Detection of Copy-Move Forgery in Digital Images Using a Novel Feature Extraction Approach

A Dissertation submitted in partial fulfillment of the requirement for the

Award of degree of

MASTER OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

GAURAV KUMAR SHAH

(2K14/CSE/07)

Under the esteemed guidance of

RITU AGARWAL

(Asst. Professor, CSE Department)

Department of Computer Engineering
Delhi Technological University
Bawana Road, Delhi – 110042
(2014-2016)

CERTIFICATE

This is to certify that **Gaurav Kumar Shah (2K14/CSE/07)** has carried out the major project titled **“Detection of Copy-Move Forgery in Digital Images Using a Novel Feature Extraction Approach ”** for the award of Master of Technology degree in Computer Science and Engineering by Delhi Technological University.

The major project is a bonafide piece of work carried out and completed under my supervision and guidance during the academic session **2014-2016**. The matter contained in this report has not been submitted elsewhere for the award of any other degree.

(Project Guide)
Ritu Agarwal
Assistant Professor
Department of Computer Science And Engineering
Delhi Technological University
Bawana Road, Delhi-110042

ACKNOWLEDGEMENT

I express my sincere thanks and deep sense of gratitude to my project guide**, Ritu Agarwal**, Assistant Professor, Department of Computer Science And Engineering, Delhi Technological University, for his valuable motivation and guidance, without which this study would not have been possible. I consider myself fortunate for having the opportunity to learn and work under his supervision and guidance over the entire period of association.

I humbly extend my words of gratitude to other faculty members of this department for providing their valuable help and time whenever it was required.

Gaurav Kumar Shah

Roll No. 2K14/CSE/07

M.Tech. (Computer science and engineering)

E-mail: gkshahh@gmail.com

ABSTRACT

Digital images are easy to manipulate and edit due to availability of powerful image processing and editing software. Nowadays, it is possible to add or remove important features from an image without leaving any obvious traces of tampering. Copy –move forgery is One of the most commonly used forgery techniques in Copy-move forgery that copies a region of an image and pastes it on the other region in the same image. Several approaches have been developed so far to detect such kind of forgery. in our approach, first of all image go under pre-processing.in which image convert RGB to grey image. after that image divide in to blocks than on each blocks apply DCT(discrete cosine transform) for feature extraction. The DCT coefficients matrix is quantized and averaging is used to extract a feature set with reduce dimensionality. each feature vector are sorted by radix-sort which is less computation complex than lexicographical sort. that vectors corresponding to similar blocks come adjacent to each other. The decision of forgery can then be taken if a number of connected/similar blocks are equidistant to each other. For better result we optimize the distance by genetic optimization algorithm than match the similar blocks which are equidistant to optimized distance. The feature extraction is vital as not only does it reduce the computation complexity but choosing the correct set of features makes an approach invariant to various image manipulations such as rotation, translation, scaling, etc.

Table of Contents

LIST OF FIGURE

Figure Caption . **page**

List of Tables

Table Caption **page**

LIST OF ACRONYMS

ACRONYM	MEANING
DCT	Discrete Cosine Transform
PCA	Principal Component Analysis
DWT	Discrete wavelet transform
LBP	Local Binary Pattern
DFT	Discrete Fourier Transform
SIFT	Scale Invariant Feature Transform
SVD	Singular Value Decomposition
FMT	Fourier Millin Transform
SVM	Support Vector Machine
DyWT	Dyadic Wavelet Transform
FVM	feature vector matrix

CHAPTER 1

INTRODUCTION

1.1 History And Background In Digital Image

According to Merriam-Webster, forgery is defined as the crime of falsely and fraudulently making or altering a document (Forgery, Merriam-Webster). So therefore, digital forgery involves falsely altering digital contents such as pictures and documents. Digital forgery has occurred for many years and still remains a relevant topic today. We see it every day in newspapers, magazines, the television, and even the internet. Whether altering the way someone looks, using digital photography in a courtroom, or even bringing a celebrity back from the dead, digital photography and digital television stimulate countless questions and queries about the ethics and morals of digital forgery, with respect to today's technology, and the involvement of digital forgery in our daily lives. The questions that arise because of digital forgery can be addressed and evaluated successfully only through consideration of the history, usage, and ethics of digital forgery in order to determine how and in what ways restriction or limitation of digital forgery should occur.

The history of photography and digital forgery helps one realize exactly how digital forgery became a commonly utilized method. Photography dates back to as far as 1826 when a French inventor, Joseph Nicephore Niepce, produced the first everlasting photograph (Photograph, Wikipedia). As time went on, photography got more advanced and more complicated. Soon enough, colour photographs were being produced. The first colour photograph was produced by a Scottish physicist named James Clerk Maxwell (Photograph, Wikipedia). After the production of colour photograph came the invention of film, which led to instant cameras, automatic cameras, and finally digital cameras. Digital photography started first in 1951 with a video tape recorder that produced live images from television cameras by altering the ability to create digital photographs opened up the doors for producing counterfeit images and made it easier for this to be done. However, it wasn't the first time we've seen manipulation in pictures. As a matter of fact, photo manipulation dates back to the 1860s when a picture of John C. Calhoun was edited to have his body with the head of Abraham Lincoln (Photo Manipulation, Wikipedia).

Picture-editing software often comes readily installed with most current computers, meaning that most people with current computers or lap tops have access to technology for digital editing. Social networking websites, such as facebook.com and myspace.com, give users the ability to post up almost any type of picture or photo, regardless of whether the picture has undergone some type of alteration. Though often misleading, especially in the cases of digital forgery with pictures of the actual user, the question of whether the altering of the picture itself is right or wrong depends on the users viewing the image and their opinions. Depending on the intent of those who partake in digital forgery, the misleading appearances of digital forgery could potentially be detrimental to the other people.

Hypothetically, a user of a social networking website might try to establish a relationship with another user based on the viewing of digitally altered pictures or photographs of that user. What is conveyed on a computer screen with a digitally forged picture may differ greatly from the actual appearance of a user, and this will likely cause a variety of problems for users who try and establish a relationship that is even partially based on the appearance of the other user. However, the user with the altered picture benefits in the sense that he or she increases the likelihood of establishing a relationship with another person based on the digitally altered picture and not the truthful portrayal of that person in reality.

Digital Image: Any such image which is derived from or processed with a digital device. The digital device or gadget may be a digital camera, computer, etc. In simpler words any image that is either derived via a digital device or an image which can be viewed in digital device are called digital images.

A digital image is a numeric representation of (normally binary) a two-dimensional image. Depending on whether the image resolution is fixed, it may be of vector or raster type. By itself, the term "digital image" usually refers to raster images or bitmapped images.

Raster images have a finite set of digital values, called picture elements or pixels. The digital image contains a fixed number of rows and columns of pixels. Pixels are the smallest individual element in an image, holding antiquated values that represent the brightness of a given colour at any specific point.

Vector images resulted from mathematical geometry (vector). In mathematical terms, a vector consists of point that has both direction and length.

Often, both raster and vector elements will be combined in one image; for example, in the case of a billboard with text (vector) and photographs (raster).

