ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645
Indian Journal of Science and Technology, *Vol 11(40), DOI: 10.17485/ijst/2018/v11i40/120482, October 2018*

Video Watermarking Algorithm: Reducing Vulnerability to Geometric Attacks

R. Maharjan[1]*, Abeer Alsadoon[1], P. W. C. Prasad[1], A. M. S. Rahma[2] and A. Elchouemi[3]

[1]School of Computing and Mathematics, Charles Sturt University, Australia;
chandanapw@yahoo.co.uk, aalsadoon@studygroup.com, cwithana@studygroup.com
[2]School of Software Engineering, University of Technology Baghdad;
monem.rahma@yahoo.com
[3] Information Technologies, Walden University, USA;
amr.elchouemi@hpe.com

Abstract

Objective: This paper proposes a new integrated system based on a video watermarking algorithm with high imperceptibility, improved security, robust against common image and video processing attacks and geometric attacks. **Methods/ Statistical Analysis:** The proposed algorithm is based on the features of a selected best algorithm by Agilandeeswari and Ganesan and its limitations. The advantages of the hybrid transform techniques of Non-subsampled Contourlet Transform, Discrete Wavelet Transform and Singular Value Decomposition are utilized for high robustness against common attacks. **Finding:** The features of angle invariance and distance invariability of Log Polar Transform and Inverse Log Polar Transform are used in the extraction process only to resist geometric attacks without affecting the imperceptibility of watermark and watermarked video**. Application/Improvements:** Modified Arnold Transformation is introduced to improve the security of the watermark. The experiment shows that the algorithm is extremely robust in terms of image and video processing, temporal, common attacks and geometric attacks covering rotation, scaling and translation attacks along with high imperceptibility and improved security.

Keywords: Geometric Attacks, Non-subsampled Contourlet Transform, Video Watermarking

1. Introduction

The fast paced advancement in the multimedia and Internet technology has significantly enhanced the level and speed of knowledge exchange. This has made it possible to exchange information in a variety of forms, including videos. Due to the evolution in multimedia technology and increment in Internet bandwidth, people nowadays can easily share or watch videos through the Internet[1].

However, this evolution has equally generated copyright and authentication issues with videos. Due to almost unlimited availability, people can easily download, edit and re-upload as if these were their own videos and sometimes they try to manipulate and destroy the original videos. Even the encrypted video has been illegally sold to unauthorized customers for business profits after decryption by those who purchased the item legally. These tampering cases have become serious problems and this vulnerability of videos needs to be addressed urgently. Therefore, the need for protecting the authenticity and integrity of videos is the focus of many research work. The Digital Watermarking Technique is one of the solutions for copyright protection[1].

Digital watermarking is the process of embedding digital watermarks in multimedia objects so as to hide the copyright information in the object and protect its ownership[1] some of the applications of this digital watermarking are in copyright protection, authentication, transaction tracking, proof of ownership, broadcast monitoring etc. The watermark is the digital data which is embedded in the multimedia objects for protection. It could be the

*Author for correspondence

owner's logo, identifier, information or content of the data to be protected, or the creator, any random sequence or image which provides integrity while authenticating the object. Digital video watermarking is the digital watermarking procedure for the protection of the authenticity of the video from being vulnerable to tampering attacks[1].

Watermarking algorithms have been classified based on the domain for embedding into either *spatial domain* or *transform (frequency) domain*. In *spatial domain*, the watermark is embedded directly by modifying the subsets of the pixels of the multimedia object. Previous research shows that the available algorithms based on *spatial domain* are simpler and faster than the algorithms based on *transform domain*[2]. However, they failed to provide better robustness. To overcome the drawbacks of algorithms based on spatial domain, watermarking types based on *transform domain* were introduced. Transform domain is transferring multimedia contents into its frequency bands using reversible *transforms*[3]. In watermarking algorithms based on *transform domain*, the watermark is embedded in the transformed coefficients of the multimedia objects which provide more robustness against various types of video processing, such as image processing and geometric attacks. The watermarking algorithm can be visible or invisible[1]. However, almost all watermarking prefers invisible types. Invisible watermarking can be either *blind* or *non blind*. *Non blind* watermarking requires an original watermark and an original video while extracting watermark, whereas *blind* method does not require that[2].

According to[1], the digital watermarking process has certain factors that are required to be considered. They are:

- Robustness: The method is said to be robust if the watermark can resist certain possible attacks like image processing attacks, geometric attacks etc.
- Imperceptibility: Imperceptibility is the ability to maintain the quality of the watermarked video as the original video.
- Security: The method is said to be secure if the watermark can resist the attempt of its removal or destruction by changing the video.
- Capacity: Capacity is the size of the watermark which should be large enough so as to uniquely identify its video owner.
- Computational Time: Computational time is the time taken for embedding the watermark in the video and extracting it from the video. Computational time is highly dependent on the computational complexity of the method.

Some of these factors are difficult to maintain in watermarking techniques. Robustness and imperceptibility are the main priority factors that should be considered in digital watermarking. However, maintaining a real time performance is also equally important. With high robustness, imperceptibility and computational time get degraded and vice versa. Likewise, with high robustness, imperceptibility may be compromised. So, equal trade-off has to be achieved for maintaining these factors. In our proposed work, enhancing robustness in terms of geometric attacks along with maintaining other attacks has been highly emphasized. While performing this feature, though the computational time has not been minimized, the previous value has still been maintained without any drastic increment in it.

The paper is organized as follows: Section 1 is the introduction and section 2 discusses the literature survey on the research topic. Brief explanations on selected best algorithms with their limitations and mitigations and the proposed algorithm with its logical design are given in sections 3 and 4 respectively, followed by detailed explanations of the proposed algorithm in sections 5, 6 and 7. Implementation and an analysis of results and the review of our contribution on current system are given in section 8 respectively. The discussion followed by the conclusion is given in section 9, together with possible future work.

2. Research Methodolgy

For the purpose of providing robust digital video watermarking, several video watermarking techniques and algorithms have been introduced focusing on one or more main factors of watermarking. Among the collected algorithms, some algorithms are based on the real time performance which is one of the factors of watermarking. In[4] presented a real time watermarking algorithm for the streaming video using a Watermark Embedding Margin of Selection (WEMF) method which supports in selecting frames without decoding the complete video and using a Just Noticeable Difference (JNF) which provides a measurement for improving invisibility of the watermark. In[5] proposed a video watermarking algorithm considering not only the aspects of robustness and invisibility but also real time detection for MPEG which has been the requirement to authenticate the source of the video for a surveillance network of a small business. This algorithm has used a segmentation of the video for reducing the

computational time and enhances the robustness against temporal attacks Huffman Table for improving the robustness and Visual Model for improving the invisibility of the watermark. However, while reducing the computational time, this algorithm has highly compromised the robustness and the imperceptibility factors of the watermarked video. Similarly, focusing on real time performance[6] came up with a video watermarking technique on a compressed domain of a High Definition video with the robustness against video processing attacks such as frame rate changing, downscaling and transcoding computation whereas, this algorithm compromises the robustness against other general attacks and visual quality.

From the research, it proved that maintaining high robustness and imperceptibility with the computational time are inverse to each other. Hence, equal trade off is required to maintain both of the factors of the watermarking algorithms.

In order to provide an effective and robust video watermarking algorithm[7] focused on a distortion resistance based on an additive spread spectrum and a periodic watermark concept to protect the piracy of the video files. However, it is less resistant to noise attacks and also the use of the spread spectrum is computationally complex. In[8] proposed a new video watermarking algorithm based on Singular Value Decomposition and slope-based embedding technique focusing on temporal dimension attacks. Whereas, this algorithm has provided less robustness against most of the common attacks.

Similarly Qi et al.[9] fused dual transform domains 2D Discrete Wavelet Transform (DWT) and 3D Discrete Cosine Transform (DCT) with Particle Swarm Optimization (PSO) in order to improve the robustness of the watermark in the video watermarking. In[10] presented a robust and hybrid non-blind video watermarking technique for MPEG based on Singular Value Decomposition of high tensor and DWT. But this algorithm degrades the video quality[11] introduced an adaptive video watermarking using a human visual system with a fuzzy interference model to provide high robustness and imperceptibility. However, it is less resistible to geometric attacks.

In[12] focused on the real time performance while maintaining the robustness and the imperceptibility by utilizing the advantages of the Scene Change Detection to reduce the computational time and the robustness against temporal attacks, fused DWT, Discrete Fourier Transform (DFT) and Singular Value Decomposition (SVD) to robust the watermarking process and used Binary Particle Swarm Optimization (BPSO) to get the robust pixel frames. However, DFT and BPSO processes have high computational complexity[13] also used the shot boundary detection which reduces the computational time and the robustness against temporal attacks and classifying blocks of the compressed video. However, this method is less robust against geometric attacks and other noise attacks. In[14] proposed a robust video watermarking scheme using hybrid techniques of Contourlet Transform (CT) and DWT for enhancing the robustness and the visual perception. This algorithm has maintained the security using Arnold transform and has achieved high imperceptibility and payload using a bit plane slicing. However, CT basically lacks the property of the shift invariance. Furthermore, the Arnold transformation used in this scheme is the traditional one which is less secure. To overcome these limitations[15] proposed a robust watermarking scheme based on nonsubsampled contourlet transform (NSCT) which is multiscale, multidirectional as well as shift invariant and uses a non-subsampled pyramid (NSP) and non-subsampled directional filter bank (NDFB). However, the robustness against geometric attacks is not highly improved.

Besides all of the image and video processing attacks, the most challenging attack to be considered are geometric attacks, also known as RST (Rotation, Scaling and Translation) attacks for providing the robust video watermarking algorithm. The schemes presented by[16–19] have focused on the robustness against geometric attacks using DFT and LPM for images. The algorithm used for images can also be applied for video frames. The watermarking scheme proposed by[16] uses Fourier-Mellin Transform which works with Log Polar Mapping (LPM) and DFT. However, the use of LPM and inverse LPM (ILPM) highly degrades the watermarked image quality[18] focused on providing a RST invariant watermarking for image using approximate ILPM to get the location to be the watermarked image. However, this scheme also had drawback of ILPM which introduces interference distortion. To overcome this drawback[19] introduced the new scheme which reduces the distortion caused by ILPM. However, this scheme failed by not being able to extract for fractional angles of rotation attacks. To reduce this limitation[20] presented color watermarking for image using Fourier transform and improved ULPM. However, improved inverse LPM still provides distortion effects. Although these schemes focused on geometric attacks,

they are all based on DFT which is less robust and more complex than other *transform domains*. Hence, to provide the robustness against RST attacks, in[21] proposed a robust video watermarking algorithm by embedding watermark with moving objects of video shots while maintaining the robustness against other attacks. But, this algorithm provided poor performance on the robustness as well as the imperceptibility. Relatively, in[22] proposed a compressed video watermarking algorithm focusing on resistance to geometric attacks with invariance of a Histogram Shape in DWT Domain and the real time performance by using a fast inter transformation between Block DCTs and One-Level DWT. However, it is less resistant against temporal attacks and the video quality is also degraded. Likewise, to achieve high robustness against geometric attacks without compromising other common attacks[23] has used a zero watermarking algorithm based on PM with 2D DWT and 3D DCT. However, this algorithm is fixed for the authentication only rather than extracting the original watermark.

