have less difference in transformed coefficients, then select them as temporally homogeneous TBs.

4.1.2 Watermark Embedding

In the proposed scheme, the watermark is embedded 4×4 luma TBs of intra predicted frame having higher NNZ values. Intuitively, TBs having higher NNZ values belong to busy areas of a video frame. These blocks are suitable for embedding, as the high frequency components of these areas essentially mask the embedding noise and thus produces relatively less visual artifacts with respect to

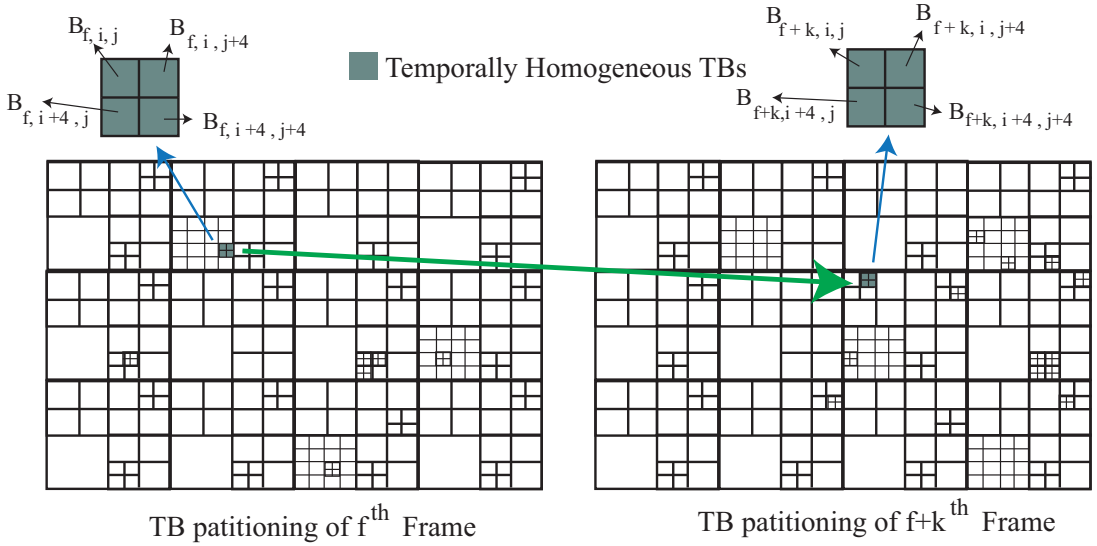


Figure 4.1: *Group of eight* 4×4 *TBs in* f^{th} *and* $(f+k)^{th}$ *frames are used to embed a bit of watermark.*

the human visual systems (HVS).

As discussed in Section 1.2.1, a CB is recursively divided into TBs for transformation of residual error to optimize the bit rate and the visual quality. Thus, an 8×8 TB is subdivided into four non-overlapping 4×4 TBs in higher textured areas. In the proposed scheme, a group of four 4×4 TBs of an intra frame is selected such that they are sub-blocks of the same 8×8 TB and they have temporally homogeneous 4×4 TBs in next intra frame. In Figure4.1, a group of such four TBs be $Q_{f,i,j} = \{B_{f,i,j}, B_{f,i+4,j}, B_{f,i,j+4}, B_{f,i+4,j+4}\}$ is shown where $B_{f,i,j}$ denotes a TB at position i, j in the f^{th} frame. The temporally homogeneous TBs of those TBs in next intra frame are $Q_{f+k,i',j'} = \{B_{f+k,i',j'}, B_{f+k,i'+4,j'}, B_{f+k,i',j'+4}, B_{f+k,i'+4,j'+4}\}$ where k is the length of the GOP and $B_{f+k,i',j'}$ denotes a TB at position i', j' in the $(f+k)^{th}$ frame. If these groups of TBs have NNZ values greater than some threshold $\mathcal{N}_{th}$ and the pseudo motion assigned to them is less than some threshold PMV_{th}, then these eight TBs are selected for embedding the watermark. As, these TBs of $Q_{f,i,j}$ and $Q_{f+k,i',j'}$ have similar NNZ values, very few alterations of zero transform coefficients are required for embedding the

watermark. The TBs with lower pseudo motion values are selected to embed the watermark to reduce the flickering artifacts in the watermarked video. After embedding, the location of these watermarked TBs are saved int a location map WL which is used at the time of extraction. As discussed in Section 3.1.1.1, if the quantization parameter (QP) has been increased during re-compression, the NNZ values for a block decreases. In this scenario, the subdivisions of a TB may result in a higher Lagrangian cost. Thus, in the case of re-compression with a higher QP value, smaller 4×4 TBs tend to be clustered with neighbor TBs to make a larger 8×8 TB. This leads to loss of embedded TBs. This cause the de-synchronization of watermark. In this scheme, as a group of 4×4 TBs are used to embed the watermark and the watermark can be extracted by comparing NNZ values of the merged 8×8 TBs from consecutive intra frames. So, the use of a grouped 4×4 TBs increases the robustness of the proposed scheme in case of re-compression attack. The step by step procedure is shown in Algorithm 3.

4.1.3 Watermark Extraction

The watermark extraction is performed on the compressed bit stream after entropy decoding. The process is quite simple. Using the location map WL locate the groups of eight watermarked TBs (i.e. $Q_{f,i,j}$ and $Q_{f+k,i',j'}$). The watermark bit is extracted from each group of eight TBs. If the group of four TBs ($Q_{f,i,j}$) is merged together during re-compression and form a 8×8 TB, then also watermark extraction is possible by comparing the NNZ value of the merged 8×8 TB. The step by step procedure is shown in Algorithm 4.

4.2 Experiment Results

The proposed algorithm is implemented using the H.265/HEVC reference software HM.10 [13]. To check the effectiveness of the scheme, the algorithm is tested on 15 standard video sequences. Details of test sequences along with experimental setup are tabulated in Table 3.4. The watermark is embedded at

ALGORITHM 3: Watermark Embedding(V, WS)

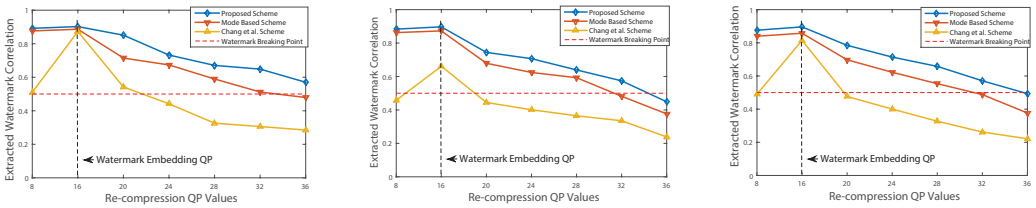
Input: V :Video to be watermarked, WS: Watermark Sequence
Output: VW : watermarked Video Sequence, WL : Location Map
begin
  Calculate pseudo motion vector and find out temporally homogeneous 4×4 TBs in intra frames of input video V. (Section 4.1.1);
  for *each unique pair of consecutive intra frame* $\{I_f, I_{f+k}\} \in V$ *where* f *denotes frame number and* k *is size of GOP* **do**
    for *each group of four TBs* $Q_{f,i,j}$ *in frame* I_f **do**
      Select corresponding group of four temporally homogeneous TBs $Q_{f+k,i',j'}$ in frame I_{f+k};
      Calculate the average pseudo motion vector (PMV_{mean}) of these selected TBs in $Q_{f,i,j}$;
      $N_{f,i,j}$ = The sum of NNZ of TBs in $Q_{f,i,j}$;
      $N_{f+k,i',j'}$ = The sum of NNZ of TBs in $Q_{f+k,i',j'}$;
      if $PMV_{mean} \leq PMV_{th}$ *and* $N_{f,i,j} \geq \mathcal{N}_{th}$ **then**
        Select watermarking bit w_b from WS;
        if $N_{f,i,j} \geq N_{f+k,i',j'}$ *and* $w_b = 0$ **then**
          $\delta = \mid N_{f,i,j} - N_{f+k,i',j'} \mid$;
          Increase NNZ values in of TBs in $Q_{f+k,i',j'}$ by altering $(\delta + 2)$ number of zero valued transformed coefficients;
        else if $N_{f,i,j} < N_{f+k,i',j'}$ *and* $w_b = 1$ **then**
          $\delta = \mid N_{f,i,j} - N_{f+k,i',j'} \mid$;
          Increase NNZ values in of TBs in $Q_{f,i,j}$ by altering $(\delta + 2)$ number of zero valued transformed coefficients;
        Save the watermark embedding location f, i, j and $f + k, i', j'$ in WL;

QP=16. The test sequences have different resolutions, textures and motions. For example, *Yachtride* and *People on street* video sequences have high resolution, *Tennis*, *BasketBall* video sequences have high motion, *Jonny* and *Kristen & Sara* sequences have lower texture variations and very less motion. In the proposed scheme, a text file for the location map which is of size around 300 bytes/GOP is needed at the time of extraction. The size of location map is negligible science it is observed as less than 0.1% of the compressed bit stream. The performance

ALGORITHM 4: Watermark Extraction(VW, WL)

Input: VW : watermarked Video Sequence, WL : Location Map
Output: WE : Extracted watermark sequence
begin
  for *each unique pair of consecutive intra frame* $\{I_f, I_{f+k}\} \in VW$ *where* f *denotes frame number and* k *is size of GOP* **do**
    Using location map WL find out the group of four 4×4 TBs $Q_{f,i,j}$ in frame I_f and corresponding group of four temporally homogeneous 4×4 TBs $Q_{f+k,i',j'}$ in frame I_{f+k};
    if *four* 4×4 *TBs* $Q_{f,i,j}$ *merged into a* 8×8 *TB* **then**
      Select the co-located 8×8 TB of frame I_f in $Q_{f,i,j}$;
      Select the co-located 8×8 TB of frame I_f in $Q_{f+k,i',j'}$;
    $N_{f,i,j}$ = The sum of NNZ of TBs in $Q_{f,i,j}$;
    $N_{f+k,i',j'}$ = The sum of NNZ of TBs in $Q_{f+k,i',j'}$;
    if $N_{f,i,j} \geq N_{f+k,i',j'}$ **then**
      Extract watermark bit w_e=1;
    else
      Extract watermark bit w_e=0;
    Save the extracted watermark bit w_e in watermarked sequence WE;

of the watermarking scheme is evaluated in terms of three metrics: robustness, visual quality and bit increase rate. For the existing scheme [57] also the same watermarking parameters are used to embed the watermark and the watermark is embedded at QP=16.



(a) BQMall (b) Kimono (c) Average

Figure 4.2: *Comparison of the robustness of the proposed scheme with mode based watermarking scheme proposed in Chapter 3 and Chang et al. scheme [57] against re-compression attack*

Table 4.1: ***Details of the Experimental Set Up***

Parameters for Watermarking	**Values Taken**
H.265/HEVC version	HM-10.0
GOP structure	I-P-P-P-P-P-P-P-P
Intra Period	8
Quantization Parameter(QP)	16
Video Sequence Used	1. BasketBall, 2. BQTerrace, 3. BQMall, 4. Cactus, 5. Chipmunks, 6. Craving, 7. FlowerVase, 8. HoneyBee, 9. Johnny, 10. Kristen & Sara, 11. Komono, 12. ParkScene, 13. People on street, 14. Tennis, 15. Yachtride
video resolutions :	416×240, 832×240, 1280×720, 1920×1080, 2560×1600
Frame Rate	Original Frame Rate of Videos
Bit Depth	8 bit/pixel

Table 4.2: ***Comparison of Robustness of the Proposed Scheme (Denoted by A) with Mode Based Watermarking Scheme Proposed in Chapter 3 (Denoted by B) and Chang et al. [57] Scheme (Denoted by C) against Re-compression Attack***

Video Sequences	Video Resolution	Embedding Capacity bits per frame	Scheme Name	Extracted Watermark correlation After Re-compression attacks with different QPs						
				QP 8	QP 16	QP 20	QP 24	QP 28	QP 32	QP 36
BQTerrace	832× 480	100	A	0.8918	0.9023	0.8509	0.7325	0.6707	0.6486	0.5701
			B	0.8762	0.8865	0.7145	0.6735	0.5896	0.5121	0.4791
			C	0.5104	0.8730	0.5420	0.4416	0.3260	0.3060	0.2850
Craving	832× 480	100	A	0.8854	0.9229	0.8415	0.7415	0.6789	0.5919	0.4864
			B	0.8576	0.8715	0.6878	0.5926	0.5303	0.4602	0.4172
			C	0.4500	0.6169	0.4831	0.4382	0.3708	0.2809	0.2360
Johnny	1280× 720	100	A	0.8600	0.8823	0.7743	0.7638	0.7013	0.6041	0.3798
			B	0.7916	0.8258	0.6872	0.6426	0.5823	0.5412	0.3512
			C	0.4474	0.8967	0.4088	0.3558	0.2802	0.2407	0.1835
Kimono	1920× 1080	100	A	0.8835	0.8972	0.7443	0.7067	0.6393	0.5741	0.4488
			B	0.8628	0.8728	0.6791	0.6243	0.5928	0.4816	0.3764
			C	0.4578	0.6613	0.4446	0.4012	0.3650	0.3353	0.2384
Kristen&Sara	1280× 720	100	A	0.8650	0.8652	0.8017	0.6982	0.6682	0.5535	0.4177
			B	0.8463	0.8563	0.7345	0.6806	0.6102	0.5556	0.3174
			C	0.4139	0.8935	0.4019	0.3292	0.2729	0.2178	0.1974
ParkScene	1920× 1080	100	A	0.8138	0.8857	0.7368	0.6462	0.5690	0.4712	0.4015
			B	0.7714	0.7943	0.6544	0.5819	0.5196	0.4478	0.3842
			C	0.5663	0.8128	0.5270	0.3622	0.2409	0.1567	0.1373
People on street	2560× 1600	100	A	0.8838	0.9190	0.7843	0.7194	0.6593	0.5122	0.4611
			B	0.8656	0.8686	0.7055	0.5999	0.4802	0.4522	0.3463
			C	0.4592	0.8930	0.4630	0.4080	0.3650	0.2640	0.2312
Tennis	1920× 1080	100	A	0.9165	0.9221	0.7448	0.7045	0.6756	0.6126	0.5406
			B	0.9043	0.9103	0.6972	0.6106	0.5302	0.4822	0.3531
			C	0.6166	0.8715	0.5385	0.4674	0.3962	0.2946	0.2621
Average		100	A	0.8760	0.8959	0.7848	0.7141	0.6578	0.5710	0.4932
			B	0.8396	0.8578	0.6975	0.6224	0.5536	0.4876	0.3768
			C	0.4902	0.8148	0.4761	0.4004	0.3271	0.2620	0.2214

4.2.1 Robustness

The main objective of this work is to embed a robust watermark in H.265/HEVC videos with minimal visual quality degradation. So, the robustness of the proposed scheme is analyzed in this subsection against the re-compression attack and common image processing attacks. In the proposed scheme, the watermark is embedded by varying the transformed coefficients of 4×4 TBs in intra frames. Similar to the proposed scheme, Chang et. al. [57] embed the watermark by altering the transformed coefficients of 4×4 TBs in intra frames of H.265/HEVC. So the robustness comparison in terms of extracted watermark correlation with original watermark, among the proposed scheme and Chang et al. scheme [57] against re-compression attack is presented in Table 4.2. The re-compression attack is performed in two steps. First, the compressed watermarked video is decompressed using H.265/HEVC decoder. Then, the decompressed raw video is compressed with different QPs. The re-compression has been performed for QP=[8, 16, 20, 24, 28, 32, 36]. Similar results for the *BQMall*, *Kimono* and *Average* of all video sequences are plotted in Figure 4.2 for more clarity. It can be observed from the Table 4.2 and Figure 4.2 that the proposed scheme is more robust against re-compression attack than the existing scheme [57]. The robustness is also compared with the prediction mode based watermarking scheme proposed in Chapter 3. It can be observed that the proposed scheme preforms better specially at higher QP values compared to the prediction mode based watermarking scheme. During the re-compression attack, the watermark is de-synchronized mainly because of the change in the quantized transformed coefficients of the TBs and the loss of embedded TBs due to merging of TBs into a larger size. As the NNZ differences are less affected by re-compression, the proposed scheme performs better than the Chang et. al. scheme [57]. Further, the proposed scheme can also extract the watermark by comparing merged 8×8 TBs of consecutive intra frames. Thus the performance of the proposed scheme is higher compared to prediction mode based watermarking scheme.

The robustness of proposed scheme is also evaluated against different image processing attacks such as the addition of salt and pepper noise, the addition of Gaussian Noise, Gaussian filtering in Table 4.3. It can be observed that the proposed scheme is resistant against these common image processing attacks and perform better than existing scheme. Further, the robustness of the proposed scheme is also tested against video transcoding attack where video formats are modified to access the content in user end devices. As H.264/AVC is one of the most popular formats, the transcoding attack is performed using it. First the watermarked video is transcoded using H.264/AVC for different QP values and GOP structure of I-P-P-P. Then, the H.264/AVC video is re-compressed with H.265/HEVC at QP 16 and the watermark is extracted from the re-compressed video. In Table 4.3, the result depicts that the proposed scheme is robust against the transcoding attack using H.264/AVC.

Table 4.3: ***Watermark Correlation against Different Attacks for the Proposed Scheme (Denoted by A) and Chang et. al.'s Scheme [57] (Denoted by B)***

Attack Type	Parameters		BQTerrace		Cactus		Johnny		People	
			A	B	A	B	A	B	A	B
	Noise strength									
Salt & Pepper Noise	0.01		0.8430	0.5445	0.8085	0.5570	0.8471	0.6316	0.8750	0.8824
	0.02		0.7961	0.4484	0.7627	0.4740	0.8027	0.5053	0.8256	0.7673
	Mean	Varience								
Gaussian Noise	0	0.0001	0.8478	0.5377	0.7975	0.5106	0.8251	0.5714	0.8594	0.4943
	0	0.0002	0.8174	0.4621	0.7755	0.4876	0.7952	0.5263	0.8201	0.4607
	Filter Size	Sigma								
	3×3	0.3	0.8384	0.5665	0.7945	0.5990	0.8394	0.6863	0.8653	0.6273
Gaussian Filter	3×3	0.5	0.7949	0.4022	0.7419	0.4202	0.7886	0.4211	0.8288	0.3848
	5×5	0.3	0.8244	0.5491	0.7716	0.5910	0.8364	0.6584	0.8353	0.6018
	5×5	0.5	0.7779	0.3764	0.7377	0.3815	0.7788	0.4026	0.7850	0.3680
	QP									
Tanscoding Attack Using H.264/AVC	16		0.8681	0.5046	0.8305	0.5677	0.8613	0.5429	0.8479	0.4368
	20		0.8179	0.3712	0.7612	0.4362	0.8070	0.2939	0.7864	0.2311
	24		0.7640	0.2972	0.7386	0.3215	0.7856	0.2359	0.7466	0.1855
	28		0.7489	0.1818	0.7018	0.2432	0.7339	0.1067	0.7014	0.1390

4.2.2 Visual Quality

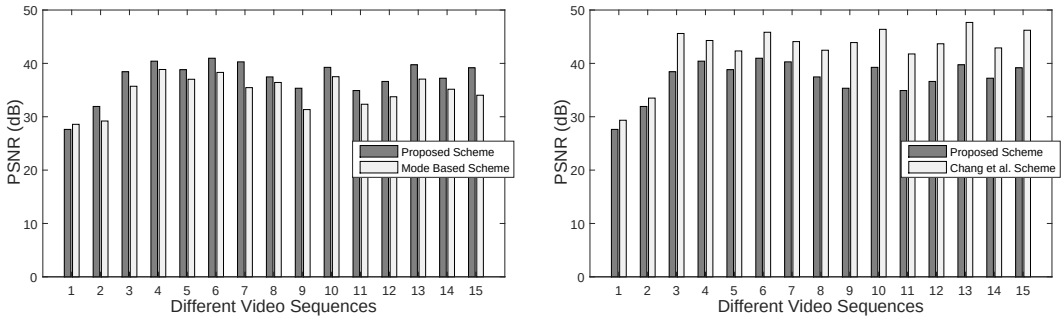
As discussed in Sec.4.1, a group of 4×4 temporally homogeneous TBs are used for embedding and a minimal number of zero valued transform coefficients are altered to embed the watermark. Intuitively, this makes the proposed scheme to perform well with respect to the visual quality of the watermarked video. An original intra frame and the corresponding watermarked frame of the *Flowervase* video sequences are shown in Figure 4.3. From this figure, it can be observed

(a) Original Frame (b) Watermarked Frame

Figure 4.3: *Cover frame of the Flowervase video sequence.*

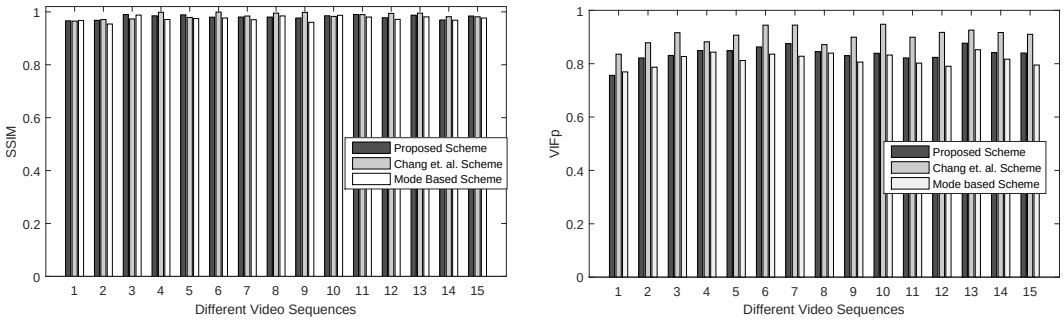
that there exist hardly any visual traces of watermark embedding. Further, the visual quality is measured using average PSNR, average SSIM index and VIFp [103] [104] [105] over all embedded frames. These metrics of the watermarked video are calculated with respect to original YUV video sequence using the VQM Tool [103]. In Figure 4.4(a), the PSNR comparisons between the proposed scheme and the mode based watermarking scheme proposed in Chapter 3 are depicted. From this figure, it can be observed that the visual quality of the proposed scheme is increased as the number of altered transformed coefficients is very less in this scheme. In Figure 4.4(b), the PSNR comparison between the proposed scheme and the existing Chang et al. scheme [57] is presented. It can be observe that the visual quality is higher in Chang et al. scheme [57]. As the watermark embedding in Chang et al. scheme [57] is drift error compensated, the visual quality is higher compared to the proposed scheme. However, the scheme [57] is more

susceptible against re-compression attack as shown in Table 4.2. In Figure 4.5, the comparison of the VIFp measure and SSIM index among the proposed scheme, the mode based scheme and Chang et. al.'s scheme [57] is depicted. From these results, it can be observed the watermarked video has decent visual quality which has improved compared to the mode based watermarking proposed in Chapter 3.



(a) Proposed Scheme & Mode Based Scheme (b) Proposed Scheme & chang et. al. scheme [57]

Figure 4.4: *Comparison of the PSNR values of the proposed scheme with the mode based scheme proposed in Chapter 3 and Chang et. al.'s scheme [57]. The video sequence index same as given in Table 4.1*



(a) SSIM (b) VIFp

Figure 4.5: *Comparison of the SSIM index and VIFp measures of the proposed scheme with the mode based scheme proposed in Chapter 3 and Chang et. al. scheme [57]. The video sequence index is same as given in Table 4.1*

4.2.3 Bit Increase Rate

In the proposed scheme, a very small number of zero valued transform coefficients need to be altered to embed the watermark in a group of 4×4 temporally homogeneous TBs (as discussed in Sec.4.1). So, the BIR due to the embedding is very less for the proposed scheme. In Figure 4.6, the comparison of BIR among the proposed scheme, the mode based scheme proposed in Chapter 3 and Chang et. al. scheme [57] is shown. The embedding is done for different video sequences with QP=16 and embedding rate is 100 bits/I Frame. From Figure 4.6, it can be observed that the BIR is very nominal and a little bit higher than the existing scheme. As the zero valued transformed coefficients are altered in the proposed scheme for embedding, the BIR increase compared with the existing scheme and mode based scheme.