1.2 Digital Image Forensics

1.2.1 Detection of Computer Generated Images

Because of the image processing and editing software, it becomes very easy for anybody to manipulate, edit change digital images. This opened up the chance not only for experts but also to starters to generate and render computer generated images with a high level of realism that makes it difficult for the viewer to detect manipulation and to distinguish them from real non computer generated picture, especially when displayed for short period of time. Figure (1) displays some of the images that are created by a professional with the result. They look like photographs "but nothing we are about to see is real".

Figure (1): Examples of computer generated images.

Many years ago the study of computer generated images began with the classification methods of the type of images (Athitsos, et al., 1997)[5]. The aim of these methods is to differentiate between graphics and photographs. However, the difference between graphic images such as cartoons and clip arts was great and can be easily detected. Since 2005, digital images forensic has begun to attract the attention of multimedia forensic community leading to many studies on the problem of identifying photorealistic computer graphics.

1.2.2 Image Source Identification: image source identification is considered one of the basic problems that the techniques of digital image forensic try to solve. This technique aims in particular to recover the used type of imaging devices such as digital cameras, scanners, mobiles, computer, and graphic technologies. By focusing on visual data, the specific model or brand of such devices can be identified, since each digital imaginary device leaves a unique imprint on the acquired image during the processing operation or the storing phases. This difference arises from the fact that images have different characteristics because of using different physical devices and different image processing techniques. Although the internal processes of imaging devices are always known, there is always a difference in the techniques and the parameters used during the image creation yielding different patterns or traces of the resulting image. Here comes the importance of digital image forensic techniques to check out such patterns and to detect the image authentication and source identification (Luo, et al., 2007)[14].

Image source identification has two functions:

(1) To determine the device used to capture the image. This can be carried out through analyzing the image and matching its characteristic with the unique characteristics of a certain device to distinguish among different devices such as digital cameras and scanners, or different models of the same devices or different examples of the same model;

(2) To prove the claim that a certain image was captured by a specific device.

1.2.3 Image Forgery Detection

It can be said that forgery is related to an altered or reproduced object. Therefore, the main purpose of forgery analysis is to determine whether any changes were made to change the meaningful content of an image.

Is every altered image a forged one? Digital image forgery means the intentional manipulation of digital image, for the purpose of changing the semantic meaning of the visual message included in that image. There are some techniques applied to the image such as cropping, rotating or applying horizon correction which are widely accepted techniques since they alter the image without necessarily forging it. Image forgery detection has become one of the main goals of image forensics since digital images were presented as evidence in courts of law, as news items, as a part of medical records or as a financial document.

1.3 Mechanisms for Image Forgery Detection

Digital image forgery detection can be classified into two different groups based on whether the original image is available or not. These are active methods and passive methods.

1.3.1 Active Methods

An active detection method which consists of adding image details in order to describe digital tampering such as name, date, signature, etc. It requires a special hardware implementation to mark the authentication of the digital image (Kaur and Kaur, 2015)[8]. There are two types of active methods: Digital Watermaking and Digital Signatures (Al-Hammadi, 2013)[1].

 Digital Watermarking: where a specific digest is embedded inside any image at the time of recording. To verify authenticity, this digest can be extracted from the image at any time. If this digest is different from the original one that was embedded inside the image at the acquisition time, this means that the image was modified after the recording time (Al-Hammadi, 2013)[1].

- **Digital signature** extracts the unique properties of an image as a signature at the image capturing end, while regenerating it using the same method .When comparing both signatures the authenticity of the image can be identified.

The active method has some advantages such as low computational cost and the availability of simplified knowledge about the original image. On the other hand, it has some drawbacks since these techniques require previous knowledge of the original image and so they are not automatic, in addition to requiring human intervention or specially-equipped cameras. Moreover, millions of digital images on the internet don’t have a digital signature or watermark. This means that the active approach could not be used to verify the authenticity of such images, especially that the digital signature scheme requires extra bandwidth to transmit the signature (Kaur and Kaur, 2015)[8].

1.3.2 Passive Methods (Blind Methods)

Passive methods of image forgery detection are regarded as a new direction of research. In recent years, there has been significant work performed in this highly active area of research. Passive approaches do not depend on hidden data to detect image forgeries, but only utilize the statistics and/or content of the image in question to verify its genuineness.

Thus, the binary information of the digital image is analyzed to detect if there is any manipulation or forgery. Passive methods may detect manipulation depending on this variance in consistency before and after the operation. When digital watermarks or signature are not available, passive methods can serve as the only method to decide upon the trustworthiness of a digital image. Passive approach of digital image forgery detection consists of two categories of techniques:(1) The Statistical Method which is based on the value of pixels in an image; (2) The Visual Method which is based on the visual cases to detect the inconsistency of the image itself such as lightening inconsistency. Statistical methods are stronger and more potent than visual methods since they depend on analyzing the digital image pixel by pixel (Al-Hammadi, 2013)[1].

The passive method has some advantages such as it does not require any sort of prior information and work purely on binary data. The detectable traces are authentic and have not been modified in image (Kirchner and Bohme, 2008). On the other hand, it has some drawbacks is their need for many prior images to estimate the internal traces, while in potential situations there is no more than the image in question. These techniques based on the assumption that digital forgeries may leave no visual clues that indicate tampering, so they require different statistics of an image. Thus it is complex (Kaur and Kaur, 2015)[8].

1.3.3 Block Based Approach

In this the image is divided into blocks of fixed size so that instead of randomly matching the pixels with each other these blocks are matched against each other. The image is divided into overlapping blocks of fixed size and it is presumed that size of duplicated region would be larger than the block's size. With this, we are sure of the fact that there are many blocks within the duplicated regions. The idea of detection is to match each block against each other and find a number of connected blocks that matches with another set of blocks that have same number of blocks and are connected. When we match two blocks, it is not the exact pixel by pixel values that we are matching since it is not computationally viable. So we extract some important features from each block, represent each block by these few important features thereby reducing the dimension of the block and hence make matching of the blocks easier and computationally viable.

1.3.4 Key-point based Approach

Key-points are those locations within an image that carry the distinct information of the image. It may be local extrema or minima or other such factor by which a region can be clearly distinguished from its neighbouring regions.

In this detection scheme, we do not divide the complete image into fixed size, overlapping blocks, instead some distinct features or region of interests are extracted from the image itself. When these locations are found, then we extract useful features from them. In this way the number of blocks reduces drastically. There are several approaches such as SIFT and MIFT to extract the features from an image. A number of features so extracted are known as key-points of the image. The region around these key-points is matched against each other for similarity. This is different from the block based approach in the way that only the key-points (area around key-points) are matched against each other instead of matching each block of image with each other. From the neighbourhood of each key-point say 128 pixel neighbourhood some characteristics features are extracted to represent each key-point. Now instead of matching the whole neighbourhood of key-points, the characteristics feature vector with respect to each key-point is matched against each other. Naturally there would be less key-point than number of overlapping blocks in an image. Hence it computational complexity is better. However there are some inherent problems with this approach. Many a times it fails to cover the whole image and there may be significant forgeries left undetected.