From the research, it has been derived that achieving high robustness against geometric attacks while maintaining the imperceptibility with other common attacks has been one of the most challenging feature in the video watermarking algorithms. Most of the schemes lacked to consider geometric attacks whereas others failed to maintain the imperceptibility with other common attacks. Apart from the robustness, only some of the algorithms have focused on providing the security whereas most of the algorithms have not focused on it.

2.1 Current Selected Video Watermarking Algorithm

The algorithm proposed by[14] has provided high robustness against image and video processing and some degrees of rotation attacks, high imperceptibility and payload along with dual security. Furthermore, this algorithm has used a color watermark to be embedded which has not been considered by any other watermarking algorithms. This is *non-blind* watermarking algorithm so for extracting the watermark, both the original watermark and the original video are required.

The algorithm has two stages which are embedding and extraction process. The watermark embedding process has sub sections such as watermark pre-processing, video pre-processing, embedding and video post processing. Similarly, watermark extraction process has sub sections such as watermark pre-processing, video pre-processing, extraction and watermark post processing[14].

According to[14], this algorithm has used the bit plane slicing method in the watermark to be embedded which has highly enhanced the imperceptibility of the watermarked video by embedding only a bit slice in each key frame. Moreover, it has also used the scene change detection method to select the key frames which reduces the computational time for embedding the watermark in all the frames as well as provides robustness against temporal attacks. The algorithm has used the hybrid transform technique of Contourlet Transform and Discrete Wavelet transform which provides high robustness against image processing and temporal attacks. The algorithm provides double security by scrambling the watermark bits using Arnold transformation and authenticating using Eigen Vector. These are the main features of the current algorithm.

The proposed solution with its limitation and possible mitigation are presented in Figure 1. Though the robustness against rotation attack has been considered in this algorithm, only some low degrees of rotations and rotation of 180 degree of the watermarked video has been able to resist. The quality of the extracted watermark from the rotation attacked watermarked videos of more than 10 degrees are very poor. Furthermore, the robustness against other geometric attacks is also not considered. Hence, possible mitigation can be achieving better robustness against geometric attacks by modifying some processes while extracting the watermark from the watermarked video.

Though this algorithm uses Arnold transformation to scramble the watermark to provide security to the watermark, the Arnold transform used is a traditional one which is less secure. Hence, possible mitigation can be modifying the Arnold transformation which is more secure than the previous one.

Hence, possible mitigation to reduce these limitations is to propose a modified video watermarking algorithm with the reduction in its limitations.

3. Proposed Work

For this work, different articles on the existing video watermarking methods have been reviewed. The positive and negative outcomes for each of them have been identified based on deep analysis of the main factors of video

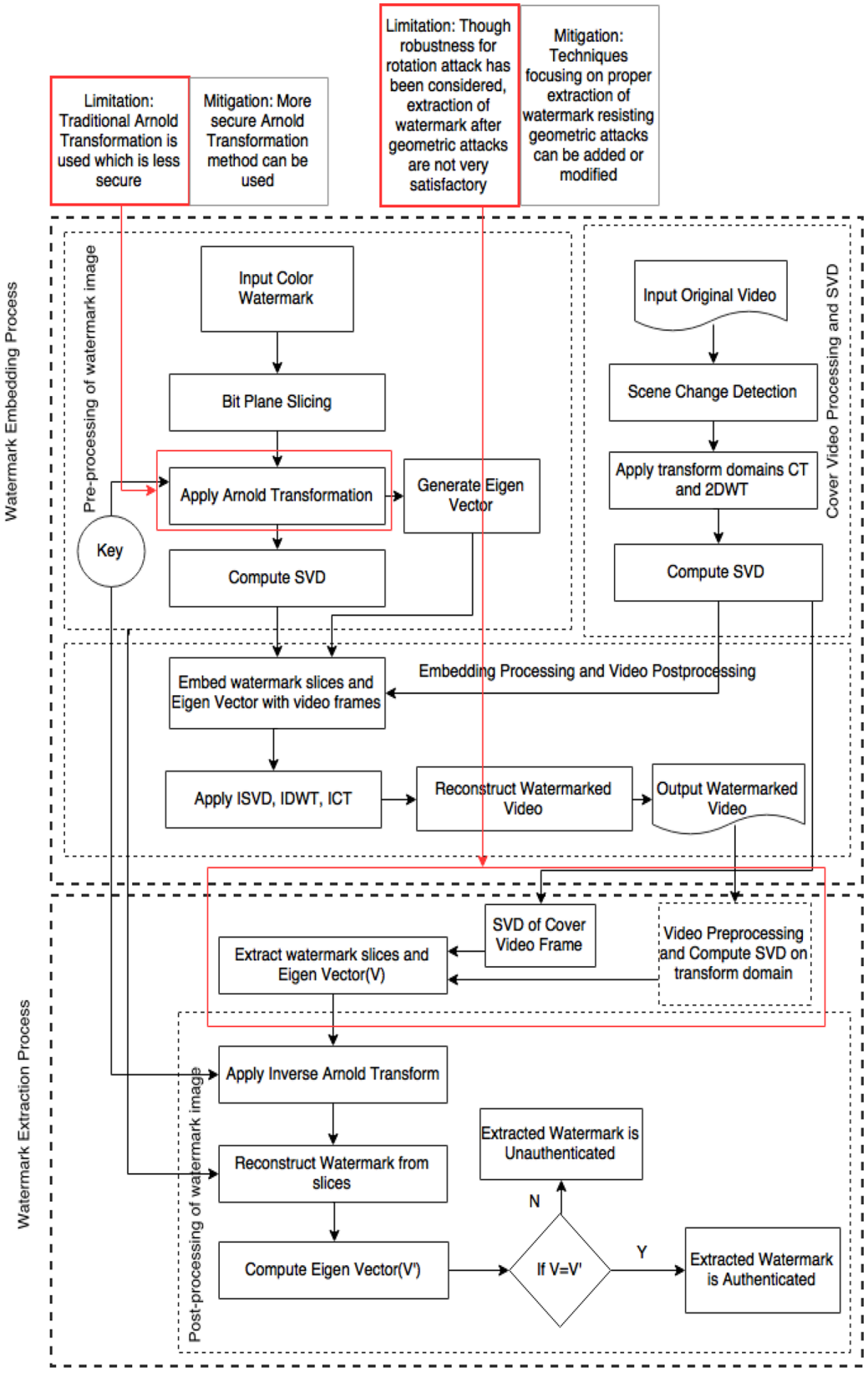


Figure 1. Current solution of video watermarking algorithm with its limitations and mitigations.

watermarking algorithms. The main factors are robustness, imperceptibility, security, computational time and payload. From the collected current solutions, the best algorithm has been selected based on the main factors in this work. A new system has been proposed that is based on the selected best solution[14] with enhanced robustness against geometric attacks utilizing the good features of Log Polar Transform and Inverse Log Polar Transform and modified Arnold Transformation.

The proposed system and the current selected one[14] have been implemented using Matlab R2015b. In these implementations, 10 test sample videos and 10 test watermark images with different sizes and formats have been used to test the proposed algorithm with the current algorithm. Finally, comparisons of proposed system with the current one and their evaluations have been done by testing both of them in different attack situations.

The proposed system, Robust Video Watermarking Algorithm: Enhanced Extraction from Geometric Attacks (PRVWA-EEfGA), aims to reduce the limitations found in the existing algorithm proposed by[14] and to propose a new system which highly focuses on the robustness in terms of geometric attacks while maintaining robustness against other common attacks. The new contribution in the proposed system is the use of Log Polar Transform (LPT) and Inverse Log Polar Transform (ILPT) in the extraction process only. The other main contribution is modifying the Arnold Transform method to be more secure than the traditional one.

For the purpose of maintaining the robustness against translation attack, Nonsubsampled Contourlet Transform has been used. For the purpose of maintaining high robustness against rotation and scaling attacks, LPT and ILPT have been used in the original video and the watermarked video respectively during the extraction process only. Furthermore, for enhancing the security, the traditional Arnold transform of the selected existing algorithm has been replaced by the modified Arnold Transform.

The proposed system; PRVWA-EEfGA, is a multi-stage system that includes embedding system as a first stage and extraction as a second stage. Figure 2 represents the transition between these two stages. In the proposed system, the first stage contains all the tasks related to embed the watermark into the video sample. The second stage contains all the tasks related to extract the watermark from the watermarked video which is obtained from the first stage. The next section gives a brief introduction of the logical design of the proposed system and the following next sections give the detail descriptions of each stage and their algorithms.

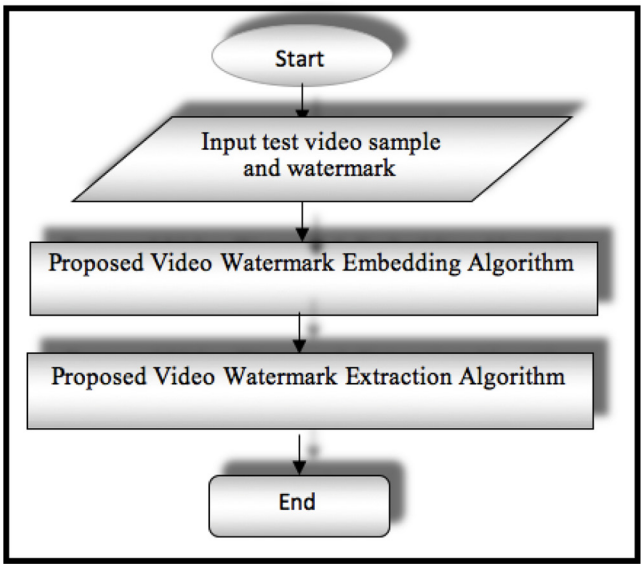


Figure 2. Transition diagram of multi- stages proposed system.

3.1 Logical Design of the Proposed System (PRVWA-EEfGA)

The block diagram of the proposed system is presented in Figure 3. Though the embedding and the extraction are two separate stages, the extraction stage occurs only after the embedding stage. The general block diagram presents the flow from an input video sample and a watermark sample followed by obtaining the watermarked video to finally achieving the extracted watermark from the watermarked video.

