VIDEO INPAINTING DETECTION USING INCONSISTENCIES IN OPTICAL FLOW

Shobhita Saxena

Computer Science

Indraprastha Institute of Information Technology, Delhi (IIIT-D), India

A Thesis Report submitted in partial fulfilment for the degree of

MTech Computer Science

29 July 2015

1. Assistant Prof. A V Subramanyam (Thesis Adviser)

2. Assistant Prof. Sambuddho Chakravarty (Internal Examiner)

2. Associate Prof. Pradeep K. Atrey(External Examiner)

Day of the defense: 29 July 2015

Signature from Post-Graduate Committee (PGC) Chair:

Abstract

In recent years due to advancement in video and image editing tools it has become increasingly easy to modify the multimedia content. The doctored videos are very difficult to identify through visual examination as artifacts left behind by processing steps are subtle and cannot be easily captured visually. Therefore, the integrity of digital videos can no longer be taken for granted and these are not readily acceptable as a proof-of-evidence in court-of-law. Hence, identifying the authenticity of videos has become an important field of information security.

In this thesis work, we present a novel approach to detect and temporally localize video inpainting forgery based on optical flow consistency. The proposed algorithm comprises of two stages. In the first step, we detect if the given video is inpainted or authentic and in the second step we perform temporal localization. Towards this, we first compute the optical flow between frames. Further, we analyze the goodness of fit of chi-square values obtained from optical flow histograms using a Guassian mixture model. A threshold is then applied to classify between authentic and inpainted videos. In the next step, we extract Transition Probability Matrices (TPMs) by modelling the optical flow as first order Markov process. SVM based classification is then applied on the obtained TPM features to decide whether a block of non-overlapping frames is authentic or inpainted thus obtaining temporal localization.

In order to evaluate the robustness of the proposed algorithm, we perform the experiments against two popular and efficient inpainting techniques. We test our algorithm on public datasets like PETS and SULFA. The results show that the approach is effective against the inpainting techniques. In addition, it detects and localizes the inpainted frames in a video with high accuracy and low false positives.

This Thesis work is dedicated to my parents and sister for their endless love and support.

Acknowledgements

I offer my sincere gratitude to my supervisor, Assistant Prof. A V Subramanyam, who has supported me thoughout my thesis with his patience and knowledge. I attribute the level of my Masters degree to his encouragement and effort and without him this thesis, too, would not have been completed or written. One simply could not wish for a better and approachable supervisor.

Besides my advisor, I would like to thank my esteemed committee members Assistant Prof. Sambuddho Chakravarty and Associate Prof. Pradeep Atrey for agreeing to evaluate my thesis. Next, I would specially like to thank Hareesh Ravi and Ambuj Mehrish for their feedbacks , support and valuable time. I thank CERC at IIIT-Delhi for providing me an exposure to share my ideas with experts from different parts of the world.

Last but not the least, I would like to extend special thanks to my family for their patient love which enabled me to complete this work.

Declaration

This is to certify that the MTech Thesis Report titled **Video Inpainting detection using inconsistencies in optical flow** submitted by **Shobhita Saxena** for the partial fulfillment of the requirements for the degree of *MTech* in *Computer Science* is a record of the bonafide work carried out by her under my guidance and supervision at Indraprastha Institute of Information Technology, Delhi. This work has not been submitted anywhere else for the reward of any other degree.

Assistant Professor A V Subramanyam
Indraprastha Institute of Information Technology, New Delhi

Contents

List of Figures **ix**

List of Tables **x**

1 Introduction **1**

1.1 Motivation 3

1.2 Thesis Objective 3

1.3 Thesis Contribution 5

1.4 Thesis organization 5

2 Related Work **6**

2.1 Video forgery detection 6

2.2 Video Inpainting detection 8

3 Research Methodology **10**

3.1 Optical flow computation 10

3.2 Video Inpainting Detection 15

3.2.1 Histogram based thresholding using Peak Ratios 21

3.2.2 Goodness-of-Fit using Guassian Mixtures 22

3.3 Video Inpainting Localizaton 25

3.3.1 Markov Chain based feature extraction 25

3.3.2 SVM Classification 28

4 Experimental Analysis and Results **29**

4.1 Dataset 29

4.2 Experimental Setup 31

4.3 Experimental Results 32

4.4 Performance Evaluation 37
4.4.1 Video Inpainting Detection 37
4.4.2 Video Inpainting Localization 38
4.5 Comparison with previous approaches 39

5 Discussion,Limitations and Future Work **45**

References **47**

List of Figures

1.1 (a) Source video frames (b) Inpainted video frames 4

3.1 Algorithm Design : Inpainting Detection and Inpainting Localization . 11
3.2 Color scheme used to represent the orientation and magnitude of optical flows 14
3.3 (a) source frames of video (b) optical flow of source frames 16
3.4 (a) coventional TCP inpainted frames (b) optical flow of inpainted frames 17
3.5 (a) source frames of video (b) optical flow of source frames 18
3.6 (a) complex TCP inpainted frames (b) optical flow of inpainted frames . 19
3.7 optical flow computation steps 20
3.8 Chisq distance computation 21
3.9 Histogram Plots for complex TCP (a) video 1 (b)video 2 23
3.10 Histogram Plots for conventional TCP (a) video 1 (b)video 2 24
3.11 Block diagram of video authentication pipeline 27

4.1 RMS values classification for Conventional TCP inpainting 34
4.2 RMS values classification for Complex TCP inpainting 34
4.3 Conventional TCP results - Video 1 (a) Source Video (b) Conventional TCP inpainted video (c) Inpainting Localization 41
4.4 Conventional TCP results - Video 2 (a) Source Video (b) Conventional TCP inpainted video (c) Inpainting Localization 42
4.5 Complex TCP results - Video 1 (a) Source Video (b) Complex TCP inpainted video (c) Inpainting Localization 43
4.6 Complex TCP results - Video 2 (a) Source Video (b) Complex TCP inpainted video (c) Inpainting Localization 44

List of Tables

3.1 Parameters of optical flow method 14

4.1 Conventional Inpainting Dataset 30
4.2 Complex Inpainting Dataset 31
4.3 RMS values of Complex TCP Inpainting 32
4.4 RMS values of Conventional TCP Inpainting 33
4.5 Spatial TPM for complex TCP inpainting 35
4.6 Spatial TPM for conventional TCP inpainting 36
4.7 Classification Results for complex TCP 37
4.8 Classification Results for conventional TCP 37
4.9 Inpainting Detection Performance 38
4.10 Classification Results for complex TCP 38
4.11 Classification Results for conventional TCP 38
4.12 Inpainting Localization Performance 39

1

Introduction

Videos are an important medium of communication and expression in today's world. Videos are widely used in a number of areas such as surveillance, movie industry, journalism, medical imaging, home videos and so on. However, in recent times, due to advancement in network technologies, low cost multimedia devices, sophisticated image/video editing software and rapid growth of digital devices, it has become very easy to produce and modify digital videos with increasing sophistication. The digital content can now be easily manipulated, synthesized and tampered in numerous ways without leaving any visible clues. The integrity of digital videos can no longer be taken for granted. It has become difficult to differentiate in between a doctored and an original video. Therefore, there is an increasing dissatisfaction and mistrust about the authenticity of these videos and these can no longer be provided as a proof-of-evidence in court-of-law [1]. This has made video forensics which evaluates the authenticity of a video and verifies its trustworthiness to become an important and interesting field.

Video forensics can be classified into two categories:

a) Active - It primarily focuses on data hiding and hashing techniques like watermarking [2] and hashing [3]. Watermarking techniques consider multimedia data as a communication channel. The embedded watermark, maybe imperceptible too, contains either a specific producer ID or some content-related codes that are used for authentication [4].

In hashing, image identifiers or image hash functions are produced using suitable algorithm which is then used for image authentication. But these approaches require source camera or video at the time of generation which makes their usage practically

limited. Most of the current video capturing devices do not have a built-in watermarking and signature computation embedded module and it may also have an adverse effect on video quality, hence these may not be preferred.

b) Passive - In passive forensics, intrinsic characteristics of digital media or acquisition device are used to detect and localize tampering. A passive approach is better suited when there is no prior information of the source media and acquisition device [5]. Hence, it detects forgery without any prior information. In this thesis, we focus on passive forensics.

In this work, we study the problem of detection and temporal localization of tampered regions resulting from removal of objects from videos, and present an effective and robust approach based on inconsistencies in optical flow. Our approach performs on video sequences tampered by well known inpainting techniques which are variants of Temporal Copy Paste inpainting (TCP). TCP maintains temporal coherence in between successive frames to create a plausible tampered video. It takes temporal domain into account and replaces inpainted regions with similar areas from the closest frame [6] [7][8][9]. In this work, we have targeted two popular and efficient inpainting techniques proposed by Patwardhan et al [7] and Alasdair et al [8] which adopt TCP inpainting scheme. In [7] preprocessing is done by segmenting each frame into static background and moving foreground. Next, Optical flow is computed to create foreground and background mosaics. It then performs priority based motion inpainting and temporal copying. As this approach is able to handle only simple motion types and can not handle all situations like multiple moving objects and dynamic background motion, hence, we name it as Conventional TCP.

In [8] authors propose an automatic video inpainting algorithm which is able to deal with a variety of challenging situations which naturally arise in video inpainting, such as the correct reconstruction of dynamic textures, multiple objects motion, moving cameras and moving background. It works on high definition videos and requires no segmentation and manual input. As this algorithm can handle multiple dynamic and complex motions, we have named this inpainting type as Complex TCP inpainting. It should be noted that proposed inpainting techniques of Patwardhan et al and Alasdair et al inpainting have been referred to as Conventional and Complex TCP inpainting types respectively in the rest of the work.

1.1 Motivation

Video Inpainting is the process of removing an unwanted object from a video or filling up missing or damaged parts of a video sequence with visually plausible information [7]. It has attracted a great deal of attention in recent years because of its powerful ability to restore damaged videos and flexibility in editing videos. But this video restoration\completion technique can also be used as a major forgery tool to perform malicious changes in videos such as object-removal and photo-montages [10] as illustrated in Figure 1.1. Inpainting alters the multimedia content in a plausible way. This technique is more sophisticated and complex than other type of forgeries like copy and move [11] because the patch size may be as small as $\langle m * n \rangle$. In such a scenario, the existing copy-move algorithms may not be easily extended as the performance of algorithm depends on size of forged patches. Also, source of copied information may lie in different frames of video and can be non-continuous. This means that an object can be filled by a set of multiple small parts located at different places in different frames of the same video.Hence, inpainting forgery is more difficult to detect than other forgery types and therefore poses a challenging research problem. The existing techniques can detect some of the popular inpainting approaches [6] [9]. These are novel inpainting methods to fill in missing background and moving foreground of real-time videos captured by moving or still camera while still maintaining spatial and temporal coherance simultaneously. However, these detection techniques do not perform well on other state-of-the-art inpainting techniques [8] which rely on the optimisation of a global, patch-based function. This in turn has necessitated the need for research in the field of video inpainting detection and localization.

1.2 Thesis Objective

This work targets two popularly used inpainting techniques for forgery detection. Thesis Objectives are :

1. Given an input video determine if its an authentic or an inpainted video.
2. For an inpainted classified video temporally localize the inpainted regions.