1.4 Digital Image Forgery Types

The semantic contents of a digital image may be altered through removing information from that image, or adding extra information to it. There are numerous ways that forgers may use to achieve that. In addition, different criteria are used to classify those techniques such as the number of images involved in the manipulation operations (Al-Hammadi, 2013)[1]. These techniques can be classified into five major categories: Image Retouching, Image Enhancing, Image Morphing, Image Splicing, and Copy-Move (Cloning), and. In this section, each of these categories is described briefly.

1.4.1 Image Retouching:

Digital images retouching is considered the less harmful kind of digital image forgery, since it does not make significant changes to the visual message of an image(Al-Hammadi, 2013)[1]. In this method, the image editors change the background, fill some attractive colors, and work with hue saturation for toning and balancing. It is used for enhances an image or reduces certain feature of an image and enhances the image quality for capturing the reader's attention as shown in Figure (2).

This does not completely change an image but alters certain features such as enhancing some features while reducing others. A lot of photographers and editors of various magazines and newspapers very often make use of this method to make their pictures more appealing and eye-catching.

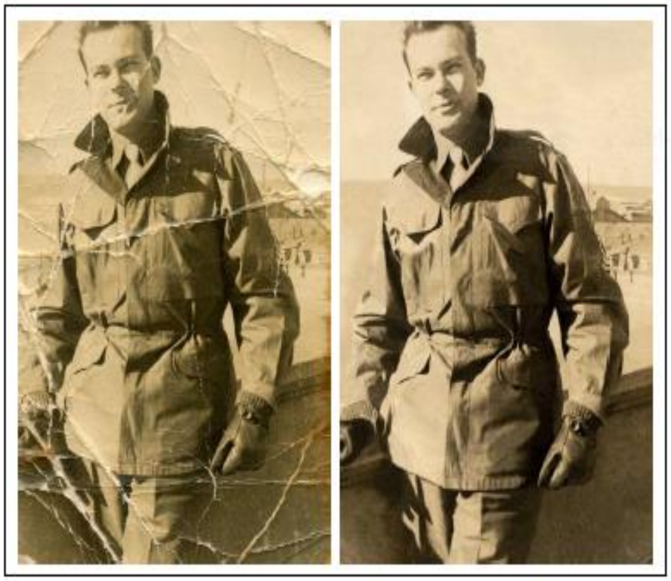


Figure (2): "Image Retouching" image forgery:
left is Original image, and right is the forged one.

1.4.2 Image Enhancing:

This technique is based on enhancing an image with the help of Photoshop such as saturation, blur and tone etc. These enhancements don't affect or alter the image meaning or appearance. Still, they can have a subtle effect on the interpretation of an image. For example, simple enhancements can obscure or exaggerate image details, or alter the time of day in which the image appears to have been taken (Granty, et al., 2010). See Figure (3).

This is a type of forgery which came into existence long ago, just about the same time when photography was discovered. Compared to other forgeries this is still less dangerous as the implications are less harmful. This does not completely change an image but alters certain features such as enhancing some features while reducing others. A lot of photographers and editors of various magazines and newspapers very often make use of this method to make their pictures more appealing and eye-catching.

Figure (3).Image Retouching

1.4.3 Image Morphing:

Image morphing is a digital technique that gradually transforms one image into another through using smooth transition between two images. An example of morphing image is shown in Figure (4), which is available at (xmorph.sourceforge.net). On the top, there are two original images overlaid with the feature correspondence required for morphing and on the below five images are overlaid from a morphed sequence.

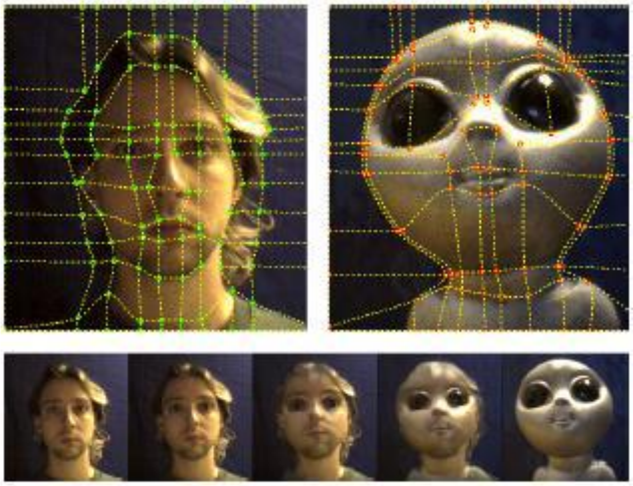


Figure (4).Image Morphing

1.4.4 Image Splicing:

Different elements from multiple images are juxtaposed in a single image to convey an idea that doesn’t reflect reality. Such splicing can usually be detected by searching the splicing boundary, or the effect of splicing on image statistics, or by considering the directions of the light incident on the image surfaces (Grantee, et al., 2010). A sample of image splicing is illustrated in Figure (5).

It is defined as the process where two or more than two images are combined and a single image is formed. In this, portion of one image is taken and combined onto another image. For example, we see in tabloid magazines face of some other celebrity on the body of some other person. When carried out cautiously, the boundary linking the spliced portions can be visually undetectable. This method is more aggressive than the previous one, that is, image retouching.

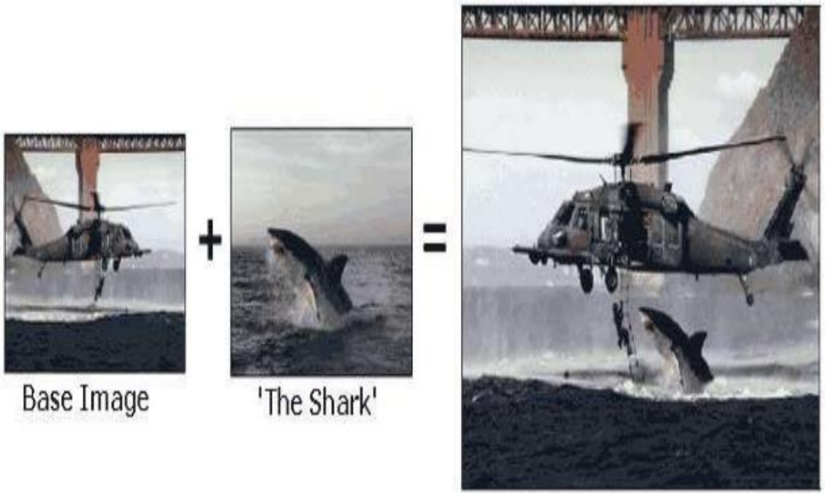


Figure (5).Image Splicing.

1.4.5 Copy-Move (Cloning):

Copy-move forgery, also known as Cloning when only one image is considered for the forging process, is more or less similar to image splicing in view of the fact that both techniques modify a certain image region with another image. One region is copied from an image and pasted onto another region of the same image. However, instead of using an external image as a source, copy-move forgery uses portions of the original base image as a source which means that the same image is both the source and the destination of the modified image. Because the copied parts come from the same image, its important properties, such as noise, texture, and brightness, will be compatible with the rest of the image making it more difficult for experts to distinguish and detect the alteration (Sridevi, et al., 2012)[11].