The flow starts with the input video sample and the watermark sample. In Embedding stage, there are further sections which are video pre-processing, watermark pre-processing and embedding process with video post-processing. In video pre-processing, there are sub-sections such as scene change detection, frame color conversion, Nonsubsampled Contourlet Transform, Discrete Wavelet transform and Singular Value Decomposition. In this video pre-processing, the tasks are performed with the original input video sample and are related to reducing the computational time and providing high robustness. In watermark pre-processing, there are sub-sections such as bit plane slicing, modified Arnold transform, Singular Value Decomposition which are done for purpose of enhancing the imperceptibility and the payload. In embedding process with video post-processing, water-

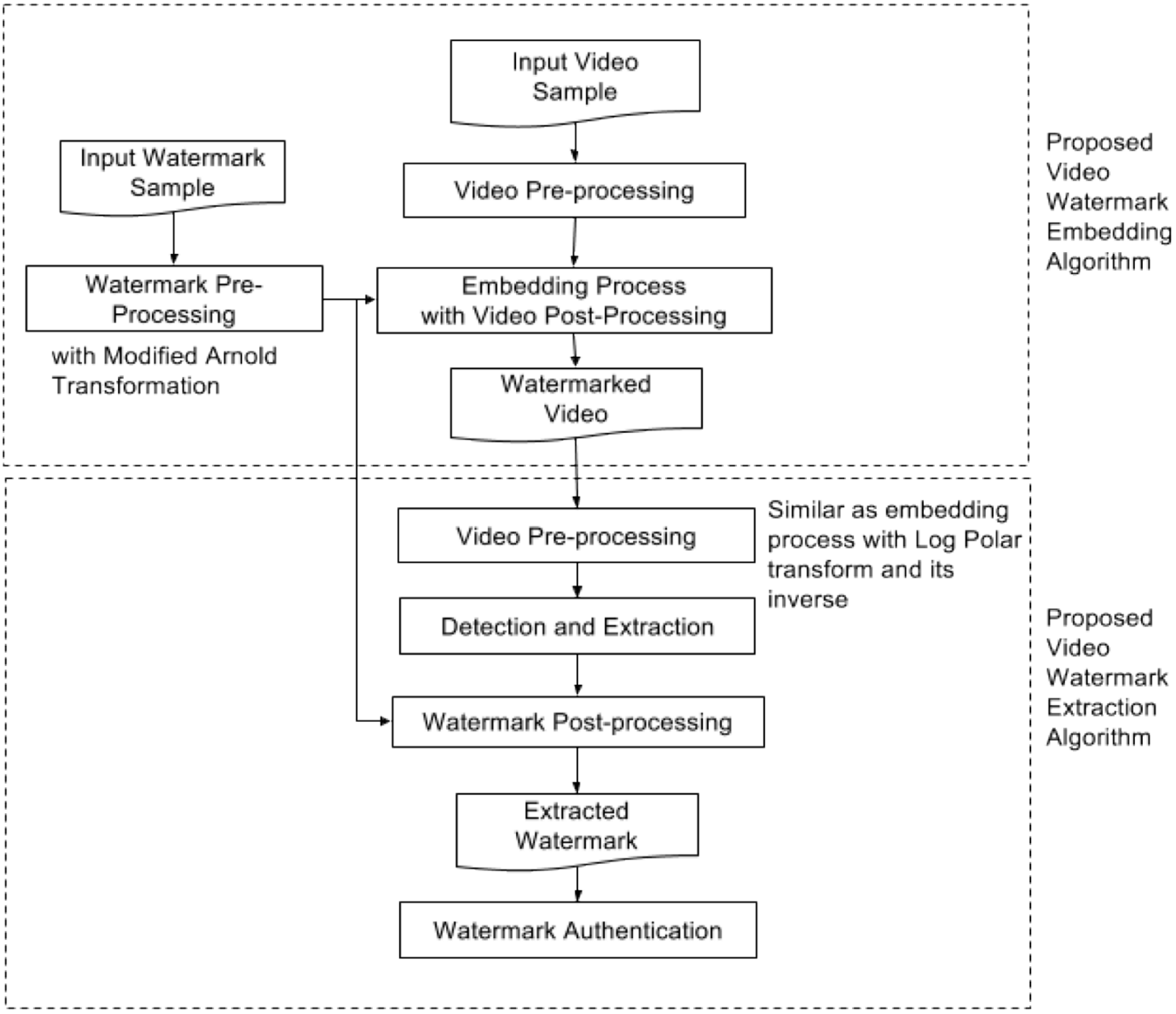


Figure 3. General block diagram of the proposed system.

mark embedding process is done with all the reversing are performed so as to obtain the final watermarked video.

In Extraction stage, similar sections such as video pre-processing and watermark pre-processing are performed along with watermark detection, extraction and post-processing. However, video-preprocessing section is performed in the watermarked video and the original video sample and while performing this section, before reaching the sub-section of Nonsubsampled Contourlet Transform, the additional sub-sections Log Polar Transform and Inverse Log Polar Transform are performed with the videos. In watermark post-processing, the processes related to descrambling and reconstructing the watermark is performed.

4. Proposed Video Watermarking Embedding Algorithm (Embedding Stage)

The embedding process has been expressed in three different steps as watermark pre-processing, video pre-processing and finally watermarks embedding with video post-processing. Figure 4 shows the block diagram of Proposed Video Watermark Embedding Process (Embedding stage) which is described in detail in the following sections. Table 1 shows one of the video frame sample and the watermark sample with its size and format of Table 1 which is going to be tested during implementation.

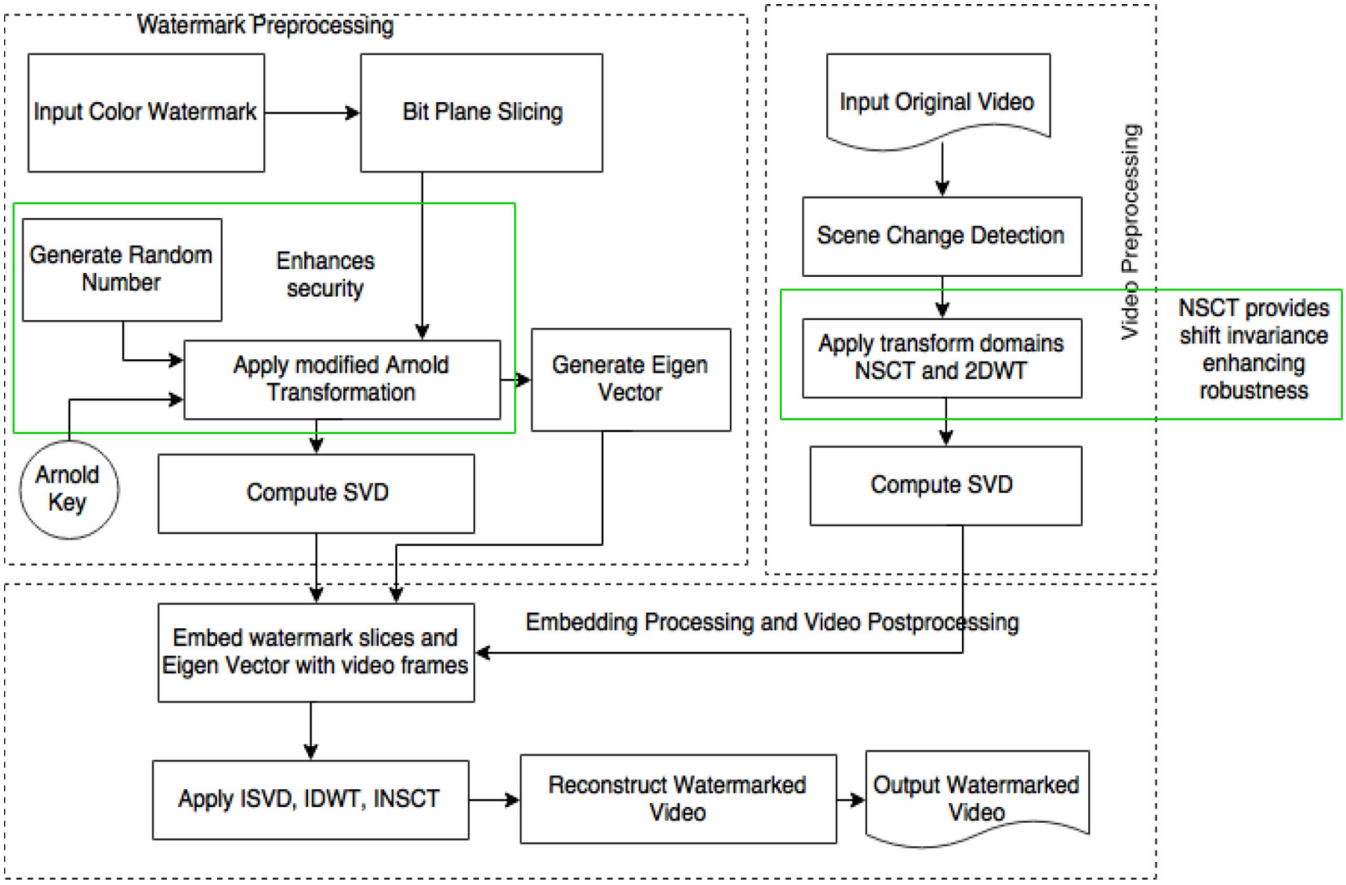


Figure 4. Proposed Video Watermark Embedding Process (Embedding stage).

Table 1. Test video frame sample of suzie.avi and Test Watermark Sample of csulogo.png

Sample Videos Frame	Video Format	Video Size	Sample Watermark	Watermark Format	Watermark Size
	avi	176x144		png	80x80

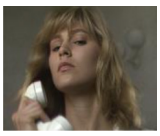
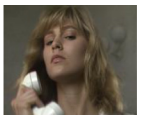
Before embedding watermark in the video frame, at first video pre-processing is done where the transform frequency coefficients of the video frames are obtained. Video pre-processing is required so as to embed watermark in the transformed coefficients of the video frame and hides the watermark information all over the pixels of the frame[13].

4.1 Video Processing

The first step in video pre-processing is the scene change detection. Scene change detection is the process of identifying the key frames of the video[13]. Since, the key frames are highly interconnected to its related frames, working with only these frames are now being a trend while working with the video related tasks[14]. This method provides high robustness against temporal attacks and maintains the real time performance which is achieved by reducing the computational time required for the watermark embedding in each of the video frames[13]. For the scene change detection, a histogram correlation method has been used. Table 2 shows some of the extracted key frames from the test video sample.