FRAME 15 FRAME 16 FRAME 17

FRAME 18 FRAME 19 FRAME 20

(a)

FRAME 15 FRAME 16 FRAME 17

FRAME 18 FRAME 19 FRAME 20

(b)

Figure 1.1: (a) Source video frames (b) Inpainted video frames

1.3 Thesis Contribution

This work proposes a novel approach to perform video inpainting detection and temporal localization of inpainted regions. It performs well in detection of popular and effective state-of-the-art-inpainting techniques. Our approach is able to detect those inpainting types where other detection algorithms fail to perform.

1.4 Thesis organization

Chapter 2 discusses about the Related Work and gives a literature review

Chapter 3 proposes the algorithm to detect inpainting in videos and also temporally localizing inpainted frames in the video.

Chapter 4 gives experimental details of the performed experiments and covers analysis and results of the proposed algorithm

Chapter 5 has discussions, limitations and future work

2

Related Work

In this chapter we discuss previous works on video forgery detection and video inpainting detection. Here, the first section focuses on video forgery detection and second section describes video inpainting detection.

2.1 Video forgery detection

There are basically two techniques for detecting tampering in videos : active and passive. Active approaches use digital signatures and watermarking techniques to maintain integrity and authenticate video data [12] [13] [14] [15]. Cox et al [12] in one of their first algorithms used global DCT to embed a robust watermark in the perceptually significant portion of the Human Visual System (HVS). A new watermarking system based on the principles of informed coding and informed embedding was also proposed by them. Martino et al [14]proposed a fragile watermarking scheme where original image was divided into smaller images called blocks and were then compressed with fuzzy transform to images of size 2*2. Chen et al [15] detected image forgery using image watermarking and alpha mattes where image watermarking based on the discrete wavelet transform , discrete cosine transform and singular value decomposition is used . But these active approaches suffer drawback as their application is greatly limited because of requirement of inbuilt watermark modules. Therefore, passive approaches which detect forgery in video data without relying on the previously embedded information are widely used and extensively studied[16] [17] [18].These passive approaches can be classified into four categories :

1. Pixel based : In pixel level tampering, pixel level anomalies caused by the tampering are examined. Cloning (copy-move), splicing and resampling are the common pixel level operations found. In [19] [20] authors propose an algorithm based on correlation between frames arising from duplicate frames insertion. Wang and Farid[19] used the high correlation in between original and forged regions to detect copy and paste forgery. They have described two techniques for detecting common tampering in video. The first technique detected entire frame duplication and the second detected if only a portion of one or more frames was duplicated. Lin et al [20] proposed a coarse-to-fine-grained approach which used the difference in between color histograms of adjacent frames to identify candidate clips from temporal domain. Block based correlation algorithm is then applied in the spatial domain to measure the similarities between the query clip and candidate clip, in order to detect the duplicated clips. Authors in [21] consider motion direction based on optical flow consistency for frame deletion detection and employ a dual-window scheme.

2. Format based : This approach exploits the properties of video compression, such as periodic properties and blocking artifacts in MPEG1 and MPEG2 videos. Wang and Farid [19] used spatial and temporal artifacts of double MPEG compression. It works on static and temporal artifacts which get introduced in the video sequence when it is subjected to double MPEG compression. Luo et al [22] [23] used the temporal patterns of blocking artifacts to determine if an MPEG video has suffered frame addition or removal before recompression.They showed that MPEG compression introduces different block artifacts into various kinds of frames. Thus, when some frames are removed from an MPEG video file and the file is recompressed, the block artifacts introduced by the previous compression remain and affect the average of block artifact strength of the recompressed one which leads to providing an evidence of tampering. In [24] authors have proposed a technique to detect forgery in MPEG videos by analzying the frame's compression noise characteristics which is extracted from spatial domain by using a Huber Markov Random Field Model.

3. Camera based : These approaches analyze the specific sensor artifacts caused by components in the imaging pipeline such as sensor noise and interlaced scanning.

In [19] Wang and Farid examined the consistency of de-interlacing parameters which are used to convert an interlaced video into a non-interlaced video. These interlaced videos contain half of vertical resolution of the original videos, therefore, de-interlacing procedure exploited interpolation, insertion and duplication of frames to produce full resolution videos. Mondaini et al.[25] detected video forgery with sensor pattern noise. Johnson et al [26] exploited cameras optical system by considering lateral chromatic aberrations which gets disturbed when image is forged or tampered. A block that deviated significantly from the global estimate was suspected to be tampered. In [1] Kobayeshi et al used photon shot noise to detect forged regions. Their work exploited the inconsistency of photon shot noise caused by different video cameras.

4. Geometry based : These approaches inspect the geometric properties of objects and their positions relative to the camera . In [5] a geometric technique to detect physically impalusible object trajectories in a video sequence has been proposed.

2.2 Video Inpainting detection

Though several approaches investigating forgeries such as copy-move [11] and splicing detection [27] are popular, very few works have been proposed in the area of video inpainting detection. This section describes previous work on video inpainting detection. Hsu et al[28] proposed an approach to locate forged regions in an inpainted video using correlation of noise residue. Their method had the assumption that tampering shall change correlation of noise residue and make it different from that of the non-tampered parts. They have modeled the distribution of correlation of temporal noise residue in a forged video as a Gaussian mixture model (GMM). A Bayesian classifier is used to find the optimal threshold value based on the estimated parameters. Two video inpainting schemes [7] [29]have been used in it.

Zhang et al [30] proposed detecting video inpainting forgery based on ghost shadow artifact. This artifact gets introduced when moving objects get removed by video inpainting. Each frames is segmented into static background and moving foreground by block matching. They build a foreground mosaic and also

compute track of moving foreground. If the foreground mosaic and track are consistent they consider input video as authentic or else it is classified as forged. Authors in [10] propose a blind detection method based on zero-connectivity feature and fuzzy membership function to detect video inpainting type forgery. Lin et al [20] propose a passive approach for detection and localization of region-level forgery. They investigate two inpainting methods for object removal, Temporal Copy Paste [7] and Texture based synthesis [29].

3

Research Methodology

Aim of the proposed algorithm is to detect video inpainting forgery and temporally localize inpainted regions in a video. The block diagram of proposed approach is given in figure 3.1.

The proposed algorithm performs video inpainting detection based on optical flow inconsistencies and takes image motion into consideration. It has two steps: video inpainting detection and temporal localization in inpainted video. These procedures are explained below.

3.1 Optical flow computation

This step involves computing optical flow $\mathbf{O_f}$ in between video frames. Let $\mathbf{O_f}$ be the optical flow between two frames, that characterizes the motion of every pixel in one frame to its corresponding location in the next frame. It performs motion estimation in between video frames and gives the distribution of the apparent velocities of objects in an image [31]. It is best applied to video frames as these are sequence of time-ordered images which allow the estimation of projected two-dimensional image motion as either instantaneous image velocities or discrete image displacements [32]. Thus, optical flow conveys significant information about 3-D environment and holds direct relevance to numerous problems in dynamic scene analysis such as moving object detection, motion based segmentation [33][34], qualitative kinematic labeling of moving objects in a scene, recovery

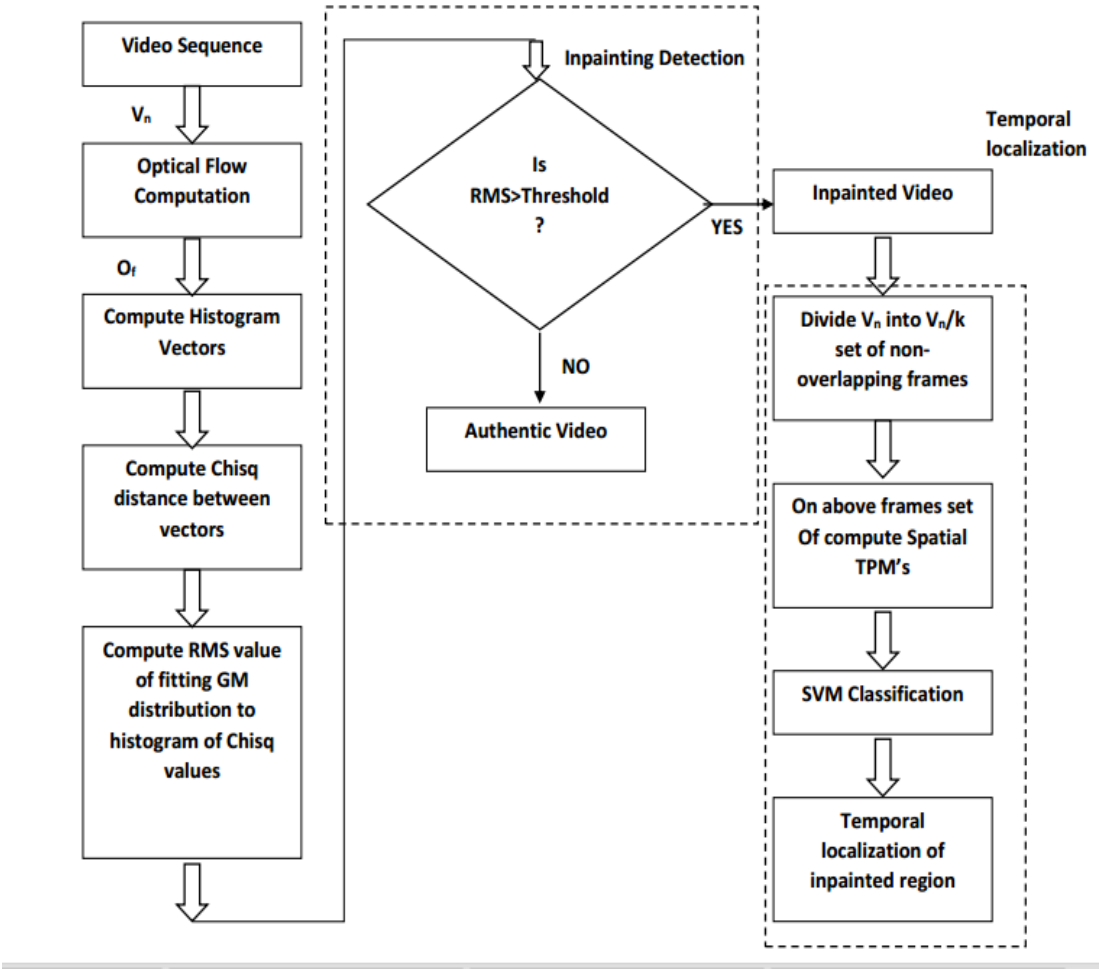


Figure 3.1: Algorithm Design : Inpainting Detection and Inpainting Localization

of 3-D motion or structure [33] [34] and image coding [35]. The Horn-Schunck method [36] of estimating optical flow is a global method which introduces a global constraint of smoothness. In it, the flow is formulated as a global energy function which is sought to be minimized. This function is given for two-dimensional image streams as:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2(\|\nabla u\|^2 + \|\nabla v\|^2)\right] \mathrm{d}x\mathrm{d}y \tag{3.1}$$

where, I_x, I_y and I_t are the derivatives of the image intensity values along the x, y and time dimensions respectively. The parameter α is a regularization constant. Larger values of α lead to a smoother flow . This function can be minimized by solving the associated multi-dimensional Euler-Lagrange equations. These are

$$\frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0 \tag{3.2}$$

$$\frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0 \tag{3.3}$$

where L is the integrand of the energy expression, giving

$$I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u = 0 \tag{3.4}$$

$$I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v = 0 \tag{3.5}$$

where subscripts again denote partial differentiation and $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ denotes the Laplace operator.