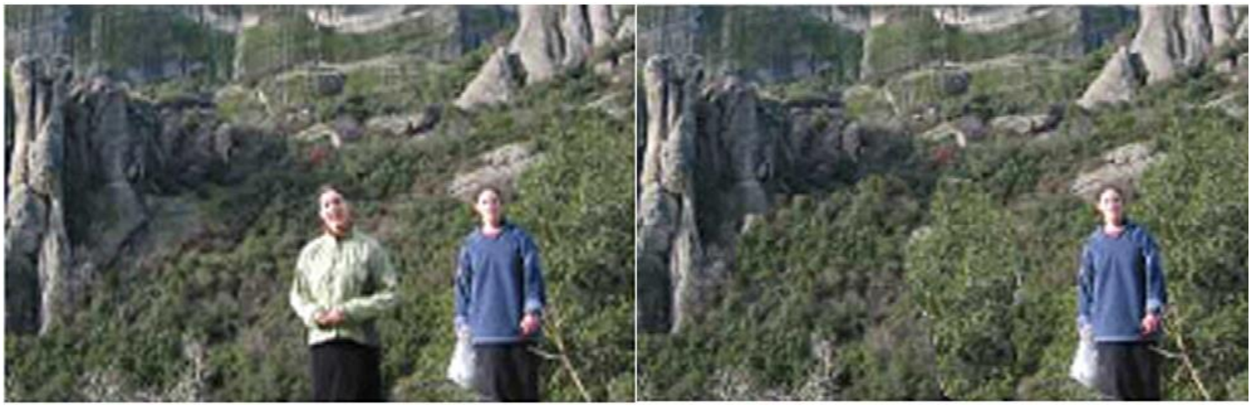


Figure (6). Copy-Move Forgery
Example(1)

Figure (5) illustrates an example of this type of forgery, which appeared in a Tunisian newspaper (Amerini, et al., 2013)[4]. The photo has been manipulated in order to make the crowd appear larger. This demonstrates that this kind of manipulation is used more and more often in news and advertising campaigns.

Iran also released an altered photograph (Figure (6)) on July 9, 2008, showing four missiles rising into the air instead of three during a test firing at an undisclosed location in the Iranian desert. Ironically, several western media, including New York Times and Los-Angeles Time published that counterfeited image as if it was an authentic image.

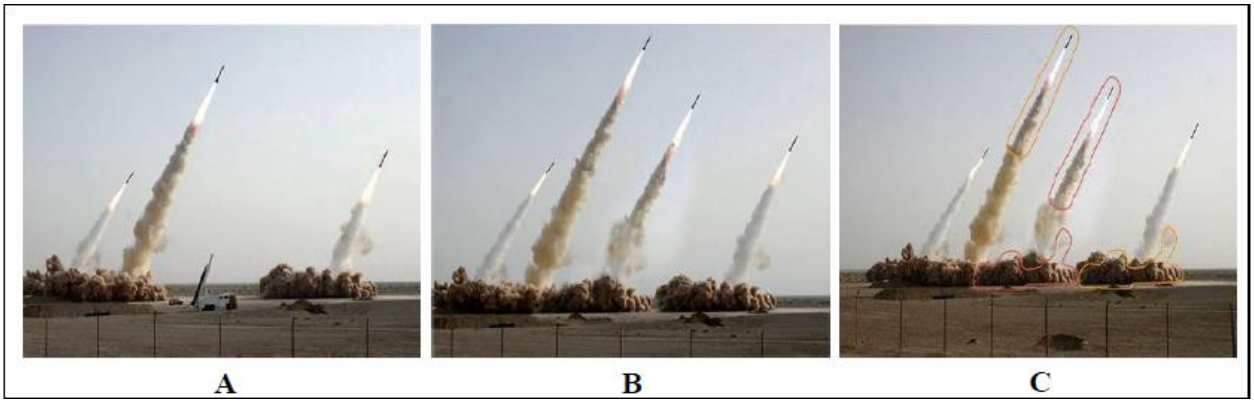


Figure (7).“A” is the original image, “B” is manipulated, and “C” The duplicated regions.

Example (2)

Chapter 2
Literature Review And Survey

The aim of copy-move forgery detection is to detect duplicated image regions, to the very slight difference in each of such regions (Bayram, et al., 2009)[2]. There are a large number of published papers on copy-move detection that can be found in the literature, and the number of those papers is increasing. This chapter gives a review of available relevant literature. It begins with a general background of the techniques on Copy- move forgery detection. Then some of the previous studies are mentioned.

2.1 Copy- Move Forgery Detection Techniques

Copy-move forgery can be implemented effectively and easily making it the most common forgery process used to alter the content of an image. However, in practical situations, forgery may involve more than a simple duplicating process. Several image processing operations could be involved in practical copy-move forgery. These operations can be divided into two groups:

A. **Intermediate processing operations** that are used to provide a type of spatial synchronization and homogeneity between the copied region and the other parts of the image. These operations include rotation, scaling, mirroring, illumination modifying, or chrominance modifying. In a practical situation, intermediate processing may include a combination of two or more operations.

B. **Post-processing operations** that are used to remove any detectable traces of the copy-move operation, such as sharp edges. These operations include the additive noise, JPEG compression or blurring. This indicates that any suggested detection algorithm had to be robust against such operations (Liu, et al., 2010)[9]. Therefore, the following requirements must be available in any detection algorithm (Fridrich, et al., 2003)[3]:

- the detection algorithm must allow for an approximate match of small image segments.

- the detection algorithm must work in a reasonable time while introducing few false positives (i.e., detecting incorrect matching areas).

- the forged segment is likely to be a connected component rather than a collection of very small patches or individual pixels, a natural assumption that has to be generally accepted.

- the detection algorithm must be robust against image processing operations.

On the other hand, copy-move forgery detection techniques can be classified into two main classes depending upon the segmentation plan:

- **“Block-based” approaches** that uniformly divide the image into small overlapping/ non-overlapping rectangular or circular partitions called “blocks” of fixed size.

- **“Key point-based” approaches** that compute their features only on image regions with high entropy and without any image subdivision.

Figure (7) shows the common processing pipeline for Copy- Move forgery detection. Additional relevant details to pipeline's steps are presented below (Christlein, et al., 2012)[6].

- **Preprocessing:** For instance, most methods operate on grayscale images, and thus the color channels evvi to be first merged.

- **Feature extraction:** Even in the case of post-processing, accuracy detection depends mainly on the ability to map the blocks in a copy-move pair to similar features. Block-based methods subdivide the image into rectangular or circular regions. For every such region, a feature vector is computed. Subsequently, similar feature vectors are matched. By contrast, key point-based methods compute their features only on image regions with high entropy, without any image subdivision (Christlein, et al., 2012).

- **Matching:** High similarity between two feature descriptors is interpreted as a cue for a duplicated region. For block-based methods, most authors propose the use of lexicographic sorting in identifying similar feature vectors. In lexicographic sorting a matrix of feature vectors is built so that every feature vector becomes a row in the matrix. This matrix is then sorted along with the direction of this row. Thus, the most similar features appear in consecutive rows. In particular, key point-based methods often use the Best-Bin-First search method derived from the k-d tree algorithm to get the approximate nearest neighbors.