The next step of video pre-processing after extracting the key frames is frame color conversion. In video watermarking, color information is not required for embedding the watermark. Only a luminance component of the frame is enough. Hence, RGB key frames are converted into YCbCr components using Matlab tool. Figure 5(a) shows the RGB key frame and Figure 5(b)

Table 2. Extraction of 8 among 24 key frames from test video sample using scene change detection

Video Sample				
Key Frames	37	41	43	45
Video Sample				
Key Frames	47	49	53	57

shows the luminance component of key frame of the sample video. Further watermarking steps are continued only with the obtained luminance component. The third step of video pre-processing is applying Non-subsampled Contourlet Transform on the obtained luminance component of the key frame which provides additional feature of shift invariance. NSCT is an improved Contourlet Transform which is multiscale, multidirectional as well as shift invariant and uses Non-Subsampled Pyramid (NSP) and Non-subsampled Directional Filter Bank (NDFB)[24]. NSCT provides feature to design better frequency filters providing improved sub-band decomposition.

(a)

(b)

Figure 5. (a) RGB frame (b) Y component of 6(a) image after conversion to YCbCr.

Algorithm 1: Video Preprocessing for Embedding Stage
INPUT: Test Video Samples (Vd)
OUTPUT: Singular values of transformed Video Key frames $S1_f$, $S2_f$

Initial k = 1
BEGIN
Step 1: Input Sample Video (Vd) to get the singular values of transformed Video Key frames
Step 2: Read RGB frames of sample video Vd, as $f = f_1, f_2 \ldots f_M$
where, M is the number of video frames of sample video Vd
Step 3: Extract key frames from the video frames
Key frames $[kf_1, kf_2 \ldots kf_p]$ = scene change detection(f)
Where, p is the number of key frames of video Vd with frames f
Step 4: Select only 24 key frames $(kf_1, kf_2 \ldots kf_{24})$
Step 5: Convert selected 24 key frames color to YCbCr
$[ykf_k, cbkf_k, crkf_k] = rgb2ycbcr(kf_k)$
where, ykf_k is luminance component
$cbkf_k$ and $crkf_{kcr}$ are chrominance components,
k = 1,2,..24
Step 6: Apply 1 level NSCT on luminance component ykf_k to get its first level
$[nctL1_k, [D1_k; D2_k; D3_k; D4_k])] = NCT(ykf_k)$
Step 7: Apply 2 level DWT on first level of the results from step 6
$[ll1_k, lh1_k, hl1_k, hh1_k] = DWT(nctL1_k)$
$[ll2_k, lh2_k, hl2_k, hh2_k] = DWT(ll1_k)$
Step 8: Apply SVD on $lh2_k$ and $hl2_k$
$[U1_k, S1_k, V1_k] = SVD(lh2_k)$
$[U2_k, S2_k, V2_k] = SVD(hl2_k)$
Where, $S1_k$ and $S2_k$ denote singular values, $U1_k$, $U2_k$, $V1_k$ and $V2_k$ denote orthogonal values of matrices $lh2_k$ and $hl2_k$
Step 9: Select next 24 key frames kf_k and repeat step 5 to step 8
Step 10: Repeat step 9 until all key frames p are covered
Step 11: End of algorithm 1
END

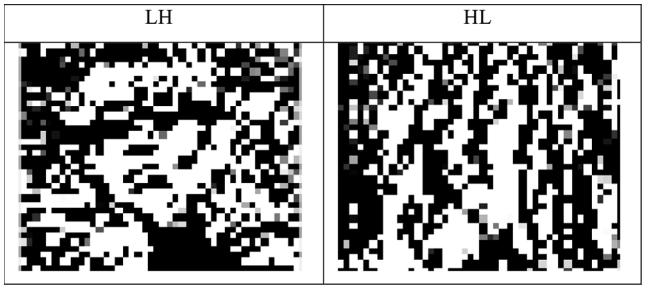


Figure 6. LH and HL Sub-Bands of DWT.

The fourth step in video pre-processing is applying Discrete Wavelet Transform (DWT) in the level 1 approximation sub-bands obtained using NSCT in previous step. DWT allows multi resolution decomposition of an image which decomposes into four parts of frequencies; one low frequency sub band and other three are high frequency sub bands[14]. In the proposed algorithm, 2 level DWT is used whose vertical and horizontal sub-bands of the second level of DWT are used for watermark embedding. The use of these powerful hybrid transform techniques, NSCT and DWT, highly strengthens the watermark to be embedded in the video frame. Figure 6 shows the obtained vertical and horizontal sub-bands of DWT applied in level 1 approximation sub-band obtained from the previous step in the real time sample video.

The final step of video pre-processing is to apply Singular Value Decomposition (SVD) on the mid frequency coefficients LH sub band and HL sub band of DWT obtained in the previous step so as to obtain their singular values where the watermark is to be embedded. SVD is a widely used technique which is applied in most of the signal processing and statistics related matrix analyses and computations[14]. Since, SVD reduces numerical errors which can highly support in extracting more accurate values of the embedded watermark. Following algorithm 1 is the detailed algorithm of video preprocessing for embedding stage.

4.2 Watermark Pre-processing

Watermark pre-processing is the next step of the embedding stage. In watermark pre-processing, the color watermark that is to be embedded is first split into 24 bit slices using the bit plane slicing method for enhancing the imperceptibility of the video to be watermarked. Bit plane slicing is the method of converting an image into multilevel binary images or representing an image with its certain number of bits[25]. Since, the color image has 24 bits, 24 bit planes are extracted ranging from least significant bit to most significant bit of each RGB components. In our proposed embedding algorithm, the CSU logo has been sliced to 24 bit slices with 4 samples. Then, to provide much variation among the bits of the watermark bit slices, each 0 bits of each slices are converted to -1. This process is also followed in watermark pre-processing step in the watermark extraction algorithm.

Algorithm 2: Watermark Preprocessing for Embedding Stage
INPUT: Test Watermark Samples (W)
OUTPUT: a) Principal components of Scrambled Watermark Bit Slices (U_w, S_w)
b) Generated Key (AK)

Initial k = 1
BEGIN
Step 1: Input Sample Watermark (W) to get the 24 scrambled bit slices and then to get the singular values for each of the slices
Step 2: Get R, G and B components of watermark (W)
Step 3: Get 8 bit plane slices for each R, G and B components

$[bs_{w1}, bs_{w2}\ldots bs_{w8}] = BPS(R)$
$[bs_{w9}, bs_{w10}\ldots bs_{w16}] = BPS(G)$
$[bs_{w17}, bs_{w18}\ldots bs_{w24}] = BPS(B)$

Step 4: Convert all 0 bits of 24 slices to -1
Get each bit of bs_{wk}, where k = 1,2…24
if bit is 0
bit = -1
end if
Step 5: Generate random number AK as the key for modified Arnold Transformation
Step 6: Scramble all 24 slices using mAT with generated key AK Get each bit slice bs_{wk}
Apply mAT $ebs_{wk} = E_{AK}(bs_{wk})$
Step 7: Apply SVD on all 24 scrambled bit slices
$[U_{wk}, S_{wk}, V_{wk}] = svd(ebs_{wk})$
Where, S_{wk} singular value and U_{wk}, V_{wk} denote orthogonal values of matrix ebs_{wk}
Step 8: Calculate Eigen Vector of the watermark (W)
Eigen Vector (Vc) = jmax(W)
Step 9: End of algorithm 2.
END

Improving the security of the watermark has done after the above step. Modified Arnold Transform method is used to scramble each of these 24-bit plane slices with the new generated key. Since, this modified Arnold Transform method uses the variable transform coefficients; this will make an intruder more difficult to descramble the watermark of the video. The detail algorithm of the new modified Arnold Transform is presented in algorithm 7 in section VII.

The next step of watermark pre-processing is obtaining a maximum Eigen Vector of a corresponding maximum Eigen Value of the watermark. This Eigen Vector of the watermark is embedded as the watermark information in the video frame so as to provide authentication of the extracted watermark[14]. Eigen Vector is the characteristic vector of a matrix whose direction does not change under related linear transformation[26].This value is obtained using the Matlab tool. Modified Arnold Transformation and Eigen Vector are used for the purpose of providing double security to the watermark.

Finally, in watermark pre-processing, Singular Value Decomposition is applied in Arnold transformed watermark bit slices as in video pre-processing step so as to obtain the principal components of the scrambled bit slices which are to be embedded in the video frame. The principal components are the first two matrices of the resulting decomposed matrices where one is the right orthogonal and other is the singular values matrix. Algorithm 2 presents the detailed of watermark preprocessing for the embedding stage.

4.3 Embedding Process with Video Post-processing

Finally, in embedding process, the principal components of the scrambled watermark bit slices are embedded with the singular values of the mid frequency coefficient (LH) sub band of DWT. Similarly, the Eigen Vector is embedded with the singular values of the mid frequency coefficient (HL) sub band using a robustness factora. After embedding, both of the mid frequencies LH and HL sub-bands of DWT are reconstructed using their new singular values. Then, the new luminance component of the video key frame is obtained by reversing back all the transformations with inverse DWT and then with inverse NSCT. Finally, the RGB key frames are reconstructed with the new Y component and the watermarked video is obtained by combining the new RGB key frames with other non-key frames. The watermarked video frame sample with its PSNR value. The result shows that the imperceptibility of the watermarked video has been highly maintained. Algorithm 3 presents the detailed algorithm of the watermark embedding and video post processing of the embedding stage.

Algorithm 3: Watermark Embedding and Video Post processing of Embedding Stage

INPUT:

a) Principal components of Scrambled Watermark Bit Slices (U_w, S_w)
b) Singular values of transformed Video Keyframes ($S1_f$, $S2_f$)
c) orthogonal components of transformed Video Keyframes, $U1_f$, $V1_f$ and $U2_f$, $V2_f$
d) Eigen Vector of the watermarks (Vc)

OUTPUT: Watermarked videos (Vd')

Initial k = 1

BEGIN

Step 1: Input Singular values of video keyframes $S1_f$ and $S2_f$ and watermark principal components U_w and S_w

Step 2: Compute new singular values of video keyframes

$$S1'_{fk} = S1_{fk} + \alpha\,(U_{wk} * S_{wk})$$

$$S2'_{fk} = S2_{fk} + \alpha\,(Vc)$$

Step 2: Compute inverse SVD

$$lh2'_k = U1_{fk} * S1'_{fk} * V1_{fk}$$

$$hl2'_k = U2_{fk} * S2'_{fk} * V2_{fk}$$

Step 3: Apply inverse DWT

$$ll1'_k = idwt(ll2_k, lh2'_k, hl2'_k, hh2_k)$$

$$nctL1'_k = idwt(ll1'_k, lh1'_k, hl1'_k, hh1_{1k})$$

Step 4: Apply inverse NSCT

$$ykf'_k = inct(nctL1'_k, [D1_k; D2_k; D3_k; D4_k])$$

Step 5: Reconstruct each RGB keyframes color from watermarked luminance component

$$kf'_k = ycbcr2rgb(ykf'_{k,}\ cbkf_{k,}\ crkf_k)$$

Step 6: Regroup all the watermarked key frames with other non watermarked frames to obtain final watermarked video Vd'.