In our algorithm, we apply Brox et al [37] implementation of optical flow in the form of energy based formulation. Given a sequence of images, $\mathbf{I} : \Omega \subset \mathbf{R}^3 \rightarrow \mathbf{R}$ of gray level values in space and time, $\mathbf{x} = (x, y, t)^{\mathbf{T}} \varepsilon \Omega$, the optical flow is defined as a dense mapping, $\mathbf{w} = (u(\mathbf{x}), v(\mathbf{x}), 1)^{\mathbf{T}}$, between the pixels of every two consecutive images. The scalar fields $u(\mathbf{x})$ and $v(\mathbf{x})$ are the x and y displacements in the 3D volume, respectively. The gradient of the optical flow is defined as:

$$\Delta = \{(u_x, u_y)^{\mathbf{T}}, (u_x, u_y, u_t)^{\mathbf{T}}\} \tag{3.6}$$

It employs the following energy model :

$$E(\mathbf{w}) = \int_{\Omega} \psi((\mathbf{I}(\mathbf{x}+\mathbf{w}) - \mathbf{I}(\mathbf{x})^2))dx + \gamma \int_{\Omega} +\psi((|\Delta\mathbf{I}(\mathbf{x}+\mathbf{w}) - \Delta\mathbf{I}(\mathbf{x}) \,|^2)dx + \alpha \int_{\Omega} \psi(|\, \Delta u \,|^2 + |\, \Delta v \,|^2)dx \tag{3.7}$$

with $\psi(s^2) = \sqrt{(}s^2+\epsilon^{\,2})$ and $\epsilon = 0.001$ a small constant. In [37] authors apply three assumptions of brightness constancy, gradient constancy and discontinuity preservation. This algorithm takes a set of grey level images as input data and computes optical flow between a pair of given images. The smoothness term α ensures that the optical flow computation is extended to textureless regions by hole filling from the neighbourhood, gradient term γ ensures that difference in image gradients is also minimised along with the intensity difference. Gradient term is useful when two images are captured under slightly varying lighting conditions. This approach is separated into two modules : one procedure that computes the optical flows in each scale, and the main algorithm that is in charge of handling the pyramidal structure. To compute optical flow, image pyramid is constructed to perform motion estimation. Optical flow computed at coarser levels are used to initialize the finer level flows. The parameter *numlevels* determines the number of levels in the image pyramid. Its computation is such that optical flow for first and last columns and rows, and the four corners of the images are computed separately. The *inneriterations* and *outeriterations* control the number of iterations to perform till threshold value ϵ is reached and η is downsampling parameter. Please refer to [37] for more information on these aspects. The parameters defined in [37] are listed in table 3.1. The color scheme used to represent the magnitude and direction of optical flows has been shown in Figure 3.2 .

Table 3.1: Parameters of optical flow method

Parameter	Explanation
α	Regularization parameter. It determines the smoothness of the output. The bigger this parameter is, the smoother the solutions obtained
γ	Gradient constancy parameter
	Number of scales in the pyramidal structure
η	Downsampling factor. It is used to downscale the original images in order to create the pyramidal structure. Its value must be in the interval (0, 1).
ξ	Stopping criterion threshold. It is the threshold used to stop the iterations of convergence
inneriterations	number of inner iterations
outeriterations	number of outer iterations

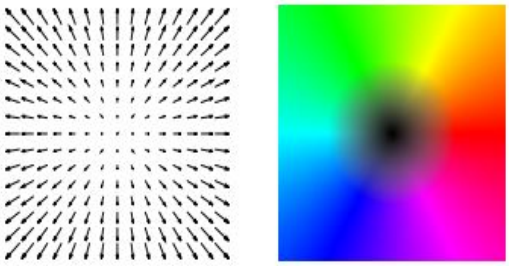


Figure 3.2: Color scheme used to represent the orientation and magnitude of optical flows

The results of optical flow computation on our test video sequence are shown in Figures 3.3 to 3.6. Optical flow is computed in between a particular frame and its next 1 to m successive frames, thus producing m optical flows for that specific frame. In Figure 3.3, source video frames of conventional TCP inpainting type

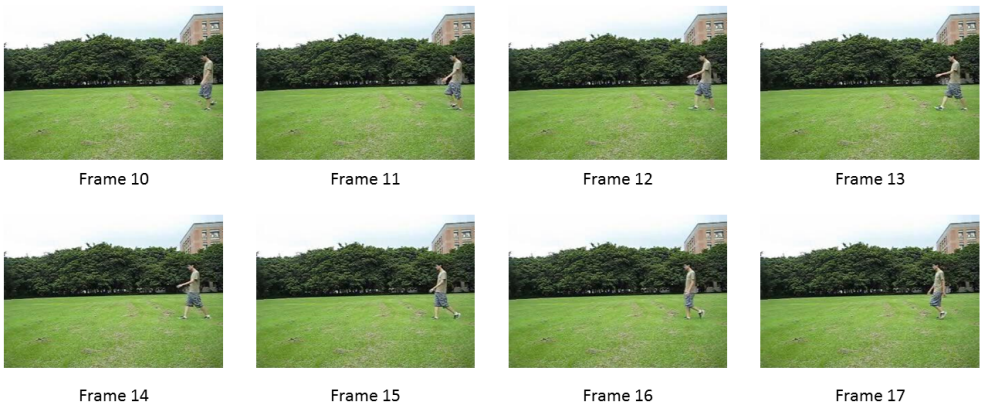
and their corresponding optical flow computations are shown. Figure 3.4 shows inpainted frames of conventional TCP and their optical flow results. Similarly Figure 3.5 and Figure 3.6 show results for complex TCP inpainting technique. The difference in optical flow results can be clearly observed in case of source frames and inpainted frames for both inpainting techniques. This occurrence of inconsistency in optical flow has been studied and analyzed by our approach for video inpainting forgery detection and localization.

!

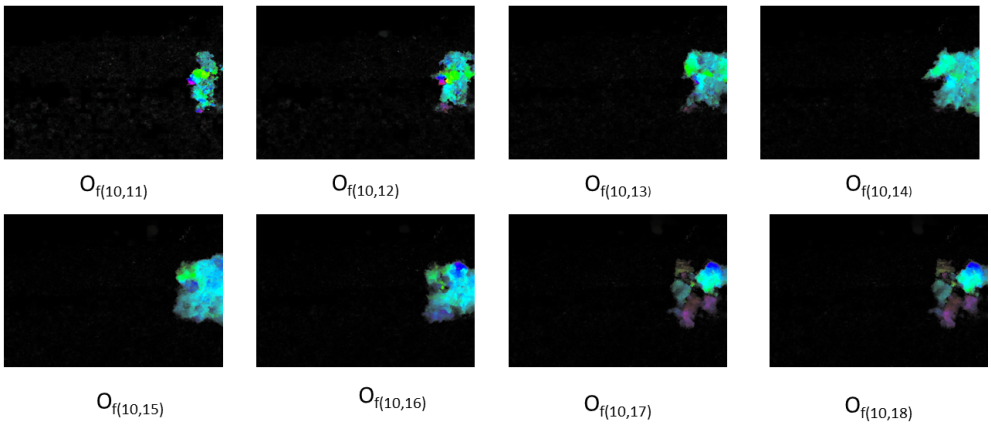
We perform optical flow computation for each test video. The objective is to produce optical flow matrices in between frames, which can be used in subsequent steps to detect inpainting forgery and temporally localize inpainted regions. The optical flow methods try to calculate the motion between two image frames which are taken at times t and $t + \Delta t$ at every voxel position. Let $\mathbf{V}$ be the video under consideration consisting of $\mathbf{N}$ frames as $\mathbf{v_1}$, $\mathbf{v_2}$, $\mathbf{v_3}$...$\mathbf{v_N}$. In our approach, optical flow $\mathbf{O_f(n, n+1)}$ between two frames characterizes the motion of a pixel $[x, y]$ in frame $\mathbf{v_n}$ to frame $\mathbf{v_{n+1}}$ by a motion vector. Similarly optical flow matrices between frame $\mathbf{v_n}$ and $\mathbf{v_{n+2}}$, $\mathbf{v_{n+3}}$ .. $\mathbf{v_{n+m}}$ are obtained to get m optical flows for any frame $\mathbf{v_n}$. In a given input video of $\mathbf{N}$ frames, we apply [37] to first $\mathbf{N - m}$ frames for optical flow computation. For each frame, $\mathbf{m}$ optical flow matrices are computed as $\mathbf{O_{f(n,n+1)}}$, $\mathbf{O_{f(n,n+2)}}$, $\mathbf{O_{f(n,n+3)}}$...$\mathbf{O_{f(n,m+m)}}$ resulting into generation of $\mathbf{(N - m) * m}$ optical flow matrices for each video. These matrices are then used in subsequent steps for detecting and localizing video inpainting forgery. The procedure for computing optical flow between frames of the video is illustrated in Figure 3.7.

3.2 Video Inpainting Detection

For video inpainting detection, thresholding based classification on two different features has been performed. We incorporate histogram peaks based and Guassian Model distribution based thresholding. For each optical flow matrix computed in between a pair of frames, we compute histogram vector to determine



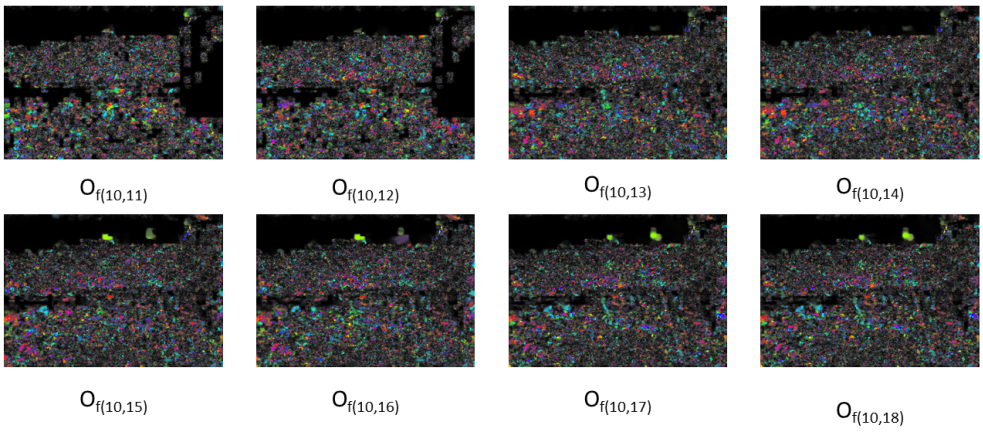
(a)



(b)