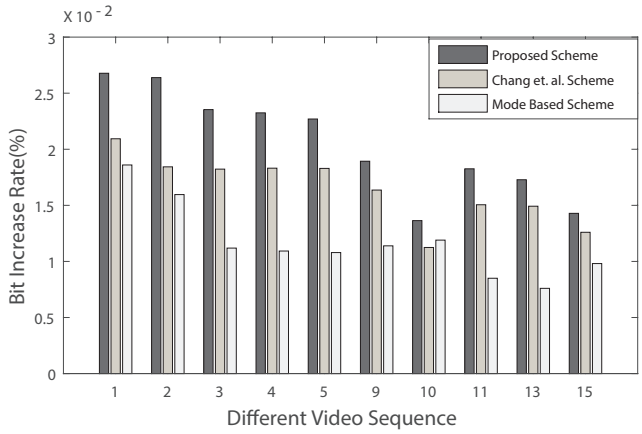


Figure 4.6: *Comparison of BIR of the proposed approach with the mode based scheme proposed in Chapter 3 and the existing scheme [57]. The video sequence index is same as given in Table 4.1*

4.3 Time Complexity Analysis

The detection of the temporally homogeneous blocks and watermark embedding are done using compressed domain parameters only without full decoding and

re-encoding of the video sequence. Thus, the scheme is not computationally complex. In the method, the detection of the temporally homogeneous 4×4 TB in two consecutive intra frame involves the time complexity of $O(f_{GOP}^2 \times n_B)$ where f_{GOP} is the size of GOP and n_B is the number of blocks in a frame. f_{GOP} varies depending on the size of the GOP and n_B varies depending on the resolution of the video. In case of the whole video sequence, this complexity can be written as $O(f_{total}^2 \times n_B)$ where f_{total} is the total number of frames in a video sequences. After the detection of temporally homogeneous 4×4 TBs, the time complexity of the watermark embedding in the intra frames is $O(f_I \times n_B)$ where f_I is the number of intra frames within a video sequence. So the total time complexity of the proposed scheme is $O(f_{total}^2 \times n_B + f_I \times n_B)$.

4.4 Summary

In this chapter, a robust watermarking scheme is proposed for H.265/HEVC compressed videos to resist against re-compression and image processing attacks. Firstly, temporally homogeneous 4×4 TBs are detected using only compressed domain parameters like motion vector, prediction modes and quantized transform coefficients. Then, two groups of such TBs from consecutive intra frames are used to embed the watermark by altering the NNZ values. The number of altered transformed coefficients is very less due to less NNZ difference in these groups. The watermark can also be extracted when the watermarked 4×4 TBs are merged into a 8×8 TB. Experimental results show that the proposed scheme is comparatively more robust than the existing schemes while maintaining decent visual quality. Further, the visual quality of the watermarked video can be improved by preventing drift error propagation. In the next chapter, first the drawback of existing drift compensation algorithms are analyzed. Then, a drift compensated watermarking scheme is proposed, which is robust against several distortions.

Chapter 5

Drift Compensated Robust Watermarking for H.265/HEVC Video Stream

In Chapter 3, a real time, low complexity and robust watermarking scheme is proposed to authenticate the H.265/HEVC compressed HD and Beyond-HD videos. In this scheme, the sustainability of the PBs and intra prediction modes have been analyzed with respect to nonzero transformed coefficients. Then, the intra prediction modes of H.265/HEVC stream are grouped to ensure that the mode change due to re-compression can be closed within a group. In Chapter 4, the robustness of the watermarking have been improved further by considering the temporal homogeneity. The motion coherent 4×4 TBs in an intra frame are detected using only compressed domain parameters like the motion vector, the prediction modes and the quantized transform coefficients. Then, the NNZ values of such TBs from consecutive intra frames are altered to embed the watermark. Though, both of the schemes are robust against re-compression and several image processing attacks, the drift error propagation can't be resisted by these schemes. As discussed in Chapter 2, the watermark embedding noise may propagate to the predicted blocks during the intra as well as the inter prediction processes. The process in which this noise accumulates and propagates to the subsequent

blocks in the prediction process until the next I-frame, is known as *drift error propagation*. In this chapter, a robust watermarking scheme for H.265/HEVC is proposed with drift error prevention for both intra and inter frames.

As discussed in the Introduction, relatively less work on drift compensated HEVC watermarking has been reported in the literature. A number of watermarking schemes with drift compensation are available for H.264/AVC. So, the effectiveness of the existing H.264/AVC based video drift compensated watermarking schemes and the extension of these schemes to the H.265/HEVC stream, are analyzed in this chapter. The research problem in this work is defined on the challenges faced in such extensions. Zhang et al. [52] have proposed a watermarking scheme in which the drift error is eliminated by subtracting it from all of the affected blocks. Huo et al. [53] have improved the Zhang et al. scheme [52] by reducing the complexity of the drift compensation process. Different drift compensations are used for different groups of watermarked AC coefficients in this scheme. Ma et al. [54] have proposed a drift compensated watermarking scheme in which pairs of AC coefficients are selected for compensating the drift error in the selected intra direction. In this scheme [54], embedding blocks are selected depending upon directions of neighboring intra predicted blocks. In [55], authors have proposed a drift compensated reversible watermarking method by selecting embeddable 4×4 blocks depending on the intra prediction mode of the neighboring blocks. If the neighboring blocks do not use the current block for intra prediction, then the current block is selected for watermark embedding. Recently, Chen et al. [56] have proposed another drift compensated watermarking scheme for H.264/AVC by altering a group of DCT coefficients which are selected such that the reference pixels do not get modified due to embedding. Chang et al. [57] have proposed an intra frame data hiding scheme for H.265/HEVC by altering a triplet of DST coefficients in 4×4 blocks and a group of DCT coefficients in larger size of blocks. Though, the scheme is drift compensated, the watermark is not robust against re-compression and common image processing attacks.

As discussed in Chapter 1, H.265/HEVC compression has some major differ-

ences with H.264/AVC that may play a critical role in the performance of the watermarking schemes. For example, the DST is used additionally in HEVC rather than the DCT which has been used in case of H.264/AVC. The number of prediction modes as well as the heterogeneity among the macro-block sizes are increased in HEVC. Moreover, the distribution of the coefficients in residual blocks becomes more sparse due to more accurate prediction in the case of the HEVC. It is experimentally observed that perturbing such sparse coefficients leads to substantial visual artifacts. So a direct extension of existing H.264/AVC based watermarking algorithms to the HEVC stream may not always give the optimal results. On the other hand, the recently proposed drift compensated H.265/HEVC watermarking scheme [57], is fragile against re-compression and common image processing attacks. The extension of this work [57] for robust watermarking, degrades visual quality of the watermarked video. The goal of this chapter is to present a drift compensated robust watermarking scheme for the H.265/HEVC video stream. The analysis of the intra prediction process of HEVC shows a designated set of spatial domain pixels is used as reference pixels for the prediction in a particular prediction mode. In this work, the pixels which are not involved in the prediction process are identified and used for embedding so that the propagation of the embedding noise can be resisted.

5.1 Problem Definitions

In literature, very few drift compensated watermarking schemes have been found for H.265/HEVC video sequences. On the other hand, the schemes for compensating the drift error for H.264/AVC video sequence can not be directly extended to the H.265/HEVC video because of the following reasons:

In the H.265/HEVC compression, the number of prediction modes and that of reference pixels used for intra-prediction are increased from the previous standards. This structural modification in HEVC ensures more accurate prediction than the previous encoders. Moreover, for HD and beyond HD videos, spatial

correlation is likely to be higher in small neighborhoods. The above two observations may be the probable causes for very sparse representation for number of nonzero transform coefficients of each transform block (TB) in HEVC encoding. Altering such sparsely distributed coefficients for watermark embedding may degrade the visual quality substantially. Moreover, the increase in the number of reference pixels causes greater accumulation of the drift error.

A few works have been reported in the literature where the drift error has been handled for H.264/AVC or the older video codecs. For example, in [55], the drift error is avoided by the careful selection of embedding blocks which are not participating in the intra prediction process. It has been observed that this scheme is not very suitable for H.265/HEVC as the number of embeddable blocks becomes too few. Intuitively, this is caused by the increased number of prediction modes and the reference pixels in HEVC.

Another interesting approach to compensate the drift error has been proposed in [54] [56] [57] where the watermark is embedded in a set of pixels /coefficients and the corresponding drift error is nullified by another set of pixels/coefficients of the same block. Let the transform coefficients Y of a 4×4 block be altered to Y' by adding the watermarking matrix W. Due to this alteration, the spatial domain residual error X will be modified to X'. Let Q be the step size defined by the *quantization parameter* (QP) and H be the transform kernel which is used to transform the prediction error residuals. Then, the distortion in the spatial domain D due to embedding can be written as follows,

$$D = \begin{bmatrix} d_{1,1} & d_{1,2} & d_{1,3} & d_{1,4} \\ d_{2,1} & d_{2,2} & d_{2,3} & d_{2,4} \\ d_{3,1} & d_{3,2} & d_{3,3} & d_{3,4} \\ d_{4,1} & d_{4,2} & d_{4,3} & d_{4,4} \end{bmatrix} = X' - X \tag{5.1}$$

Using the inverse transformation, D can be obtained from watermarking matrix W as follows [56],

$$D = H^T \times (W \times Q) \times H \tag{5.2}$$

For drift compensated watermarking, W should be chosen such that D has zero

values at the last row (i.e. $d_{4,1} = d_{4,2} = d_{4,3} = d_{4,4} = 0$) or last column (i.e. $d_{4,4} = d_{3,4} = d_{2,4} = d_{1,4} = 0$). In addition, to maintain low BIR due to embedding, W is chosen in such a way that it has the minimum number of nonzero elements. In [54], W is chosen with two nonzero elements and in [56], W is chosen with four nonzero elements to cancel the distortion in the last column and the last row of the 4×4 spatial domain block for H.264/AVC videos. Since in H.265/HEVC, 4×4 TBs use the DST-VII transform whose transform kernel is not symmetric (even or odd) around its middle as DCT, embedding watermark in pair of coefficients (or quadruple of coefficients) like the schemes [54] [56] cannot fully cancel the drift distortion.

In another work [57], a drift compensated information hiding scheme is proposed for H.265/HEVC videos, where the triplets of DST coefficient are perturbed by ±1 embedding. This scheme is fragile against re-compression attack as shown in the experimental section (Please refer to Sec 5.3, Table 5.2). Now, let this scheme be extended for robust watermarking by altering the triplet of coefficients by a larger amount, say $\pm\delta$. Then the chosen W can be presented by,

$$W = \begin{bmatrix} \delta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\delta & 0 & 0 & 0 \\ \delta & 0 & 0 & 0 \end{bmatrix} \tag{5.3}$$

Using Eq. 1.4 and Eq. 1.5, the distortion added in spatial domain can be written as,

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ d_{3,1} & d_{3,2} & d_{3,3} & d_{3,4} \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.4}$$

where, values of $d_{3,1},\ d_{3,2},\ d_{3,3},\ d_{3,4}$ are given by,

$$\begin{aligned} d_{3,1} &= (\lfloor(29 \times \lfloor(111 \times \delta \times Q)/64 + 1/2\rfloor)/4096 + 1/2\rfloor) \\ d_{3,2} &= (\lfloor(55 \times \lfloor((111 \times \delta \times Q)/64 + 1/2\rfloor)/4096 + 1/2\rfloor) \\ d_{3,3} &= (\lfloor(37 \times \lfloor(111 \times \delta \times Q)/64 + 1/2\rfloor)/2048 + 1/2\rfloor) \\ d_{3,4} &= (\lfloor(21 \times \lfloor(111 \times \delta \times Q)/64 + 1/2\rfloor)/1024 + 1/2\rfloor) \end{aligned} \tag{5.5}$$

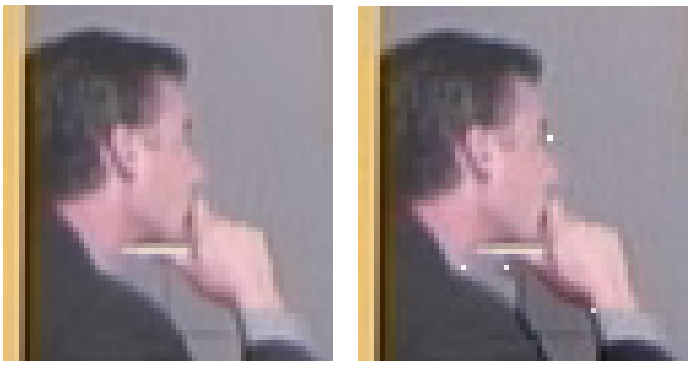
where, $\lfloor\ \rfloor$ gives the floor value of the argument. So total distortion added in spatial domain is $\Delta = \mid d_{3,1} \mid + \mid d_{3,2} \mid + \mid d_{3,3} \mid + \mid d_{3,4} \mid$. It can be observed that the Δ is quite high for small value of δ. For example, when $\delta = 5$ and quantization parameter is 16, the distortion added in the spatial domain is

$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 8 & 15 & 20 & 23 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ and } \Delta = 66 \tag{5.6}$$

Due to this large amount of distortion, the visual quality of the watermarked video may be substantially degraded. In Figure 5.1, the part of original frame and watermarked frame have been shown, where the watermark is embedded using the Chang et al. scheme [57] with parameter $\delta = 5$ and quantization parameter =16. From the figure, it can observed that some pixels exceed the boundary value of the dynamic range $0 - 255$ due to large amount of distortion. On the other hand, this scheme embeds in TBs of all sizes and the number of coefficients altered for embedding is also very high. This may also cause degradation in visual quality and increase in bit rate. So, Chang et al. scheme [57] can not be directly extended for the robust drift compensated watermarking of H.265/HEVC video stream.

5.2 Proposed Scheme

From Section 5.1, it can be observed that existing watermarking schemes are not suitable for drift compensated watermarking of H.265/HEVC. In this section, a drift compensated robust watermarking scheme is proposed for HD and beyond-HD videos using only compressed domain parameters of H.265/HEVC. Intuitively, a small size TB with higher NNZ values belongs to a higher texture region of the video frame. As the perturbation in this type of blocks is less sensitive to the *Human Visual System* (HVS), watermark embedding causes less visual artifacts [32]. So, the luma components of such 4×4 TBs in the intra frames in HEVC compression are used for embedding in this work to minimize the visual



(a) Original Frame (b) Watermarked Frame

Figure 5.1: *Degradation of visual quality in a part of intra frame of Kristen & Sara video sequence are depicted in the case where the Chang et al. scheme [57] is extended for robust watermarking. The watermark is embedded with $\delta = 5$ and quantization parameter =16*

artifacts. *Henceforth, a luma 4×4 TB will be denoted as LTB in this chapter.* In the proposed scheme, the watermark is embedded by altering the transform coefficients of the selected LTB of the intra predicted frames. The embedding coefficients are selected in such a way that the drift error is nullified in the intra prediction process. For P frame drift compensation, two sets of blocks have been used. One set is used for embedding and the other set is used for the inter prediction process. In this section, first the prevention of I frame drift errors and P frame drift errors are first discussed. Then, these drift error prevention processes are used for watermark embedding. The watermark extraction process is explained in the subsequent subsection.

5.2.1 Prevention of the Drift Error for Intra-Predicted (I) Frames

In Section 5.1, it is observed that the drift error can be compensated by altering a selected triplet of transformed DST coefficients in 4×4 TBs of H.265/HEVC [57]. It was also pointed out that the distortion in the pixel domain due to transform domain embedding may be quite high and sometimes it may cross the dynamic

range of the pixels (or boundary values of pixels). This may cause visible artifacts. In this chapter, a drift compensated watermarking scheme is proposed by eliminating the last row and last column of a TB such that a robust watermarking can be achieved without substantial degradation of the watermarked video. The process is explained as follows:

Let $Y_{4\times4}$ be a transformed coefficient matrix of a residuals matrix $X_{4\times4}$ which is found out by applying inverse 1D DST-VII in column and row wise on the de-quantized values of $Y_{4\times4}$ as discussed in Sec. 1.2.1.2. The residuals matrix $X_{4\times4}$ can be written as,

$$X_{4\times4} = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix} \tag{5.7}$$

After eliminating the residuals which will take part in intra prediction, the matrix $X_{3\times3}$ is constructed from $X_{4\times4}$ as follows,

$$X_{3\times3} = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} \\ x_{2,1} & x_{2,2} & x_{2,3} \\ x_{3,1} & x_{3,2} & x_{3,3} \end{bmatrix} \tag{5.8}$$

If the watermark is embedded by manipulating $X_{3\times3}$ then the drift error is nullified. In the proposed scheme, $X_{3\times3}$ is altered in such a way that distortion affects fewer coefficients and the distortion added in spatial domain of an LTB is also very little. The basis vectors of DST-I are symmetric around middle as the DST-I is odd symmetric transform. So, the amount of distortion added due to altering a pair of DST coefficients is less than that due to altering a pair of DCT coefficients. The 3×3 transform matrix H_1 of DST-I is given by,

$$H_1 = \sqrt{\frac{1}{2}} \times \begin{bmatrix} \sin(\pi/4) & \sin(\pi/2) & \sin(3\pi/4) \\ \sin(\pi/2) & \sin(\pi) & -\sin(\pi/2) \\ \sin(3\pi/4) & -\sin(\pi/2) & \sin(\pi/4) \end{bmatrix} \sim \left[\sqrt{\frac{2}{N+1}} \sin\left(\frac{\pi ij}{N+1}\right) \right] \tag{5.9}$$

where i,j denotes the position in matrix H_1 and i,j=1,...,N and N=3

To embed the watermark, the $X_{3\times3}$ is forward transformed by H_1 to obtain the transformed coefficient matrix $\widetilde{Y}_{3\times3}$. The forward and inverse transforms can

be written as follows,

$$\begin{aligned} \widetilde{Y}_{3\times3} &= H_1 \times X_{3\times3} \times H_1^T \\ X_{3\times3} &= H_1^T \times \widetilde{Y}_{3\times3} \times H_1 \end{aligned} \tag{5.10}$$

After the forward transform, the transformed coefficient matrix $\widetilde{Y}_{3\times3}$ is used for embedding the watermark matrix $W_{3\times3}$. The inverse DST-I transformation is used after embedding to get the spatial distorted matrix $X'_{3\times3}$ which can be presented as the sum of $X_{3\times3}$ and the distortion $D_{3\times3}$ as follows,

$$\begin{aligned} X'_{3\times3} &= \begin{bmatrix} x'_{1,1} & x'_{1,2} & x'_{1,3} \\ x'_{2,1} & x'_{2,2} & x'_{2,3} \\ x'_{3,1} & x'_{3,2} & x'_{3,3} \end{bmatrix} \\ &= H_1^T \times (\widetilde{Y}_{3\times3} + W_{3\times3}) \times H_1 \\ &= H_1^T \times \widetilde{Y}_{3\times3} \times H_1 + H_1^T \times W_{3\times3} \times H_1 = X_{3\times3} + D_{3\times3} \\ &\quad (\textit{using linearity of DST}) \end{aligned} \tag{5.11}$$

In the proposed scheme, $W_{3\times3}$ is chosen in such a way that not only a few number of pixels are altered but the amount of distortion to be added is also relatively small. Let $W_{3\times3}$ be chosen from two matrices in the set $\mathcal{W}$,

$$\mathcal{W} = \left\{ \begin{bmatrix} 0 & \delta & 0 \\ 0 & 0 & 0 \\ 0 & -\delta & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ \delta & 0 & -\delta \\ 0 & 0 & 0 \end{bmatrix} \right\} \tag{5.12}$$

where, δ is the variation of coefficients due to embedding. If $W_{3\times3}$ is taken from Eq. 5.12, the spatial distortion matrix $D_{3\times3}$ due to embedding in $\widetilde{Y}_{3\times3}$ can be written using Eq. 5.11 as,

$$D_{3\times3} = H_1^T \times \begin{bmatrix} 0 & \delta & 0 \\ 0 & 0 & 0 \\ 0 & -\delta & 0 \end{bmatrix} \times H_1 = \begin{bmatrix} 0 & 0 & 0 \\ -\delta & 0 & \delta \\ 0 & 0 & 0 \end{bmatrix} \tag{5.13}$$

In this case, only two pixels are altered and the total distortion added in spatial domain Δ_3 is given by,

$$\Delta_3 = (\mid \delta \mid + \mid -\delta \mid) = 2 \times \delta \tag{5.14}$$

From the above discussion, it can be concluded that if the watermark is embedded in $X_{3\times3}$ then the distortion added in LTB is less. For example, when $\delta = 5$ and quantization parameter is 16, the distortion added in the spatial domain

$$D = \begin{bmatrix} 0 & 0 & 0 \\ -5 & 0 & 5 \\ 0 & 0 & 0 \end{bmatrix} \text{ and } \Delta_3 = 10 \tag{5.15}$$

Due to this reduced amount of distortion, visual quality is maintained in proposed scheme. After embedding in $X_{3\times3}$, the watermarked residuals of a 4×4 TB is formed by appending the last row and the last column of $X_{4\times4}$ to $X'_{3\times3}$. As, $X'_{4\times4}$ has last row and column unaltered, the drift is nullified within intra frames. The watermarked residuals $X'_{4\times4}$ of a 4×4 TB can be written as,

$$X'_{4\times4} = \begin{bmatrix} x'_{1,1} & x'_{1,2} & x'_{1,3} & x_{1,4} \\ x'_{2,1} & x'_{2,2} & x'_{2,3} & x_{2,4} \\ x'_{3,1} & x'_{3,2} & x'_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \end{bmatrix} \tag{5.16}$$

$X'_{4\times4}$ is forward DST-VII transformed and quantized similar to the H.265/HEVC encoding and the watermarked transformed coefficients are saved in compressed H.265/HEVC stream. The overall drift error free watermark embedding process is summerized as follows:

1. Firstly, spatial domain residual error values ($X_{4\times4}$) are obtained from the 4×4 TB using 4×4 inverse DST-VII transformation as given in Eq. 1.4 and Eq. 1.5.

2. The residual error pixels which do not participate in the intra prediction process are selected for embedding. In another words, pixels from last row and last column of $X_{4\times4}$ residual error matrix are not considered for embedding as they are involved in intra prediction process. Thus a matrix of dimension 3×3 (say $X_{3\times3}$) is used for embedding.

3. The DST coefficients $\widetilde{Y}_{3\times3}$ are obtained from $X_{3\times3}$ block by applying 3×3 DST-I transformation. The watermark is embedded by varying selected transformed coefficients pair of $\widetilde{Y}_{3\times3}$ [please refer Eq. 5.11 and Eq. 5.12].

4. The watermarked residuals error coefficients $X'_{3\times3}$ are obtained using IDST-I on the $(\widetilde{Y}_{3\times3} + W_{3\times3})$ values.

5. The last row and the last column $X_{4\times4}$ are appended with $X'_{3\times3}$ to get the watermarked residual matrix $X'_{4\times4}$.

6. Finally, the watermarked residual error blocks $(X'_{4\times4})$ are subjected to DST-VII and the quantization process to get watermarked TB.

5.2.2 Prevention of the Drift error for Inter-Predicted (P) Frames

In the inter-prediction process, frames are divided into non-overlapping blocks and each block is predicted from the reference frame blocks. If these reference frame blocks are watermarked, then these watermark noise propagates to the predicted frames during the prediction process. Thus, inter frame drift error propagation degrades the visual quality of the predicted frames. In this section, a scheme is proposed for drift compensation for the inter-predicted P-frames based on the concept of *homogeneous zones*. A region in a frame is called *spatially homogeneous* if pixels within it are highly correlated. Within a spatially homogeneous region, most of the blocks have high inter block correlation. It has been observed that the temporal prediction process can be done from any block within such homogeneous zone without considerable deviation from the actual prediction. In this scheme, this observation is capitalized and the intra-predicted reference blocks which are watermarked, are not used in the temporal prediction process. Thus within a homogeneous zone, the blocks with high NNZ values are selected for embedding and the rest of the blocks are used for motion prediction. This results in a substantial reduction in inter frame drift propagation. The overall process is shown in Figure 5.2 where the marked region is a homogeneous region in the reference frame. Block B is predicted from block A. If A is used for watermarking, then block A' (which has very high spatial correlation with A) can be used for

inter prediction of B. Thus the noise added due to watermarking in A does not propagate to B.