Step 7: Repeat the steps from step 1 to step 6 for all keyframes

Step 8: End of algorithm 3.

END

5. Proposed Video Watermarking Extraction Algorithm (Extracting Stage)

For extraction of the watermark, this algorithm has three steps: video pre-processing, watermark pre-processing, detection and extraction and finally post processing and authentication of the extracted watermark. Since the proposed system is also *non-blind*, both the original video and the original watermark are required in this stage. In extraction stage, before proceeding to its steps, the presence of the watermark is tested by calculating a correlation between the extracted key frame of the watermarked video and the key frame of the original video. If the correlation value is greater than the pre-defined threshold, then the watermark is concluded to be present else no watermark is present. Figure 7 shows the block diagram of the proposed video watermark extraction algorithm.

Algorithm 4: Video Preprocessing for Extraction Stage
INPUT: Test Watermarked Video Samples (Vd')
Original video samples (Vd)
OUTPUT: Singular values of transformed Watermarked Video Keyframes ($S1'_f$, $S2'_f$)

Singular values of transformed Original Video Keyframes ($S1_f$, $S2_f$)
Initial k = 1
BEGIN
Step 1: Input original Video sample (Vd) and watermarked video sample (Vd')
Step 2: Follow step 2 to step 5 of algorithm 1 for both videos Vd and Vd'
(ykf_k, $cbkf_k$, $crkf_k$) = result from step 2 for Vd
($ywkf_k$, $cbwkf_k$, $crwkf_k$) = result from step 2 for Vd'
where, ykf_k and $ywkf_k$ are luminance component

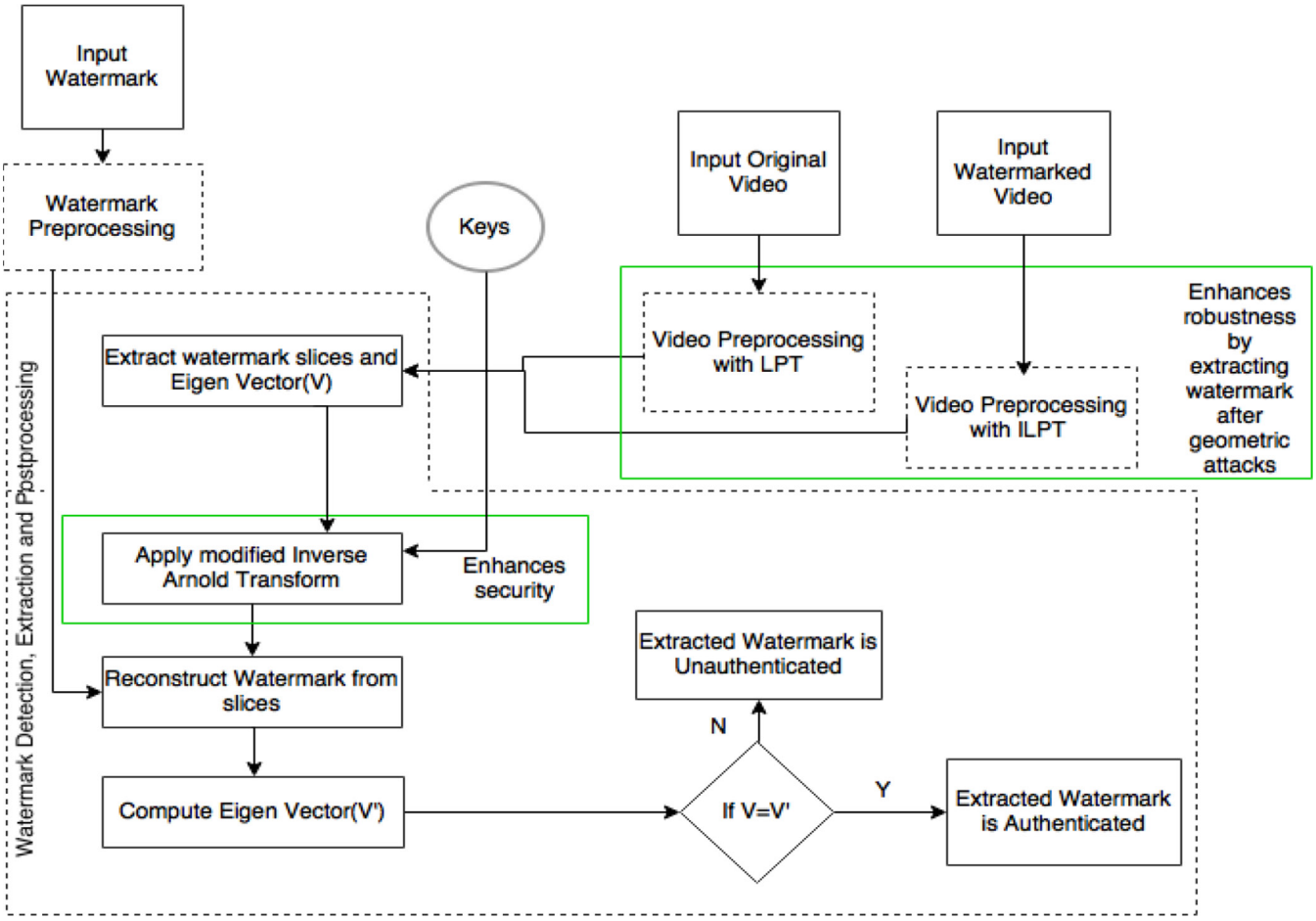


Figure 7. Proposed Video Watermark Extraction Algorithm (Extraction stage).

$cbkf_k$, $cbwkf_k$, $crkf_k$ and $crwkf_k$ are chrominance components
k = 1,2…24
Step 3: Apply LPT on luminance components of Vd keyframes
$lpt_k = LPT(ykf_k)$
Assign $ykf_k = lpt_k$
Step 4: Follow step 6 to step 8 of algorithm 1 for result of step 3
$[S1_{f1}, S1_{f2} \ldots, S1_{f24}]$ = result from step 4
$[S2_{f1}, S2_{f2} \ldots, S2_{f24}]$ = result from step 4
Step 5: Apply ILPT on luminance components of Vd' keyframes
$ilpt_k = ILPT(ywkf_k)$
Assign $ykf_k = ilpt_k$
Step 6: Follow step 6 to step 8 of algorithm 1 for result of step 5
$[S1'_{f1}, S1'_{f2} \ldots, S1'_{f24}]$ = result from step 6
$[S2'_{f1}, S2'_{f2} \ldots, S2'_{f24}]$ = result from step 6
Step 7: End of algorithm 4.
END

5.1 Video Processing

The first step in the extraction stage is also video pre-processing. In this step, video pre-processing is done for both the original and the watermarked videos. Similar procedures of video pre-processing are followed as in the embedding stage to identify the singular values of the transformed original video frames and the watermarked video frames. However, in video pre-processing of this stage, before computing NSCT, Inverse Log Polar Transform and Log Polar Transform are applied on the luminance component of the watermarked video key frame and the original video key frame respectively.

The use of Log Polar Transform (LPT) and Inverse Log Polar Transform (ILPT) in the extraction process only is the main contribution of our proposed algorithm. Figure 8 (a) shows the result of LPT applied in the Y component of the original sample video key frame and Figure 8 (b) shows the result of ILPT applied in the Y component of the watermarked sample video key frame. Log Polar Transform is a conformal mapping of Cartesian plane points (x, y) to Log Polar plane points (ρ,θ) where r is the logarithmic distance between origin and the given point and q is the angle between the line of point from an origin and a line of reference. Whereas, Inverse Log Polar Transform is used for mapping Cartesian plane points (x, y) from Log Polar plane points (ρ,θ).

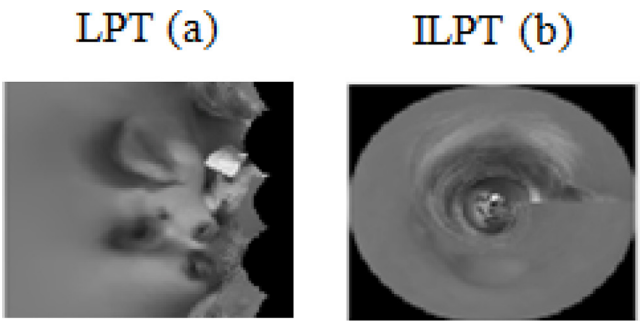


Figure 8. (a) LPT in original video frame (b) ILPT in watermarked video frame.

The features of angle invariance and distance invariability provided by LPT are used to resist rotation and scaling attacks of geometric attacks[27]. Both LPT and ILPT uses bilinear interpolation while transforming. According to[20], Inverse Log Polar Mapping(ILPM) done by ILPT uses bilinear interpolation and introduces distortion in the watermarked image and based on[28], the watermarked bits will be scattered in multiple frequencies while applying ILPM[29] has expressed that geometric distortion can be reversed back using the original image. Hence, the discretization property features of angle and distance invariability of LPT while resampling and distortion property of ILPT has been utilized. This process is applied in extraction stage only where quality of watermarked video does not have to be maintained so the features of LPT and ILPT are utilized and the quality of watermarked video is also preserved. Implementing this step has highly provided the robustness against geometric attacks maintaining other image processing and video processing attacks. Algorithm 4 shows the detailed steps for video preprocessing for extraction stage.

5.2 Watermarking Pre-processing

The next step in the extraction stage is also watermarking pre-processing. Since our proposed system is *non-blind*, the original watermark is required. In this step, similar procedures of watermark pre-processing in the embedding stage is done to obtain the right orthogonal value of the original watermark which is used in watermark post processing step of the extraction stage. However, the process of generating the key is not done in this step of the extraction stage. Instead, the key generated in watermark

pre-processing step of the embedding stage is used for modified Arnold Transformation. Algorithm 5 shows the detailed steps for watermark preprocessing for extraction stage.

Algorithm 5: Watermark Preprocessing for Extraction Stage
INPUT: a) Test Watermark Samples (W)
b) Generated Key (AK)
OUTPUT: a) Left Orthogonal component of Scrambled Watermark Bit Slices (V_w)

BEGIN
Step 1: Input Sample Watermark (W) and generated key AK to get the 24 scrambled bit slices and then to get the left orthogonal values for each of the slices
Step 2: Follow step 2 to step 4 of algorithm 2 of embedding stage
Step 3: Scramble each generated slices using mAT with input generated key AK and follow step 6 and step 7
Step 4: End of algorithm 5
END

5.3 Watermarking Pre-processing

The final step of the extraction stage is detection, extraction and post processing of the watermark. From the obtained singular values of vertical mid frequency sub-band LH of the original and the watermarked video key frames in the previous step, the principal components of the watermark are extracted. Similarly, from the obtained singular values of horizontal mid frequency sub-band HL of the original and the watermarked video key frames in the previous step, the embedded Eigen Vector of the watermark is extracted using same robustness factor a.