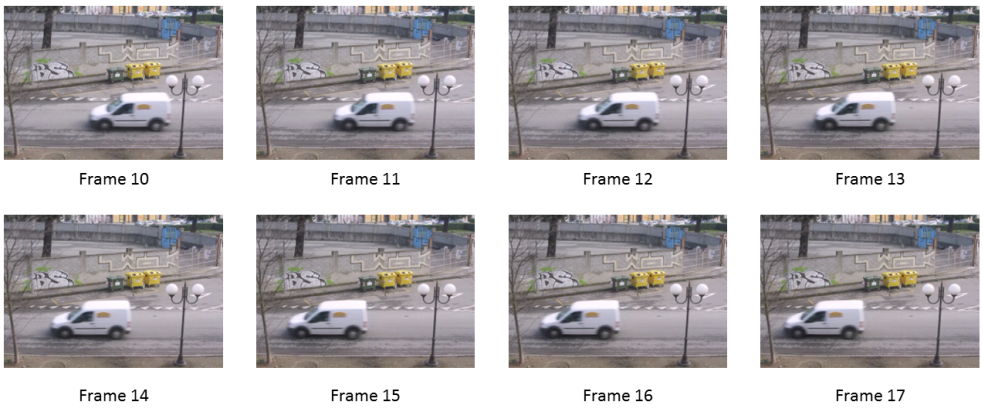
Figure 3.3: (a) source frames of video (b) optical flow of source frames

(a)

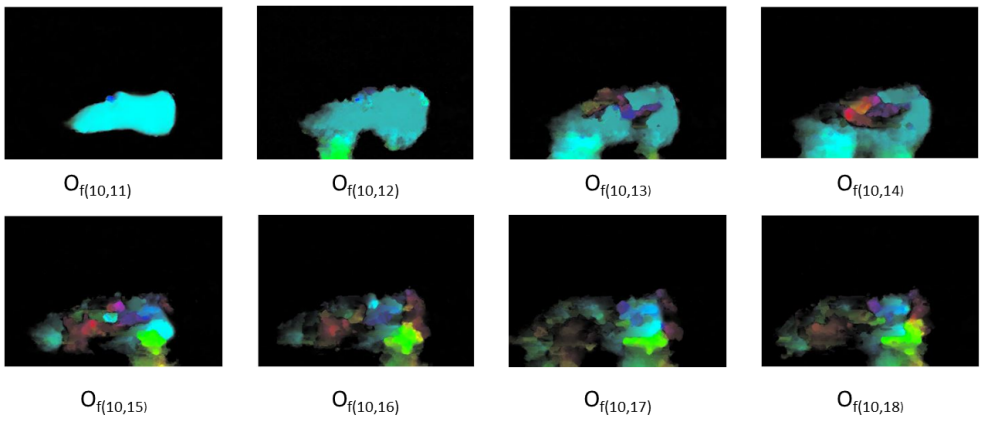


(b)

Figure 3.4: (a) coventional TCP inpainted frames (b) optical flow of inpainted frames



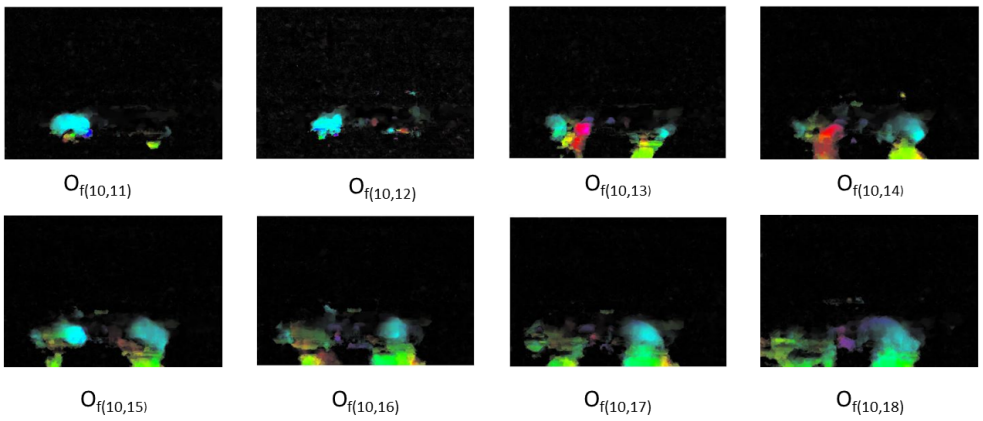
(a)



(b)

Figure 3.5: (a) source frames of video (b) optical flow of source frames

(a)



(b)

Figure 3.6: (a) complex TCP inpainted frames (b) optical flow of inpainted frames

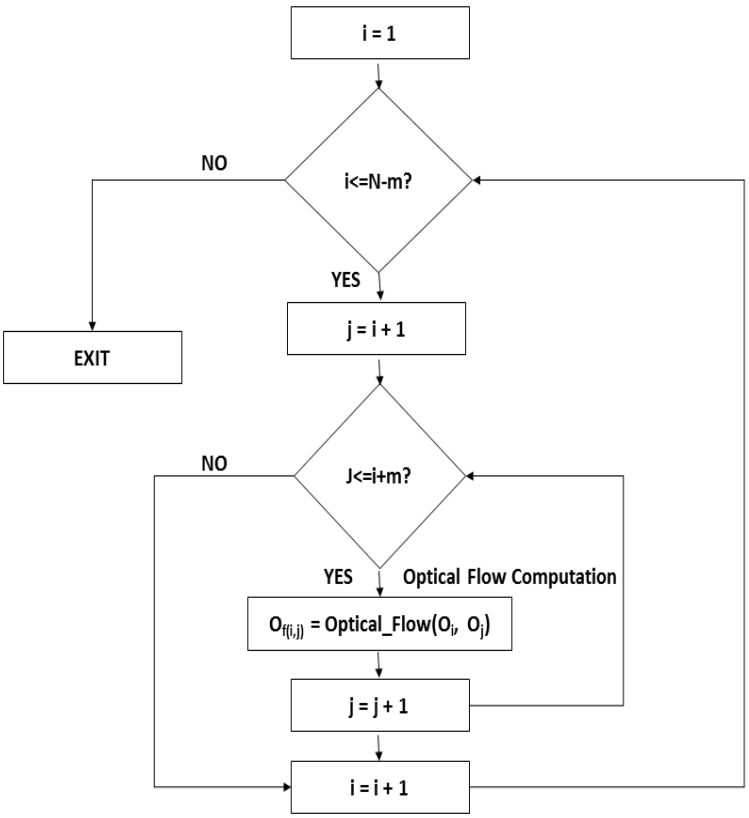


Figure 3.7: optical flow computation steps

probability distribution of data. To compare these computed histograms, we calculate chi-square distance (**chisq**) in between these histogram vectors. **chisq** between two histogram vectors each having s bins is defined as [38]:

$$d(x, y) = sum((x_i - y_i)^2/(x_i + y_i))/2; \tag{3.8}$$

We compute histogram vector for each optical flow matrix $\mathbf{O_f}$ generated for each frame of a video sequence. Hence, in total, $(\mathbf{N} - \mathbf{m}) * \mathbf{m}$ histogram vectors are produced. **chisq** is then computed in between histogram vector pairs as illustrated in figure 3.8. **chisq** matrix of $(\mathbf{N} - \mathbf{m}) * (\mathbf{m} - \mathbf{1})$ values is obtained and normalized for each video sequence.

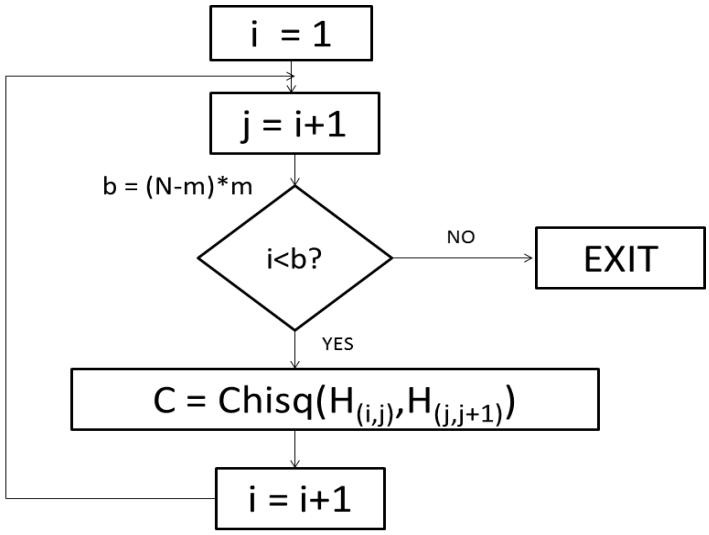


Figure 3.8: Chisq distance computation

After computing normalized chisq values we perform feature extraction explained in section below.

3.2.1 Histogram based thresholding using Peak Ratios

Once **chisq** values are obtained, a histogram vector is computed and histogram peaks are calculated. For each chisq distance based histogram vector, two his-

togram peaks have been obtained : Ratio between highest peak to second highest peak , Ratio between highest to third highest peak and so on.

$Ratio1 = P1/P2$

$Ratio2 = P1/P3$

After observing ratio values of a number of source and inpainted videos a global threshold value θ is set to differentiate in between authentic and inpainted video sequence. Source and inpainted videos histogram plots for Conventional and Complex TCP are given in Figure 3.9 and 3.10. As observed from the below plots, the Peak Ratios for inpainted videos are much greater than Peak Ratios of source video. This nature of source and inpainted videos is used to set value of θ for classification.

3.2.2 Goodness-of-Fit using Guassian Mixtures

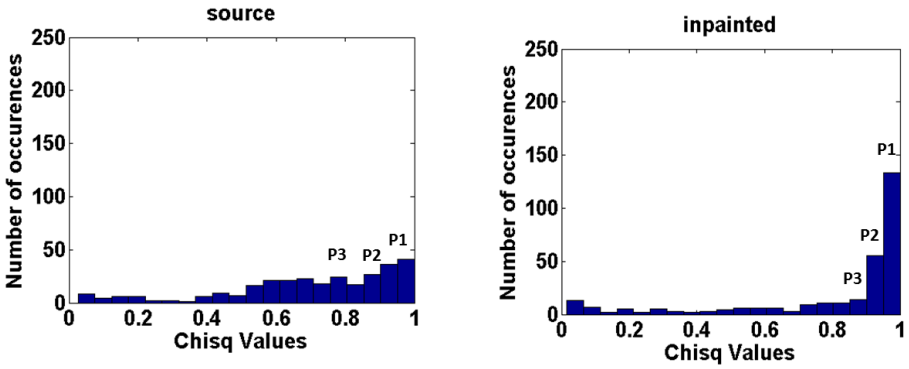
A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities [39]. It is represented by the equation

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i g(x|\mu_i, \Sigma_i) \tag{3.9}$$