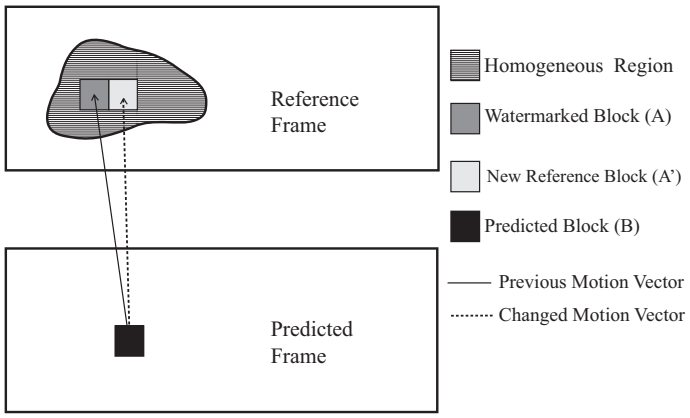


Figure 5.2: *Homogeneous blocks are used for drift compensation*

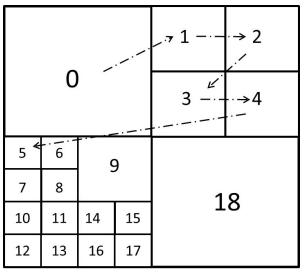


Figure 5.3: *Z-Scanning order of CBs within a CTU.*

5.2.2.1 Prevention of drift Error

To prevent the drift error in the inter frame, the watermarked blocks of the intra frame should not be used in the inter prediction process. After detecting homogeneous regions in the intra frame, it is checked if a PB of a P-frame is predicted using a watermarked block of intra frame. If it is predicted using the watermarked block, the motion vector of the current PB is changed to the closest motion vector such that it is now being predicted from a block which belongs to the homogeneous zone containing this watermarked block, and not directly from the watermarked block. The drift error prevention process has been shown in Figure 5.2 and the corresponding algorithm is described in Algorithm 6.

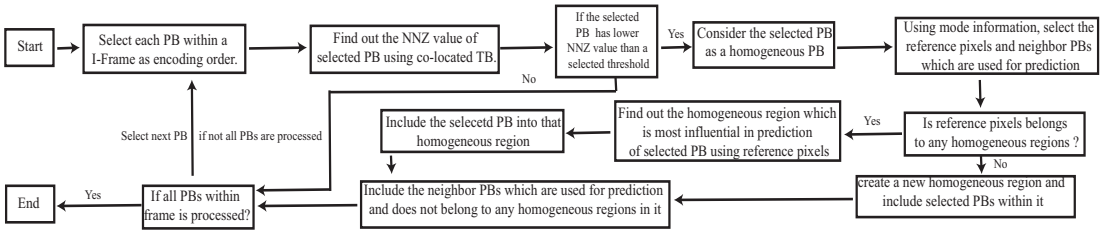


Figure 5.4: *Block diagram of Homogeneous Zone detection process in intra frame*

ALGORITHM 5: fH_region_I (**CTU**) /* Find Homogeneous region in an I frame */

Input: **CTU** : CTUs of an I frame
Output: **HR** : Homogeneous Regions for selected I frame
begin
Create a vector **HR** to store homogeneous region number for each PB within a frame;
Initialize all values of HR to zero;
$count = 0$; `/* count is used for numbering the homogeneous regions within a frame */`
`/* CTUs are selected in raster order */`
for *each* $\mathbf{PB_j}$ *in* $\mathbf{CTU_i}$ *where* $\mathbf{PB_j}$ *represents the* j^{th} *PB in* $\mathbf{CTU_i}$ **do**
`/* PBs are selected in z-scan order */`
Find the NNZ value of $\mathbf{PB_j}$ $\mathcal{N}_j$, using the TBs which are co-located with $\mathbf{PB_j}$;
Select NNZ threshold value $\mathcal{N}_{th}$, depending on size of $\mathbf{PB_j}$;
if $\mathcal{N}_j < \mathcal{N}_{th}$ **then**
`/* PBj has a homogeneous texture. So it is selected as a homogeneous block */`
Find the Prediction mode m of $\mathbf{PB_j}$;
Create a vector **P** to keep the reference pixels of $\mathbf{PB_j}$;
if $m \geq 2$ *and* $m \leq 9$ **then**
Select all left reference pixels $\Re_{0,0}$ to $\Re_{0,2N}$ in **P**;
else if $m \geq 27$ *and* $m \leq 34$ **then**
Select all top reference pixels $\Re_{0,0}$ to $\Re_{2N,0}$ in **P**;
else if $m \geq 11$ *and* $m \leq 25$ **then**
Select top reference pixels $\Re_{0,0}$ to $\Re_{N,0}$ and left reference pixels $\Re_{0,0}$ to $\Re_{0,N}$ in **P** ;
else if $m = 10$ **then**
Select left reference pixels $\Re_{0,0}$ to $\Re_{0,N}$ in **P**;
else if $m = 26$ **then**
Select top reference pixels $\Re_{0,0}$ to $\Re_{N,0}$ in **P** ;
else
Select all reference pixels $\Re_{2N,0}$ to $\Re_{0,2N}$ in **P**; `/* for prediction mode 0 and 1 */`
Create a vector **NB** to keep the neighbor PBs of $\mathbf{PB_j}$;
Select the PBs in which the pixels of **P** belongs in **NB**;
Using vector **HR**, find out the homogeneous region **hr** in which maximum pixels of **P** belongs ;
if *the* **NB** *PBs do not belongs to any homogeneous region* **then**
$count = count + 1$; `/* increment the homogeneous region counter */`
$HR_{i,j} = count$; `/* assign PBj new homogeneous number */`
for *each PB* $\mathbf{NB}_l$ *in* **NB do**
Assign $\mathbf{NB}_l$, same homogeneous region number *count* ;
else
Find the homogeneous region number k of **hr**;
$HR_{i,j} = k$; `/* assign homogeneous region number` k `to the` PB_j `*/`
for *each PB* $\mathbf{NB}_l$ *in* **NB do**
if $\mathbf{NB}_l$ *does not belong to any homogeneous regions* **then**
assign $\mathbf{NB}_l$, same homogeneous region number k ;

ALGORITHM 6: fDEP_Inter_Frame(**F** , **RF**, **HR**) /*Drift error prevention in inter frame*/

Input: **F** : P frame of video, **RF** : reference frame of **F**, **HR** : Homogeneous regions of reference frame **RF**
Output: **F′** : Inter drift compensated frame
begin
  for *each* $\mathbf{PB}_j$ *in the inter frame* **F** *where* $\mathbf{PB_j}$ *represents the* j^{th} *PB in* **F do**
    if $\mathbf{PB}_j$ *is predicted using watermarked blocks* $\mathbf{PB}_w$ *of* ***RF*** **then**
      Find the homogeneous region **hr** in which watermarked blocks $\mathbf{PB}_w$ belong;
      Select the homogeneous region excluding the watermarked blocks i.e. select $\mathbf{hr} - \mathbf{PB}_w$ for motion prediction of $\mathbf{PB}_j$;
      Change motion vector of the PB_j such that it is now being predicted from $\mathbf{hr} - \mathbf{PB}_w$;

5.2.3 Watermark Embedding

In this scheme, the watermark is embedded only in the intra predicted frames as they are the most important frames in a compressed stream. The drift error is compensated both for intra and inter predicted frames. As discussed in Sec. 5.2, 4×4 TBs with higher NNZ values of the intra frame are used to embed the watermark. Balancing between robustness and the number of embeddable bits per frames, the threshold for NNZ value ($\mathcal{N}_{th}$) is chosen from a comprehensive set of experimental results. The 4×4 TBs having higher NNZ than $\mathcal{N}_{th}$ are selected for embedding. The selected 4×4 TBs for embedding are subjected to de-quantization and the subsequent IDST-VII to get the spatial residual block coefficients $X_{4\times4}$. Now the subset $X_{3\times3}$ where the pixels are not involved for next prediction, are used for embedding as discussed in Sec. 5.2.1. The watermark is embedded in the DST-I coefficients $\widetilde{Y}_{3\times3}$ of $X_{3\times3}$. For embedding, a pair of coefficients is selected such a way that perturbing those coefficients results least distortion in spatial domain as discussed in Sec 5.2.1. The watermark matrix $W_{3\times3}$ is chosen from the two matrices given in Eq. 5.12. The amount of alteration

ALGORITHM 7: fTB_Watermarking(TB, b_w) /* Embed watermark in a TB*/

Input: TB : TB to be watermarked, b_w : Watermark bit
Output: TB_w : Watermarked TB
begin

Select quantized transformed coefficients $Y_{4\times4}$ of TB;
Find dequantize values $Y^{dq}_{4\times4}$ of $Y_{4\times4}$ using H.265/HEVC dequantization function;
Apply inverse DST-VII Transform on $Y^{dq}_{4\times4}$ and obtain residual error $X_{4\times4}$ of TB;
Select the coefficients $X_{3\times3}$ which will not participate in prediction process i.e. select the values from $X_{4\times4}$ without last row and column;
Apply forward DST-I Transform $X_{3\times3}$ and obtain transform coefficients $\widetilde{Y}_{3\times3}$;
Select $W_{3\times3}$ from Eq. 5.12;
Select pair of efficients $\widetilde{Y}_{3\times3}(m_1, n_1)$ and $\widetilde{Y}_{3\times3}(m_2, n_2)$ from $\widetilde{Y}_{3\times3}$ depending on $W_{3\times3}$;
Find the magnitude of altering the coefficients δ using Eq. 5.17;
if $b_w = 1$ *and* $\widetilde{Y}_{3\times3}(m_1, n_1) < \widetilde{Y}_{3\times3}(m_2, n_2)$ **then**
    `/* Embed the watermark by altering coefficients */`
    $\widetilde{Y}'_{3\times3} = \widetilde{Y}_{3\times3} + \alpha * W_{3\times3}$;
if $b_w = 0$ *and* $\widetilde{Y}_{3\times3}(m_1, n_1) > \widetilde{Y}_{3\times3}(m_2, n_2)$ **then**
    `/* Embed the watermark by altering coefficients */`
    $\widetilde{Y}'_{3\times3} = \widetilde{Y}_{3\times3} - \alpha * W_{3\times3}$;
Apply inverse DST-I Transform on $\widetilde{Y}'_{3\times3}$ and obtain watermarked spatial coefficients $X'_{3\times3}$;
Add last row and last column of $X_{4\times4}$ with $X'_{3\times3}$ and get the watermarked residual prediction error $X'_{4\times4}$;
Apply forward DST-VII Transform on $X'_{4\times4}$ using H.265/HEVC Transform function and get transformed coefficients $Y^{dq'}_{4\times4}$;
Quantize $Y^{dq'}_{4\times4}$ to obtain the watermarked Quantized Transformed coefficient $Y'_{4\times4}$;
Save $Y'_{4\times4}$ in TB and denote it as TB_w;

δ is calculated based on the difference between the selected pair coefficient, given as,

$$\begin{aligned}\textbf{If}\Big(W_{3\times3} = \begin{bmatrix}0 & \delta/2 & 0\\ 0 & 0 & 0\\ 0 & -\delta/2 & 0\end{bmatrix}\Big)\{\\ \delta = \mid \widetilde{Y}_{3\times3}(1,2) - \widetilde{Y}_{3\times3}(3,2) \mid +\Delta_{th}\}\\ \textbf{Else}\{\\ \delta = \mid \widetilde{Y}_{3\times3}(2,1) - \widetilde{Y}_{3\times3}(2,3) \mid +\Delta_{th}\}\end{aligned} \tag{5.17}$$

where, Δ_{th} is a positive constant which is used to make the difference when the selected coefficients of $\widetilde{Y}_{3\times3}$ have same magnitude. The embedding is done by changing values of $\widetilde{Y}_{3\times3}$ as follows:

$$\begin{aligned}&\textbf{If}(watermark\ bit = 1)\\ &\quad \widetilde{Y}'_{3\times3} = \widetilde{Y}_{3\times3} + \alpha \times W_{3\times3}\\ &\textbf{Else}\\ &\quad \widetilde{Y}'_{3\times3} = \widetilde{Y}_{3\times3} - \alpha \times W_{3\times3}\end{aligned} \tag{5.18}$$

where $\alpha \geq 1$ is the watermarking strength. In the proposed scheme, the security of embedding procedure may be increased by choosing TBs according to a secret key. The locations of watermarked TBs are saved into a location map WL which is used at the time of extarction. The watermark embedding procedure in the selected TB is elaborated step by step in Algorithm 7 and corresponding block diagram has been shown in Figure 5.5. For inter frame drift compensation, the embeddable TBs are chosen from homogeneous regions of intra frame. After watermark embedding, the motion vector of inter predicted PB is altered, if the PB is predicted using watermarked regions. The watermark embedding procedure for an input video is elaborated step by step in Algorithm 8 and and corresponding block diagram has been shown in Figure 5.6.

5.2.4 Watermark Extraction

The extraction of the watermark is relatively simpler than embedding process. A location map describing embedding location WL, is used to find the watermarked TBs. Using de-quantization and inverse DST-VII, the spatial values $X'_{4\times4}$ of a

ALGORITHM 8: Watermark Embedding(V, WS)

Input: V : Video to be watermarked, WS : Watermark Sequence
Output: VW : Watermarked Video, WL = Watermark Location Map
begin
  for *each frame* F_i *of input video* V **do**
    if F_i *is a intra frame* **then**
      Select the CTUs, $\mathbf{CTU}^i$ of F_i;
      HR= fH_region_I ($\mathbf{CTU}^i$);
        `/* Find out the homogeneous regions` HR^i `of` F_i `using Algorithm 5; */`
      for *each luma* TB_j *within* F_i *where* TB_j *is* j^{th} *luma TB in* F_i **do**
        if *the size of* TB_j *is* 4×4 **then**
          Find the size of NNZ value $\mathcal{N}_j$ of TB_j;
          Generate a pseudo random bit b_r using a key;
          if $\mathcal{N}_j \geq \mathcal{N}_{th}$ *and* $b_r = 1$ **then**
            `/* Embed watermark in` TB_j `*/`
            Select the watermarking bit b_w from WS;
            fTB_Watermarking(TB_j, b_w)
            `/* Embed watermark` b_w `in` TB_j `using Algorithm 7 */`
            Save location i, j in WL;
    if F_i *is a inter frame* **then**
      Select the reference frame **RF** of F_i and homogeneous regions **HR** of **RF** ;
      fDEP_Inter_Frame(F_i , **RF**, **HR**)
        `/* Do drift compensation in` F_i `using Algorithm 6 */`

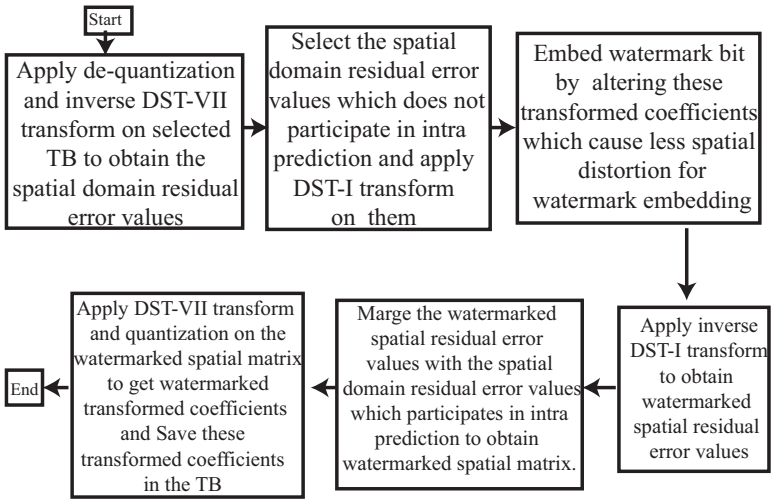


Figure 5.5: *Block diagram of the watermark embedding process in a TB of proposed scheme*

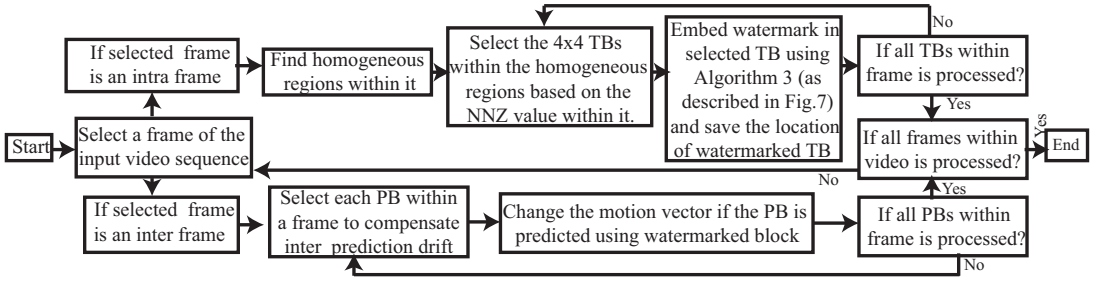


Figure 5.6: *Block diagram of the watermark embedding process of proposed scheme for a input video*

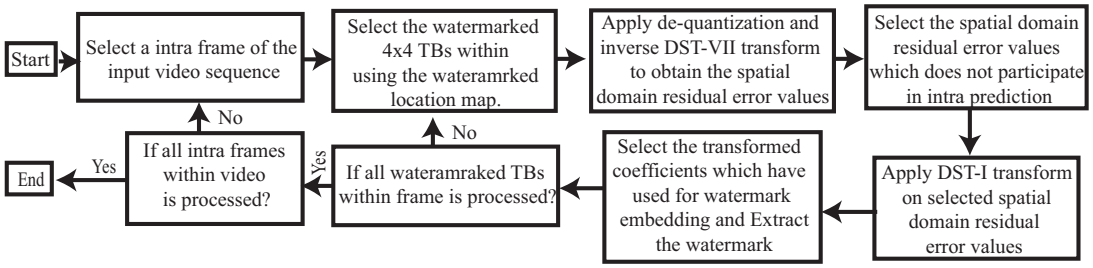


Figure 5.7: *Block diagram of the watermark extraction process of proposed scheme*

watermarked TB is found. The subset of residuals $X'_{3\times3}$ is obtained by eliminating the last row and last column of the original residual $X'_{4\times4}$. Applying the forward DST-I on the $X'_{3\times3}$ block, the transformed coefficients $\widetilde{Y}'_{3\times3}$ are found. Select the watermarked coefficient position $(m_1,\ n_1)$, $(m_2,\ n_2)$ depending on watermarking matrix $W_{3\times3}$. Then watermark is extracted by comparing values of the coefficient pair $\widetilde{Y}'_{3\times3}(m_1, n_1)$ and $\widetilde{Y}'_{3\times3}(m_2, n_2)$, using the following rule,

$$\begin{aligned} &\textbf{If}(\widetilde{Y}'_{3\times3}(m_1, n_1) \geq \widetilde{Y}'_{3\times3}(m_2, n_2)) \\ &\qquad \text{watermark bit=1} \\ &\textbf{Else} \\ &\qquad \text{watermark bit=0} \end{aligned} \tag{5.19}$$

The extraction algorithm is described in Algorithm 9 and corresponding block diagram has been shown in Figure 5.7.

5.3 Experimental Results

The proposed algorithm is implemented using the H.265/HEVC reference software HM.10 [13]. To check the effectiveness of the scheme, the algorithm is tested on standard video sequences with different resolutions, textures and motions. Details of test sequences along with experimental setup are listed in Table 5.1. The watermark is embedded at QP=16. The NNZ threshold for selecting embeddable TB (i.e. $\mathcal{N}_{th}$) is chosen based on the number of bits embedded per frame and the visual quality degradation due to embedding. The performance of the watermarking scheme is evaluated in terms of three metrics: robustness, visual quality and BIR.

5.3.1 Robustness

The robustness of the proposed scheme is analyzed against the re-compression attack and common image processing attacks in this subsection. In the proposed scheme, the watermark is embedded by varying the residuals of 4×4 TU

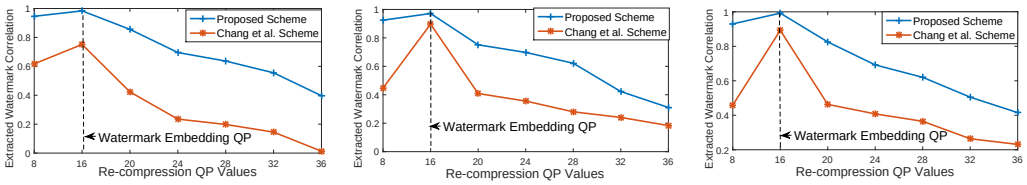
ALGORITHM 9: Watermark Extraction(VW, WL)

Input: VW :Watermarked Compressed Bit stream, WL: Watermark Location

Output: WE:Extracted Watermark sequence

begin

- Read the WL for watermark location;
- **for** *each intra coded video frame* F_i *of input video* VW **do**
  - **for** *each watermarked Transform Block* TB_j *within Frame* F_i **do**
    - Select the coefficients $Y'_{4\times4}$ of TB_j;
    - Dequantize values of $Y_{4\times4}$ and obtain $Y^{dq'}_{4\times4}$ using H.265/HEVC Dequantize function;
    - Apply inverse DST-VII Transform on $Y^{dq}_{4\times4}$ and obtain $X'_{4\times4}$ using H.265/HEVC Transform function;
    - Select the coefficients $X'_{3\times3}$ from $X'_{4\times4}$ without last row and last column of $X'_{4\times4}$;
    - Apply forward DST-I Transform on $X'_{3\times3}$ and obtain $\widetilde{Y}'_{3\times3}$;
    - Select the pair of watermarked efficient $\widetilde{Y}'_{3\times3}(m_1, n_1)$ and $\widetilde{Y}'_{3\times3}(m_2, n_2)$ based on selected $W_{3\times3}$ at the time of embedding;
    - **if** $\widetilde{Y}'_{3\times3}(m_1, n_1) \geq \widetilde{Y}'_{3\times3}(m_2, n_2)$ **then**
      - extracted watermark bit=1;
    - **else**
      - extracted watermark bit=0;
    - save extracted bit in WE;



(a) BQMall (b) Johnny (c) PeopleOnStreet

Figure 5.8: *Comparison of robustness of the proposed scheme with Chang et al. scheme [57] against re-compression attack*

blocks. The residuals are chosen in such a way that drift can be compensated. As in earlier case, Chang et al. [57] scheme has been chosen for comparison.

Table 5.1: ***Details of the Experimental Set Up***

Parameters for Watermarking	**Values Taken**
H.265/HEVC version	HM-10.0
GOP structure	One I frame and consecutive 8 P frames
Intra Period	8
Quantization Parameter(QP)	16
Video Sequence Used	1. BasketBall, 2.BlowingBubbles, 3.BQMall, 4.BQTerrace, 5.Cactus, 6.Chipmunks, 7.Jonny, 8. Kristan& Sara, 9. FourPeople, 10. shields, 11.HoneyBee, 12.YachtRide 13. PeopleOnStreet 14.Jocky 15.ReadySteadyGo
video resolutions :	416×240, 832×240, 1280×720 1920×1080, 2560×1600, 3840×2160
Bit Depth	8 bit/pixel
Frame Rate	30 Frame per Sec. : Video No. 13, 50 Frame per Sec. : Video No. 1, 2, 5, 10 60 Frame per Sec. : Video No. 3, 4, 6, 7, 8, 9 120 Frame per Sec. : Video No. 11, 12, 14, 15
NNZ Threshold (NNN_{th})	12
Watermarking Strength (α)	1

As, Chang et al. [57] proposed a drift compensated data hiding scheme in intra frames of H.265/HEVC and the proposed scheme also embeds watermark in H.265/HEVC without drift error. The robustness of the proposed scheme is evaluated in terms of the correlation coefficient which is an objective metric for self similarity between original watermark and the extracted watermark. In Table 5.2, the robustness comparison against re-compression attack is presented. The re-compression attack performed in two steps. First the compressed watermarked video is decompressed using H.265/HEVC decoder. Then, the decompressed raw video is again compressed with different QPs. The re-compression has been performed for QP=[8, 16, 20, 24, 28, 32, 36]. Similar results for the *Basketball*, the *Jonny* and the *PeopleOnStreet* video sequences are plotted in Figure 5.8 for more clarity. Table 5.2 and Figure 5.8 shows that the proposed scheme is more robust against re-compression attack than the Chang et al. scheme [57]. The robustness of the watermarking scheme can be further improved by increasing the watermark strength (α). The extracted watermark correlations for different watermark strength (α) have been tabulated in Table 5.3. During re-compression attack, the watermark is de-synchronized mainly because of two reasons. First,

the change in the QP values during re-compression change the prediction error values, which lead to change in watermarked coefficients. Secondly, the loss of embedded TBs due to merging of TBs into a larger size also de-synchronizes the watermark. If the watermark is embedded using higher α, the robustness of the watermark increases as change in QP values will have less affect the watermarked coefficients. But the rate of loss of the embedded TBs remains almost the same. Thus, it may be observed that the extracted watermark correlations do not improve much when the watermark strength reaches a certain threshold.