- **Filtering:** Filtering schemes have been proposed in order to reduce the possibility of making false matches. For instance, a common noise suppression measure involves the removal of matches between spatially close regions. Neighbouring pixels often have similar intensities, which may lead to false forgery detection (Christlein, et al., 2012)[6].

- **Post-processing:** The goal of this last step is to only preserve matches that exhibit a common behavior. If a set of matches belong to a copied region, these matches are expected to be spatially close to each other in both the source and the target blocks (or key points). Furthermore, matches that originate from the same copy-move action should exhibit similar amounts of translation, scaling and rotation (Christlein, et al., 2012)[6].

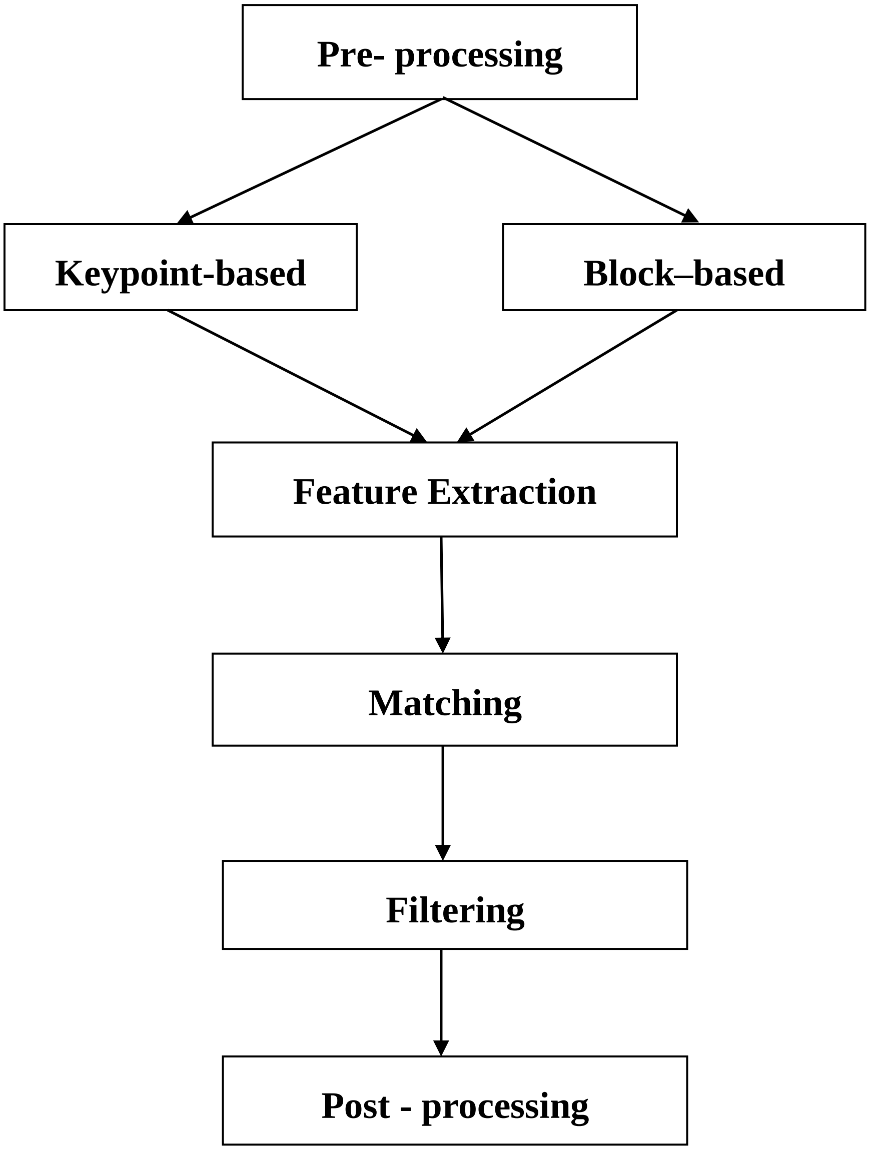


Figure (8): Common processing pipeline for the detection of copy-move forgeries.

2.2 Literature Review of Copy-move Forgery Detection

In literature, there are large and increasing numbers of published papers on passive copy-move detection that require a huge amount of effort and time to be reviewed. First of all, a direct solution to this problem would be an exhaustive search involving comparison of the image to every cyclic-shifted version of itself. However, this approach would be computationally very expensive and would take (*MN*) 2 steps for an image of size M × N. Moreover, this type of search might not work in the case of having modifications made on the copied area (Bayram, et al., 2008). Other techniques aim to increase the efficiency and minimize the complexity of the algorithm. The main difference between the existing methods lies in the type and size of the features used for matching the image blocks. Some of these techniques are summarized below:

2.2.1 DCT-Based Methods

Among the initial attempts, (Fredrich, et al., 2003)[3] proposed a method to detect copy-move forgery. Discrete Cosine Transform (DCT) of the image blocks was used and their lexicographical sorting was considered to avoid the computational burden. Once sorted, the adjacent identical pair of blocks is considered to be copy-moved blocks. A drawback of this method is that it cannot detect small duplicate regions.

Cao et al. (2012), present region duplication detection algorithm which depends on improved DCT and exhibits low computational complexity. The profound difference between this method and the other DCT-based methods is that here the quantized block is characterized by a circle block. The circle block is then divided into a fixed number of parts, for which the feature vectors are calculated. Euclidean distance between adjacent pairs is calculated after lexicographic sorting of vectors. The actual distance between the similar vectors is also considered before the final call on duplication is made. This method is capable of identifying multiple region duplications and is also robust against blurring and additive noise but it has poor performance with poor image quality. It is not robust to geometrical operation either.

Zhao and Guo (2013)[12], proposed a robust method to detect copy-move forgery based on DCT and SVD. The image is divided into fixed-size overlapping blocks and 2D-DCT is applied to each block. The DCT coefficients are then quantized to obtain a more robust representation of each block followed by dividing these quantized blocks into non overlapping sub-blocks. SVD is applied to each sub-block. Afterwards, features are extracted to reduce each block dimension using its largest singular value. Finally, feature vectors are lexicographically sorted, and the duplicated image blocks are matched by predefined shift frequency threshold. Experimental results showed that the proposed method can detect copy-move forgery even when an image was distorted by Gaussian blurring; Additive White Gaussian Noise (AWGN), JPEG compression or any other related mixed operations.

2.2.2 PCA-based method

Popescu and Farid (2004)[10] suggested a method using Principal Component Analysis (PCA). In this method the image is transformed into grayscale and separated into many parts represented into vectors. These parts or blocks are organized lexicographically and PCA is used to represent the dissimilar blocks in a substitute mode. It is proficient for detecting even minor variations resulting from noise or wasted compression. Moreover, this technique is far efficient for grey scale images. It is better for detecting copy-move forgeries and gives less number of false positives. The computational cost and the number of computations required are considerably reduced O(*NtN logN*), where *Nt* is the dimensionality of the truncated PCA representation and *N* is the number of image pixels. Although this method has reduced complexity and is highly discriminative for large block size, its accuracy is reduced considerably for small block sizes and low JPEG qualities. The other main benefit of PCA is its ability to compress the data that is once patterns are identified in the data, it uses these patterns to compress the data that is it reduces the number of dimensions without any significant loss of information.