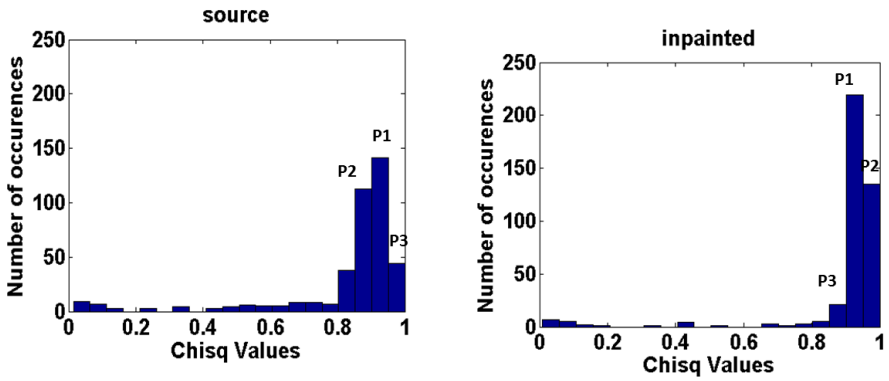
where x is the random variable modeled as GMM, where w_i, i = 1,2...M, are the mixture weights and $g(x|\mu_i, \Sigma_i)$, i = 1,2...M are the component Guassian densities. Each component density is a Guassian function of the form,

$$g(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\{-\frac{1}{2}(x-\mu_i)^{'}\Sigma_i(x-\mu_i)\} \tag{3.10}$$

We have considered M=2 for Guassian distribution. This parametric model has been applied on above obtained normalized chisq values to estimate Guassian parameters. Parametric Probability Guassian Mixture Model(set M=2) is fitted over histogram of normalized chisq values obtained above. RMSE (Root Mean Square Error) is a frequently used measure of difference between values predicted by a model and the values actually observed from the environment that is being modelled. These individual differences are also called residuals, and the RMSE

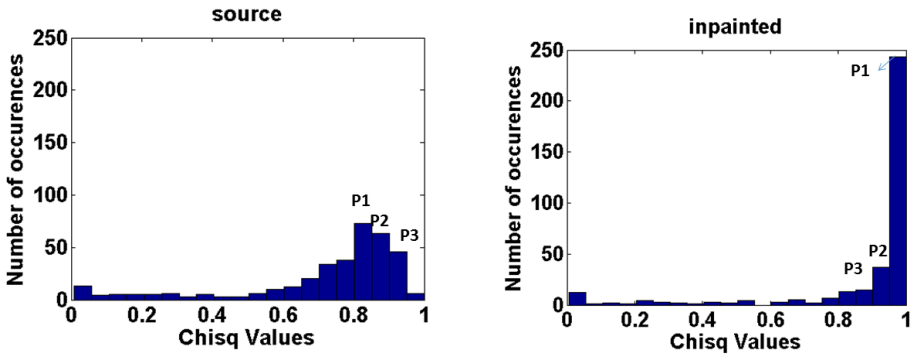


(a)

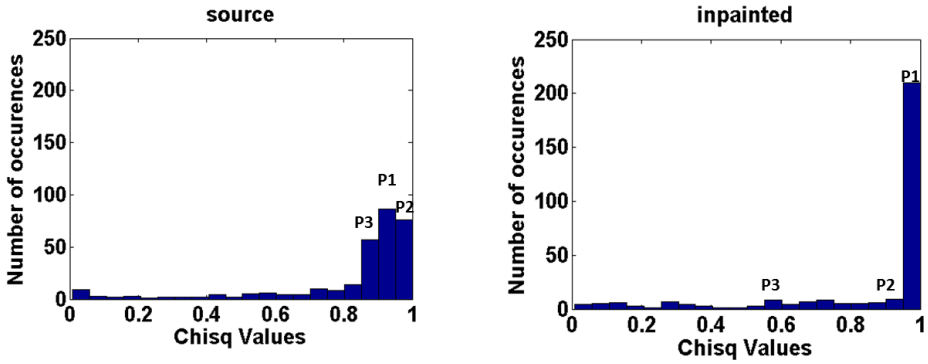


(b)

Figure 3.9: Histogram Plots for complex TCP (a) video 1 (b)video 2



(a)



(b)

Figure 3.10: Histogram Plots for conventional TCP (a) video 1 (b)video 2

serves to aggregate them into a single measure of predictive power [38]. The RMSE of an observed variable with respect to estimated variable model is defined as square root of the mean squared error:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (X_{obs,i} - X_{model,i})^2}{n}} \tag{3.11}$$

where X_{obs} are observed values and X_{model} is estimated value at time i. It is calculated as a measure of goodness of fit for authentic and inpainted videos. On the basis of goodness-of-fit of inpainted and authentic videos in Guassian Mixture distribution, RMS errors are calculated. It was observed that RMSE of inpainted videos was higher than RMSE of authentic videos and this nature of authentic and inpainted videos was used in our approach for classification.

After obtaining goodness of fit, RMS values of source and inpainted videos are analyzed and a global threshold value τ is set for classifying a video as inpainted or authentic.

Both of the above methods for feature extraction show similar results in video classification on the given dataset. Hence, we have chosen only one of the above methods in our final implemented algorithm. GMM based parameter estimation has been adopted in our proposed approach for video inpainting detection. Next section explains video inpainting localization.

3.3 Video Inpainting Localizaton

In inpainted region localization, we have modeled optical flow matrix as spatial Markov chain. The procedure is explained in the section below.

3.3.1 Markov Chain based feature extraction

Markov statistics have proven to be a distinguishable and effective feature in the field of video forensics. In [40] these features have been used to detect whether a given video is authentic or inpainted. First we have employed spatial markov chain model of [40] as follows:

The optical flow matrix $\mathbf{O_f}$ computed for video frames can be modeled as a first order ergodic spatial Markov chain such that,

$p(X_{t+1} = x|X_1 = x_1, X_2 = x_2, ..., X_t = x_t) = p(X_{t+1} = x|X_t = x_t)$,

where X_{t+1} is the present state and $(X_1, X_2, ..., X_t)$ are the previous states. The features that we use to characterize this optical flow matrix is Transition Probability Matrix (TPM).

The states of the chain are obtained from the absolute values of optical flow matrix . Representation along the right direction is shown hereafter and those along other directions can be obtained in a similar way. Values in each optical flow matrix $\mathbf{O_f^{\rightarrow}}$ are rounded off to the nearest integer to get integer value states and then truncated between $-T_r$ to T_r before extracting the transition probabilities,

$$\tilde{\mathbf{D}}_{\mathbf{c}}^{\rightarrow}(i,j) = \begin{cases} -T_r, \mathbf{O_f^{\rightarrow}}(i,j) < -T_r \\ +T_r, \mathbf{O_f^{\rightarrow}}(i,j) > +T_r \\ \mathbf{O_f^{\rightarrow}}(i,j), otherwise \end{cases} \tag{3.12}$$

This provides us with $(2T_r + 1)$ different states to model the Markov chain. In our model T_r is set to 5 as 95% of the values in the optical flow matrix are distributed within [-5, 5]. Also, experiments were performed with values between 3 to 30 for T_r and it was found that saturation in performance reached after 5. Moreover, for values lower than 5, percentage of values considered as is, dropped significantly, and so did the accuracy. Now, TPM is constructed as,

$$P_{u,v}^{\rightarrow} = Pr(\tilde{\mathbf{O}}_{\mathbf{f}}^{\rightarrow}(i, j+1) = u|\tilde{\mathbf{O}}_{\mathbf{f}}^{\rightarrow}(i,j) = v) \tag{3.13}$$

where, $u, v \in [-T_r, T_r]$, and $u, v \in \mathbb{Z}$. Similarly, the probabilities can be obtained for other directions. The size of each TPM will be 1×847. This gives us $(2T_r + 1) \times (2T_r + 1) = 121$ transition probabilities for each matrix. The TPMs along the eight directions are averaged to get 121-D feature vector. Since , as described in above section there will be k optical flow matrixes for each frame. The total size of the feature vector will be $121 \times k$ obtained by concatenating the k TPM's to get the final feature which is N-D, and this remains the same irrespective of the size of the frame.

Further, to take motion dependency of frames into consideration, we have modified the above applied approach of spatial Markov chain Model [40] to a temporal domain. In it, TPM is constructed in between successive frames , as follows:

$$P^{\rightarrow}_{u,v} = Pr(\tilde{\mathbf{O}}^{\rightarrow}_{\mathbf{f}}(i,j) = u | \tilde{\mathbf{O}}^{\rightarrow}_{\mathbf{f}}(i,j) = v) \tag{3.14}$$

Temporal TPM's do not show much accurate results because of less number of frames present in the video resulting in producing less samples for Feature extraction. Hence, in our final algorithm only spatial TPM's have been employed for feature extraction.

The Research Method pipeline for video authentication enlisting all the above applied methods has been shown in figure 3.11.

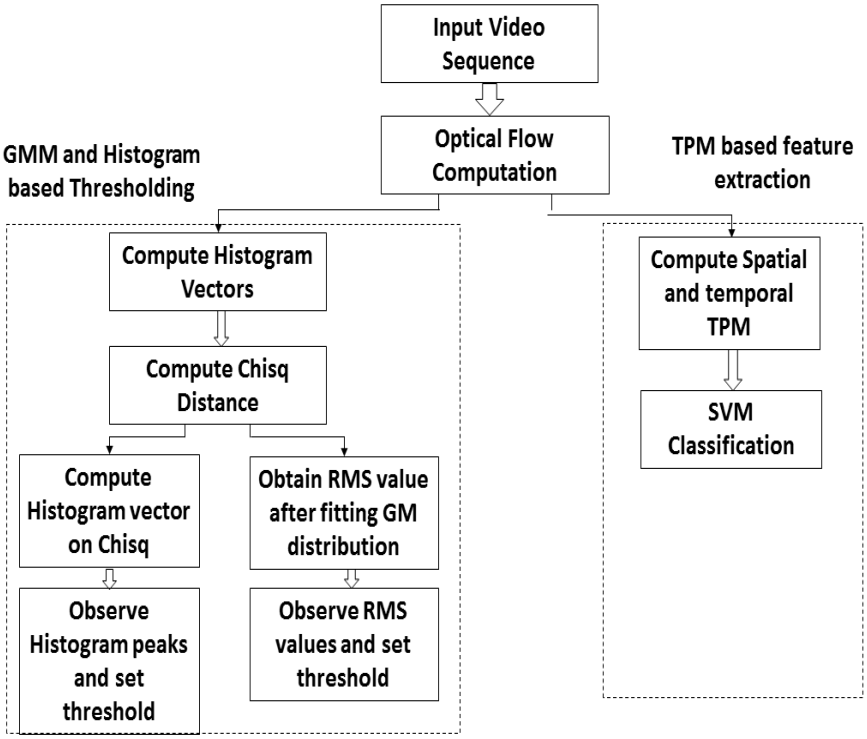


Figure 3.11: Block diagram of video authentication pipeline

These features are then used to train an SVM model and classify a test video as original or inpainted using the trained model.

3.3.2 SVM Classification

Video frames are divided into ($\mathbf{N}/k$) sets of non-overlapping frames. The optical flow matrices of these sets are then modeled as spatial markov chain , such that, Transition Probability Matrix features can be obtained as described above. We use RBF kernel binary classification SVM from svm libary [41]. We have employed Leave-one-out cross validation approach [42]. The dataset is divided into p subsets, where p is equal to number of videos and a holdout method is repeated p times. Each time, one of the p subsets is used as the test set and the other $p-1$ subsets are put together to form a training set. Then the average error across all p trials is computed. The SVM model after getting trained by these features is then employed to classify a set of frames as inpainted or authentic based on the voting mechanism.