Table 5.2: ***Comparison of Robustness of the Proposed Scheme (Denoted by A) with Chang et al. [57] Scheme (Denoted by B) against Re-compression Attack***

Video Sequences	Video Resolution	Embedding Capacity bits per frame	Scheme Name	Extracted Watermark correlation After Re-compression attacks with different QPs						
				QP 8	QP 16	QP 20	QP 24	QP 28	QP 32	QP 36
BasketballPass	416× 240	100	A	0.9288	0.9815	0.8632	0.7404	0.7047	0.6106	0.4736
			B	0.5729	0.9010	0.4721	0.4140	0.4091	0.3492	0.3121
BlowingBubbles	416× 240	100	A	0.9464	0.9915	0.8796	0.7760	0.7181	0.6222	0.4984
			B	0.5245	0.8250	0.4323	0.3791	0.3457	0.2973	0.2576
BQMall	832× 480	100	A	0.9461	0.9836	0.8556	0.6956	0.6364	0.5552	0.3968
			B	0.6171	0.7515	0.4230	0.2350	0.1981	0.1450	0.0130
BQTerrace	832× 480	100	A	0.9381	0.9828	0.8360	0.7096	0.6792	0.6372	0.5276
			B	0.5104	0.8730	0.5420	0.4416	0.3260	0.3060	0.2850
Cactus	832× 480	100	A	0.9323	0.9819	0.8561	0.7780	0.6105	0.5331	0.3899
			B	0.5161	0.7654	0.4127	0.4029	0.3699	0.3425	0.2645
Chipmunks	832× 480	100	A	0.9588	0.9838	0.8784	0.7951	0.6973	0.5676	0.4054
			B	0.4500	0.6169	0.4831	0.4382	0.3708	0.2809	0.2360
Johnny	1280× 720	100	A	0.9245	0.9716	0.7513	0.6980	0.6217	0.4227	0.3108
			B	0.4474	0.8967	0.4088	0.3558	0.2802	0.2407	0.1835
Kristen& Sara	1280× 720	100	A	0.8804	0.9848	0.8265	0.6766	0.6052	0.4801	0.3316
			B	0.4139	0.8935	0.4019	0.3292	0.2729	0.2178	0.1974
FourPeople	1280× 720	100	A	0.9134	0.9871	0.8182	0.7838	0.6237	0.5054	0.3871
			B	0.4400	0.8384	0.5443	0.4722	0.4401	0.3642	0.2430
shields	1280× 720	100	A	0.9493	0.9959	0.9024	0.8608	0.7143	0.5444	0.4126
			B	0.6092	0.6837	0.5408	0.4694	0.3980	0.2959	0.2633
HoneyBee	1920× 1080	100	A	0.9794	0.9873	0.8529	0.7353	0.5412	0.4706	0.2059
			B	0.4500	0.5500	0.4370	0.3943	0.3240	0.3024	0.2200
YachtRide	1920× 1080	100	A	0.9501	0.9985	0.9144	0.7884	0.6256	0.5456	0.3228
			B	0.5154	0.8434	0.5005	0.4516	0.3711	0.3463	0.2520
PeopleOnStreet	2560× 1600	100	A	0.9301	0.9923	0.8254	0.6921	0.6201	0.5051	0.4153
			B	0.4592	0.8930	0.4630	0.4080	0.3650	0.2640	0.2312
Jockey	3840 × 2160	100	A	0.9232	0.9967	0.8333	0.7143	0.5574	0.3133	0.2167
			B	0.5107	0.8232	0.4753	0.3266	0.2172	0.1096	0.0845
ReadySteadyGo	3840 × 2160	100	A	0.9198	0.9939	0.8833	0.7714	0.5410	0.3097	0.2601
			B	0.4811	0.8671	0.4451	0.2998	0.1811	0.1006	0.0776

The robustness of proposed scheme is evaluated against different image pro-

Table 5.3: ***Extracted Watermark Correlation of the Proposed Scheme for Different α Values.***

α values	Extracted watermark correlation after re-compression attack with QP 24							
	BQMall	BQTerrace	Carving	Chipmunks	Flowervase	Johnny	HoneyBee	PeopleOnStreet
1.0	0.6956	0.7096	0.7511	0.7951	0.7076	0.6980	0.7353	0.6900
1.2	0.7053	0.7142	0.7556	0.8100	0.7175	0.7106	0.7523	0.7101
1.4	0.7156	0.7188	0.7601	0.8249	0.7274	0.7232	0.7692	0.7423
1.6	0.7341	0.7280	0.7742	0.8272	0.7437	0.7482	0.7778	0.7602
1.8	0.7418	0.7577	0.7750	0.8380	0.7494	0.7621	0.7833	0.7836
2.0	0.7423	0.7692	0.7785	0.8405	0.7514	0.7636	0.7836	0.7897

cessing attacks such as the addition of *salt and pepper noise*, *the addition of Gaussian Noise*, *Gaussian filtering*, *scaling* and *contrast change*. Table 5.4 shows the correlations between the original watermark and the watermark extracted from the attacked video sequences. It can be observed that the proposed scheme is robust against these common image processing attacks and performs better than the existing scheme [57]. The performance of the proposed scheme is also tested against video attacks such as temporal frame averaging, frame dropping and transcoding. In this chapter, the temporal frame averaging attack is performed by averaging a frame with its two nearest neighbor frames to replace the current one. From Table 5.4, it can be concluded that proposed scheme is robust against temporal frame averaging. Next, the robustness of the proposed scheme against frame dropping attack has been analyzed. Similar to the work [89], a 15 bit binary sequence is embedded as a synchronization sequence before the watermark sequence to identify the watermarked frame in case of frame dropping attack. At the time of extraction, the watermark frame is first identified using the synchronization sequence. Then, the watermark is extracted from the identified watermarked frame. The extracted correlation in case of 25% of random frame dropping has been presented in Table 5.4. In the proposed scheme, the watermark correlation is only affected due to loss of intra frames in the frame dropping attack. Video transcoding attack is next considered. As H.264/AVC is one of the most popular format, the transcoding attack is performed using it. First the watermarked video is transcoded using H.264/AVC for different QP values and GOP structure of I-P-P-P. Then, the H.264/AVC video is re-compressed

with H.265/HEVC at QP 16 and the watermark is extracted from it. The result shows that the proposed scheme is robust against the transcoding attack using H.264/AVC.

Table 5.4: ***Watermark Correlation against Different Attacks of Proposed Scheme (Denoted by A) and Chang et al. Scheme [57] (Denoted by B)***

Attack Type	Parameters		BQMall		Cactus		Jonny		YachtRide		People		Average	
			A	B	A	B	A	B	A	B	A	B	A	B
salt & pepper Noise	Noise strength													
	0.01		0.840	0.538	0.856	0.557	0.906	0.632	0.912	0.625	0.882	0.597	0.868	0.580
	0.02		0.746	0.443	0.744	0.474	0.805	0.505	0.824	0.510	0.767	0.435	0.756	0.462
Gaussian Noise	Mean	Variance												
	0	0.0001	0.853	0.447	0.831	0.446	0.854	0.571	0.875	0.543	0.854	0.494	0.849	0.482
	0	0.0002	0.837	0.392	0.818	0.402	0.816	0.526	0.839	0.468	0.792	0.461	0.814	0.458
	0	0.001	0.809	0.280	0.733	0.361	0.771	0.429	0.781	0.388	0.748	0.397	0.733	0.397
Gaussian Filter	Filter Size	Sigma												
	3×3	0.3	0.845	0.516	0.886	0.546	0.891	0.686	0.891	0.607	0.883	0.627	0.882	0.571
	3×3	0.5	0.840	0.433	0.832	0.506	0.774	0.421	0.842	0.419	0.791	0.385	0.809	0.419
	5×5	0.3	0.883	0.515	0.882	0.543	0.891	0.658	0.891	0.605	0.882	0.602	0.874	0.565
	5×5	0.5	0.874	0.442	0.822	0.506	0.774	0.403	0.806	0.410	0.785	0.368	0.808	0.406
Scaling	1.2		0.820	0.429	0.897	0.423	0.843	0.474	0.876	0.476	0.888	0.453	0.870	0.432
	0.9		0.831	0.436	0.909	0.430	0.856	0.482	0.887	0.484	0.899	0.468	0.867	0.452
	0.7		0.688	0.342	0.691	0.278	0.725	0.289	0.653	0.317	0.702	0.316	0.707	0.314
Contrast Change	0.1		0.889	0.537	0.916	0.526	0.875	0.622	0.906	0.484	0.943	0.543	0.893	0.520
	0.2		0.808	0.462	0.849	0.443	0.857	0.546	0.842	0.417	0.878	0.468	0.837	0.448
	0.3		0.786	0.429	0.791	0.402	0.814	0.467	0.812	0.386	0.846	0.434	0.805	0.415
Frame Average			0.796	0.477	0.779	0.423	0.833	0.511	0.745	0.479	0.792	0.463	0.792	0.458
Frame Dropping	25%		0.8589	0.3098	0.8751	0.2588	0.8482	0.2942	0.7827	0.2860	0.8527	0.2649	0.8616	0.3165
Trans-coding Using H.264/AVC	QP													
	16		0.871	0.453	0.870	0.416	0.939	0.543	0.912	0.425	0.886	0.437	0.896	0.469
	20		0.853	0.333	0.809	0.251	0.882	0.294	0.882	0.318	0.818	0.231	0.843	0.243
	24		0.830	0.196	0.743	0.202	0.842	0.236	0.759	0.262	0.745	0.185	0.782	0.192
	28		0.783	0.120	0.725	0.091	0.720	0.107	0.709	0.161	0.694	0.139	0.705	0.128

5.3.2 Visual Quality

As discussed in Sec. 5.2.3, the watermark bit is embedded in the pixels which do not participate in the intra prediction process. Thus the drift error is prevented for intra frames. After watermark embedding in intra frame, the drift error is resisted for the inter frame by varying the motion vector. In each TB, only one

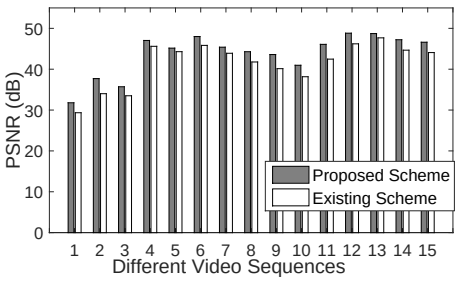
(a) Original Frame

(b) Watermarked Frame

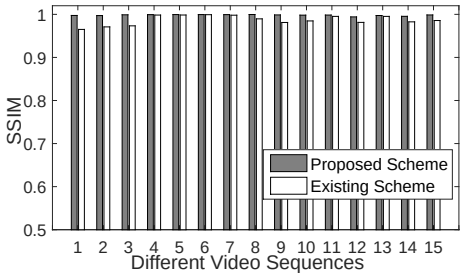
(c) Original Frame

(d) Watermarked Frame

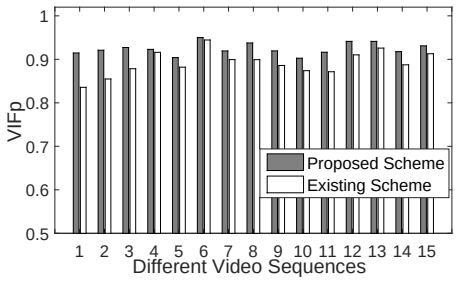
Figure 5.9: *Cover frame of Johnny and Flowervase video sequences.*



(a) PSNR

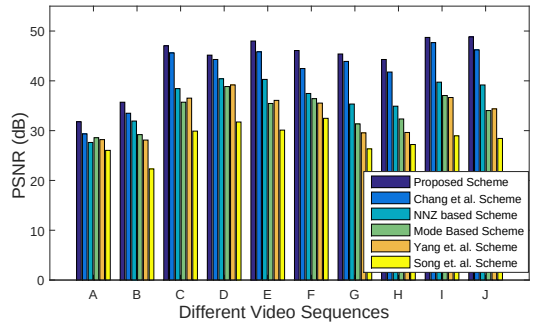


(b) SSIM

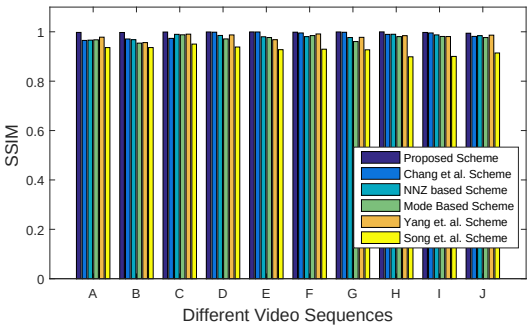


(c) VIFp

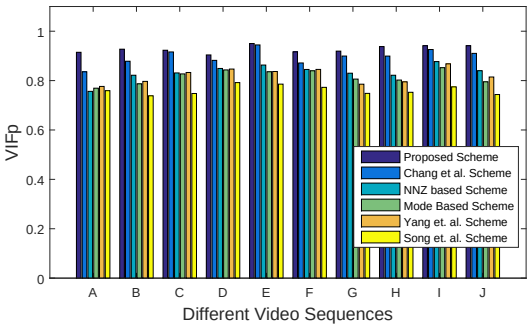
Figure 5.10: *Comparison of the PSNR, SSIM index, VIFp measures of the proposed scheme with Chang et al. scheme [57]. The video sequence index is same as shown in Table 5.1*



(a) PSNR



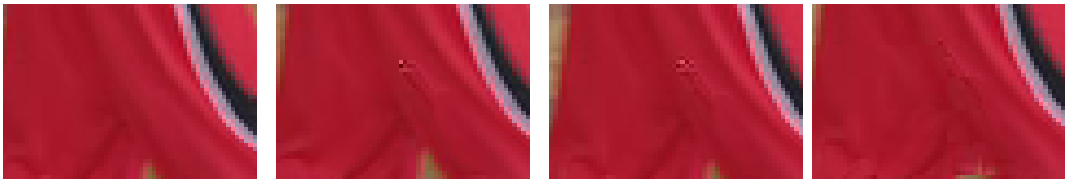
(b) SSIM



(c) VIFp

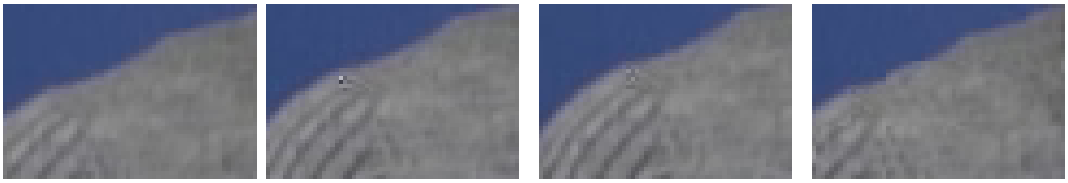
Figure 5.11: *Comparison of visual quality of the proposed drift compensated scheme with Chang et al. scheme [57], NNZ based scheme proposed in Chapter 4, mode based scheme proposed in Chapter 3, Yang et al. scheme [34] and Song et al. Scheme [45]. The video sequences are, A: BasketBall, B. BQMAll, C. BQTerrace, D. Cactus, E. Chipmunks, F. HoneyBee, G. Johnny, H. Kristen & Sara, I. PeopleOnStreet, and J. YachtRide*

bit is embedded to achieve minimal visual distortion. An original intra frame and the corresponding watermarked frame of the *Flowervase* and the *Johnny*



(a) Original P Frame (b) Reference Watermarked I Frame (c) P Frame with Drift Error (d) Drift Compensated P Frame

Figure 5.12: *Depiction of the P frame drift compensation process of proposed scheme on 1^{st} I frame and 2^{nd} P frame of Basketball video sequence. The watermark is embedded by altering last five high frequency coefficients of a 4×4 TB.*



(a) Original P Frame (b) Reference Watermarked I Frame (c) P Frame with Drift Error (d) Drift Compensated P Frame

Figure 5.13: *Depiction of the P frame drift compensation process of proposed scheme on 5^{st} I frame and 6^{nd} P frame of Kristen & Sara video sequence. The watermark is embedded by altering last five high frequency coefficients of a 4×4 TB.*

video sequences are shown in Figure 5.9. It can be observed that there exist hardly any visual traces for watermark embedding. The average PSNR, SSIM index and VIFp [104] [105] over all embedded intra frames in a video sequence are used to measure visual quality of the watermarked video. These metrics of the watermarked video are calculated with respect to original YUV video sequence using the Video Quality Measurement Tool (`http://mmspg.epfl.ch/vqmt`). Figure 5.10 presents the comparison of the average PSNR, average SSIM and average VIFp between the proposed scheme and Chang et al. scheme [57]. The results shows that the proposed scheme shows better visual quality. Further, the visual quality performance of all proposed schemes (the drift compensated scheme proposed in this chapter, the NNZ based scheme proposed in Chapter 4, the mode based scheme proposed in Chapter 3) and existing schemes [57] [34] [45] has been compared in Figure5.11. The results show that proposed drift

compensated watermarking scheme achieves better visual quality.

The performance of inter frame drift compensation process is shown in Figure 5.12 and Figure 5.13 for two video sequences. For more clarity, a large amount of distortion is added to the intra frame TBs by altering the last five high frequency coefficients of TBs during watermarking. In Figure 5.12(c) and Figure 5.13(c), the drift error in P-frame can be observed. Figure 5.12(d) and Figure 5.13(d) depict the drift compensation in an inter frame using the proposed algorithm. In Table 5.5, the improvement of the visual quality of inter frames using proposed scheme is given in terms of the PSNR, SSIM index and VIFp. From these results, it can be concluded that proposed scheme compensates the drift error in inter frames.

Table 5.5: ***Improvement of visual quality for P-frames before and after P-frame drift compensation***

Video Sequence	Before P Frame Drift Compensation			After P Frame Drift Compensation		
	PSNR (dB)	SSIM	VIFp	PSNR (dB)	SSIM	VIFp
BasketballPass	35.97280	0.99709	0.90971	37.63724	0.99726	0.91384
BlowingBubbles	37.47983	0.99653	0.82101	38.95879	0.99706	0.83096
BQMall	43.72958	0.99886	0.90997	44.25853	0.99920	0.92113
BQTerrace	43.66774	0.99926	0.91608	44.47284	0.99938	0.92251
Cactus	43.72504	0.99935	0.91197	44.09318	0.99945	0.92836
Chipmunks	47.55823	0.99941	0.92271	48.25518	0.99945	0.94448
Johnny	42.47064	0.99857	0.89587	44.32513	0.99906	0.91809
Kristan& Sara	47.75353	0.99973	0.94453	49.02036	0.99981	0.96993
FourPeople	43.30264	0.99729	0.91944	44.78824	0.99803	0.93507
shields	40.66503	0.99794	0.89152	41.81662	0.99806	0.92705
HoneyBee	46.43374	0.99357	0.86442	48.45063	0.99643	0.88858
YachtRide	47.46875	0.99416	0.88504	49.29985	0.99661	0.91524
PeopleOnStreet	46.99560	0.98999	0.91898	47.93990	0.99928	0.92911
Jocky	45.53788	0.99317	0.88871	47.37732	0.99531	0.90318
ReadySteadyGo	45.16374	0.99807	0.91083	46.08856	0.99835	0.92092

5.3.3 Bit Increase Rate

In the proposed scheme, the watermark is embedded in the 4×4 TBs whose NNZ values are quite high (as discussed in Sec. 5.2.3). Thus, NNZ values of 4×4 TBs do not vary much due to watermark embedding. The BIR due to embedding is very less for the proposed scheme. In Figure 5.14, the BIR is compared with that for the Chang et al. scheme [57]. The embedding is done for different video sequences with a QP=16 and an embedding rate of 100 bits/I frame. It can be observed that the BIR is negligible and is similar to the Chang et al. scheme [57]. Similar to the present scheme, the transform coefficients are altered in NNZ based watermark embedding proposed in previous chapter. The BIR of NNZ based scheme is also compared with proposed drift compensated scheme. The results show that the proposed drift compensated watermarking scheme achieve better BIR performance compared to the NNZ based scheme proposed in the previous chapter as the it does not alter zero valued transformed coefficients. Additionally, a text file for the location map with size around 300 bytes/GOP size is needed at the time of extraction in proposed scheme. The size of the location map is negligible science it is observed as 0.1% of the compressed bit stream.

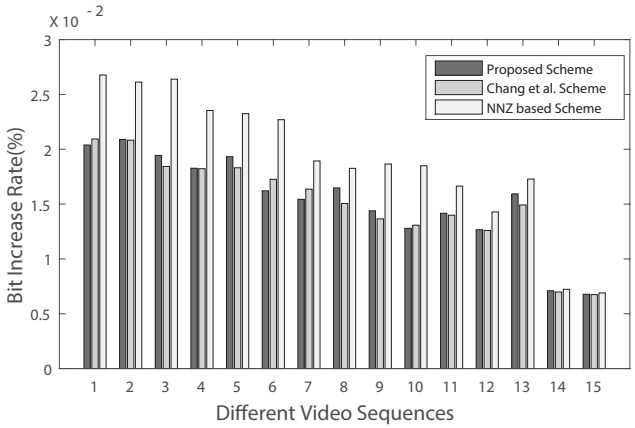


Figure 5.14: *Comparison of BIR of the proposed approach with Chang et al. scheme [57]. The video index is same as shown in Table 5.1*

5.3.4 Discussion

During re-compression, first the compressed H.265/HEVC bit stream is uncompressed into a raw video sequence using H.265 decoder. Then, this raw video sequence is compressed again using different QP values. During these operations, the transform coefficients and the block structure may be altered to create more compressed bit stream. Chang et al. [57] proposed a drift compensated watermarking scheme for H.265/HEVC videos where the triplet of DST coefficients in a 4×4 TBs are perturbed by ± 1 for embedding one bit of watermark. As the amount of alteration in the DST coefficients is very minor in the existing scheme [57], the scheme is fragile against re-compression and other common image and video processing attacks. As discussed in Sec. 5.1, this scheme may be extended for drift compensated robust watermarking for H.265/HEVC videos by altering the triplet of coefficients by a larger amount. But this extension causes larger amount of distortions in the spatial domain and causes visual artifacts as shown in Figure 5.1. So, this modification is not suitable for H.265/HEVC videos. In the proposed scheme, the watermark is embedded in 4×4 TBs of H.265/HEVC video sequences. The drift compensation is achieved in such a way that not only an acceptable visual quality has been maintained but the proposed scheme is also robust against re-compression and common image processing attacks. The careful choice of the embedding coefficients (those are not participating in intra prediction process) and the amount of embedding alteration may be an intuitive reason that the proposed scheme outperforms the recent existing schemes.

5.4 Time Complexity Analysis

In the proposed scheme, the watermark is embedded into 4×4 TBs of intra frame and the drift error is prevented in both intra and inter frames. As discussed in Sec. 5.2.1, the drift error is resisted at the time of embedding for intra frames. To embed the watermark, all 4×4 TBs in a intra frame are processed only once. After selecting a 4×4 TB, embedding can be done in a constant time (please ref.

to Algorithm 8). If the number of intra frames in a video sequence is f_I and the maximum number of 4×4 TB in a intra frame is n_{LTB}, the time complexity of intra frame drift compensated embedding is $\mathcal{O}(f_I \times n_{LTB})$. The drift compensation for P frames is done in two steps. First, the homogeneous regions are detected in I frames using Algorithm 5. This process has time complexity of $\mathcal{O}(f_I \times n_{PB})$ where n_{PB} is the maximum number of PB in a frame. In the second step, inter frame drift compensation is done for PBs in inter frame using Algorithm 6. This drift compensation process has the time complexity of $\mathcal{O}(f_P \times n_{PB})$ where f_P is the number of inter frames in a video sequence. The total time complexity can be written now as $\mathcal{O}(f_I \times n_{LTB} + f_I \times n_{PB} + f_P \times n_{PB})$. In the case of a video sequence, the number of inter frame is greater than the intra frame (i.e. $f_P > f_I$) and the number of 4×4 TBs is less than total number of PBs in a frame (i.e. $n_{LTB} < n_{PB}$). Therefore, the worst case time complexity can be written as $\mathcal{O}(f_I \times n_{LTB} + f_I \times n_{PB} + f_P \times n_{PB}) \simeq \mathcal{O}(f_P \times n_{PB})$. On the other hand, the existing scheme [57] embeds watermark in 4×4 and 8×8 TBs of intra frames. So, total time complexity is $\mathcal{O}(f_I \times N')$ where N' is the maximum number of 4×4 and 8×8 TBs in an intra frame . So, the processing time is higher for the proposed scheme as it also handles the inter drift prevention.