2.2.3 LBP-based method:

Al-Sawadi et al. (2013)[15], presented a copy-move image forgery detection method based on Local Binary Pattern (LBP) and neighborhood clustering. In the proposed method, an image is first decomposed into three color components. LBP histograms are then calculated from the overlapping blocks of each component. The histogram distance between the blocks is calculated and the block-pairs having the minimal distance are retained. If the retained block-pairs are present in all the three color components, they are selected as primary candidates. Eight-connected neighborhood clustering is then applied to refine the candidates. Experimental results show improvement in reducing the false positive rates over some recent related methods. The performance of the methods degrades when the pasted parts undergo both rotation and scaling.

Proposed a tampering detection method based on LBP. This algorithm can detect copied regions even if the geometry of the forged region is further polluted by noise, blurring, JPEG compression, scaling or rotation in multiples of 90-degree. In this algorithm the image is translated into gray scale and is then subdivided into overlapping blocks. Multi-resolution Local Binary Pattern (MLBP) features are identified for each block by applying different types of LBP operators. The feature vectors are put together to form feature matrices which the number is equal to the number of LBP operators employed. Feature matrices are lexicographically sorted and k-d tree method is used for determining the matching blocks. RANdom Sample Consensus (RANSAC) algorithm is then used to eliminate false matches. However, the method is still time consuming for forgery detection in high resolution images, and it cannot detect duplicated regions with arbitrary rotation angles either.

2.2.4 FFT-Based Methods:

Bayram, et al. (2009)[2], conducted a study to detect copy-move forgery by using Fourier-Mellin Transform (FMT). They choose FMT because it is robust to loss JPEG compression, blurring, noise, scaling and translation effects applied as post-processing. At the beginning, the image is divided into several small sized blocks and the Fourier Transform of each block is calculated. By doing so, they ensured that transform is translation invariant. Then the resulting magnitude values are re-sampled, projected and quantized into log polar coordinates to get feature vectors. These feature vectors made rotation invariant to small rotation angles. Then they are matched to find similar feature vectors by using either lexicographic sorting or counting bloom filters. Even a natural image may have several similar blocks. Hence, forging is verified only when there are a certain number of connected blocks within the same distance. This process reduces false positives making the technique more efficient. This method could detect forgeries involving blocks with rotations of up to 10 degrees and a scaling of 10%. Their algorithm is also robust to JPEG compression.

Shao et al. (2012), proposed an algorithm which is computationally a complex copy - move forgery detection algorithm. These algorithm dependents on circular window expansion and phase correlation. The image is scanned by a circular window which is then expanded into a normalized rectangular block using bi-linear interpolation. Discrete Fourier Transform (DFT) is calculated for these expanded blocks to obtain the phase correlation matrix. Enhanced peak values reflect the similarity in regions. A band limitation procedure is applied to the DFT in order to remove the high frequency components as they do not make any constructive contribution towards the calculation of peak values. This method also identifies copied-rotated - moved regions in the image. This method proves to be accurate in forgery detection even after the forged region has undergone rotation, blurring, JPEG compression, and variations in luminance. The drawbacks of this method are represented in the fact that it is not computationally fast and is also not scale invariant.

2.2.5 Wavelet-Based Methods:

Muhammad, et al. (2012)[1], proposed a method using un-decimated Dyadic Wavelet Transform (DyWT), which was chosen because of its property of shifting invariance and thus being more suitable than Discrete Wavelet Transform (DWT) for data analysis. First, the input image is decomposed into approximation (LL1) and detail (HH1) sub-bands. Then LL1 and HH1 sub-bands are divided into overlapping blocks and the similarity between those blocks is calculated. The main idea is that there should be high similarity between the copied and moved blocks from the LL1 sub-band, but low similarities among those from HH1 sub-band because of noise inconsistency in the moved block. Therefore, pairs of blocks are sorted based on high similarity using LL1 sub-band and high dissimilarity using HH1 sub-band. Using thresholding, these matched pairs are obtained from the sorted list as copied and moved blocks. Experimental results show the effectiveness of this proposed method over the other competitive methods using DWT, in addition to reduced false positives. However, it is tested only for small rotation angle and good quality images Gomase and Wankhade (2014), proposed a technique in which DWT is applied to find out the intensity local changes in the image. Then a median filter is applied to remove the noise. . For detection process, the image is divided into overlapping blocks, store them in a matrix and then sort the matrix. Finally, the matrix is used to locate the copy-move regions through pixel matching. This method is useful when images are preprocessed, but only shifting of copied regions must be taken into account.

2.2.6 Singular Value Decomposition (SVD) based method:

SVD served to produce algebraic and geometric invariant and feature vectors .Experimental results demonstrate the validity of the proposed approach to tampered images undergone some attacks like Gaussian blur filtering, Gaussian white noise contamination, lossy JPEG compression, etc. Singular value decomposition (SVD) is a matrix factorization and provides a new way for extracting algebraic and geometric features from an image. SVD has been used in many fields such as data compression, signal processing and pattern analysis. Among excellent references on SVD are (Kang 2008[8]).

2.2.7 Moment-Based Methods:

Mohamadian (2013) presented a method that was based on mixed moments. First, The Gaussian pyramid transform was used to extract the low-frequency information from the image which was then divided into overlapping blocks; Secondly, the eigenvector of block composed by the exponenti-fourier moments and histogram moments is lexicographically sorted; thirdly, tampered region was positioned precisely and quickly according to the Euclidean distance and space distance. Experimental results show that this method can successfully detect the forged part with translation, rotation, scaling and mixed operation tamper when the image is changed by brightness variation and contrast adjustment. But the qualitative evaluation, rotation angle and scaling factor are not specified.

2.2.8 SIFT point-Based Methods:

Hsu and Wang (2012) proposed a new forgery detection system based on Gabor filter. The Gabor filter with different scaling factors, rotation angles and frequencies are considered to generate the Gabor feature representation of an image. For comparing two images, their Gabor features are applied to find if there is any similarity between them. To reduce the processing time and detect the small copy and move area, an image is divided into small blocks. In each block, a new descriptor is extracted and key points from the Gabor feature of the block image are defined. H. Huang (2008), proposed a copy-move forgery detecting method based on local invariant feature matching. This method locates copied and pasted regions by matching feature points. It detects feature points and extracts local features using SIFT algorithm. Matching local features is based on k-d tree and Best-Bin-First method. The computational complexity of the proposed method is similar to the existing block-matching methods, but it has better locating accuracy. Experiments show that this method cannot only detect copy-move forgery, but it also detects copy regions with geometrical deformation and some post-operations such as JPEG compression and Gaussian blurring.

Amerini et al. (2013)[4], employed a Scale Invariant Feature Transform (SIFT) for feature extraction, combined with localization based on the J-Linkage algorithm for detecting tampering. SIFT features are extracted for the image. Afterwards the feature vectors are matched using g2NN algorithm. The Coordinates of the matched vectors are considered likely candidates for clustering, which are performed using J-linkage algorithm. The result of clustering reveals the copied regions. Because the method adopts SIFT features, it is capable of detecting forgeries involving scaling and rotation. The method is successful in detecting multiple duplications and is also able to localize tampered regions with a high degree of precision.