4

Experimental Analysis and Results

In this section, we describe experiments performed on 24 test videos. We measure the performance of our approach on the given videos and evaluate our algorithm's advantages over other previously proposed inpainting detection approaches.

4.1 Dataset

Experiments have been conducted on test videos of two inpainting techniques, complex TCP inpainting [8] and conventional TCP inpainting [7] which are variants of TCP inpainting. We collect two datasets each of 12 videos for both complex and conventional inpainting techniques. Database contains all static camera videos having moving foreground and stationary background. Inpainted test videos have got varied kinds of objects removed. Details of test videos for both the inpainting types have been summarized in table 4.1 and 4.2 . Video sequences in complex TCP inpainting dataset have been obtained from SULFA 2012 [43] database and wisdom weizmann videos dataset [44]. Further details of these videos can be found on SULFA website [43]. Videos in conventional inpainting dataset have been taken from PETS database [45] and Canon IXUS 750 [46]. These videos have got different resolutions and consist of sequence of JPEG images. We extract consecutive frames from each video sequence, then alter the

content in each frame to form a forged video sequence. We run all experiments on a PC with 2.2 GHz with Intel(R)Xeon(R) CPU and 128 GB RAM, using the MATLAB 64 bit software development tool. Figure 4.1 consists of snapshots of sampled frames from 4 selected test videos, original and inpainted frames.

Table 4.1: Conventional Inpainting Dataset

Video ID	Size	Video Description	ObjectsRemoved	Source
1	320 x 240	Man walking on grass	1	CANON IXUS 750
2	320 x 240	2 cyclists on road	2	CANON IXUS 750
3	320 x 240	3 Men walking in park	3	CANON IXUS 750
4	320 x 240	2 Men crossing	2	CANON IXUS 750
5	320 x 240	2 cyclists on road	1	CANON IXUS 750
6	320 x 240	2 Men crossing	1	CANON IXUS 750
7	768 x 576	6 men walking in campus	6	PETS
8	768 x 576	6 men walking in campus	3	PETS
9	768 x 576	10 people crossing	10	PETS
10	768 x 576	8 people walking on road	8	PETS
11	768 x 576	8 people walking on road	5	PETS
12	768 x 576	6 people walking	6	PETS

Table 4.2: Complex Inpainting Dataset

Video ID	Size	Video Description	ObjectsRemoved	Source
1	320 x 156	Girl walking	1	SULFA
2	320 x 240	Van moving on road	1	SULFA
3	320 x 240	Man coming out of lift	1	SULFA
4	384 x 192	Man waving his hand	1	SULFA
5	320 x 240	Car moving on road	1	SULFA
6	320 x 240	Man bending down	1	SULFA
7	320 x 154	Cat walking on floor	1	SULFA
8	320 x 189	Wool rolling on floor	1	SULFA
9	280 x 184	Man sitting on bench	1	SULFA
10	180 x 144	Girl walking on road	1	Wisdom Weizman
11	320 x 240	Man walking on grass	1	SULFA
12	180 x 144	Man walking on road	1	Wisdom Weizman

4.2 Experimental Setup

In our experiments, for optical flow computation, the value of α is set to 15 and γ is set to 30 in eq(3.7). The *inneriterations*, *outeriterations*, Downsampling factor η and pyramid levels are set to 500, 3, 0.95 and 40 respectively. Value of k for determining the number of sets of overlapping frames in a video is set to 10. Here number of frames n in each video is 47. The number of components for Guassian Mixture is chosen to be 2. The RMS threshold value τ is emperically set to 3.5. We perform grid search on svm classifier with rbf kernel which gives values of parameters *cc* and *gg* as 5.1 and 0.32 respectively. The value of m is chosen to be 7.

4.3 Experimental Results

The RMS values of source and inpainted videos of conventional TCP inpainting and complex TCP inpainting are summarized in table 4.3 and 4.4 respectively. According to the set threshold value of $\tau = 3.5$, in conventional inpainting method, 2 source videos and 3 inpainted videos are wrongly classified out of total 24 source and inpainted videos. In complex inpainting method, 1 source video and 2 inpainted videos are misclassified out of 24 videos of source and inpainted.

Table 4.3: RMS values of Complex TCP Inpainting

Video ID	Source RMS value	Inpainted RMS value
1	3.01	3.55
2	2.82	3.57
3	1.87	2.28
4	2.84	5.18
5	3.32	3.72
6	3.97	2.96
7	3.43	3.74
8	3.41	4.18
9	3.14	3.82
10	2.17	4.4
11	2.9	3.99
12	3.34	4.95

Table 4.4: RMS values of Conventional TCP Inpainting

Video ID	Source RMS value	Inpainted RMS value
1	2.97	5.02
2	1.91	5.18
3	3.85	7.43
4	3.37	9.53
5	1.91	2.62
6	3.37	4.34
7	2.97	12.2
8	2.97	2.81
9	2.56	5.31
10	0.6948	2.32
11	0.6948	11.8
12	3.88	18.98

The thresholding results of Complex TCP and Conventional TCP based on RMS values are shown in Figure 4.1 and 4.2 respectively.

For video inpainting localization, spatial TPM's have been calculated. For each set of 10 frames in a video, we classify it as inpainted or authentic by voting mechanism. If number of frames correctly classified are ≥ 5, we say that the set is correctly classified. The results for complex TCP inpainting and conventional TCP inpainting have been summarized in table 4.5 and 4.6 respectively.

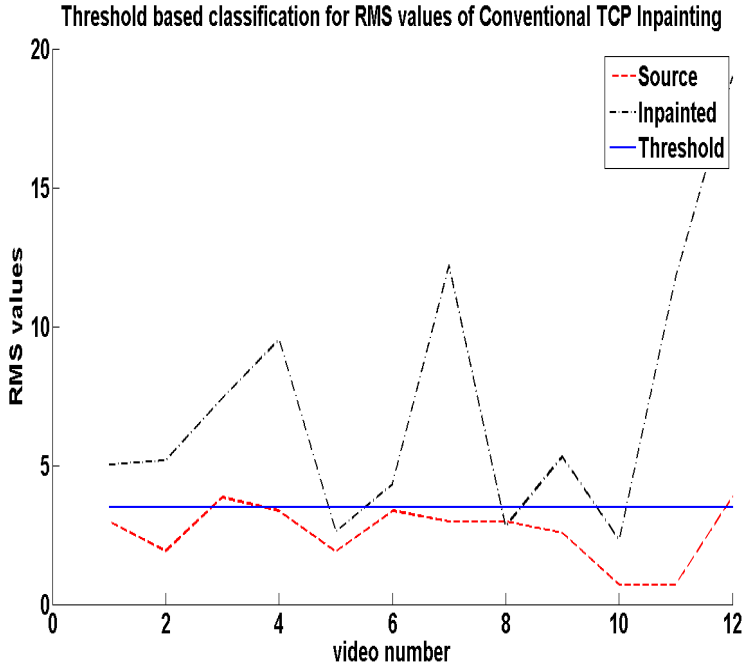


Figure 4.1: RMS values classification for Conventional TCP inpainting

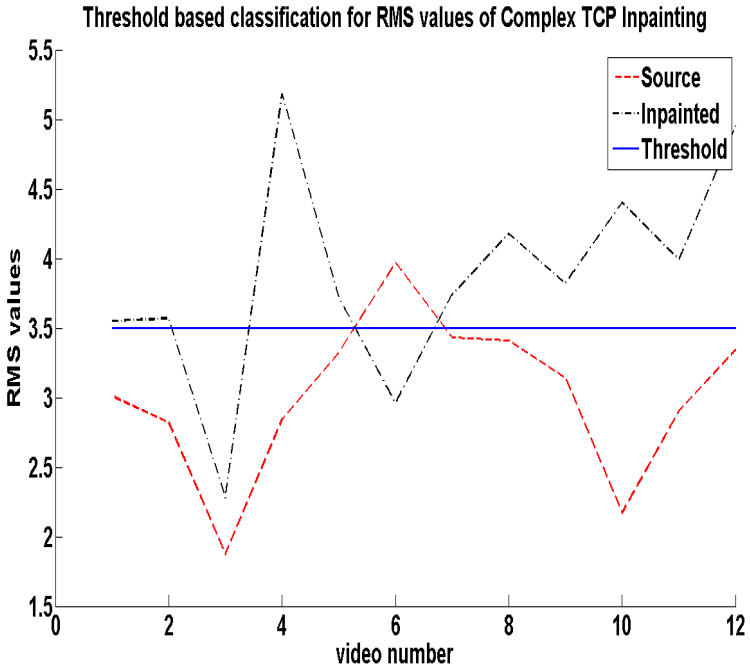


Figure 4.2: RMS values classification for Complex TCP inpainting

Table 4.5: Spatial TPM for complex TCP inpainting

Video ID	Source Frames Classification	Inpainted Frames Classification
Vid1 Set1	10	0
Vid1 Set2	10	4
Vid1 Set3	10	3
Vid1 Set4	10	2
Vid2 Set1	4	10
Vid2 Set2	2	10
Vid2 Set3	1	10
Vid2 Set4	1	8
Vid3 Set1	1	10
Vid3 Set2	6	10
Vid3 Set3	9	10
Vid3 Set4	10	9
Vid4 Set1	6	2
Vid4 Set2	9	0
Vid4 Set3	7	0
Vid4 Set4	4	0
Vid5 Set1	9	7
Vid5 Set2	10	9
Vid5 Set3	10	8
Vid5 Set4	9	5
Vid6 Set1	8	10
Vid6 Set2	10	10
Vid6 Set3	10	10
Vid6 Set4	9	10
Vid7 Set1	6	7
Vid7 Set2	5	8
Vid7 Set3	5	5
Vid7 Set4	4	10
Vid8 Set1	5	10
Vid8 Set2	10	10
Vid8 Set3	8	10
Vid8 Set4	2	10
Vid9 Set1	0	10
Vid9 Set2	2	10
Vid9 Set3	7	10
Vid9 Set4	10	10
Vid10 Set1	9	10
Vid10 Set2	10	10
Vid10 Set3	10	9
Vid10 Set4	10	10
Vid11 Set1	7	10
Vid11 Set2	9	10
Vid11 Set3	10	10
Vid11 Set4	10	10
Vid12 Set1	9	10
Vid12 Set2	10	10
Vid12 Set3	7	10
Vid12 Set4	8	10

Table 4.6: Spatial TPM for conventional TCP inpainting

Video ID	Source Frames Classification	Inpainted Frames Classification
Vid1 Set1	8	10
Vid1 Set2	10	10
Vid1 Set3	10	10
Vid1 Set4	10	10
Vid2 Set1	4	10
Vid2 Set2	8	10
Vid2 Set3	5	10
Vid2 Set4	3	10
Vid3 Set1	0	10
Vid3 Set2	0	10
Vid3 Set3	0	10
Vid3 Set4	3	10
Vid4 Set1	10	10
Vid4 Set2	10	10
Vid4 Set3	9	10
Vid4 Set4	8	9
Vid5 Set1	8	10
Vid5 Set2	4	2
Vid5 Set3	7	0
Vid5 Set4	4	6
Vid6 Set1	5	10
Vid6 Set2	8	5
Vid6 Set3	7	0
Vid6 Set4	8	0
Vid7 Set1	10	10
Vid7 Set2	10	10
Vid7 Set3	10	10
Vid7 Set4	10	10
Vid8 Set1	10	0
Vid8 Set2	10	0
Vid8 Set3	10	0
Vid8 Set4	10	0
Vid9 Set1	10	0
Vid9 Set2	10	0
Vid9 Set3	10	0
Vid9 Set4	10	0
Vid10 Set1	10	10
Vid10 Set2	10	10
Vid10 Set3	10	10
Vid10 Set4	10	10
Vid11 Set1	0	8
Vid11 Set2	0	9
Vid11 Set3	0	8
Vid11 Set4	0	3
Vid12 Set1	0	0
Vid12 Set2	0	0
Vid12 Set3	0	0
Vid12 Set4	8	0

4.4 Performance Evaluation

The performance is measured by precision, recall and accuracy. We say that a frame is a positive instance if it is tampered; otherwise it is a negative instance. Let TP be the number of true positives, i.e the number of positive instances which are detected as tampered, and FP be the number of false positives, i.e the number of instances that are detected as tampered. Similarly we define TN to be the number of true negatives and FN to be the number of false positives. Accuracy gives what percentage of predictions were correct. We have the following definitions

$Precision(P) = TP/(TP + FP)$

$Recall(R) = TP/(TP + FN)$

$Accuracy(A) = (TP + TN)/(TP + TN + FP + FN)$

4.4.1 Video Inpainting Detection

First we evaluate the performance of video inpainting detection step. Table 4.7 and 4.8 gives the classification of video results for complex TCP and conventional TCP inpainting techniques, respectively. For classification, total number of test videos are 12.