5.5 Summary

In this chapter, a robust watermarking scheme is proposed for H.265/HEVC bit stream with low visual distortion and low BIR. One of the major contributions of this work is to prevent the drift error in both of intra and inter frames. The intra drift error prevention is achieved by excluding those residual coefficients which are used in the intra prediction process. The inter frame drift error has been prevented by excluding the watermarked blocks from the inter prediction process. Rather a block having high spatial correlation with the watermarked block is used for the inter prediction process. So, the embedding noise can not propagate to the predicted blocks. The embedding capacity can be increased by altering more

pairs of DST coefficients per block. The experimental results show that proposed scheme is also robust against different noise addition and re-compression attacks and outperforms the related existing scheme. In the next chapter, the scope of this dissertation is extended to the watermarking of uncompressed HD video and a robust watermarking scheme is proposed against cam-cording attack.

Chapter 6

A Robust Watermarking Scheme for HD videos against Camcording Attack

With the recent advancement in the display devices and the camcording devices, HD and beyond-HD videos can be copied efficiently using advanced camcording devices. Thus, the camcorder theft is one of the biggest problems that the film industries are facing and it is one of the largest sources of video piracy. In cam-corded videos (i.e. camcorder recorded videos), the embedded watermark suffers from geometric, projective and temporal modifications due to digital to analog and then analog to digital conversions [95]. Additionally, different noise elimination filters are applied to these recorded videos for improving quality. These attacks cause de-synchronization of the watermark in cam-corded videos. Thus, the camcorder based video copying becomes a serious threat over digital watermarking based authentication system.

As in the literature survey, several schemes are proposed to handle the attacks during the camcording. Most of the schemes alter the mean luminance value in a group of consecutive frames to embed the watermark. Do et al. [88] altered the frame luminance by adding/subtracting a watermark pattern based on the pixel-value histogram. The pattern is subtracted/added in a group of consecutive frames (named as Watermarking Group of Pictures or WGP) to embed one bit of watermark information. In a similar direction, Li et al. [89] recently proposed an-

other watermarking scheme against camcording attack by altering the luminance relationship of the consecutive frames. In this scheme, each frame is divided into 8×8 blocks and the DC coefficients of those blocks are altered to embed the watermark. The Watson model [40] is used to lower the flickering effect due to embedding. Since, the large number of pixels are altered to embed the watermark, the visual quality of watermarked video may degrade in these schemes [87–89]. A few of the existing watermarking schemes also restore the geometrically distorted videos using different geometric transforms such as affine, bilinear, curved transforms etc. and extract watermark from it. Lee et al. [91] proposed a watermarking scheme in which the local auto-correlation function (LACF) is used to estimate the projective distortion in a cam-recorded video. Then, the watermark pattern is restored by applying the inverse projective transform on distorted videos and the watermark is detected from those restored videos. In [86], the authors have extended the above scheme [91] to the composite projective distortion. However, it is difficult to estimate the watermark as the peaks of the LACF are not clear after several signal-processing attacks.

The above discussion shows that most of the existing camcording resilient watermark schemes only considers geometric distortion as one of the main distortions in the cam-recorded video. But the watermark can also suffer from desynchronization due to frame bending attacks, frame dropping, automatic gain control and automatic white balancing (Please ref to Sec. 6.1) during camcording. This chapter proposes a novel zone selection algorithm to find spatially and temporally distortion invariant regions of HD videos and use them to embed a robust watermark against camcording attack. In the next section, a brief overview of camcording distortions will be presented.

6.1 Camcording Distortions

The process of camcording of a digital video may cause different distortions in recorded video due to the digital to analog/ analog to digital conversions, the

(a) Original Frame (b) 10° horizontal projection (c) 5° vertical projection (d) composite projection

(e) Bayer Color Filter (f) Frame Blending (g) Automatic Gain Control (h) Automatic White Balance

Figure 6.1: *Different distortions are caused to the* 59^{th} *Frame of BasketBall video sequence due to camcording attack.*

position of the camcorder with respect to the display, the camera sensor subsampling, additional image processing inside the camera logic etc.. In [95], the authors have modeled the different types of distortions caused by camcording. These distortions are briefly discussed below.

6.1.1 Spatial Distortion

At the time of the video acquisition process using a cam-recorder, the video gets spatially distorted due to several reasons. The zooming in / zooming out functionality of the camcorder may change the scale of the video. Moreover, the position of the camcorder with respect to the screen and the lens distortion can affect the output video by non-linear geometric transforms. In [86], the authors have presented three different types of projective distortion i.e horizontal, vertical and composite projection (please refer to Figure 6.1(b), 6.1(c) and 6.1(d)) which are caused by the uneven scaling and slight rotation due to angle of the camera with respect to the projection. Apart from these geometric de-synchronizations, the inbuilt image processing functionality of camcording devices (such as *Automatic Gain Control* (AGC), *Automatic White Balance* (AWB) etc. also causes spatial

distortion. The AGC mechanism in a camcorder always tries to maintain the average luminance of a frame within certain predefined range. Thus, the average luminance of a frame of a video may suffer from drastic or gradual changes which may later become the reason for poor watermark extraction. The AWB [108] tries to adjust the image so that it looks as if it is taken under canonical light. In this process, it alters the color components of pixels such that the average color of the image is closer to natural. This process sometimes may create a bluish color cast on the stones as shown in Figure 6.1(h). Moreover, the most of the camcorders do not use a full size sensor which is capable of capturing each color channel for each pixel. Instead, each sensor measures the intensity of one color channel and the reaming colors for that pixel are interpolated using the neighbor pixels. An optical anti-aliasing filter is employed to reduce the sub-sampling artifacts. Such a sub-sampling color sensor which is called the Bayer color filter array, is shown in Figure 6.2. The distorted image after anti-aliasing filtering is shown in Figure 6.1(e). These pixel alterations caused by AGC, AWB, interpolation of sensors, may degrade the embedded watermark in the recorded video. There may be several image processing operations after the acquisition process to improve the visual quality of captured video. These operations may further impair the embedded watermark.

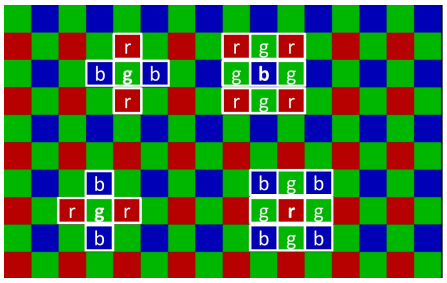


Figure 6.2: *A typical Bayer Colour Filter of camcording devices [95]*

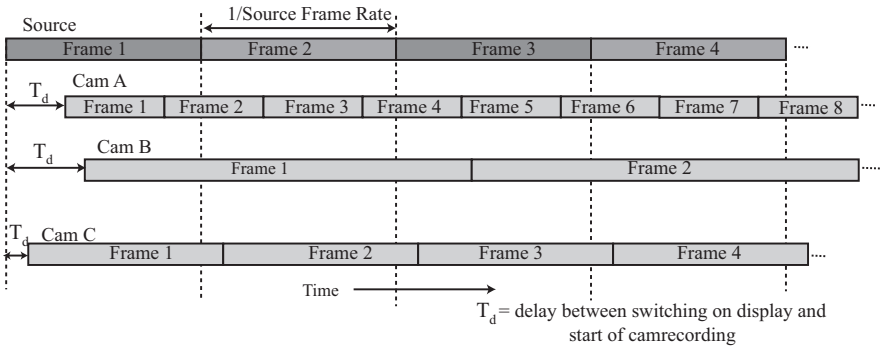


Figure 6.3: *Frame blending due to camcording devices, [95], Cam-A has a higher frame rate than the source frame rate, Cam-B has a lower frame rate than the source frame rate and Cam-C has same frame rate as the source frame rate*

6.1.2 Temporal Distortion

Temporal distortion is mainly caused by time de-synchronization between the camcording device and the display. At the time of camcording, the difference between the source frame rate and the recording frame rate creates the temporal distortion generally called *frame blending*. In Figure 6.3, the different situations of frame blending have been depicted where Cam-A has a higher frame rate than the source and Cam-B has a lower frame rate than the source. Even if the cam-recorder uses the same frame rate as the source video (in case of Cam-C), frame blending may occur. The slight timing variations between the camera's shutter opening and switching on the display, may cause de-synchronization between the source frame and the cam-corded frame. This de-synchronization results frame blending as depicted in Figure 6.3. Other types of temporal distortions can be caused due to frame dropping, frame swapping, frame insertion, frame averaging etc. during the camcording.

6.2 Proposed Scheme

It is observed that the cam-corded video mainly suffers from distortions due to frame blending and projective distortions. It is found in the literature that

the most the schemes [86, 90, 91] embed watermark frame by frame by varying luminance to resist the projective distortion. A few schemes [88, 89] are reported to resist the temporal distortion but these schemes are mainly resilient against frame dropping and frame averaging. Recently, a number of watermarking schemes [58, 69–72] are proposed to resist the geometric distortion. As, the temporal distortion during camcording may de-synchronize the watermark, these schemes are not directly applicable to resist the camcording attacks. The main goal of this chapter is to propose a scheme which is resilient against temporal distortions such as frame blending and spatial distortions such as projective distortions, AGC, image processing attacks etc..

Due to frame blending, each frame of cam-corded video may be constructed from several consecutive frames of the source video. If different watermarks are embedded in consecutive frames, the embedded signal is distorted due to overlapping of the frames. To overcome this, the same watermarking information needs to be embedded in a set of consecutive frames. In this case, such a group of consecutive frames should be correctly identified to avoid temporal de-synchronization noise at the time of watermark extraction. To achieve this temporal synchronization, the source video is segmented into small shots using geometric invariant features such that these shot segments are invariant to both geometric and temporal distortions. Since these shots can be identified even after temporal and geometric de-synchronization, the above problem can be solved if the same watermark is injected into all frames of a same shot. As a result, the watermark can be extracted from the cam-recorded video. In the proposed scheme, the Scale Invariant Feature Transform (SIFT) [96] is used to find the geometric invariant features of a video.

6.2.1 Resilience against Temporal Distortion

To extract the watermark information, the cam-corded video frames should be temporally synchronized with the original video frames. For a blind scheme, this

temporal synchronization should be achieved in such a way that the watermark can be extracted without the cover video. On the other hand, as discussed in Sec. 6.1.2, frame blending is a serious problem as it affects the watermark even if the video is recorded with a same frame rate. In the proposed scheme, the temporal synchronization is achieved based on the fact that the frames within a close neighborhood are highly correlated if there are no scene changes. These neighboring frames with high correlations can be grouped together to form a segment using some video features which are robust against the spatial distortion, such that the video segment remains unchanged during camcording. The reconstructed frames within this segment are almost similar to the original frame after frame blending because the neighbor frames within this segment are highly correlated. Thus, the watermark can be accurately extracted from these segments. The cam-corded video also suffers from luminance variation due to the inbuilt AGC property of the cam-corder. So, the pixel correlation based segmentation is not used in the proposed scheme. Instead, the SIFT feature points of a frame, which are invariant to the geometric distortion, are used to segment the source video. To make the segments invariant to the AGC, only robust SIFT points are chosen by keeping the *peak threshold* value quite high. To segment the video sequence, firstly, the SIFT features are obtained for each frame. Let SD_i denote the set of the SIFT descriptors for the interest points of $i\ th$ frame. Let $j\ th$ frame is the starting frame of the segment. By comparing SD_i and SD_j, a matching percentage p_i of the SIFT features is calculated. If the matching percentage p_i is greater than some threshold p_{th}, then $i\ th$ frame belongs to the same segment as the $j\ th$. Otherwise, the $i\ th$ frame is treated as the first frame of the next segment. The step by step segmentation algorithm is given in Algorithm 10

6.2.2 Resilience against Spatial Distortion

As discussed in Sec. 6.1.1, different spatial distortions such as the projective distortion, the luminance change and the geometric distortion, can occur in cam-

ALGORITHM 10: Video Segmentation(V)

Input: V:Input Video Sequence
Output: B: Segment Boundary
begin
    /* Initialize the parameters */
    Find the SIFT descriptor SD_1 of the first frame F_1 of video sequence V and save them in SD_m;
    m=1; /* m save the start index of segment */
    for *each frame* F_i *in remaining video Sequence* $\{V - F_1\}$ *where* i *represents frame index in* V **do**
        Find SIFT points and corresponding SIFT descriptor SD_i of F_i;
        calculated matching percentage p_i of SIFT feature descriptor SD_i and SD_m between current frame F_i and starting frame of segment F_m;
        if $p_i < p_{th}$ **then**
            /* p_{th} is threshold of matching percentage of SIFT descriptor */
            Save index pair $(m,\ i-1)$ in B as a segment;
            Clear SIFT descriptors saved in SD_m respectively;
            m=i; /* i is the start index of next segment */
            save the SIFT descriptor SD_i of F_i in SD_m;
    Return B;

corded videos. In the proposed scheme, the pixels in the local neighborhood of the SIFT points are used to embed the watermark. As, the SIFT points are robust against these spatial deformations, the embedding locations can be identified easily at the time of extraction from the spatially distorted cam-corded video. Thus, the watermark remains spatially synchronized with embedding locations and can be extracted from the cam-corded video.

6.3 Watermark Embedding Method

During camcording a display, the recorded video is mainly affected by frame blending and projection distortions. It is found in the literature that the most of the schemes are focused to resist geometric distortions. This work proposes a

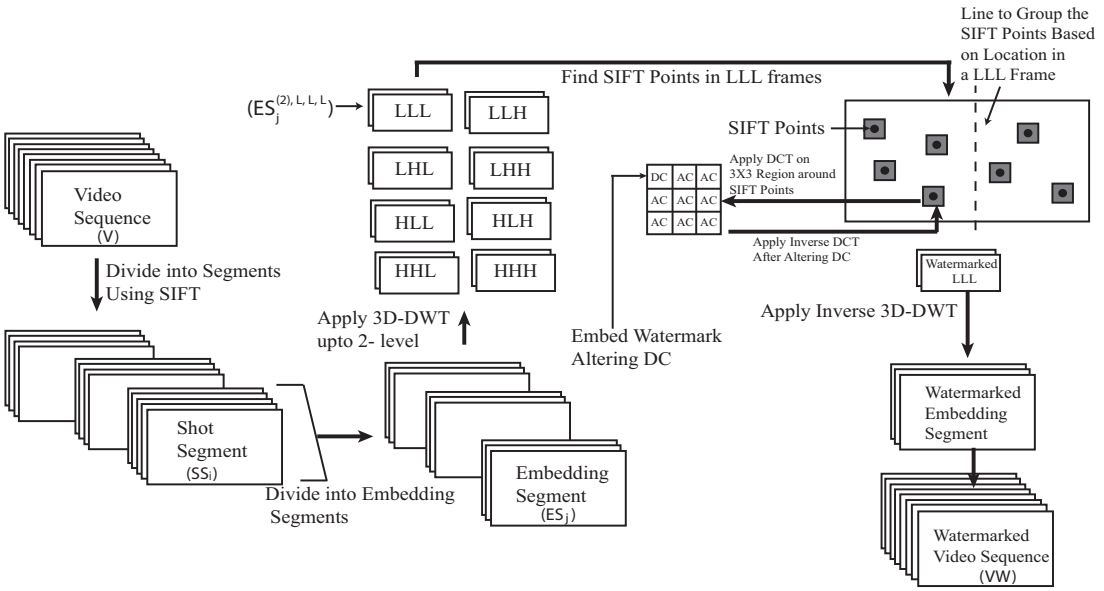


Figure 6.4: *Watermark embedding process*

watermarking scheme which is robust against both the projective distortion and frame blending. As discussed earlier, a cam-corded frame can be obtained using multiple frames of the source video due to frame blending when the frame rate of the cam-corded video is lower than that of the source video. On the other hand, if the camcorder frame rate is higher than the source frame rate, a source frame can occur multiple times in the recorded video. In both the cases, the frame by frame synchronization with the source video is not possible. Thus, the frame by frame watermarking may be distorted at the time of extraction. In the proposed scheme, a watermark bit is embedded in a set of frames such that the frame by frame synchronization is not required at the time of extraction. The watermark can be extracted from the cam-recorded video frames which are constructed using this set of frames of the source video. So, only the temporal synchronization is needed for a group of frames. To achieve temporal synchronization, the source video is divided into *shot segments* using Algorithm 10. The SIFT features having high sustainability against luminance variations and geometric distortions, are chosen for segmentation. These SIFT features can be selected by high *peak threshold*

and *edge threshold*. As, these shot segments are not changed after the camcording process, the watermark can be embedded in these shot segments to achieve temporal synchronization. In the proposed scheme, such a shot segment is used to embed multiple watermark bits. To make the scheme robust against temporal distortions, consecutive frames within a shot segment are used to embed a watermarking bit. This set of consecutive frames is called an *embedding segment*. The size of an embedding segment is selected based on the number of bits to be embedded and the size of the source video. Suppose,

SS_i: The *i th shot segment* of the source video V,

ES_j: The *j th* embedding segment of the shot segment SS_i,

N_{ES}: The number of frames in an *embedding segment*. So, the total number of watermarking bits n_i embedded in the shot segment SS_i is given by

$$n_i = \frac{\mid SS_i \mid}{N_{ES}}$$

where $\mid SS_i \mid$ denotes the number of frames in SS_i. From the cam-recorded video, a shot segment SS_i' is found out corresponding to SS_i. Each bit of watermark can be extracted from the changed embedding segment ES_j' which now consists of $\frac{|SS_i'|}{n_i}$ number of frames.

In this work, the watermark bit is embedded using the 3D discrete wavelet transform (3D-DWT) coefficients to exploit both the spatial and the temporal characteristics of the embedding segment. The 3D-DWT is a separable dyadic tree structured transform where a video segment of size $n_1 \times n_2 \times n_3$ is decomposed into multiple sub-bands based on variations both in the spatial and the temporal dimension. Given an embedding segment $ES_j[n_1, n_2, n_3]$, its one level 3D DWT is obtained by applying 1-D analysis filter banks, first on the rows, then on the columns, and finally on the temporal dimensions, thus obtaining the eight subbands $ES_j^{(1),a,\,b,\,c}[\frac{n_1}{2}, \frac{n_2}{2}, \frac{n_3}{2}]$ where a, b, and c refer to the lowpass (L) and to the high pass (H) representations of the given signal along the rows, columns, and the temporal dimension respectively. Thus, after l level decomposition of $ES_j[n_1, n_2, n_3]$, the subbands can be written as $ES_j^{(l),a,\,b,\,c}[\frac{n_1}{2^l}, \frac{n_2}{2^l}, \frac{n_3}{2^l}]$ and the low-

est frequency subband can be presented as $ES_j^{(l),L,L,L}[\frac{n_1}{2^l}, \frac{n_2}{2^l}, \frac{n_3}{2^l}]$. It is observed that generally the LLL subband remains more robust to various types of spatial and temporal distortions than any other sub band. Hence, the LLL subband of an embedding segment ES_j is chosen for embedding.

To make the scheme more robust, the 3D-DWT is applied up to 2-level on a selected embedding segment ES_j. Thus, the watermark is embedded by altering coefficients of $ES_j^{(2),L,L,L}[\frac{n_1}{4}, \frac{n_2}{4}, \frac{n_3}{4}]$. In these sub bands, each frame of size $\frac{n_1}{4} \times \frac{n_2}{4}$ contains information of n_3 number of frames. Thus, the watermark information in these scaled frames get distributed in n_3 frames after inverse 3D-DWT. To make the scheme robust against the spatial distortion, the SIFT is applied on each the n_3 frames. By matching the SIFT descriptors, common SIFT points are selected from these n_3 frames. In each scaled frame, the SIFT points are divided into two groups based on their locations in the frame. After that a 3×3 region is selected around the SIFT point for embedding. In the proposed scheme 3×3 region around the SIFT points is chosen for embedding based on the experimental results. If a smaller size block is selected, then the watermark is affected by image processing attacks and blocks larger than this selected size cause visible distortions in the embedded video.Suppose,

F_m: The $m\ th$ scaled frame of size $\frac{n_1}{4} \times \frac{n_2}{4}$ in $ES_j^{(2),L,L,L}[\frac{n_1}{4}, \frac{n_2}{4}, \frac{n_3}{4}]$.

P_k: The $k\ th$ SIFT point in F_m.

e_k: The energy of 3×3 block around P_k. e_k is found out by applying 2D-Discrete Cosine Transform (2D-DCT) on that 3×3 block.

$e(P_k)$: The function for calculating energy e_k.

$c_{u,v}$: The 2D-DCT coefficient at (u, v) position of 3×3 transformed block. Then, the energy e_k of P_k can be written as,

$$e_k = e(P_k) = \sqrt{\sum_{u=1}^{3}\sum_{v=1}^{3}(c_{u,v})^2} \tag{6.1}$$

Based on the position within a frame, the SIFT points P_ks are grouped into

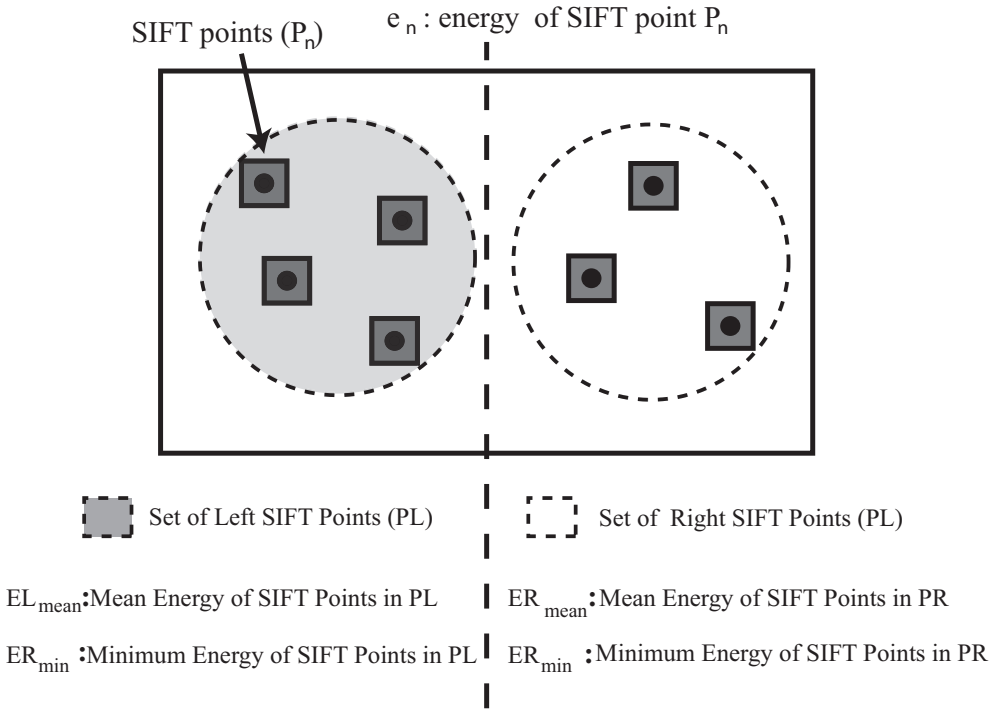


Figure 6.5: *A frame in low frequency LLL sub-band of ES_j*

two sets PL and PR (as shown in Figure 6.5). PL denotes the set of left SIFT points and PR denotes the set of right SIFT points. The mean EL_{mean} and the minimum energy EL_{min} of SIFT points in PL are calculated.

$$\begin{aligned} EL_{mean} &= \frac{1}{\mid PL \mid} \sum_{\forall P_k \in PL} e(P_k), \quad \text{and} \\ EL_{min} &= \min\{e(P_k) : \forall\ P_k \in PL\} \end{aligned} \tag{6.2}$$

Similarly, the mean ER_{mean} and the minimum energy ER_{min} of SIFT points in PR are calculated.

$$\begin{aligned} ER_{mean} &= \frac{1}{\mid PR \mid} \sum_{P_k \in PR} e(P_k), \quad \text{and} \\ ER_{min} &= \min\{e(P_k) : \forall\ P_k \in PR\} \end{aligned} \tag{6.3}$$

To embed the watermark bit, the energy e_ks around the SIFT points are altered. If the watermarking bit = 1, the energy around the left SIFT points are increased such that they become greater than the mean energy around the right SIFT

points. The embedding policy for watermarking bit = 1 of the proposed scheme can be described as follows,

$$\begin{aligned} &if(\text{Watermark bit} = 1)\{ \\ &\quad \forall P_k \in PL \quad \text{do the following}\{ \\ &\qquad \text{Calculate } e_k = e(P_k); \\ &\quad \text{Find difference } \delta_k = (e_k - ER_{mean}) \\ &\quad if(\delta_k \leq \delta_{th})\{e_k = e_k + \mid \delta_k \mid + \Delta_{th}\}\} \\ &\qquad \} \end{aligned} \tag{6.4}$$

δ_{th} is a positive value which is chosen based on the robustness of the watermark and the visual quality of the watermarked video. $\mid \delta_k \mid$ represents the absolute value of the difference between e_k and ER_{mean}. Δ_{th} is a positive value used to increase robustness of the watermark. Similarly, for embedding watermarking bit = 0, the energy around the right SIFT points are increased such that they become greater than the mean energy of around the left SIFT points. The embedding policy for the watermarking bit = 0 is,

$$\begin{aligned} &if(\text{Watermark bit} = 0)\{ \\ &\quad \forall P_k \in PR \quad \text{do the following}\{ \\ &\qquad \text{Calculate } e_k = e(P_k); \\ &\quad \text{Find difference } \delta_k = (e_k - EL_{mean}) \\ &\quad if(\delta_k \leq \delta_{th})\{e_k = e_k + \mid \delta_k \mid + \Delta_{th}\}\} \\ &\qquad \} \end{aligned} \tag{6.5}$$

The increase in the energy e_k by an amount ($\mid \delta_k \mid + \Delta_{th}$), is achieved by altering of the DC component of the 3×3 region. Only the DC components have been changed because the neighboring pixels are used for calculating SIFT feature points. Any change in the AC component will change the relative difference between neighbor pixels which will cause the change in SIFT feature points and as a result the watermark can be lost. After embedding the watermark, the

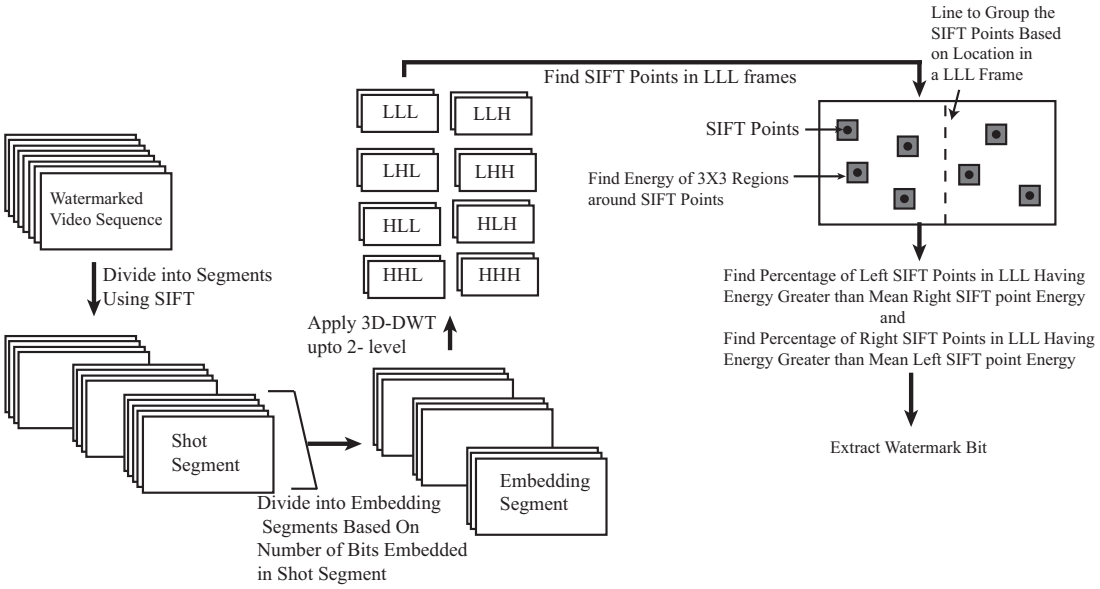


Figure 6.6: *Watermark extraction process*

inverse 3D-DWT is applied to get back the watermarked video sequence. The step by step embedding process is given in Algorithm 11.