Chapter 3

DISCRETE COSINE TRANSFORM
AN OVERVIEW

3.1 Introduction

The Discrete Cosine Transform (DCT) was first proposed by Ahmed et al. (1974)[16], and it has been more and more important in recent years. DCT has been widely used in signal processing of image data, especially in coding for compression, especially in loss compression, for its near-optimal performance. Because of the wide-spread use of DCT's, research into fast algorithms for their implementation has been rather active ,and also, since the DCT is computation intensive, the development of high speed hardware and real-time DCT processor design have been object of research .

Discrete cosine transform (DCT) is widely used in image processing, especially for compression. Some of the applications of two-dimensional DCT involve still image compression and compression of individual video frames, while multidimensional DCT is mostly used for compression of video streams. DCT is also useful for transferring multidimensional data to frequency domain, where different operations, like spread-spectrum, data compression, data watermarking, can be performed in easier and more efficient manner. A number of papers discussing DCT algorithms is available in the literature that signifies its importance and application.

Hardware implementation of parallel DCT transform is possible, that would give higher throughput than software solutions. Special purpose DCT hardware decreases the computational load from the processor and therefore improves the performance of complete multimedia system. The throughput is directly influencing the quality of experience of multimedia content. Another important factor that influences the quality is the finite register length effect that affects the accuracy of the forward-inverse transformation process.

Hence, the motivation for investigating hardware specific DCT algorithms is clear. As 2-D DCT algorithms are the most typical for image compression, the main focus of this chapter will be on the efficient hardware implementations of 2-D DCT based compression by decreasing the number of computations, increasing the accuracy of reconstruction, and reducing the chip area. This in return reduces the power consumption of the compression technique. As the number of applications that require higher-dimensional DCT algorithms are growing, a special attention will be paid to the algorithms that are easily extensible to higher dimensional cases. The JPEG standard has been around since the late 1980's and has been an effective first solution to the standardization of image compression. Although JPEG has some very useful strategies for DCT quantization and compression, it was only developed for low compressions. The 8×8 DCT block size was chosen for speed (which is less of an issue now, with the advent of faster processors) not for performance.

Like other transforms, the Discrete Cosine Transform (DCT) attempts to decor relate the image data. After decor relation each transform coefficient can be encoded independently without losing compression efficiency. This section describes the DCT and some of its important properties

3.2 The One-Dimensional DCT

The most common DCT definition of a 1-D sequence of length N is

$$\mathrm{C(u)}=\alpha(u)\sum_{x=0}^{N-1} f(x)\cos[\frac{\pi(2\mathrm{x}+1)\mathrm{u}}{2N}] \qquad (1)$$

for u = 0,1,2,…., N — 1. Similarly, the inverse transformation is defined as

$$f(x)=\sum_{u=0}^{N-1}\alpha(u)c(u)\cos[\frac{\pi(2x+1)}{2N}] \qquad (2)$$

for x = 0,1,2..,, N — 1. In both equations (1) and (2) *α(u)* is defined as

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & for \quad u = 0 \\ \sqrt{\frac{2}{N}} & for \quad u \neq 0. \end{cases} \qquad (3)$$

It is clear from (1) that for $u = 0$, c(u=0)=$\sqrt{\frac{1}{N}}\sum_{x=0}^{N-1} f(x)$ Thus, the first transform coefficient is the average value of the sample sequence. In literature, this value is referred to as the *DC Coefficient.* All other transform coefficients are called the *AC Coefficients.*

To fix ideas, ignore the *f(x)* and *α(u)* component in (1). The plot of $\sum_{x=0}^{N-1} \cos[\frac{\pi(2x+1)u}{2N}]$ for N = 8 and varying values of u is shown in Figure 1. In accordance with our previous observation, the first the top-left waveform *(u* = 0) renders a constant (DC) value, whereas, all other waveforms *(u* = 1,2,...,) give waveforms at progressively increasing frequencies . These waveforms are called the *cosine basis function.*

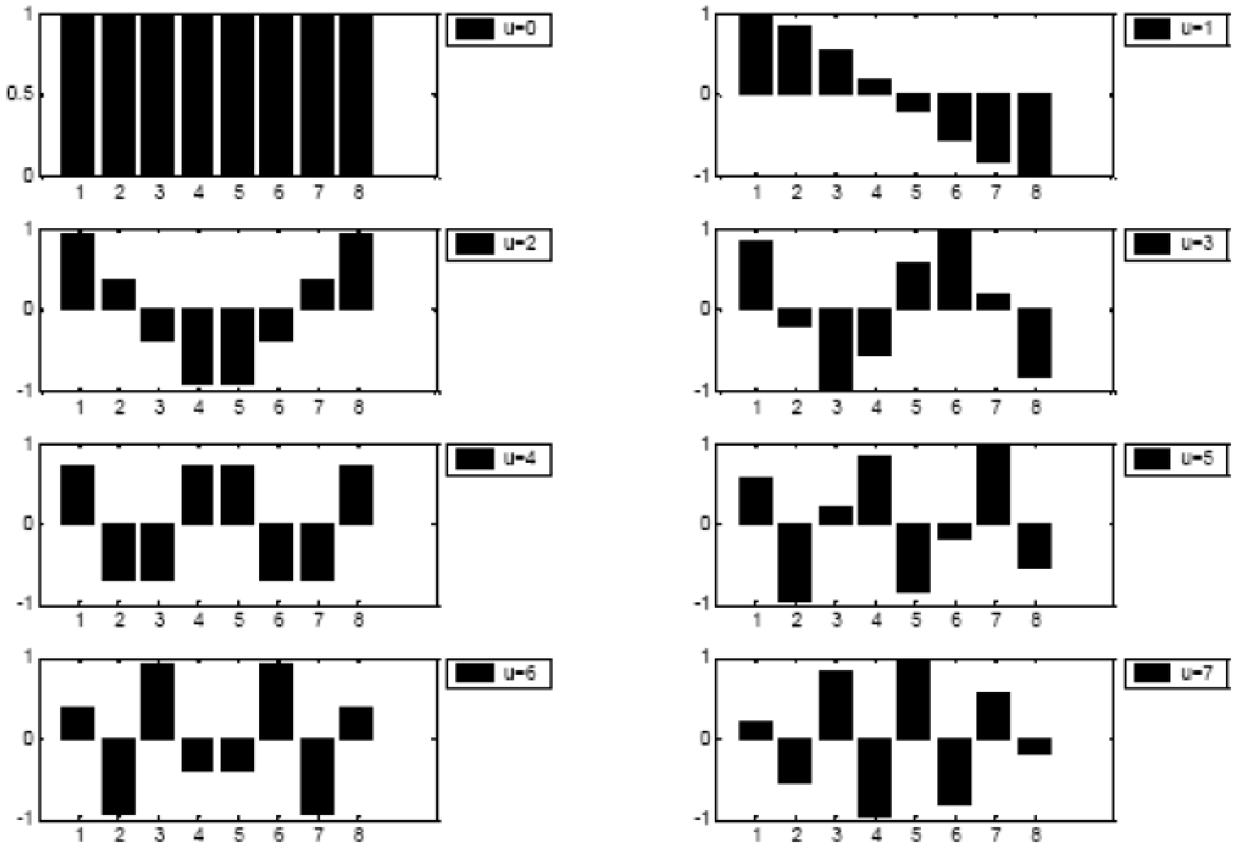


Figure (9).One Dimensions cosine functions.