Table 4.7: Classification Results for complex TCP

Classification	Inpainted	Authentic
Inpainted	10	2
Authentic	1	11

Table 4.8: Classification Results for conventional TCP

Classification	Inpainted	Authentic
Inpainted	9	3
Authentic	2	10

On the basis of above classification results performance of detection algorithm on two targeted inpainting techniques is given in table 4.9.

Table 4.9: Inpainting Detection Performance

Inpainting type	*P*	*R*	*A*
Complex TCP	83.33%	90.9%	87.5%
Conventional TCP	75%	81.81%	79.16%

4.4.2 Video Inpainting Localization

In this section, we evaluate performance on the basis of correct classification of sets of non overlapping frames in a video. Size of each set is 10 frames and there are 48 such sets for each inpainting type. Table 4.10 and 4.11 gives the classification of frames set results for complex TCP and conventional TCP inpainting techniques, respectively.

Table 4.10: Classification Results for complex TCP

Classification	**Inpainted**	**Authentic**
Inpainted	38	10
Authentic	11	37

Table 4.11: Classification Results for conventional TCP

Classification	**Inpainted**	**Authentic**
Inpainted	30	18
Authentic	15	33

On the basis of above classification results performance of temporal localization algorithm on two targeted inpainting techniques is given in table 4.12.

Table 4.12: Inpainting Localization Performance

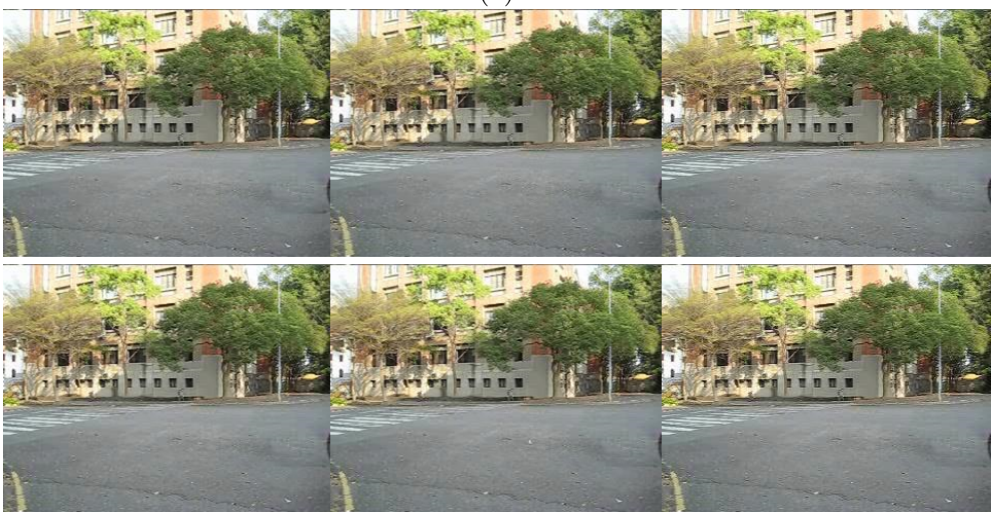
Inpainting type	*P*	*R*	*A*
Complex TCP	79.16%	77.55%	78.125%
Conventional TCP	62.5%	66.66%	65.62%

4.5 Comparison with previous approaches

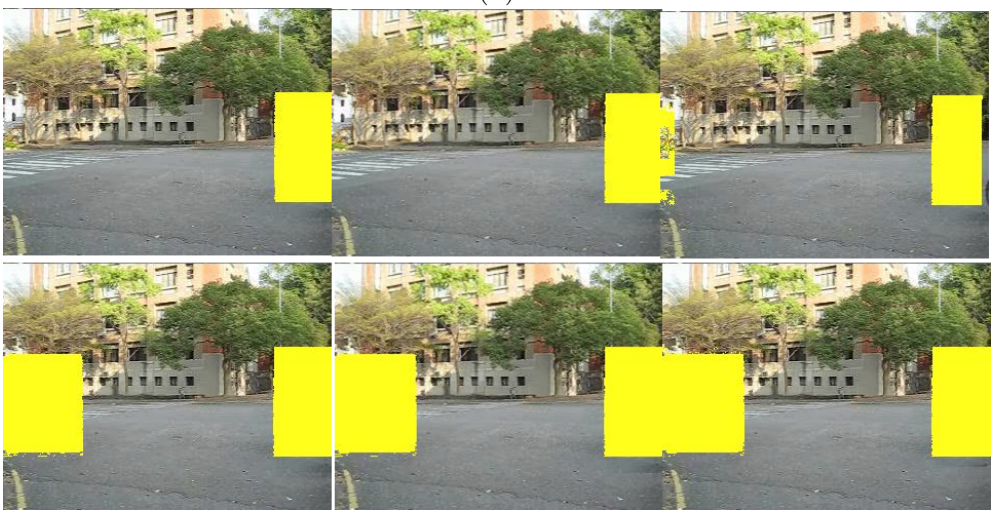
In previous methods, [28] [20] video inpainting detection and spatial localization have been performed. Hsu et al. [28] proposed an approach for locating inpainted regions in a video using correlation of noise residue. The method is based on the assumption that tampering will change the correlation of noise residue and make it different from that of the non-tampered parts. Simulation results show that the noise correlation is fairly reliable feature in fine-quality video, but it is sensitive to quantization noise. Furthermore, the extraction of noise residue process is very complex. Moreover, they have performed their experiments on only 3 videos, which is a small dataset. In [20] Lin et al approach, even though detection of conventional TCP inpainting [7] is reasonably good as shown in Figure 4.3 and 4.4, their algorithm fails to incorporate the detection of latest and very effective state-of-the-art-inpainting technique like complex TCP inpainting [8] which deals with challenging situations like multiple objects motion, moving cameras, moving backgrounds and reconstruction of dynamic textures in a video. This spatio-temporal coherance based approach [20] was when implemented by us over complex TCP inpainting dataset, the results obtained were quite sporadic as shown in Figure 4.5 and 4.6. It is evident from the Figures below that the same algorithm that performs well with conventional TCP, fails to perform on complex inpainted videos. Also, though our algorithm does not perform spatial localization as in [20] on videos, our temporal localization scheme of identifying which frames in a video are inpainted , using spatial TPM performs equally well in both conventional TCP and complex TCP. This occurs because of the efficient characterization of optical flow of video frames by spatial TPM features. The

proposed algorithm is the first method to evaluate its performance on both state-of-the-art inpainting methods of complex [8] and conventional TCP [7].

(a)



(b)

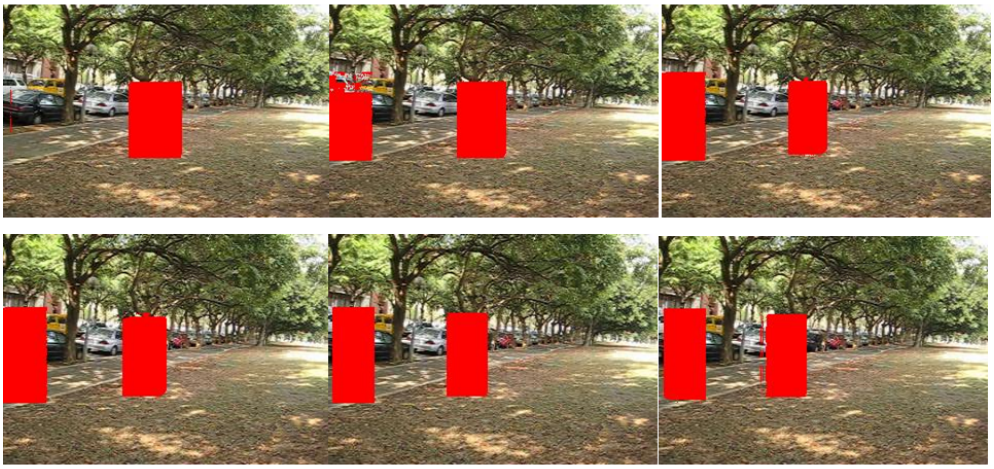


(c)

Figure 4.3: Conventional TCP results - Video 1 (a) Source Video (b) Conventional TCP inpainted video (c) Inpainting Localization 41

(a)

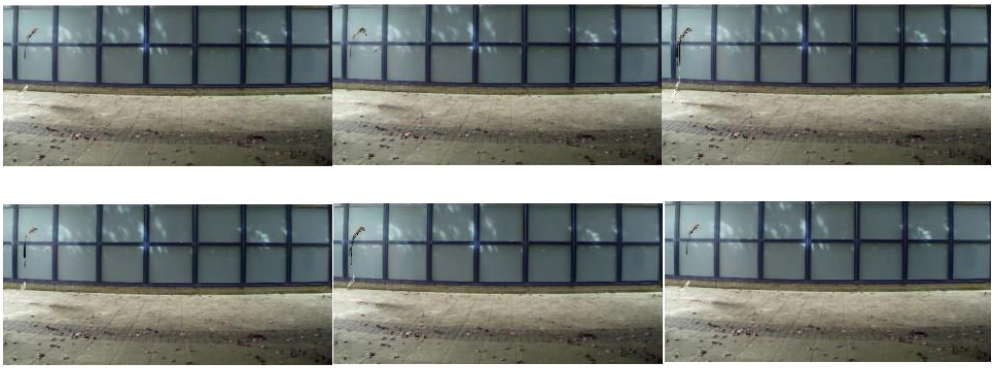
(b)



(c)

Figure 4.4: Conventional TCP results - Video 2 (a) Source Video (b) Conventional TCP inpainted video (c) Inpainting Localization

(a)



(b)

(c)

Figure 4.5: Complex TCP results - Video 1 (a) Source Video (b) Complex TCP inpainted video (c) Inpainting Localization

(a)

(b)

(c)

Figure 4.6: Complex TCP results - Video 2 (a) Source Video (b) Complex TCP inpainted video (c) Inpainting Localization

5

Discussion,Limitations and Future Work

In this study, we have proposed an algorithm for video inpainting detection and temporal localization based on optical flow inconsistencies. First, we begin by giving an overview of video forensics and highlighting how advancement in network technologies, low cost multimedia devices and digital media editing tools have made it easy to produce and modify digital videos with increasing sophistication. We then discuss about the existing forgery detection techniques and their applied research methodologies.