Finally, in watermark post processing, the scrambled bit slices of the watermark are reconstructed using the extracted principal components of the watermark and the orthogonal value obtained from the previous watermark preprocessing step. Then, the watermark bit planes are reconstructed from the scrambled bit slices using modified Inverse Arnold Transformation with the same generated key by an authorized user only. Figure 9 (a) shows the extracted scrambled bit slice and Figure 9 (b) is the result of the modified Inverse Arnold Transform.

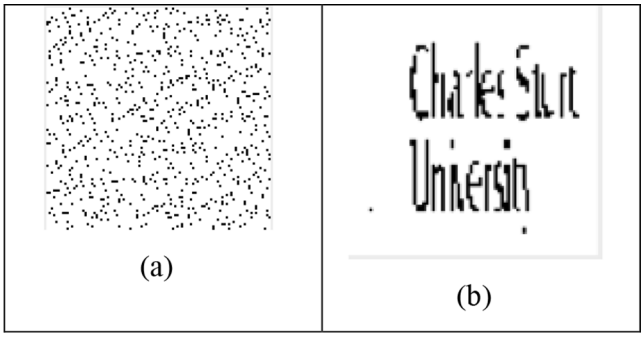


Figure 9. (a) Scrambled 8 Bit Slice of the Watermark (b) Result of Modified Inverse Arnold Transform applied to 9(a).

All the extracted 24 watermark bit slices are grouped together to obtain the image of watermark. Figure 10 shows the extracted watermark with its NCC value which shows that high robustness of the watermark has been maintained for no attacks. Furthermore, the Eigen Vector of the extracted watermark is calculated and compared with extracted Eigen Vector to verify the authenticity of the video. If the vector is same, then the video content is considered to be unchanged else it is assumed that there have been some changes in the video contents. Algorithm 6 shows the detailed steps for detection, extraction and preprocessing of watermark for extraction stage.

Extracted Watermark	Robustness (NCC)
	0.9973

Figure 10. Extracted watermark with its NCC value.

Algorithm 6: Detection, Extraction and Preprocessing of watermark for Extraction Stage
INPUT:
a) Singular values of transformed Watermarked Video Keyframes ($S1'_f$, $S2'_f$)
b) Singular values of transformed Original Video Keyframes ($S1_{if}$, $S2_{if}$)
c) Left Orthogonal component of Scrambled Watermark Bit Slices (V_w)
d) Arnold Key (AK)
OUTPUT: Extracted watermark (W')

Initial k = 0

BEGIN
Step 1: Enter all the inputs
Step 2: Extract principal component of bit slice of watermark and Eigen Vector

$$U'_{wk} * S'_{wk} = (S1'_{fk} - S1_{fk}) / a$$

$$Vc'_k = (S2'_{fk} - S2_{fk}) / a$$

Where, a is robustness factor
Step 6: Apply inverse SVD

$$ebs'_{wk} = U'_{wk} * S'_{wk} * V_{wk}{}^T$$

Step 7: Apply ImAT using key AK
Start loop
$bs'_{wk} = D_{AKk}(ebs'_{wk})$
end for loop
Step 8: Calculate threshold of matrix values of bs'_{wk}
if bs'_{wk} > = threshold/2, then $bs'_{wk} = 1$
else $ebs'_{wk} = 0$
Step 9: Repeat step 3 to step 8 to get all other remaining 23 slices of extracted watermark bits
Step 10: Group first 8 slices, second 8 slices and third 8 slices to extract RGB watermark W'
Step 11: Calculate Eigen Vector Vc" of extracted watermark W'
Step 12: Compare Vc" with Vc', as,
comp = Vc'- Vc"
If comp = 0,
Extracted watermark is authentic
Else
Extracted watermark is not authentic
Step 13: Repeat the steps from step 1 to step 13 for all keyframes
Step 14: End of algorithm
END

6. Modified Arnold Transformation

6.1 Arnold Transform

In our proposed solution, instead of using a Traditional Arnold Transform which is used in the algorithm by[14], the modified Arnold Transform method is used to scramble each of these plane slices with the new generated key to improve the security of the watermark.

6.1.1 Traditional Arnold Transform

Arnold Transform is a simple tool used for changing one matrix form into other matrix by randomly shuffling the pixels of an image for a certain period[14]. Arnold transformation is applied to N*N images as in given formula 1.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \mathrm{Mod}(\mathrm{N}) \quad (1)$$

And Inverse Arnold Transformation is obtained by using given formula 2.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \mathrm{Mod}(\mathrm{N}) \quad (2)$$

where,
x, y => coordinates of the original image
x', y' => coordinates of transformed image
N => size of image to be transformed
The main limitation of using the traditional Arnold Transformation is that the transform coefficients used are all fixed and if the use of traditional Arnold Transform is known, then one can somehow descramble the image using these fixed coefficients[30].

6.1.2 Modified Arnold Transform (mAT)

Based on the theory provided by[30,31], modified Arnold Transformation is introduced in our proposed algorithm. The feature of a matrix determinant has been utilized to modify this algorithm. In this algorithm, the transform coefficients of Arnold transform are extracted in such a way that the determinant of matrix coefficients is 1 as shown in formula 3.

$$a_{00} * a_{11} - a_{10} * a_{01} = 1 \quad (3)$$

where, a_{00}, a_{11}, a_{10} and a_{01} are the transform coefficients
At first, a key is given as an Arnold Key AK_1 which is used as a period for the scrambling watermark. Then, a random number is generated which is represented as generated key GK_1 and the next determinant value of the transform coefficient is calculated by subtracting the given key by 1. Then, the two highest multipliers of the given key and the resulting value are determined. Finally, replace all the transform coefficients of the Arnold formula are replaced by the extracted multipliers and the transformation is performed using the new obtained period using the given formula 4.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \mathrm{Mod}(\mathrm{N}) \quad (4)$$

where, x, y and x', y' are the coordinates of the original and scrambled bit respectively, N is the size of watermark and AK_1 is the period of scrambling the watermark.

Modification of Arnold Transformation is the next contribution of our proposed algorithm. This algorithm uses variable transform coefficients instead of static values of 1 and 2. Below algorithm 7 is the detailed algorithm of modified Arnold Transformation.

Algorithm 7: Modified Arnold Transform
INPUT: Bit Slices of test watermark sample bs = {bs_{w1}, bs_{w2}… bs_{w24}}
Arnold Key AK as Arnold Period
Generated key GK
OUTPUT: Scrambled slices of test watermark sample ebs = {ebs_{w1}, ebs_{w2}… ebs_{w24}}

BEGIN
Initial k = 1
Step 1: Let a_{00} * a_{11} = GK
Step 2: Then, a_{10} * a_{01} = GK'$_=$ GK-1
Step 3: Calculate highest multipliers of GK and GK' to obtain a_{00}, a_{11}, a_{10} and a_{01} respectively
Step 4: Get scrambled slices ebs_{wk} by computing modified Arnold Transformation for each bit of bs_{wk} for AK periods using the formula,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a00 & a01 \\ a10 & a11 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \mathrm{Mod}(\mathrm{N})$$

where,
x, y => coordinates of the original image
x', y' => coordinates of the transformed image
N => size of image to be transformed
Step 5: End of algorithm 7

7. Modified Insert Arnold Transformation

The proposed inverse Arnold Transform is a modified version of the traditional inverse Arnold transform. Similarly, as modified Arnold Transform, the inverse also has varying transform values but the orientation of those values are different using the given formula 8.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a00 & a01 \\ a10 & a11 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \mathrm{Mod}(\mathrm{N}) \quad (5)$$

The detail algorithm 8 is the modified Inverse Arnold Transform algorithm (mIAT).

Algorithm 8: Modified Inverse Arnold Transform
INPUT: Scrambled slices of test watermark sample ebs = {ebs_{iw}, ebs_{w2}… ebs_{w24}}
Arnold Key AK as Arnold Period
Generated key (while embedding process) GK
OUTPUT: Bit Slices of extracted watermark sample bs = {bs_{w1}, bs_{w2}… bs_{w24}}

initial k = 1
BEGIN
Step 1: Repeat step 1 to step 3 as in modified Arnold transformation
Step 2: Get bit slices bs_{wk} by computing modified Inverse Arnold Transformation for each bit of ebs_{wk} for AK periods using the formula,

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a11 & -a10 \\ -a01 & a00 \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \mathrm{Mod}(\mathrm{N})$$

where, x, y => coordinates of original image
x', y' => coordinates of transformed image
N => size of image to be transformed
Step 4: End of algorithm 8.
END

8. Results and Discussion

The implementation was carried out using Matlab R2015b on 10 sample videos and 10 color images as the watermark of different sizes and formats. The sample videos and watermarks are collected from Google database. Table 1 is one of them. The performance of video watermarking algorithm is measured in terms of robustness, imperceptibility, security, computational time and capacity.

Since our proposed system is highly focused on improving the robustness against geometric attacks without affecting the robustness against other attacks, the results and discussions presented below are focused on the comparisons of robustness of the current and proposed system in different attack situations such as several image processing, video processing and geometric attacks.

8.1 Quality Metrics

a) Peak Signal-to-Noise Ratio (PSNR):

Peak Signal-to-Noise Ratio is the common metric used to measure the quality of the watermarked video

frame with the original video frame. This metric is used to measure the imperceptibility or invisibility of the watermark in the video frame.

It is calculated using the given equation 6.

$$\text{PSNR} = 10 \lg \left(\frac{N * M * 255^2}{\sum_{i=1}^{N} \sum_{j=1}^{M} (X_w(i,j) - X(i,j))^2} \right) \quad (6)$$

Where, X(i,j) is the original pixel, X_w(i,j) is the watermarked pixel and N*M is the size of the video frame. The result of PSNR of watermarked video of proposed system is shown in Figure 11.

Watermarked Video Frame	Imperceptibility (PSNR)
	64dB

Figure 11. Watermarked Video Frame Sample with its PSNR Value.

b) Normalized Correlation Coefficient (NCC):

Normalized Correlation Coefficient is another common metric which is used to evaluate the correlation of the extracted watermark with the original watermark. If they are correlated, then the value will be closer to 1 else it will be closer to 0. NCC value can be calculated by using equation 7.

$$\text{NCC} = \frac{\sum \left((OW_i - OW_m)(EW_i - EW_m) \right)}{\sqrt{\sum (OW_i - OW_m)^2} \sqrt{\sum (EW_i - EW_m)^2}} \quad (7)$$

Where,

OW_i and EW_i are the intensity value of i^{th} pixel in the original watermark and the extracted watermark and OW_m and EW_m are the mean intensity value of the original watermark and the extracted watermark.

8.2 Results and Discussion

The results and discussions presented below are focused on the comparisons of robustness of the current and the proposed system in different attack situations. For measuring the performance of robustness, quality metric Normalized Correlation Coefficient (NCC) is used.