6.4 Watermark Extraction Method

The extraction process is similar to the embedding scheme. The process is depicted in Figure 6.6. Based on the SIFT features, the shot segments are found out from the cam-recorded video. Then the embedding segments are obtained using the number of bits embedded in that segment. The 2-level 3-D DWT is applied on shot segments to find the LLL sub-band. Let ES'_j be the j th embedding segment of size $n'_1 \times n'_2 \times n'_3$. The size of the shot and the embedding segment may be altered due to frame blending during camcording. The LLL sub-band $ES_j^{'(2),L,L,L}$ is found out after two level of 3D-DWT on ES'_j. The robust SIFT feature points P_k in all scaled frames in $ES_j^{'(2),L,L,L}$ are found out. Suppose,

F'_m: The m^{th} scaled frame of size $\frac{n'_1}{4} \times \frac{n'_2}{4}$ in $ES_j^{'(2),L,L,L}$.

P'_k: The k^{th} SIFT point in F'_m.

e'_k: The energy of 3×3 block around P'_k. e'_k is found out by applying 2D-Discrete

ALGORITHM 11: Watermark Embedding(V, WS, N_{ES})

Input: V:Input Video Sequence, WS= Watermark Sequence, N_{ES}= Embedding Segment Size

Output: VW: Watermarked Video Sequence, NE:Number of bits embedded in each shot segment

begin

- Find shot segments using algorithm 10 of input video sequence V;
- **for** *Each shot segment* $SS_i \in V$ *where* i *denotes index of shot segment* **do**
  - Divide each segment depending upon embedding segment size N_{ES};
  - **for** *Each embedding segment* ES_j *in* SS_i *where* j *denotes index of embedding segment* **do**
    - Apply forward 3D-DWT up-to 2-level in selected embedding segment ES_j;
    - Select the low frequency sub-band $ES_j^{(2),L,L,L}$;
    - **for** *Each Frame* F_m *in low frequency sub-band* $ES_j^{(2),L,L,L}$ *where* m *denotes index of frame* **do**
      - Find robust rotation invariant feature points by applying SIFT in F_m ;
      - **for** *Each SIFT point* P_k *in* F_m *where* k *denotes index of SIFT point* **do**
        - Select 3×3 region around the SIFT point and apply 2D-DCT transform;
        - Find out energy e_k using DCT coefficients of selected 3×3 region;
      - Divide the SIFT feature points of F_m into two set PL and PR based on location as shown in Figure 6.5;
      - Calculate mean energy EL_{mean} and minimum energy EL_{min} of left feature points PL using Eq. 6.2;
      - Calculate mean energy ER_{mean} and minimum energy ER_{min} of right feature points RL using Eq. 6.3 ;
      - Select watermarking bit W_j from embedding watermarking sequence WS;
      - **if** $EL_{min} < ER_{mean}$ *and* $W_j == 1$ **then**
        - **for** *each left feature point* $P_k \in PL$ **do**
          - Select energy e_k of P_k ;
          - **if** $e_k < ER_{mean}$ **then**
            - Increase energy by modifying DC value of 3×3 region around the SIFT point P_k;
            - Apply Inverse 2D-DCT on modified DCT coefficients of 3×3 region and update values in F_m;
      - **if** $EL_{mean} > ER_{min}$ *and* $W_j == 0$ **then**
        - **for** *each right feature point* $P_k \in RL$ **do**
          - Select energy e_k of P_k ;
          - **if** $e_k < EL_{mean}$ **then**
            - Increase energy by modifying DC value of 3×3 region around the SIFT point P_k;
            - Apply Inverse 2D-DCT on modified DCT coefficients of 3×3 region and update values in F_m;
    - Apply inverse 3D-DWT and update the values in selected embedding segment ES_j;
  - Save the number of bits embedded in SS_i in NE;

Cosine Transform (2D-DCT) on that 3×3 block.

$e(P'_k)$: The function for calculating energy e'_k.

Based on the location in the frame, the SIFT points are grouped in two sets PL' and PR'. The energy e'_k of 3×3 region around the SIFT points P'_k are calculated based on the 2D-DCT coefficients of that region. Then, the mean energy EL'_{mean} for the SIFT points of PL' is calculated as

$$EL'_{mean} = \frac{1}{\mid PL' \mid} \sum_{\forall P'_k \in PL'} e(P'_k) \tag{6.6}$$

Similarly, the mean energy ER'_{mean} for the SIFT points of PR' is calculated as

$$ER'_{mean} = \frac{1}{\mid PR' \mid} \sum_{\forall P'_k \in PR'} e(P'_k) \tag{6.7}$$

Let $PL'_{(e'_k \geq ER'_{mean})}$ denote the set of SIFT points of PL' having an energy value greater than the right mean ER'_{mean}. Then, the ratio of SIFT points in PL' having energy value greater than the right mean ER'_{mean} in F'_m is given by,

$$rl_m = \frac{\mid PL'_{(e'_k \geq ER'_{mean})} \mid}{\mid PL' \mid} \tag{6.8}$$

For the embedding segment ES'_j, the total ratio of the left SIFT points having energy grater than the mean energy of right SIFT points are calculated as,

$$\eta_{Left} = \sum_{\forall F'_m \in ES'^{(2),L,L,L}_j} rl_m \tag{6.9}$$

Similarly, let $PR'_{(e'_k \geq EL'_{mean})}$ denote the set of SIFT points of PR' having greater energy value than the left mean EL'_{mean}. Then, the ratio of SIFT points in PR' having greater energy value than the right mean EL'_{mean} in F'_m is,

$$rr_m = \frac{\mid PR'_{(e'_k \geq EL'_{mean})} \mid}{\mid PR' \mid} \tag{6.10}$$

For the embedding segment ES'_j, the total ratio of right SIFT points having grater energy than the mean energy of left SIFT points are calculated as,

$$\eta_{Right} = \sum_{\forall F'_m \in ES'^{(2),L,L,L}_j} rr_m \tag{6.11}$$

Using η_{Right} and η_{Left}, the watermark bit is extracted from the embedding segment ES'_j based on the following condition,

$$\begin{aligned} &if(\eta_{Left} \geq \eta_{Right}) \\ &\quad \text{Watermark bit} = 1; \\ &else \\ &\quad \text{Watermark bit} = 0; \end{aligned} \tag{6.12}$$

The watermark extraction process is given step by step in Algorithm 12.

6.5 Experimental Results

The proposed method is tested on ten standard videos with different resolutions. To check the effectiveness the algorithm, it is tested on standard video sequences with different resolutions, textures and motions. Details of test sequences along with the experimental setup are listed in Table 6.1. In the proposed scheme, the watermark is embedded after segmenting the video into multiple shot segments by using Algorithm 10. Each of the shot segment is further divided into multiple embedding segments. Each of the embedding segment is used to embed a single watermark bit. As, the size of each shot segment may not be same, different numbers of bits can be embedded in each shot segment. A random binary bit sequence is used as the watermark sequence. The performance of the proposed watermarking scheme is evaluated in terms of two metrics: robustness and visual quality.

6.5.1 Robustness

As camcording video is mostly affected by the frame blending and the projection attack, the main objective of the proposed scheme is to resist these attacks. The required temporal synchronization to resist these attacks, is achieved in two steps. First, the use of shot segments which are robust against the geometric distortion and frame blending, maintains the synchronization of a cam-corded video with

ALGORITHM 12: Watermark Extraction(VW, NE)

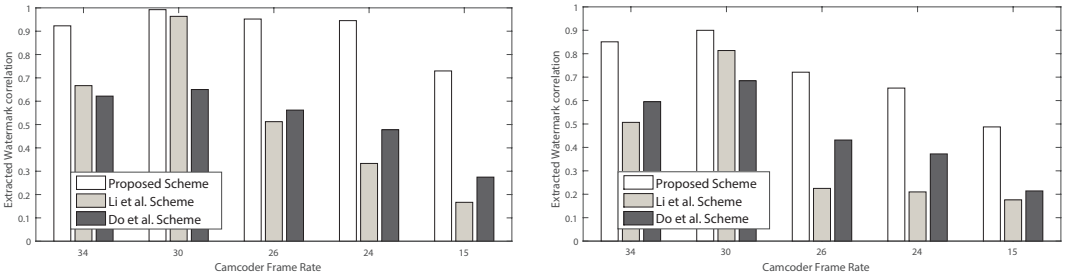
Input: VW: Watermarked Video Sequence, NE:Number of bits embedded in each shot segment
Output: WE: Watermarked Extracted Sequence
begin
- Find Shot Segments Using Algorithm 10 of input video VW;
- **for** *Each shot segment* $SS'_i \in VW$ *where* i *is the index of shot segment* **do**
  - Find out number of bits embedded in SS'_i using NE;
  - Divide each shot segment into embedding segment depending upon number of bits embedded in it;
  - **for** *Each embedding segment* ES'_j *in* SS'_i *where* j *is the index of embedding segment* **do**
    - Apply forward 3D-DWT up-to 2-level in selected embedding segment ES'_j;
    - Select the low frequency sub-band $ES_j^{'(2),LLL}$;
    - $\eta_{Left} = 0$; `/* initialization of parameters */`
    - $\eta_{Right} = 0$;
    - **for** *Each Frame* F'_m *in low frequency sub-band* $ES_j^{'(2),LLL}$ *where* m *is the index of scaled frame* **do**
      - Find robust rotation invariant feature Points by applying SIFT Transform in F'_m ;
      - **for** *Each SIFT point* P'_k *in* F'_m *where* k *denotes index of SIFT point* **do**
        - Select 3×3 region around the SIFT point and apply 2D-DCT transform;
        - Find out energy e'_k using DCT coefficients of selected 3×3 region;
      - Divide the SIFT feature points of F'_m into two set PL' and PR' based on location;
      - Calculate mean energy EL'_{mean} of left feature points PL' using Eq. 6.6;
      - Calculate mean energy ER'_{mean} of right feature points RL' using Eq. 6.7;
      - $countleft = 0$; `/* initialization of parameters */`
      - $countright = 0$;
      - **for** *each left feature point* $P'_k \in PL'$ **do**
        - Select energy e'_k of P'_k ;
        - **if** $e'_k >= ER'_{mean}$ **then**
          - $countleft = countleft + 1$;
      - **for** *each right feature point* $P'_k \in PR'$ **do**
        - Select energy e'_k of P'_k ;
        - **if** $e'_k >= EL'_{mean}$ **then**
          - $countright = countright + 1$;
      - $\eta_{Left} = \eta_{Left} + (countleft/|PL|)$;
      - $\eta_{Right} = \eta_{Right} + (countright/|PR|)$;
    - **if** $\eta_{Left} \geq \eta_{Right}$ **then**
      - Extract the watermarking bit $W_j = 1$
    - **else**
      - Extract the watermarking bit $W_j = 0$
    - Save the W_j in extracted watermark sequence WE;

Table 6.1: ***Details of the Experimental Set Up***

Parameters for Watermarking	**Values Taken**
Video Sequence Used	1. Akiyo, 2.BQTerrace, 3.Cactus, 4.Flowervase, 5.Jonny, 6. Kristan& Sara, 7.shields, 8.HoneyBee, 9.YachtRide, 10.PeopleOnStreet
video resolutions :	352×288, 416×240, 832×240, 1280×720, 1920×1080, 2560×1600
Bit Depth	8 bit/pixel
Source Frame Rate	30 Frame per Sec.

the original video. Secondly, the embedding of a bit of watermark in the low frequency sub-band of consecutive frames makes the scheme more robust against frame blending. The use of the SIFT regions in the low frequency sub-band also makes the scheme robust against geometric distortion. Recently, Li et al. [89] proposed a watermarking scheme against camcording attack. In this scheme, the watermark is embedded by altering luminace of consecutive frames such that the watermark can survive the geometric distortion. If the camcording is done using a frame rate greater than the source one, the watermark can be extracted in this existing scheme [89] from the cam-corded video after down scaling in the temporal direction. Thus, the scheme [89] is also partially robust against frame blending. So, the proposed scheme is compared with the Li et al. scheme [89]. Do et al. [88] proposes another watermarking scheme against camcording attack by temporal modulation of the frames. A watermark pattern which is generated based on pixel histogram of the frame, is added or subtracted from the consecutive frames for temporal modulation. Based on the correlation of watermark pattern and the frame, the watermark is extracted from the cam-corded video. As, the pixel histogram is used to generate the watermarking pattern, the extraction process is free from geometric synchronization. This makes it partially robust against temporal distortion specially when camcording frame rate is higher than the source frame rate. This scheme [88] is also used in comparison. The result of the frame blending attacks at different frame rates in terms of the corelation

between the original and the extracted watermarks is given in Table 6.2. Similar results for the *Akiyo* and the *PeopleOnStreet* video sequences are plotted in the Figure 6.7 for more clarity. It can be observed from Table 6.2 and Figure 6.7 that the proposed scheme is more robust against the blending attack than the existing schemes [88, 89]. During camcording, the projection distortion is caused



(a) Akiyo Sequence (b) PeopleOnStreet Sequence

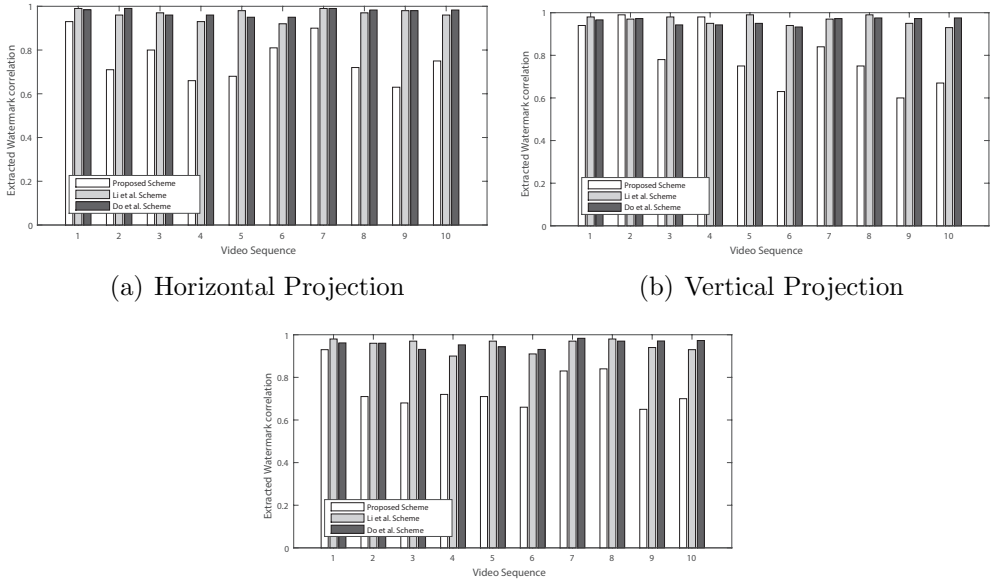
Figure 6.7: *Comparison of robustness of the proposed scheme with the existing schemes [89] [88] after frame blending attack for different video sequences.*

by recording the source video from different angles. Due to the projection attack, the frame is slightly rotated and unevenly scaled both in horizontal and vertical directions. The correlation between the embedded watermark and the extracted watermark after projection attack is shown in Figure 6.8. It can be observed from the Figure 6.8, the proposed scheme is quite robust against projective distortion due to the use of rotation and scaling invariant SIFT regions. In the existing schemes, the mean luminance [89] or pixel histogram [88] of frame is altered to embed the watermark. As the mean luminance or pixel histogram of a frame is quite robust against the geometric distortion, the existing schemes perform better if only projection attack occurs in camcording (as observed from the Figure 6.8). The frame blending attack is unavoidable during camcording because of the temporal de-synchronization caused by a slight time variation in shutter opening/closing time and switching on time of display. In Figure 6.9, the comparison is done when blending and projection occur together. These figures show that the proposed scheme performs better than the existing schemes [88, 89]

against these two main distortions occur together in cam-corded video. The proposed scheme is also robust to the video if it affected by the AGC characteristic of the cam-corder. The comparison result between the proposed scheme and the existing schemes after AGC attack is shown in the Figure 6.10. Due to AGC, the mean luminance of a frame may be changed. It may also change the pixel value differences in consecutive frames. So, the proposed scheme performs better than the existing schemes [88, 89].

Table 6.2: ***Comparison of Robustness of Proposed Scheme (Denoted by A) with Li et al. Scheme [89] (Denoted by B) and Do et al. Scheme [88] (Denoted by C) against the Frame Blending Attack***

Video Sequences	Source Frame Rate	Scheme Name	Extracted Watermark Correlation after Frame Blending Attack with Different Framerate				
			Camcoder Frame Rate 34	Camcoder Frame Rate 30	Camcoder Frame Rate 26	Camcoder Frame Rate 24	Camcoder Frame Rate 15
Akiyo	30	A	0.9230	0.9923	0.9521	0.9453	0.7297
		B	0.6667	0.9636	0.5123	0.3333	0.1667
		C	0.6218	0.6501	0.5619	0.4777	0.2747
BQTerrace	30	A	0.8554	0.9288	0.8624	0.8155	0.7416
		B	0.6269	0.9538	0.4874	0.4107	0.0881
		C	0.6572	0.7779	0.5113	0.4025	0.3031
Cactus	30	A	0.8108	0.8715	0.8333	0.7083	0.4334
		B	0.4693	0.9453	0.2162	0.2017	0.1486
		C	0.6818	0.7727	0.5091	0.4755	0.1901
Flowervase	30	A	0.7958	0.8235	0.7647	0.6470	0.5294
		B	0.6667	0.9372	0.2700	0.2163	0.1427
		C	0.6172	0.7098	0.4475	0.3951	0.2243
Johnny	30	A	0.7214	0.8118	0.7597	0.7056	0.6667
		B	0.7667	0.9722	0.3410	0.2953	0.2167
		C	0.6607	0.7658	0.4834	0.3965	0.2652
Kristan& Sara	30	A	0.8118	0.9029	0.7106	0.6642	0.6191
		B	0.6675	0.8375	0.2503	0.2109	0.1978
		C	0.7520	0.8444	0.5404	0.4280	0.3485
Shields	30	A	0.9098	0.9166	0.8809	0.8166	0.6286
		B	0.6175	0.8945	0.4511	0.3441	0.1697
		C	0.7299	0.8026	0.5659	0.4884	0.3415
HoneyBee	30	A	0.8136	0.8952	0.8264	0.7717	0.5682
		B	0.6113	0.9188	0.4921	0.4408	0.1966
		C	0.6693	0.7951	0.5126	0.4204	0.2977
YachtRide	30	A	0.8626	0.9342	0.7697	0.6832	0.4935
		B	0.5253	0.8474	0.2928	0.2607	0.1745
		C	0.6196	0.6917	0.4748	0.3714	0.2879
PeopleOnStreet	30	A	0.8508	0.8999	0.7211	0.6533	0.4875
		B	0.5069	0.8136	0.2250	0.2100	0.1760
		C	0.5952	0.6845	0.4315	0.3722	0.2142
Average	30	A	0.8355	0.8977	0.8081	0.7411	0.6010
		B	0.6125	0.9084	0.3538	0.2924	0.1677
		C	0.6605	0.7495	0.5038	0.4228	0.2747



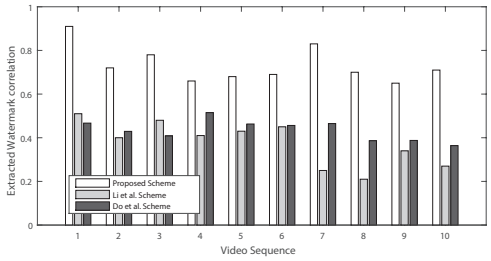
(a) Horizontal Projection

(b) Vertical Projection

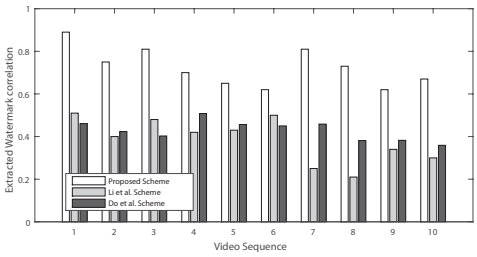
(c) Composite Projection

Figure 6.8: *Correlation of extracted watermark after only Projection attack (with angle* 3° *) for different video sequences. The video sequence index is same as depicted in Table 6.1*

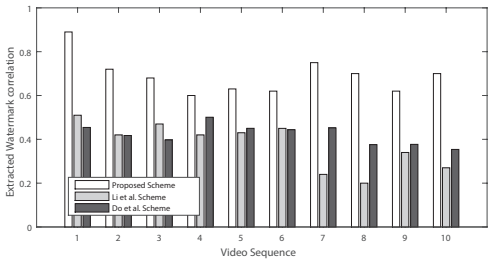
The robustness of the proposed scheme is also evaluated against different types of image processing attacks such as the addition of salt-&-pepper noise, scaling, luminance change, contrast change, median and Gaussian filtering. The correlations of the extracted watermark with the original watermark is presented in the Table 6.3. It can be observed that the proposed scheme is robust against these common image processing attacks. The performance of the proposed scheme is also tested against video attacks such as temporal frame averaging and frame dropping. In this chapter, temporal frame averaging attack is performed by averaging the current frame with its two nearest neighbor frames. From Table 6.3, it can be concluded that the proposed scheme is robust against temporal frame averaging. Next, the robustness of the proposed scheme against frame dropping attack has been analyzed. Similar to the work [89], a binary sequence is used to synchronize watermark sequence within a shot segment. The extracted correlation in case of 25% of random frame dropping has been tabulated in Table 6.3.