In next part of the study we discuss our approach based on optical flow inconsistency. We briefly explain Brox method of optical flow computation employed in our research method. We have implemented our algorithm in two steps. In first step we perform video inpainting detection in the given video and in second step we temporally localize inpainted regions present. For video inpainting detection we perform GMM based feature extraction to estimate parameters. On estimated RMS values we set a threshold for classifying a video as either inpainted or authentic. Next, for localization we model optical flow as first order Markov process to extract TPM features. Then SVM classification is performed on these extracted TPM features to decide whether a block of non overlapping frames is authentic or inpainted.

In chapter 4, we present our results on both targeted inpainting techniques, complex and conventional TCP. We also compare our algorithm with previously existing inpainting detection algorithm and show that our algorithm is able to detect and temporally localize complex TCP inpainting in a much better way in its comparison. We have considered varied types of object motion and in conventional TCP inpainting datset we have considered cases where multiple objects have been removed.

The limitation of this proposed work is that it considers only static camera videos with no background motion. In complex TCP inpainting dataset, we have taken only one object removal case and cases where multiple objects may be inpainted is not taken into account.

In future, we would like to extend this algorithm on videos with camera motion and dynamic background motion. We would like to work on detection of more inpainting techniques perform spatial localization too in addition to temporal localization in videos.

References

[1] Michihiro Kobayashi, Takahiro Okabe, and Yoichi Sato. **Detecting forgery from static-scene video based on inconsistency in noise level functions**. *Information Forensics and Security, IEEE Transactions on*, **5**(4):883–892, 2010. 1, 8

[2] Joachim J Eggers and Bernd Girod. **Blind watermarking applied to image authentication**. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, **3**, pages 1977–1980. IEEE, 2001. 1

[3] Ramarathnam Venkatesan, S-M Koon, Mariusz H Jakubowski, and Pierre Moulin. **Robust image hashing**. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, **3**, pages 664–666. IEEE, 2000. 1

[4] Ching-Yung Lin. *Watermarking and digital signature techniques for multimedia authentication and copyright protection*. PhD thesis, Columbia University, 2000. 1

[5] Valentina Conotter, Giulia Boato, Hany Farid, and E Roosevelt. **Active and passive multimedia forensics**. 2011. 2, 8

[6] Yuping Shen, Fei Lu, Xiaochun Cao, and Hassan Foroosh. **Video completion for perspective camera under constrained motion**. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, **3**, pages 63–66. IEEE, 2006. 2, 3

[7] Kedar Patwardhan, Guillermo Sapiro, Marcelo Bertalimo, et al. **Video inpainting under constrained camera motion**. *Image Processing, IEEE Transactions on*, **16**(2):545–553, 2007. 2, 3, 8, 9, 29, 39, 40

[8] Alasdair Newson, Andrés Almansa, Matthieu Fradet, Yann Gousseau, and Patrick Pérez. **Video inpainting of complex scenes**. *arXiv preprint arXiv:1503.05528*, 2015. 2, 3, 29, 39, 40

[9] Chih-Hung Ling, Chia-Wen Lin, Chih-Wen Su, Yong-Sheng Chen, and Hong-Yuan Mark Liao. **Virtual contour guided video object inpainting using posture mapping and retrieval**. *Multimedia, IEEE Transactions on*, **13**(2):292–302, 2011. 2, 3

[10] Sreelekshmi Das Gopu Darsan and Shreyas L Divya Devan. **Blind detection method for video inpainting forgery**. 2012. 3, 9

[11] A Jessica Fridrich, B David Soukal, and A Jan Lukáš. **Detection of copy-move forgery in digital images**. In *in Proceedings of Digital Forensic Research Workshop*. Citeseer, 2003. 3, 8

[12] J.A. Bloom I.J. Cox, M.L. Miller. **Digital watermarking and fundamentals**. 2002. 6

[13] Huiping Guo, Yingjiu Li, Anyi Liu, and Sushil Jajodia. **A fragile watermarking scheme for detecting malicious modifications of database relations**. *Information Sciences*, **176**(10):1350–1378, 2006. 6

[14] Ferdinando Di Martino and Salvatore Sessa. **Fragile watermarking tamper detection with images compressed by fuzzy transform**. *Information Sciences*, **195**:62–90, 2012. 6

[15] Wu-Chih Hu, Wei-Hao Chen, Deng-Yuan Huang, and Ching-Yu Yang. **Effective image forgery detection of tampered foreground or background image based on image watermarking and alpha mattes**. *Multimedia Tools and Applications*, pages 1–22, 2015. 6

[16] Hany Farid. **Image forgery detection**. *Signal Processing Magazine, IEEE*, **26**(2):16–25, 2009. 6

[17] Babak Mahdian and Stanislav Saic. **A bibliography on blind methods for identifying image forgery**. *Signal Processing: Image Communication*, **25**(6):389–399, 2010. 6

[18] Simone Milani, Marco Fontani, Paolo Bestagini, Mauro Barni, Alessandro Piva, Marco Tagliasacchi, and Stefano Tubaro. **An overview on video forensics**. *APSIPA Transactions on Signal and Information Processing*, **1**:e2, 2012. 6

[19] Weihong Wang and Hany Farid. **Exposing digital forgeries in interlaced and deinterlaced video**. *Information Forensics and Security, IEEE Transactions on*, **2**(3):438–449, 2007. 7, 8

[20] Cheng-Shian Lin and Jyh-Jong Tsay. **A passive approach for effective detection and localization of region-level video forgery with spatio-temporal coherence analysis**. *Digital Investigation*, **11**(2):120–140, 2014. 7, 9, 39

[21] WT Zhang, CH Feng, and ZQ Xu. **A dual-window detection scheme considered motion direction based on optical flow consistency for frame deletion detection**. In *Multimedia, Communication and Computing Application: Proceedings of the 2014 International Conference on Multimedia, Communication and Computing Application (MCCA2014), Xiamen, China, Oct 16-17, 2014*, page 277. CRC Press, 2015. 7

[22] Weiqi Luo, Min Wu, and Jiwu Huang. **MPEG recompression detection based on block artifacts**. In *Electronic Imaging 2008*, pages 68190X–68190X. International Society for Optics and Photonics, 2008. 7

[23] Weiqi Luo, Zhenhua Qu, Jiwu Huang, and Guoping Qiu. **A novel method for detecting cropped and recompressed image block**. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, **2**, pages II–217. IEEE, 2007. 7

[24] Hareesh Ravi, AV Subramanyam, Gaurav Gupta, and B Avinash Kumar. **Compression noise based video forgery detection**. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5352–5356. IEEE, 2014. 7

[25] N Mondaini, Roberto Caldelli, Alessandro Piva, Mauro Barni, and Vito Cappellini. **Detection of malevolent changes in digital video for forensic applications**. In *Electronic Imaging 2007*, pages 65050T–65050T. International Society for Optics and Photonics, 2007. 8

[26] Micah K Johnson and Hany Farid. **Exposing digital forgeries through chromatic aberration**. In *Proceedings of the 8th workshop on Multimedia and security*, pages 48–55. ACM, 2006. 8

[27] Weihong Wang and Hany Farid. **Exposing digital forgeries in video by detecting duplication**. In *Proceedings of the 9th workshop on Multimedia & security*, pages 35–42. ACM, 2007. 8

[28] Chih-Chung Hsu, Tzu-Yi Hung, Chia-Wen Lin, and Chiou-Ting Hsu. **Video forgery detection using correlation of noise residue**. In *Multimedia Signal Processing, 2008 IEEE 10th Workshop on*, pages 170–174. IEEE, 2008. 8, 39

[29] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. **Region filling and object removal by exemplar-based image inpainting**. *Image Processing, IEEE Transactions on*, **13**(9):1200–1212, 2004. 8, 9

[30] Jing Zhang, Yuting Su, and Mingyu Zhang. **Exposing digital video forgery by ghost shadow artifact**. In *Proceedings of the First ACM workshop on Multimedia in forensics*, pages 49–54. ACM, 2009. 8

[31] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. **A database and evaluation methodology for optical flow**. *International Journal of Computer Vision*, **92**(1):1–31, 2011. 10

[32] Steven S. Beauchemin and John L. Barron. **The computation of optical flow**. *ACM Computing Surveys (CSUR)*, **27**(3):433–466, 1995. 10

[33] Gilad Adiv. **Determining three-dimensional motion and structure from optical flow generated by several moving objects**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (4):384–401, 1985. 10, 12

[34] David W Murray and Bernard F Buxton. **Scene segmentation from visual motion using global optimization**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (2):220–228, 1987. 10, 12

[35] Patrick Bouthemy and Edouard François. **Motion segmentation and qualitative dynamic scene analysis from an image sequence**. *International Journal of Computer Vision*, **10**(2):157–182, 1993. 12

[36] Berthold K Horn and Brian G Schunck. **Determining optical flow**. In *1981 Technical symposium east*, pages 319–331. International Society for Optics and Photonics, 1981. 12

[37] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. **High accuracy optical flow estimation based on a theory for warping**. In *Computer Vision-ECCV 2004*, pages 25–36. Springer, 2004. 12, 13, 15

[38] ://vision.ucsd.edu/ pdollar/toolbox/doc/classify/pdist2.html. 21, 25

[39] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn. **Speaker verification using adapted Gaussian mixture models**. *Digital signal processing*, **10**(1):19–41, 2000. 22

[40] Hareesh Ravi, AV Subramanyam, and Sabu Emmanuel. **SPATIAL DOMAIN QUANTIZATION NOISE BASED IMAGE FILTERING DETECTION**. 25, 27

[41] Chih-Chung Chang and Chih-Jen Lin. **LIBSVM: A library for support vector machines**. *ACM Transactions on Intelligent Systems and Technology (TIST)*, **2**(3):27, 2011. 28

[42] Gavin C Cawley. **Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs**. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, pages 1661–1668. IEEE, 2006. 28

[43] **Surrey University Library for Forensics Analysis**. `http://sulfa.cs.surrey.ac.uk/`, 2012. 29

[44] `http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html#ClassificationDatabase`, 2012. 29

[45] **Performance evaluation of tracking systems 2009 dataset**. `http://www.pets2009.net`, 2009). 29

[46] ://sites.google.com/site/multimediaforensic/. 29

[47] Michihiro Kobayashi, Takahiro Okabe, and Yoichi Sato. **Detecting video forgeries based on noise characteristics**. In *Advances in Image and Video Technology*, pages 306–317. Springer, 2009.

[48] Weihong Wang and Hany Farid. **Exposing digital forgeries in video by detecting double MPEG compression**. In *Proceedings of the 8th workshop on Multimedia and security*, pages 37–47. ACM, 2006.