1. ***No attacks:*** The evaluation of the effects of the watermark embedding in the video frame can be done for the robustness when there are no attacks. Table 3 shows the result of NCC value of one of the 10 sample videos with sample watermark without any attacks of Table 2. The table shows that with no attacks, nearly 99% of the watermark can be extracted.
2. ***Image Processing Attacks:*** For evaluating the robustness of the proposed algorithm, various image processing attacks such as Salt and Pepper noise attack with noise density 0.01 and 0.03, Gaussian noise attack with variance of 0.01 and 0.1, Poisson noise attack, Median Filtering attack, Contrast Adjustment attack and Histogram attack Gaussian are performed for both the current and the proposed system and evaluated. Figure 13 shows the results of extracted watermarks with its NCC values for all the mentioned image processing attacks tested in one of the 10 test video samples or the proposed algorithm while comparing the same processes with the current algorithm. From the current algorithm, the percentage of watermark extracted from several image processing attacked videos are 75% to 99% from *salt and pepper noise attacks*, 70% to 99% from *Gaussian noise attacks*, 94% to 99% from *Poisson noise attacks*,

Table 3. Results of extracted watermark with ncc value applied in 10 sample watermarked video frames with no attacks.

Proposed Algorithm (PRVWA-EEfGA) **: No attacks**				
Sample Watermark	**Sample Watermark**	**Watermarked Video Frame**	**Extracted Watermark**	**Robustness (NCC)**
				0.9973

80% to 99% from *median filtering attacks*, 86% to 99% from *contrast adjustment attacks* and *Histogram attacks*. Moreover, from the proposed algorithm, the percentage of watermark extracted from several image processing attacked videos are 83% to 99% from *salt and pepper noise attacks*, 81% to 99% from *Gaussian noise attacks*, 93% to 99% from *Poisson noise attacks* and *median filtering attacks*, 80% to 99% from *contrast adjustment attacks* and *Histogram attacks.* The table clearly shows that the robustness for all the image processing attacks which has been achieved in the current algorithm has also been maintained in our proposed algorithm.

3. ***Temporal Attacks:*** The performance of the video watermarking algorithm is also evaluated based on temporal attacks such as frame dropping and swapping. From the result shown in Table 3 for proposed algorithm, we were able to extract 98% of the watermark by dropping 21% frames and minimum of 80% of the watermark by dropping 96% frames and for current algorithm, we were able to extract around 99% of the watermark by dropping 21% frames and minimum of 82% of the watermark by dropping 96% frames. Furthermore, from the result shown in Table 3 for proposed algorithm, we were able to extract minimum 64% to maximum 96% of the watermark after frame swapping by 25% and for the current algorithm, we were able to extract around minimum 65% to 97% of the watermark.
4. ***Geometric attacks:*** Robustness against geometric attack is one of the main issue that our proposed video watermarking algorithm has focused. To evaluate the performance of our proposed algorithm, three main geometric attacks: rotation attack, scaling attack and transition attack have been tested. Like other attacks, all these geometric attacks tests were conducted for one of the 10 watermarked videos for both the proposed and the current algorithms.

a. Rotation attack: For evaluating rotation attack, NCC values of the extracted watermarks from the different angles of the rotated watermarked videos have been measured. Figure 14 shows the result for the attacked videos with different geometric attacks for the current and proposed algorithm. From the table, the resulting values of NCC clearly shows that our proposed algorithm is able to extract minimum of 93% and maximum of 99% of the watermark for all rotated watermarked videos. However, the resulting NCC values of current algorithm is 99% only for rotation of 180^0 and maximum of 95% extraction for 1^0. For all other degrees of rotation for current algorithm, the NCC values ranged from 20% to 88%. The comparison table clearly shows that our proposed algorithm has highly enhanced robustness against rotation attack for all the degrees of rotation.

b. Scaling attack: For evaluating scaling attack, we resized the watermarked video to 100*100 pixels less than the original size of video and then resized back to its original size, extracted the watermark from this scaled watermarked video and calculated NCC value of the extracted watermark for measuring robustness. From the Table Figure 14, the resulting values of NCC clearly shows that our proposed algorithm is able to extract minimum of 97% and maximum of 99% of the watermark for all scaled watermarked videos. However, the resulting NCC values of the current algorithm is maximum around 95%. For most of the scaled watermarked video for the current algorithm, the NCC values ranged from 68% to 84%. The comparison table clearly shows that our proposed algorithm has highly enhanced robustness against scaling attack.

c. Translation attack: For evaluating translation attack, we translated the watermarked video by 10.3 in the x-direction and -10.1 in the y-direction, extracted the watermark from this translated watermarked video and calculated NCC value of the extracted watermark for measuring robustness. From the Figure 14, the resulting values of NCC clearly shows that our proposed algorithm is able to extract minimum of 94% and maximum of 99% of the watermark for all translated watermarked videos. However, the resulting NCC values of the current algorithm is maximum around 88%. For most of the translated watermarked video for current algorithm, the NCC values ranged from 38% to 80%. The comparison table clearly shows that our proposed algorithm has highly enhanced robustness against translation attack.

Hence, Tables 3–6 shows the clear comparison tables between the current algorithm and the proposed system for the NCC values calculated for image processing and temporal attacks, rotation attacks, scaling attacks and transition attacks respectively. Figure 12 represents the comparison of NCC values for different image processing and temporal attacks between the current algorithm and the proposed algorithm where the NCC values for each attack is the average of NCC values of 10 test video samples. From the Table 3 and Figure 12 with the above result discussions for each attack cases in image process-

ing and temporal attacks, the results clearly show that the proposed system has maintained the high NCC values as in current algorithms. Minimum average of 80% and maximum of 99% of the watermark can be extracted from the different image processing and temporal attacks from the watermarked video using the proposed system as well as the current algorithm which shows that the robustness against image processing attacks and temporal attacks are moreover similar.

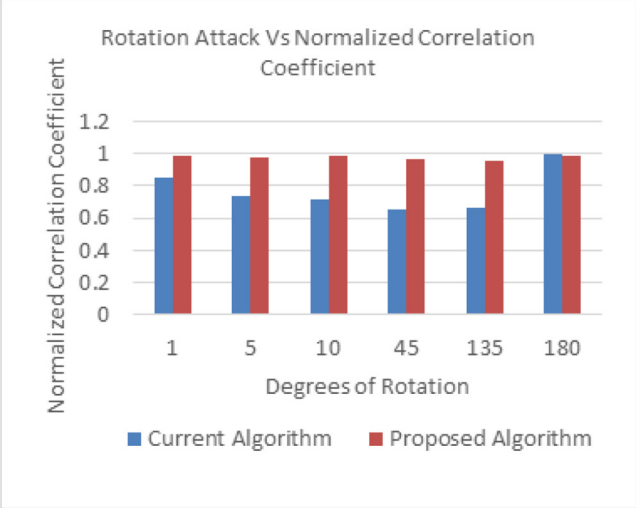


Figure 12. Comparison of NCC values for different degrees of rotation between current algorithm and proposed algorithm.

Since our system is more focused on geometric attacks, we have created graphs for all geometric attacks vs NCC to show further comparisons from the result of 10 sample videos. Figure 13 represents the comparison of NCC values for different degrees of rotation between the current algorithm and the proposed algorithm where the NCC values for each degree is the average of NCC values of 10 test video samples. From the comparison Figure 13, we can conclude that our proposed system has highly enhanced the robustness against rotation attack by increasing the NCC value of watermark from 0.65 to 0.99. Hence, using our proposed system, 99% of watermark can be extracted from any degrees of rotation attacked watermarked videos while current system can only extract around 65% of the watermark.

Moreover, Figure 14 represents the comparison of the NCC values for scaling attack and transition attack between the current algorithm and the proposed algorithm where the NCC values for scaling attack and transition attack are the average of NCC values of 10 test video samples. From the comparison Figure 14, we can conclude that our proposed system has highly enhanced the robustness against scaling attack by increasing the NCC value of watermark from 0.77 to 0.99. Hence, using our proposed system, 99% of the watermark can be extracted from any scaling attacked videos while current system can only extract around 77% of watermark.

Furthermore, from the comparison Figure 14, we can also conclude that our proposed system has also highly enhanced the robustness against transition attack by increasing the NCC value of the watermark from 0.64 to 0.99. Hence, using our proposed system, 99% of the watermark can be extracted from any transition attacked videos while current system can only extract around 64% of the watermark.

Therefore, from the comparison tables and Figure with graphs, we can conclude that enhancing the feature of shift invariance of NSCT while embedding and using the features angle invariance and distance invariance of LPT and distortion of ILPT in the original video and watermarked video respectively while extraction, has highly enhanced the robustness of geometric attacks from minimum 64% to 99% of NCC values.

Furthermore, while increasing the robustness of video watermarking algorithm, we have maintained the computational time as low as possible. Figure 14 shows the computational time for watermark embedding and extraction for the current and the proposed algorithms. From the Figure, we can view that only slight increment in extraction time has occurred which is negligible.

For measuring the performance of imperceptibility, quality metrics Peak Signal to Noise Ratio (PSNR) is used. The imperceptibility of the proposed system has been highly maintained with PSNR value of around 64dB which is shown in Figure 11. Moreover, the variable transform coefficients of modified Arnold Transformation ensure high security of the watermark than traditional Arnold Transformation. Also, for capacity evaluation, since 24 bit slices of color watermark are embedded with 24 key frames of the video with 'N' number of frames, maximum of (N-non key frames)/24 RGB images can be embed as in the current algorithm.

Hence, from the above results, discussion and comparisons, our proposed system has successfully achieved high robustness against geometric attacks maintaining high robustness for other attacks, imperceptibility, payload and computational time along with more secure Arnold Transform.

Table 4. Results for robustness against different image processing and temporal attacks for current algorithm and proposed algorithm

Sample Rotated Video Frames	Current Algorithm[14]		Proposed Algorithm (PRVWA-EEfGA)	
	Extracted Watermark	Robustness (NCC)	Extracted Watermark	Robustness (NCC)
Robustness Against Different Rotation Attacks with different angles in degree				
 1		0.8635	 1	0.9972
 5		0.3013	 5	0.9968
 10		0.3249	 10	0.9970
 45		0.4057	 45	0.9971
 135		0.5310	 135	0.9967
 180		0.9999	 180	0.9968
Robustness Against Scaling Attack:(Resized by Width-100, Height-100				
		0.6815		0.9973
Robustness Against Different Transition Attacks with transition Value = x=10.3, y=-10.1				
		0.7266		0.9977

9. Conclusion

In conclusion, the need for digital video watermarking in today's evolving technological environment is high and development of robust mechanisms is a priority in terms of copyright protection and authentication. A significant number of algorithms have been introduced in order to provide the best possible features required for video watermarking purposes. The existing algorithm proposed by Agilandeeswari[14] have also provided the best possible results required for video watermarking. However, this algorithm lacked to provide some of the most important features. Our proposed system is based on this algorithm and has highly reduced the limitations found in the current algorithm.