(a) Horizontal Projection



(b) Vertical Projection



(c) Composite Projection

Figure 6.9: *Correlation of extracted watermark under the combined projection attack (angle of projection is 5°) and frame blending attack (cam-recoding rate 24 frames/sec). The video sequence index is same as depicted in Table 6.1*

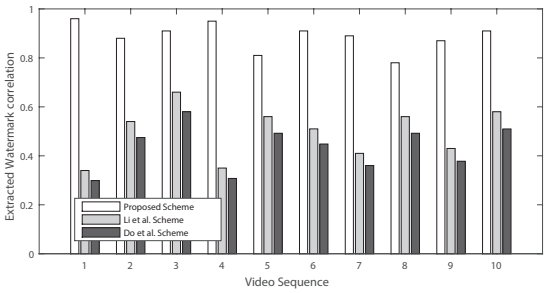


Figure 6.10: *Correlation of extracted watermark after Automatic Gain Control. The video sequence index is same as depicted in Table 6.1*

The robustness of the proposed scheme is tested against re-encoding attack by compressing the watermarked video with different quantization step size of H.265/HEVC encoder. The correlation of the extracted watermark from those compressed bit stream have been tabulated in Table 6.4. As H.265/HEVC is most recent compression standard with higher compression efficiency than pre-

Table 6.3: ***Watermark Correlation against Different Attacks for the Proposed Scheme***

Attack Type	Parameters		Basketball Pass	BQ Mall	Kristen & Sara	YachtRide	Average
salt pepper noise	Noise Strength						
	0.01		0.9730	0.7656	0.8540	0.9290	0.9059
	0.02		0.8649	0.6924	0.8080	0.8472	0.8262
Median Filter	Filter Size						
	3×3		0.9189	0.6486	0.7848	0.8393	0.8197
	5×5		0.8900	0.5946	0.6750	0.7703	0.7524
Gaussian Filter	Filter Size	Sigma					
	3×3	0.3	0.9730	0.7218	0.9590	0.9464	0.9244
	3×3	0.5	0.9730	0.6510	0.9459	0.9127	0.8929
	5×5	0.3	0.9439	0.6216	0.9109	0.9041	0.8632
	5×5	0.5	0.9189	0.6216	0.8919	0.8943	0.8491
luminance Change	20		0.9531	0.9266	0.9467	0.9584	0.9464
	10		0.9733	0.9359	0.9634	0.9745	0.9604
	-10		0.9733	0.9405	0.9597	0.9716	0.9674
	-20		0.9459	0.9259	0.9446	0.9470	0.9398
Scaling	0.5		0.8459	0.7278	0.8548	0.8469	0.8218
	0.7		0.8959	0.8089	0.8615	0.8970	0.8702
	1.2		0.9559	0.8785	0.9142	0.9237	0.9066
	1.5		0.9563	0.8785	0.9143	0.9203	0.8963
Contrast Change	0.1		0.9613	0.8334	0.8944	0.9054	0.9071
	0.2		0.9401	0.7825	0.8479	0.8534	0.8695
	0.3		0.9089	0.7103	0.8031	0.8379	0.8233
Frame Average			0.9525	0.8628	0.9048	0.8833	0.9143
Frame Dropping	25%		0.8222	0.7768	0.8033	0.8372	0.8150

vious encoders (such as H.264/AVC, MPEG-4 etc.), it is used to compress the watermarked video. The high correlation coefficient of the extracted watermark from highly compressed (with QP greater that 30) video advocates the decent robustness of the proposed scheme against the video compression attack.

6.5.2 Visual Quality

The visual quality of the watermarked video can be assessed by calculating the PSNR, the SSIM index and the VIFp [104] [105] of the watermarked video using

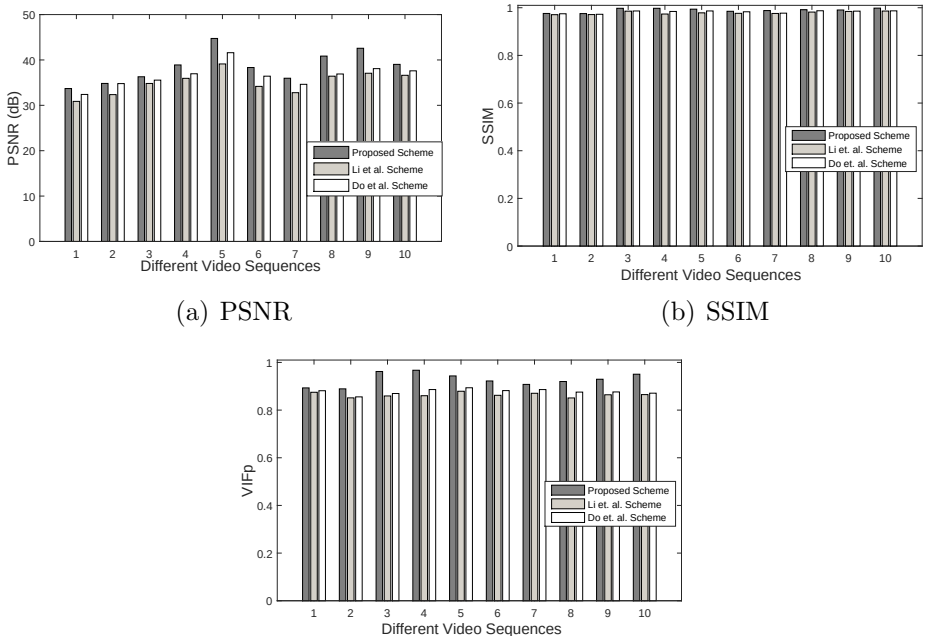
Table 6.4: ***Correlation of Extracted Watermark for Proposed Scheme against Compression using H.265/HEVC***

Compression QP	Akiyo	BQTerrace	Cactus	Flowervase	Jonny	YachtRide	People On Street
16	0.9731	0.9479	0.9259	0.8571	0.8787	0.9082	0.9445
20	0.9459	0.9186	0.8888	0.8014	0.8575	0.8871	0.9441
24	0.9189	0.8410	0.8518	0.8001	0.8507	0.8804	0.8881
28	0.9109	0.8333	0.8512	0.7491	0.7856	0.8159	0.8823
32	0.8919	0.8042	0.8148	0.7427	0.7852	0.8155	0.8821
36	0.8649	0.7967	0.8141	0.7342	0.7146	0.7488	0.8682

Video Quality Measurement Tool [103]. Due to watermark embedding in the low frequency components, the watermarked video gets slightly distorted. The visual quality can further be improved by decreasing the watermarking strength. The comparison of the PSNR, the SSIM index and the VIFp of the proposed and the existing schemes [88, 89] is shown in Figure 6.12. In the proposed scheme, the pixels are altered in very few SIFT regions of frames. But in the existing schemes [88, 89], the alteration of pixels occurs in most of the frame as the watermark is embedded by altering the mean luminance or the pixel histogram of a frame. This may be the intuitive reason that the proposed scheme shows higher visual quality than the existing schemes [88, 89]. The watermarked video frames and the original video frames of Cactus video sequence are also depicted in Figure 6.11. These figures show that the visual quality is not affected due to watermark embedding.

(a) Original Frame (b) Watermarked Frame

Figure 6.11: 8^{th} *frame of Original and watermarked Cactus video sequence.*



(a) PSNR (b) SSIM

(c) VIFp

Figure 6.12: *Comparison of the visual quality of Proposed Scheme with existing schemes [88, 89]. The video sequence index is same as depicted in Table 6.1*

6.5.3 Time Complexity Analysis

In the proposed scheme, first the input video is segmented into shots by comparing SIFT features of the frames. There are different stages for finding the SIFT features and corresponding descriptor of a frame [109]. These stages and the corresponding time complexity are given as follows. First, the Gaussian filter of size $w \times w$ is applied on each frame of size $M \times N$ to create scale space of that frame. The time complexity of creating s blurred images is $\mathcal{O}(h^2 \times w^2 \times s)$ where $h = max(M, N)$. Next, the difference between the blurred images are calculated with complexity $\mathcal{O}(h^2 \times s)$. To find the key point or interest point (SIFT points), the extrema within the difference are detected with time complexity of $\mathcal{O}(h^2 \times s)$. Only a small fraction of these extrema will qualify as keypoints. The time complexity for keypoint detection will be $\mathcal{O}(h^2 \times s)$. Further, to generate the SIFT descriptors, $2x \times 2x$ neighborhood pixels of the keypoint are processed with

time complexity $\mathcal{O}(h^2 \times s \times x)$. So, total time complexity of SIFT can be written as $\mathcal{O}(h^2 \times w^2 \times s + h^2 \times s + h^2 \times s + h^2 \times s \times x)$. The Gaussian kernel size w, the number of blurred images s in scale space and the number of neighborhood pixels x to create the feature vector, are fixed in the proposed scheme. So, the time complexity for finding the SIFT feature become $\mathcal{O}(h^2)$ [109]. Each SIFT feature of a frame is compared with another frame's SIFT features to find the similarity between two frames. This time complexity can be written as $\mathcal{O}(d^2)$ where d is the number of feature in a frame. Thus, the total time complexity of the Algorithm 10 is $\mathcal{O}(h^2 \times t + d^2 \times t)$. In the next phase, 3D-DWT is applied on embedding segment and the SIFT is applied in the low frequency components. The time complexity of 3D-DWT is $\mathcal{O}(h^2 \times t_1)$ where t_1 is embedding segment length. The worst case time complexity for applying SIFT and changing low frequency component of embedding segment is $\mathcal{O}(h^2 \times t_1 + d_1 \times t_1)$ where d_1 is total number of SIFT points in a low frequency frame of an embedding segment. So, the time complexity for watermark embedding in a single segment is $\mathcal{O}(h^2 \times t_1 + d_1 \times t_1)$. If t_2 number of embedding segments are presented in a video, then the total time complexity becomes $\mathcal{O}(h^2 \times t_1 \times t_2 + d_1 \times t_1 \times t_2) = \mathcal{O}(h^2 \times t + d_1 \times t)$. (as $t_1 \times t_2 =$ total number of frames in video $= t$). So, the time complexity becomes $\mathcal{O}(h^2 \times t + d^2 \times t + h^2 \times t + d_1 \times t)$. Now, for any videos, the usual number of SIFT descriptors is higher in the frame than its low frequency component i.e. $d > d_1$ and the frame resolution is higher than the number of descriptor in it i.e. $h > d$. So the time complexity of proposed scheme can be written as $\mathcal{O}(h^2 \times t)$. On the other hand, the existing scheme [89] divides each frame into small 8×8 blocks and DC coefficients of those blocks are altered to embed the watermark. So time required for processing a single frame can be written as $(\frac{(M \times N)}{64} \times C)$ where $(M \times N)/64$ is the number of the blocks in a frame and C is the time required to perform 2D-DCT on 8×8 blocks and altering coefficients for embedding. As, C is constant as the fixed size DCT is used, the time complexity for embedding in a video can be written as $O(h^2 \times t)$ where t represents total number of frames in a video. Do et al. embeds the watermark by finding pixel histogram and altering

each pixel value. So it also has the time complexity of $O(h^2 \times t)$. From the above discussion it can be observed that the proposed scheme has similar time complexity as existing schemes. But, the proposed scheme takes a little longer processing time as finding SIFT regions require higher time.

6.6 Summary

This chapter proposes a watermarking scheme which is robust against video camcording. From the literature, it has been found that there exists very few watermarking schemes which could survive frame blending which is an immediate effect of video camcording. In this work, a novel blind watermarking scheme is proposed which are robust against frame blending and projection attacks which are the main distortions caused during camcording. The required temporal synchronization is done by segmenting the video using geometric invariant SIFT features. The SIFT regions of the low frequency coefficients in both temporal and spatial directions are used for embedding. Thus, the scheme becomes robust against geometric and temporal distortions. The scheme is tested for various camcorder attacks such as frame blending, distortion by AGC and for different types of common image processing attacks. From the results, it can be observed that the proposed scheme is quite robust against these attacks. Further, the robustness of the proposed scheme is also tested for the re-compression attack.

Chapter 7

Conclusion and Future Works

7.1 Summary of the Contributions

The recent advancement in the communication technology and the availability of the cheaper and wider varieties of video playing devices have made HD and beyond HD video transmission more popular. With the emergence of such high definition video, the security issues like copyright protection, content authentication etc. have become important requirements. In this dissertation, the entire work is primarily motivated to propose robust watermarking solutions for both compressed and uncompressed HD representations. For HD video compression, the H.265/HEVC compression standard has emerged with the compression efficiency double of the previous standard H.264/AVC. Very few watermarking scheme exists in the literature for H.265/HEVC videos and those watermarking schemes fail to sustain against intentional or unintentional attacks such as re-encoding, image and video processing attacks etc.. It is also observed that the existing HD video watermarking schemes fail to sustain against the camcording attack. In this thesis, these aforementioned issues are considered and corresponding robust watermarking schemes are investigated. A brief summary of contribution is narrated as follows:

7.2 Prediction Mode Based H.265/HEVC Video Watermarking Resisting Re-compression Attack

In the first contributory chapter, a compressed domain blind watermarking scheme is proposed, which is robust against re-encoding attacks. In this scheme, the watermark is embedded by altering the intra prediction modes of selected 4×4 intra prediction blocks of the H.265/HEVC video. Though, several watermarking schemes exist for the H.264/AVC videos using the intra prediction mode, they are all fragile against the re-compression attacks. Moreover, the spatial correlation between prediction planes generated by two consecutive prediction modes in HEVC has become higher due to the presence of a large number of prediction modes. This increased correlation raises the probability of the mode change during re-compression. If modes based watermarking schemes of H.264/AVC are directly applied to H.265/HEVC videos, the watermark is likely to be lost during re-compression. To achieve robustness, the 4×4 luma PBs of the intra frames are first chosen for embedding based on sustaining probability after re-compression. Then, the intra prediction mode altering probability due to re-compression is determined. Based on these results, the intra prediction modes of H.265/HEVC are grouped in a fashion such that each group represents two watermark bits. The groups ensure that the mode change due to re-compression can be closed within a group. Experimental results on various test sequences show that the scheme is robust to re-compression and has a very low effect on the bitrate and the visual quality of the watermarked video.

7.3 Robust Watermarking of H.265/HEVC Videos using Temporally Homogeneous Blocks

In the next chapter, a compressed domain watermarking scheme is proposed which embeds the watermark by altering the NNZ of 4×4 transform blocks of the

HEVC video sequence. The performance of the watermarking scheme is improved in terms of the visual quality and robustness by using temporally homogeneous blocks in the consecutive intra frames. As, consecutive frames within a small temporal neighborhood have a very high correlation when no scene change is detected, many blocks can be found within a short temporal neighborhood having similar textures. Thus, they have similar NNZ values. Because of a very less NNZ difference, altering of transform coefficients due to watermark embedding is very minimal. This leads to very less degradation in the visual quality. In this scheme, these temporally homogeneous blocks are found out using the motion characteristics of the I frame which is determined by the motion information of the Inter (P or B) predicted frames of its close neighborhood. Then, a pair of temporally homogeneous blocks is selected from consecutive I-frames and the NNZ values of them are altered to embed the watermark. It is experimentally shown that the proposed scheme performs well against re-compression attack and image processing attacks while maintaining a decent visual quality.

7.4 Drift Compensated Robust Watermarking for H.265/HEVC Video Stream

Although the proposed scheme performs well in terms of robustness, the visual quality can be improved further by preventing drift error propagation. In this scenario, the visual quality of non watermarked regions degrades due to watermarking noise propagation during prediction. The existing drift error handling strategies for recent video standards such as H.264/AVC may not be directly applicable for the upcoming HD video standards (such as HEVC) due to different compression architectures. Specially, the use of DST transformation for 4×4 TBs in intra frame instead of the DCT restrict the use of H.264/AVC based drift compensated algorithm in H.265/HEVC. In this phase, a compressed domain watermarking scheme is proposed for H.265/HEVC bit stream, which can handle the drift error propagation both for intra and inter prediction processes. The intra

drift error prevention is achieved by excluding those residual coefficients which are used in the intra prediction process. The inter frame drift error has been prevented by excluding the watermarked blocks from the inter prediction process. Rather a block having high spatial correlation with the watermarked block is used for the inter prediction process. So, the embedding noise can not propagate to the predicted blocks. Additionally, the proposed scheme shows adequate robustness against re-compression attack as well as common image processing attacks while improves the visual quality of watermarked video with respect to the existing schemes.

7.5 A Robust Watermarking Scheme for HD videos against Camcording Attack

In this final contributory chapter, a blind watermarking scheme is presented, which is robust against the video camcording attack. Camcorder based video piracy is one of the biggest threats that the film industries are facing in recent time. During camcording, the captured video is mostly affected by the frame blending and the projection attacks which cause both spatial and temporal desynchronization of embedded watermark. Though, most of the existing camcording resilient watermarking are robust against spatial distortions, they are not robust against frame blending which is an immediate affect of camcording. In this work, the required temporal synchronization is achieved in two steps. First, the use of shot segments which are robust against geometric distortion and frame blending, maintains the synchronization of a cam-corded video with the original video. Secondly, the shot segment is divided into embedding segments and a bit of watermark is embedded into the low frequency sub-band of an embedding segment. Thus the use of low frequency in temporal direction makes the scheme more robust against frame blending. Further, the SIFT regions in the low frequency sub-band are used to make the scheme robust against the geometric distortion. The effectiveness of the scheme is experimentally verified against the camcording

attacks.

7.6 Future Works

The present study of this dissertation can be extended further in several directions as listed below:

- In the last contributory chapter, the proposed watermarking scheme is mainly restricted to the uncompressed domain watermarking in the case of camcording attack. Since uncompressed domain schemes are a bit slow because of decoding and further re-encoding, the equivalent study to find temporal and spatial invariant features using only compressed domain parameters may have an interesting future scope. The scheme can be extended to make it robust against other camera attacks such as Automatic White Balance, Bayer Color Filter etc.

- The proposed watermarking scheme can be combined with encryption and may provide an end-to-end commutative security system which is compliant with HD video distribution. These types of securities can be used in military applications where high-security measures are required.

- The H.265/HEVC standard has recently been extended to support efficient compression of multiview 3D videos. In this case, the encoding allows efficient coding of multiple camera views and associated auxiliary pictures. So, the extension of these schemes to the 3D multiview video sequences may also be a good topic for further research.

References

[1] I. J. Cox, M. Miller, and J. Bloom, "Watermarking applications and their properties," in *Information Technology: Coding and Computing, 2000. Proceedings. International Conference on*, 2000, pp. 6–10. [Pg.1], [Pg.2], [Pg.3], [Pg.4]

[2] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (hevc) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, Dec 2012. [Pg.2], [Pg.5], [Pg.6], [Pg.7], [Pg.8], [Pg.9], [Pg.10], [Pg.11], [Pg.13], [Pg.43], [Pg.44], [Pg.75]

[3] S.-J. Lee and S.-H. Jung, "A survey of watermarking techniques applied to multimedia," in *Industrial Electronics, 2001. Proceedings. ISIE 2001. IEEE International Symposium on*, vol. 1, 2001, pp. 272–277 vol.1. [Pg.3], [Pg.4], [Pg.20]

[4] X. Chang, W. Wang, J. Zhao, and L. Zhang, "A survey of digital video watermarking," in *Natural Computation (ICNC), 2011 Seventh International Conference on*, vol. 1, 2011, pp. 61–65. [Pg.3], [Pg.4]

[5] N. Thilagavathi, D. Saravanan, S. Kumarakrishnan, S. Punniakodi, J. Amudhavel, and U. Prabu, "A survey of reversible watermarking techniques, application and attacks," in *Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering & Technology (ICARCSET 2015)*, ser. ICARCSET '15.

New York, NY, USA: ACM, 2015, pp. 37:1–37:7. [Online]. Available: http://doi.acm.org/10.1145/2743065.2743102 [Pg.3], [Pg.4]

[6] P. Singh and R. Chadha, "A survey of digital watermarking techniques, applications and attacks," *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 2, no. 9, pp. 165–175, 2013. [Pg.3], [Pg.4]

[7] G. C. Langelaar, I. Setyawan, and R. L. Lagendijk, "Watermarking digital image and video data. a state-of-the-art overview," *IEEE Signal Processing Magazine*, vol. 17, no. 5, pp. 20–46, Sep 2000. [Pg.4], [Pg.20]

[8] R. B. Wolfgang and E. J. Delp III, "Fragile watermarking using the vw2d watermark," pp. 204–213, 1999. [Online]. Available: http://dx.doi.org/10.1117/12.344670 [Pg.5]

[9] J. Haitsma and T. Kalker, "A watermarking scheme for digital cinema," in *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, vol. 2, Oct 2001, pp. 487–489 vol.2. [Pg.5]

[10] S. Emmanuel and M. S. Kankanhalli, "Mask-based interactive watermarking protocol for video," pp. 247–258, 2001. [Online]. Available: http://dx.doi.org/10.1117/12.448209 [Pg.5]

[11] G. Coatrieux, L. Lecornu, B. Sankur, and C. Roux, "A review of image watermarking applications in healthcare," in *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug 2006, pp. 4691–4694. [Pg.5]

[12] Heinrich hertz institute, hevc software doxygen generated documentation. [Online]. Available: http://hevc.hhi.fraunhofer.de/ [Pg.5]

[13] Hevc software repository. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/ [Pg.5], [Pg.40], [Pg.66], [Pg.80], [Pg.109]

[14] *High Efficiency Video Coding (HEVC) text specification draft 10 (for FDIS & Last Call)*. [Pg.6]

[15] J. Ohm, G. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards x2014;including high efficiency video coding (hevc)," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1669–1684, Dec 2012. [Pg.6], [Pg.8], [Pg.43], [Pg.75]

[16] A. Abramowski, "Towards h.265 video coding standard," pp. 800 819–800 819–7. [Pg.7], [Pg.8], [Pg.9]

[17] I. E. Richardson, *The H.264 Advanced Video Compression Standard*, 2nd ed. Wiley Publishing, 2010. [Pg.6], [Pg.17], [Pg.43], [Pg.44]

[18] D. Le Gall, "Mpeg: A video compression standard for multimedia applications," *Commun. ACM*, vol. 34, no. 4, pp. 46–58, Apr. 1991. [Online]. Available: http://doi.acm.org/10.1145/103085.103090 [Pg.6]

[19] J. Lainema, F. Bossen, W.-J. Han, J. Min, and K. Ugur, "Intra coding of the hevc standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1792–1801, Dec 2012. [Pg.12], [Pg.62]

[20] G. Pastuszak and A. Abramowski, "Algorithm and architecture design of the h.265/hevc intra encoder," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 210–222, Jan 2016. [Pg.12]

[21] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the h.264/avc video coding standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 7, pp. 560–576, July 2003. [Pg.13], [Pg.43], [Pg.44]

[22] A. Saxena and F. Fernandes, "Dct/dst-based transform coding for intra prediction in image/video coding," *Image Processing, IEEE Transactions on*, vol. 22, no. 10, pp. 3974–3981, Oct 2013. [Pg.14]

[23] M. Alghoniemy and A. H. Tewfik, "Geometric invariance in image watermarking," *IEEE Transactions on Image Processing*, vol. 13, no. 2, pp. 145–153, Feb 2004. [Pg.15]

[24] M. Barni, "Effectiveness of exhaustive search and template matching against watermark desynchronization," *IEEE Signal Processing Letters*, vol. 12, no. 2, pp. 158–161, Feb 2005. [Pg.15]

[25] P. Bas, J. M. Chassery, and B. Macq, "Geometrically invariant watermarking using feature points," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1014–1028, Sep 2002. [Pg.15]

[26] G. Caner, A. M. Tekalp, G. Sharma, and W. Heinzelman, "Local image registration by adaptive filtering," *IEEE Transactions on Image Processing*, vol. 15, no. 10, pp. 3053–3065, Oct 2006. [Pg.15]

[27] J. L. Dugelay, S. Roche, C. Rey, and G. Doerr, "Still-image watermarking robust to local geometric distortions," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2831–2842, Sept 2006. [Pg.15]

[28] C.-T. Hsu and J.-L. Wu, "Hidden digital watermarks in images," *IEEE Transactions on Image Processing*, vol. 8, no. 1, pp. 58–68, Jan 1999. [Pg.15]

[29] S. Xiang, H. J. Kim, and J. Huang, "Invariant image watermarking based on statistical features in the low-frequency domain," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 6, pp. 777–790, June 2008. [Pg.15]