Table 5. Result for robustness against different geometric attacks for current algorithm and proposed algorithm

Sample Video Frame	Sample Watermark	Robustness for Image Processing Attacks and Temporal Attacks (NCC value)							
		Salt & Pepper Attack (var = 0.03) (var = 0.01)	Gaussian Noise Attack (var = 0.1) (var = 0.01)	Poisson noise Attack	Median Filtering Attack	Contrast Adjustment	Histogram Attack	Frame Dropping Attack 21%,58%,96%	Frame Swapping Attack 25%
Current Algorithm[14]									
		 0.8438  0.9557	 0.7025  0.7497	 0.9822	 0.9852	 0.9993	 0.9678	 0.9525  0.9368  0.8200	 0.9622
Proposed Algorithm (PRVWA-EEfGA)									
		 0.9972  0.9973	 0.9968  .9974	 0.9974	 0.9974	 0.9975	 0.9976	 0.9405  0.9259  0.7999	 0.9622

Table 6. Result for computational time taken by each frame for embedding and extraction by current algorithm and proposed algorithm

Sample Video Frame	Sample Watermark	Embedding and Extraction Time	
		Current Algorithm[14]	Proposed Algorithm (PRVWA-EEfGA)
		0.07s 0.06s	0.07s 0.093s

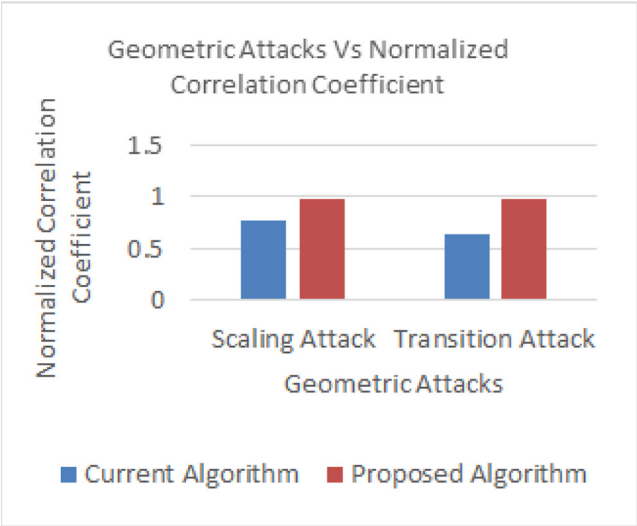


Figure 13. Comparison of NCC values for scaling attack and transition attack between current algorithm and proposed algorithm.

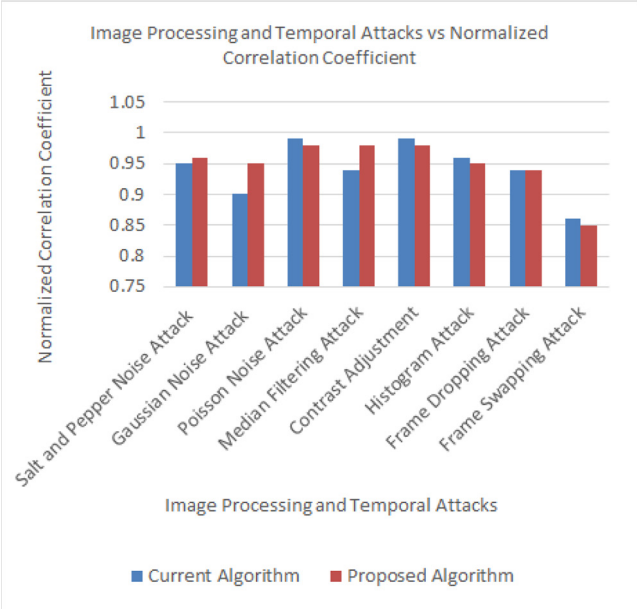


Figure 14. Comparison of NCC values for image processing and temporal attacks between current algorithm and proposed algorithm.

10. References

1. Lamberti F, Sanna A, Paravati G. Computer-assisted analysis of painting brushstrokes: digital image processing for unsupervised extraction of visible features from van Gogh's works. EURASIP Journal on Image and Video Processing. 2014; 53(1):1–17. https://doi.org/10.1186/1687-5281-2014-53
2. Padmapriya P, Manikandan K, Jeyanthi K, Renuga V, Sivaraman J. Detection and classification of brain tumor using radial basis function. Indian Journal of Science and Technology. 2016; 9(1):1–5. https://doi.org/10.17485/ijst/2016/v9i1/85758
3. Cheng B, Yang J, Yan S, Fu Y, Huang T. Learning with L1- graph for Image Analysis. IEEE Transactions Image Processing. 2010; 19(4):858–66. https://doi.org/10.1109/TIP.2009.2038764. PMid:20031500
4. SenthilKumar NK, Kumar KK, Rajkumar N, Amsavalli K. Search engine optimization by fuzzy classification and prediction. Indian Journal of Science and Technology. 2016; 9(2):1–12.
5. Condorovici R, Florea C, Vertan C. Author identification for digitized paintings collections. IEEE International Symposium in Signals Circuits and Systems; 2013. p. 1–4. https://doi.org/10.1109/ISSCS.2013.6651197
6. Rashmi A. Design and development of data classification methodology for uncertain data. Indian Journal of Scienceand Technology. 2016; 9(3):1–12.
7. Qiao L, Chen S, Tan X. Sparsity preserving projections with application to face recognition. Pattern Recognition. 2011; 33(1):331–41. https://doi.org/10.1016/j.patcog.2009.05.005
8. Sivakumar S, Selvaraj R. Predictive modeling of students performance through the enhanced decision tree. Advances in Electronics. Communication and Computing; 2017. p. 21–36.
9. Qi H, Taeb A, Hughes S M. Visual stylometry using background selection and wavelet-HMT-based Fisher information distances for attribution and dating of impressionist paintings. Signal Processing. 2013; 93(3):541–53. https://doi.org/10.1016/j.sigpro.2012.09.025
10. Amarnath S, Appavu S. Metaheuristic approach for efficient feature selection: A data classification perspective. Indian Journal of Science and Technology. 2016; 9(4):1–6. https://doi.org/10.17485/ijst/2016/v9i4/87039

11. Chandrappa DN, Ravishankar M, Babe RDR. Face detection in color images using skin color model algorithm based on skin color information. 3rd International Conference on Electronics Computer Technology; 2011. p. 254–58. https://doi.org/10.1109/ICECTECH.2011.5941600
12. Uma KV, Appavu S. Classification of adverse event thyroid cancer using naïve entropy and association function. Indian Journal of Science and Technology. 2016; 9(4):1–7. https://doi.org/10.17485/ijst/2016/v9i4/87042
13. Abry P, Wendt H, Jaffard S. When Van Gogh meets Mandelbrot: Multifractal classification of painting's texture. Signal Processing. 2013; 93(3):554–72. https://doi.org/10.1016/j.sigpro.2012.01.016
14. Zhipeng C, Junda J, Hu H, Wenbin Z. Face detection system based on skin color model. IEEE International Conference on Networking and Digital Society; 2010. p. 664–67. https://doi.org/10.1109/ICNDS.2010.5479392
15. Khan FS, Beigpour S, Weijer J, Felsberg M. Painting-91: A large scale database for computational painting categorization. Machine Vision and Applications. 2014; 25(6): 1385–97. https://doi.org/10.1007/s00138-014-0621-6
16. Tayal Y, Lamba R, Padhee S. Automatic face detection using color based segmentation. International Journal of Scientific and Research Publications. 2012; 2(6):1–7.
17. Saleh B, Abe K, Arora R, Elgammal A. Toward automated discovery of artistic influence. Multimedia Tools and Applications; 2014. p. 1–27.
18. Hemalatha G, Sumathi CP. A Study of techniques for facial detection and expression classification. International Journal of Computer Science and Engineering Survey. 2014; 5(2):27–37. https://doi.org/10.5121/ijcses.2014.5203
19. Cetinic E, Grgic S. Automated painter recognition based on image feature extraction. 55th IEEE International Symposium in ELMAR; 2013. p. 19–22.
20. Jacobsen CR, Nielsen M. Stylometry of paintings using hidden Markov modelling of contourlet transforms. Signal Processing. 2013; 93(3):579–91. https://doi.org/10.1016/j.sigpro.2012.09.019
21. Kim M, Kim J. Quantitative analysis of artists' characteristic styles through biologically-motivated image processing techniques: Uncovering a mentor to Johannes Vermeer. HCI International Posters Extended Abstracts; 2013. p. 258–62.
22. Zheng Y, Nie X, Meng Z, Feng W, Zhang K. Layered modeling and generation of Pollock's drip style. The Visual Computer. 2014; 31(5):589–600. https://doi.org/10.1007/s00371-014-0985-7
23. Atharifard A, Ghofrani S. Robust component-based face detection using color feature. World Congress on Engineering. 2011; 2:6–8.
24. HYang H, Lu J, Brown WP, Daubechies I, Ying L. Quantitative canvas weave analysis using 2-D synchrosqueezed transforms: Application of time-frequency analysis to art investigation. IEEE Signal Processing Magazine. 2015; 32(4):55–63. https://doi.org/10.1109/MSP.2015.2406882
25. Chitra S, Balakrishnan G. Comparative study for two color spaces HSCbCr and YCbCr in skin color detection. Applied Mathematical Sciences. 2012; 6(85):4229–38.
26. Manzo M, Petrosino A. Attributed relational sift-based regions graph for art painting retrieval. Image Analysis and Processing–ICIAP; 2013. p. 833–42. https://doi.org/10.1007/978-3-642-41181-6_84
27. Jankowski M. Erosion, dilation and related operators. 8th International Mathematica Symposium; 2006. p. 1–6.
28. Liu H, Chan RH, Yao Y. Geometric tight frame based stylometry for art authentication of van Gogh paintings. arXiv preprint; 2014. p. 1–14.
29. Hughes JM, Mao D, Rockmore DN, Wang Y, Wu Q. Empirical mode decomposition analysis for visual stylometry. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2012; 34(11):2147–57.
30. Wu T, Polatkan G, Steel D, Brown W, Daubechies I, Calderbank R. Painting analysis using wavelets and probabilistic topic models. 20th IEEE International Conference on Image Processing; 2013. p. 3264–8. https://doi.org/10.1109/ICIP.2013.6738672
31. Ramalingam V, Balasubramanian C, Palanivel S. Tumor diagnosis in MRI brain image using ACM segmentation and ANN-LM classification techniques. Indian Journal of Science and Technology. 2016; 9(1):1–12.