[30] X. yang Wang, Y. nan Liu, S. Li, H. ying Yang, and P. pan Niu, "Robust image watermarking approach using polar harmonic transforms based geometric correction," *Neurocomputing*, vol. 174, Part B, pp. 627 – 642, 2016. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0925231215014150 [Pg.15]

[31] M. Noorkami and R. Mersereau, "Compressed-domain video watermarking for h.264," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2, 2005, pp. II–890–3. [Pg.16]

[32] A. Mansouri, A. Aznaveh, F. Torkamani-Azar, and F. Kurugollu, "A low complexity video watermarking in h.264 compressed domain," *Information*

Forensics and Security, IEEE Transactions on, vol. 5, no. 4, pp. 649–657, Dec 2010. [Pg.16], [Pg.19], [Pg.37], [Pg.40], [Pg.44], [Pg.96]

[33] Y. Hu, C. Zhang, and Y. Su, "Information hiding based on intra prediction modes for h.264/avc," in *Multimedia and Expo, 2007 IEEE International Conference on*, July 2007, pp. 1231–1234. [Pg.16], [Pg.17], [Pg.20], [Pg.44], [Pg.45]

[34] G. Yang, J. Li, Y. He, and Z. Kang, "An information hiding algorithm based on intra-prediction modes and matrix coding for h.264/avc video stream," {*AEU*} - *International Journal of Electronics and Communications*, vol. 65, no. 4, pp. 331 – 337, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1434841110001056 [Pg.16], [Pg.17], [Pg.20], [Pg.44], [Pg.45], [Pg.68], [Pg.69], [Pg.70], [Pg.71], [Pg.73], [Pg.116], [Pg.117]

[35] G. Feng and G.-Z. Wu, "Motion vector and mode selection based fragile video watermarking algorithm," in *Anti-Counterfeiting, Security and Identification (ASID), 2011 IEEE International Conference on*, June 2011, pp. 73–76. [Pg.16], [Pg.18], [Pg.44]

[36] S. K. Kapotas and A. N. Skodras, "Real time data hiding by exploiting the ipcm macroblocks in h.264/avc streams," *Journal of Real-Time Image Processing*, vol. 4, no. 1, pp. 33–41, 2009. [Online]. Available: http://dx.doi.org/10.1007/s11554-008-0100-2 [Pg.16]

[37] D. Xu, R. Wang, and J. Wang, "A novel watermarking scheme for h.264/avc video authentication," *Signal Processing: Image Communication*, vol. 26, no. 6, pp. 267 – 279, 2011. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0923596511000464 [Pg.16]

[38] S. Swati, K. Hayat, and Z. Shahid, "A watermarking scheme for high efficiency video coding (hevc)," *PLoS ONE*, vol. 9, no. 8, pp. 1–8, 08 2014. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pone.0105613 [Pg.16], [Pg.19]

[39] M. Noorkami and R. M. Mersereau, "Towards robust compressed-domain video watermarking for h.264," pp. 60 721A–60 721A–9, 2006. [Online]. Available: +http://dx.doi.org/10.1117/12.642535 [Pg.16], [Pg.19]

[40] A. Watson, "Visual optimization of dct quantization matrices for individual images," in *9th Computing in Aerospace Conference*, 1993, p. 4512. [Pg.16], [Pg.21], [Pg.124]

[41] J. Zhang, A. T. S. Ho, G. Qiu, and P. Marziliano, "Robust video watermarking of h.264/avc," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 2, pp. 205–209, Feb 2007. [Pg.16], [Pg.19]

[42] M. Noorkami and R. Mersereau, "Digital video watermarking in p-frames with controlled video bit-rate increase," *Information Forensics and Security, IEEE Transactions on*, vol. 3, no. 3, pp. 441–455, 2008. [Pg.17]

[43] ——, "A framework for robust watermarking of h.264-encoded video with controllable detection performance," *Information Forensics and Security, IEEE Transactions on*, vol. 2, no. 1, pp. 14–23, March 2007. [Pg.17], [Pg.37], [Pg.43], [Pg.44], [Pg.76]

[44] D. Zou and J. Bloom, "H.264/avc stream replacement technique for video watermarking," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, March 2008, pp. 1749–1752. [Pg.17], [Pg.20], [Pg.44]

[45] X. Song, S. Lian, W. Hu, and Y. Hu, "Digital video watermarking based on intra prediction modes for audio video coding standard," *Multimedia Systems*, vol. 20, no. 2, pp. 195–202, 2014. [Online]. Available: http://dx.doi.org/10.1007/s00530-012-0301-1 [Pg.17], [Pg.44], [Pg.67], [Pg.68], [Pg.69], [Pg.70], [Pg.71], [Pg.73], [Pg.116], [Pg.117]

[46] Wang, W. Jiaji, L. Rangding, X. Wei, H. Dawen, and Meiling, "A large-capacity information hiding method for hevc video," *International Conference on Computer Science and Service System (CSSS)*, 2014. [Pg.17], [Pg.44]

[47] N. Mehmood and M. Mushtaq, "A fragile watermarking scheme using prediction modes for h.264/avc content authentication," in *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on*, Oct 2012, pp. 1014–1021. [Pg.18], [Pg.20], [Pg.45]

[48] K. Swaraja, Y. Latha, V. S. K. Reddy, and A. V. Paramkusam, "Video watermarking based on motion vectors of h.264," in *India Conference (INDICON), 2011 Annual IEEE*, 2011, pp. 1–4. [Pg.18]

[49] Y. Guo and F. Pan, "Information hiding for h.264 in video stream switching application," in *2010 IEEE International Conference on Information Theory and Information Security*, Dec 2010, pp. 419–421. [Pg.18]

[50] Y. Tew and K. Wong, "Information hiding in hevc standard using adaptive coding block size decision," in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014, pp. 5502–5506. [Pg.18], [Pg.19]

[51] X. Gong and H.-M. Lu, "Towards fast and robust watermarking scheme for h.264 video," in *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, Dec 2008, pp. 649–653. [Pg.18]

[52] L. Zhang, Y. Zhu, and L.-M. Po, "A novel watermarking scheme with compensation in bit-stream domain for h.264/avc," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, March 2010, pp. 1758–1761. [Pg.19], [Pg.92]

[53] W. Huo, Y. Zhu, and H. Chen, "A controllable error-drift elimination scheme for watermarking algorithm in h.264/avc stream," *Signal Processing Letters, IEEE*, vol. 18, no. 9, pp. 535–538, Sept 2011. [Pg.19], [Pg.92]

[54] X. Ma, Z. Li, H. Tu, and B. Zhang, "A data hiding algorithm for h.264/avc video streams without intra-frame distortion drift," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, no. 10, pp. 1320–1330, Oct 2010. [Pg.19], [Pg.36], [Pg.76], [Pg.92], [Pg.94], [Pg.95]

[55] Y. Liu, Z. Li, X. Ma, and J. Liu, "A novel data hiding scheme for h.264/avc video streams without intra-frame distortion drift," in *Communication*

Technology (ICCT), 2012 IEEE 14th International Conference on, Nov 2012, pp. 824–828. [Pg.19], [Pg.92], [Pg.94]

[56] W. Chen, Z. Shahid, T. Stütz, F. Autrusseau, and P. Le Callet, "Robust drift-free bit-rate preserving h.264 watermarking," *Multimedia Systems*, vol. 20, no. 2, pp. 179–193, 2014. [Online]. Available: http://dx.doi.org/10.1007/s00530-013-0329-x [Pg.19], [Pg.92], [Pg.94], [Pg.95]

[57] P.-C. Chang, K.-L. Chung, J.-J. Chen, C.-H. Lin, and T.-J. Lin, "A dct/dst-based error propagation-free data hiding algorithm for {HEVC} intra-coded frames," *Journal of Visual Communication and Image Representation*, vol. 25, no. 2, pp. 239 – 253, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320313001879 [Pg.19], [Pg.76], [Pg.82], [Pg.83], [Pg.84], [Pg.85], [Pg.86], [Pg.87], [Pg.88], [Pg.92], [Pg.93], [Pg.94], [Pg.95], [Pg.96], [Pg.97], [Pg.110], [Pg.111], [Pg.112], [Pg.113], [Pg.114], [Pg.115], [Pg.116], [Pg.117], [Pg.119], [Pg.120], [Pg.121]

[58] S. Gaj, A. S. Patel, and A. Sur, "Object based watermarking for h.264/avc video resistant to rst attacks," *Multimedia Tools and Applications*, vol. 75, no. 6, pp. 3053–3080, 2016. [Online]. Available: http://dx.doi.org/10.1007/s11042-014-2422-3 [Pg.19], [Pg.77], [Pg.78], [Pg.128]

[59] H. Shojanazeri, W. A. W. Adnan, and S. M. S. Ahmad, "Video watermarking techniques for copyright protection and content authentication," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 5, pp. 652–660, 2013. [Pg.20]

[60] G. Gupta, V. Gupta, and M. Chandra, "Review on video watermarking techniques in spatial and transform domain," in *Information Systems Design and Intelligent Applications*. Springer, 2016, pp. 683–691. [Pg.20]

[61] A. Khan, A. Siddiqa, S. Munib, and S. A. Malik, "A recent survey of reversible watermarking techniques," *Information Sciences*, vol. 279, pp. 251 – 272, 2014. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0020025514004150 [Pg.20]

[62] L. Pérez-Freire, P. Comesana, J. R. Troncoso-Pastoriza, and F. Pérez-González, "Watermarking security: a survey," in *Transactions on Data Hiding and Multimedia Security I.* Springer, 2006, pp. 41–72. [Pg.20]

[63] X. N. Martin, M. Schmucker, and C. Busch, "Video watermarking resisting to rotation, scaling, and translation," in *Proc. SPIE Security Watermarking of Multimedia Contents IV*, 2002, pp. 512–519. [Pg.20]

[64] C. Vural and B. Baraklı, "Reversible video watermarking using motion-compensated frame interpolation error expansion," *Signal, Image and Video Processing*, vol. 9, no. 7, pp. 1613–1623, 2015. [Pg.20]

[65] Y.-Y. Lee, S.-U. Park, C.-S. Kim, and S.-U. Lee, "Temporal feature modulation for video watermarking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 4, pp. 603–608, 2009. [Pg.20]

[66] C.-T. Hsu and J.-L. Wu, "Dct-based watermarking for video," *IEEE Transactions on Consumer Electronics*, vol. 44, no. 1, pp. 206–216, Feb 1998. [Pg.20]

[67] H. Liu, N. Chen, J. Huang, X. Huang, and Y. Q. Shi, "A robust dwt-based video watermarking algorithm," in *2002 IEEE International Symposium on Circuits and Systems. Proceedings (Cat. No.02CH37353)*, vol. 3, 2002, pp. 631–634. [Pg.20]

[68] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Multiresolution scene-based video watermarking using perceptual models," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 4, pp. 540–550, May 1998. [Pg.20]

[69] H.-Y. Yang, X.-Y. Wang, P.-P. Niu, and A.-L. Wang, "Robust color image watermarking using geometric invariant quaternion polar harmonic transform," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 11, no. 3, pp. 40:1–40:26, Feb. 2015. [Online]. Available: http://doi.acm.org/10.1145/2700299 [Pg.20], [Pg.128]

[70] X.-C. Yuan and C.-M. Pun, "Feature based video watermarking resistant to geometric distortions," in *Trust, Security and Privacy in Computing and*

Communications (TrustCom), 2013 12th IEEE International Conference on, July 2013, pp. 763–767. [Pg.20], [Pg.128]

[71] T. M. Thanh, P. T. Hiep, T. M. Tam, and K. Tanaka, "Robust semi-blind video watermarking based on frame-patch matching," {*AEU*} - *International Journal of Electronics and Communications*, vol. 68, no. 10, pp. 1007 – 1015, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1434841114001393 [Pg.20], [Pg.128]

[72] X.-C. Yuan and C.-M. Pun, "Feature extraction and local zernike moments based geometric invariant watermarking," *Multimedia Tools and Applications*, vol. 72, no. 1, pp. 777–799, 2014. [Online]. Available: http://dx.doi.org/10.1007/s11042-013-1405-0 [Pg.20], [Pg.128]

[73] L. Jing, "A novel scheme of robust and blind video watermarking," in *2009 International Forum on Information Technology and Applications*, vol. 1, May 2009, pp. 430–434. [Pg.20]

[74] Z. Dawei, C. Guanrong, and L. Wenbo, "A chaos-based robust wavelet-domain watermarking algorithm," *Chaos, Solitons & Fractals*, vol. 22, no. 1, pp. 47 – 54, 2004. [Online]. Available: //www.sciencedirect.com/science/article/pii/S0960077903006726 [Pg.20]

[75] M. Asikuzzaman, M. J. Alam, A. J. Lambert, and M. R. Pickering, "A blind digital video watermarking scheme with enhanced robustness to geometric distortion," in *2012 International Conference on Digital Image Computing Techniques and Applications (DICTA)*, Dec 2012, pp. 1–8. [Pg.20]

[76] Y. Zhao and R. L. Lagendijk, "Video watermarking scheme resistant to geometric attacks," in *Proceedings. International Conference on Image Processing*, vol. 2, 2002, pp. II–145–II–148 vol.2. [Pg.20]

[77] Y. Shao, L. Zhang, G. Wu, and X. Lin, "A novel frequency domain watermarking algorithm with resistance to geometric distortions and copy attack," in *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, vol. 2, May 2003, pp. II–940–II–943 vol.2. [Pg.20]

[78] D. He and Q. Sun, "A rst resilient object-based video watermarking scheme," in *Image Processing, 2004. ICIP '04. 2004 International Conference on*, vol. 2, Oct 2004, pp. 737–740 Vol.2. [Pg.20]

[79] Y. Liu and J. Zhao, "Rst invariant video watermarking based on 1d dft and radon transform," in *2008 5th International Conference on Visual Information Engineering (VIE 2008)*, July 2008, pp. 443–448. [Pg.20]

[80] ——, "Rst invariant video watermarking based on log-polar mapping and phase-only filtering," in *2010 IEEE International Conference on Multimedia and Expo*, July 2010, pp. 1305–1310. [Pg.20]

[81] H. Zhang, J. Li, and C. Dong, "Multiple video zero-watermarking based on 3d dft to resist geometric attacks," in *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, April 2012, pp. 1141–1144. [Pg.20]

[82] Z. Li, L. F. Liu, and C. X. Jiang, "A self-adaptive video dual watermarking based on the motion characteristic and geometric invariant for ubiquitous multimedia," in *2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, Dec 2015, pp. 28–32. [Pg.20]

[83] R. Maharjan, A. Alsadoon, P. W. C. Prasad, A. M. S. Rahma, A. Elchouemi, and S. A. Senanayake, "A proposed robust video watermarking algorithm: Enhanced extraction from geometric attacks," in *2016 First International Conference on Multimedia and Image Processing (ICMIP)*, June 2016, pp. 45–50. [Pg.20]

[84] J. D. Koch and H.-K. Mike D. Smith Rahul Telang, "Camcording and film piracy in asia-pacific economic cooperation economies," August 2011. [Pg.20]

[85] J. Moreira-Perez, B. Chupeau, G. Doerr, and S. Baudry, "Exploring color information to characterize camcorder piracy," in *Information Forensics and Security (WIFS), 2013 IEEE International Workshop on*, Nov 2013, pp. 132–137. [Pg.21]

[86] M.-J. Lee, K.-S. Kim, Y.-H. Suh, and H.-K. Lee, "Improved watermark detection robust to camcorder capture based on quadrangle estimation," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, Nov 2009, pp. 101–104. [Pg.21], [Pg.22], [Pg.124], [Pg.125], [Pg.128]

[87] A. van Leest, J. Haitsma, and T. Kalker, "On digital cinema and watermarking," pp. 526–535, 2003. [Online]. Available: http://dx.doi.org/10.1117/12.476856 [Pg.21], [Pg.22], [Pg.124]

[88] H. Do, D. Choi, H. Choi, and T. Kim, "Digital video watermarking based on histogram and temporal modulation and robust to camcorder recording," in *Signal Processing and Information Technology, 2008. ISSPIT 2008. IEEE International Symposium on*, Dec 2008, pp. 330–335. [Pg.21], [Pg.22], [Pg.123], [Pg.124], [Pg.128], [Pg.141], [Pg.142], [Pg.143], [Pg.147], [Pg.148]

[89] L. Li, Z. Dong, J. Lu, J. Dai, Q. Huang, C.-C. Chang, and T. Wu, "{AN} h.264/avc {HDTV} watermarking algorithm robust to camcorder recording," *Journal of Visual Communication and Image Representation*, vol. 26, pp. 1 – 8, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320314001400 [Pg.21], [Pg.22], [Pg.38], [Pg.113], [Pg.123], [Pg.124], [Pg.128], [Pg.141], [Pg.142], [Pg.143], [Pg.144], [Pg.147], [Pg.148], [Pg.149]

[90] D. Delannay, J.-F. Delaigle, B. M. M. Macq, and M. Barlaud, "Compensation of geometrical deformations for watermark extraction in digital cinema application," pp. 149–157, 2001. [Online]. Available: http://dx.doi.org/10.1117/12.435395 [Pg.22], [Pg.128]

[91] M.-J. Lee, K.-S. Kim, H.-Y. Lee, T.-W. Oh, Y.-H. Suh, and H.-K. Lee, "Robust watermark detection against d-a/a-d conversion for digital cinema using local auto-correlation function," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, Oct 2008, pp. 425–428. [Pg.22], [Pg.124], [Pg.128]

[92] Y. Wang, "A blind mpeg-2 video watermarking robust to regular geometric attacks," in *Open-source Software for Scientific Computation (OSSC), 2009 IEEE International Workshop on*, Sept 2009, pp. 169–171. [Pg.22]

[93] J. Stankowski, T. Grajek, and M. Domanski, "Fast watermarking of mpeg-4 avc/h.264 encoded hdtv video bitstreams," in *Picture Coding Symposium (PCS), 2012*, May 2012, pp. 265–268. [Pg.22]

[94] J. Siast, J. Stankowski, T. Grajek, and M. Domanski, "Digital watermarking with local strength adjustment for avc-compressed hdtv bitstreams," in *Picture Coding Symposium (PCS), 2013*, Dec 2013, pp. 53–56. [Pg.22]

[95] P. Schaber, S. Kopf, C. Wesch, and W. Effelsberg, "Cammark: A camcorder copy simulation as watermarking benchmark for digital video," in *Proceedings of the 5th ACM Multimedia Systems Conference*, ser. MMSys '14. New York, NY, USA: ACM, 2014, pp. 91–102. [Online]. Available: http://doi.acm.org/10.1145/2557642.2557644 [Pg.23], [Pg.24], [Pg.38], [Pg.123], [Pg.125], [Pg.126], [Pg.127]

[96] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004. [Online]. Available: http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94 [Pg.31], [Pg.128]

[97] K. Mikolajczyk, "Detection of local features invariant to affine transformations," Ph.D. dissertation, Institut National Polytechnique de Grenoble, 2011. [Pg.32]

[98] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Proceedings of the 7th European Conference on Computer Vision-Part I*, ser. ECCV '02. London, UK, UK: Springer-Verlag, 2002, pp. 128–142. [Online]. Available: http://dl.acm.org/citation.cfm?id=645315.649184 [Pg.32]

[99] M. Vishwanath, R. M. Owens, and M. J. Irwin, "Vlsi architectures for the discrete wavelet transform," *IEEE Transactions on Circuits and Systems*

II: Analog and Digital Signal Processing, vol. 42, no. 5, pp. 305–316, 1995. [Pg.34]

[100] C. S. Burrus, R. A. Gopinath, and H. Guo, "Introduction to wavelets and wavelet transforms: a primer," 1997. [Pg.34]

[101] M. Weeks and M. Bayoumi, "3d discrete wavelet transform architectures," in *Circuits and Systems, 1998. ISCAS '98. Proceedings of the 1998 IEEE International Symposium on*, vol. 4, May 1998, pp. 57–60 vol.4. [Pg.35]

[102] P. W. Chan, M. R. Lyu, and R. T. Chin, "A novel scheme for hybrid digital video watermarking: approach, evaluation and experimentation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 12, pp. 1638–1649, Dec 2005. [Pg.37]

[103] (2013) Video quality measurement tool. [Online]. Available: http://mmspg.epfl.ch/vqmt [Pg.38], [Pg.70], [Pg.86], [Pg.147]

[104] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004. [Pg.38], [Pg.39], [Pg.70], [Pg.86], [Pg.117], [Pg.146]

[105] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, Feb 2006. [Pg.38], [Pg.39], [Pg.70], [Pg.86], [Pg.117], [Pg.146]

[106] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, Jan 2009. [Pg.39]

[107] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation, scale, and translation resilient watermarking for images," *IEEE Transactions on Image Processing*, vol. 10, no. 5, pp. 767–782, May 2001. [Pg.40]

[108] C.-C. Weng, H. Chen, and C.-S. Fuh, "A novel automatic white balance method for digital still cameras," in *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, May 2005, pp. 3801–3804 Vol. 4. [Pg.126]

[109] P. Vinukonda, "A study of the scale-invariant feature transform on a parallel pipeline," Master's thesis, Louisiana State University, 2011. [Pg.148], [Pg.149]

Appendix: A

Journal Publications:

- **Sibaji Gaj**, Aditya Kanetkar, Arijit Sur, and Prabin Kumar Bora. ***"Drift-Compensated Robust Watermarking Algorithm for H.265/HEVC Video Stream".*** ACM Trans. Multimedia Comput. Commun. Appl. 13, 1, Article 11 (January 2017), 24 pages. DOI: https://doi.org/10.1145/3009910

- **Sibaji Gaj**, Ashish Singh Patel and Arijit Sur, ***"Object Based Watermarking for H.264/AVC Video Resistant to RST Attacks".*** Multimedia Tools and Applications, Springer. March 2016, Volume 75, Issue 6, pp 30533080. DOI 10.1007/s11042-014-2422-3.

- **Sibaji Gaj**, Anoop Kumar Rathore, Arijit Sur and Prabin Kumar Bora, ***"A Robust Watermarking Scheme Against Frame Blending and Projection Attack".*** Multimedia Tools and Applications, Springer. pp 1-25. DOI:10.1007/s11042-016-3961-6. (Accepted)

- **Sibaji Gaj**, Arijit Sur and Prabin Kumar Bora, ***"Prediction Mode Based H.265/HEVC Video Watermarking Resisting Re-compression Attack".*** Journal of Visual Communication and Image Representation, Elsevier. (Major Revision Submitted.)

Conference Publications:

- **Sibaji Gaj**, Shuvendu Rana, Arijit Sur and Prabin Kumar Bora, ***"A Drift Compensated Reversible Watermarking Scheme for H.265/HEVC".*** 2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP), Montreal, Canada, 2016, pp. 1-5.doi: 10.1109/MMSP.2016.7813358

- Shuvendu Rana, **Sibaji Gaj**, Arijit Sur and Prabin Kumar Bora, ***"Detection of Fake 3D Video Using CNN".*** 2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP), Montreal, Canada, 2016, pp. 1-5. doi: 10.1109/MMSP.2016.7813368

- Shuvendu Rana, **Sibaji Gaj**, Arijit Sur and Prabin Kumar Bora, ***"Segmentation Based 3D Depth Watermarking using SIFT".*** 2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP), Montreal, Canada, 2016, pp. 1-5. doi: 10.1109/MMSP.2016.7813367

- **Sibaji Gaj**, Shuvendu Rana, Arijit Sur and Prabin Kumar Bora, ***"A Robust Watermarking Scheme against Frame Blending, Projection and Content Adaptation Attacks".*** 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 2016, pp. 1-8. doi: 10.1109/DICTA.2016.7796996

- Shuvendu Rana, **Sibaji Gaj** and Arijit Sur, ***"View Invariant 3D Video Watermarking Using Depth Based Embedding".*** 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 2016, pp. 1-8.
doi: 10.1109/DICTA.2016.7797095

- **Sibaji Gaj**, Arijit Sur and Prabin Kumar, ***"A Robust Watermarking Scheme Against Re-compression Attack for H.265/HEVC".*** 2015 Fifth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), Patna, 2015, pp. 1-4. doi: 10.1109/NCVPRIPG.2015.7490065

- **Sibaji Gaj**, Shuvendu Rana, Anirban Lekharu, Arijit Sur and Prabin Kumar Bora, ***"RST Invariant Multi View 3D Image Watermarking using DWT and SVD".*** 2015 Fifth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), Patna, 2015, pp. 1-4. doi: 10.1109/NCVPRIPG.2015.7490066