If the input sequence has more than *N* sample points then it can be divided into sub-sequences of length *N* and DCT can be applied to these chunks independently. Here, a very important point to note is that in each such computation the values of the basis function points will not change. Only the values of f(x) will change in each sub-sequence. This is a very important property, since it shows that the basic functions can be pre-computed offline and then multiplied with the sub-sequences. This reduces the number of mathematical operations (i.e., multiplications and additions) thereby rendering computation efficiency.

3.3 The Two-Dimensional DCT:

The objective of this document is to study the efficacy of DCT on images. This necessitates the extension of ideas presented in the last section to a two-dimensional space. The 2-D DCT is a direct extension of the 1-D case and is given by

$$c(u,v) = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\,\frac{(cos2x+1)u\pi}{2N}\,\frac{(cos2y+1)v\pi}{2N} \qquad (4)$$

for u,v = 0,1,2,….,N — 1 and *α(u)* and *α(v)* are defined in (3). The inverse transform is define as.

$$f(x,y) = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} c(u,v)\alpha(u)\alpha(v)\cos[\frac{(2x+1)u\pi}{2N}]\cos[\frac{(cos2y+1)v\pi}{2N}] \qquad (5)$$

for x,y = 0,1,2,…N —*1* . The 2-D basis functions can be generated by multiplying the horizontally oriented 1-D basis functions (shown in Figure 10) with vertically oriented set of the same functions [13]. The basis functions for N = 8 are shown in. Again, it can be noted that the basis functions exhibit a progressive increase in frequency both in the vertical and horizontal direction. The top left basis function of results from multiplication of the DC component in Figure 11 with its transpose. Hence, this function assumes a constant value and is referred to as the DC coefficient.

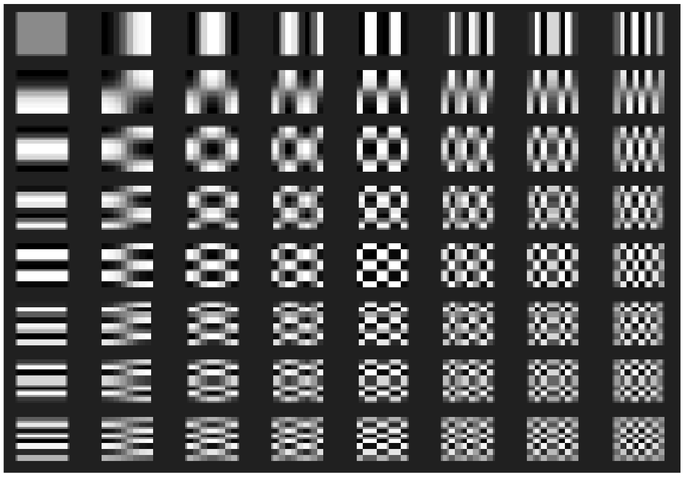


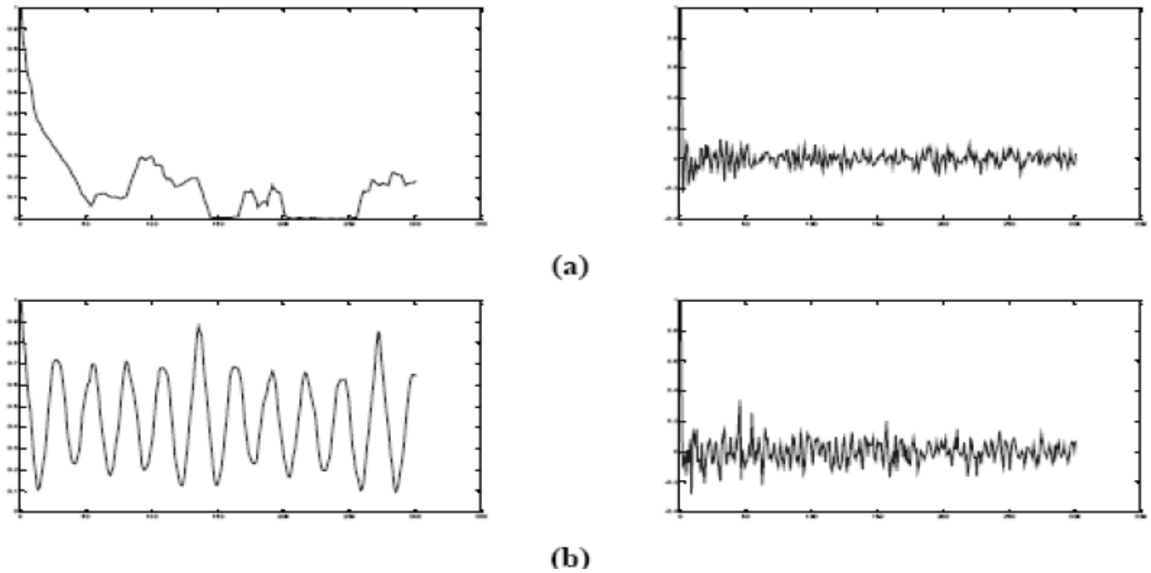
Figure (10). Two dimensional DCT basis functions (N = 8). Neutral gray represents zero, white represents positive amplitudes, and black represents negative amplitude.

3.4 Properties of DCT

Discussions in the preceding sections have developed a mathematical foundation for DCT. However, the intuitive insight into its image processing application has not been presented. This section outlines (with examples) some properties of the DCT which are of particular value to image processing applications.

3.4.1 Decor relation

As discussed previously, the principle advantage of image transformation is the removal of redundancy between neighboring pixels. This leads to uncorrelated transform coefficients which can be encoded independently. The normalized autocorrelation of the images before and after DCT is shown in Figure 3. Clearly, the amplitude of the autocorrelation after the DCT operation is very small at all lags. Hence, it can be inferred that DCT exhibits excellent decor relation properties.


(a)

(b)

Figure.(11) (a) Normalized autocorrelation of uncorrelated image before and after DCT; (b) Normalized autocorrelation of correlated image before and after DCT.

3.4.2 Symmetry

Another look at the row and column operations in Equation 6 reveals that these operations are functionally identical. Such a transformation is called a *symmetric transformation.* A separable and symmetric transform can be expressed in the form .

$$T = AfA \qquad (6)$$

Where A is an N XN symmetric transformation matrix with entries $\alpha(i,j) = \alpha(j)\sum_{u=0}^{N-1}\cos[\frac{\pi(2j+1)N}{2N}]$ and f is the *NXN* image matrix.

This is an extremely useful property since it implies that the transformation matrix can be pre-computed offline and then applied to the image thereby providing orders of magnitude improvement in computation efficiency.

3.4.3 Orthogonality

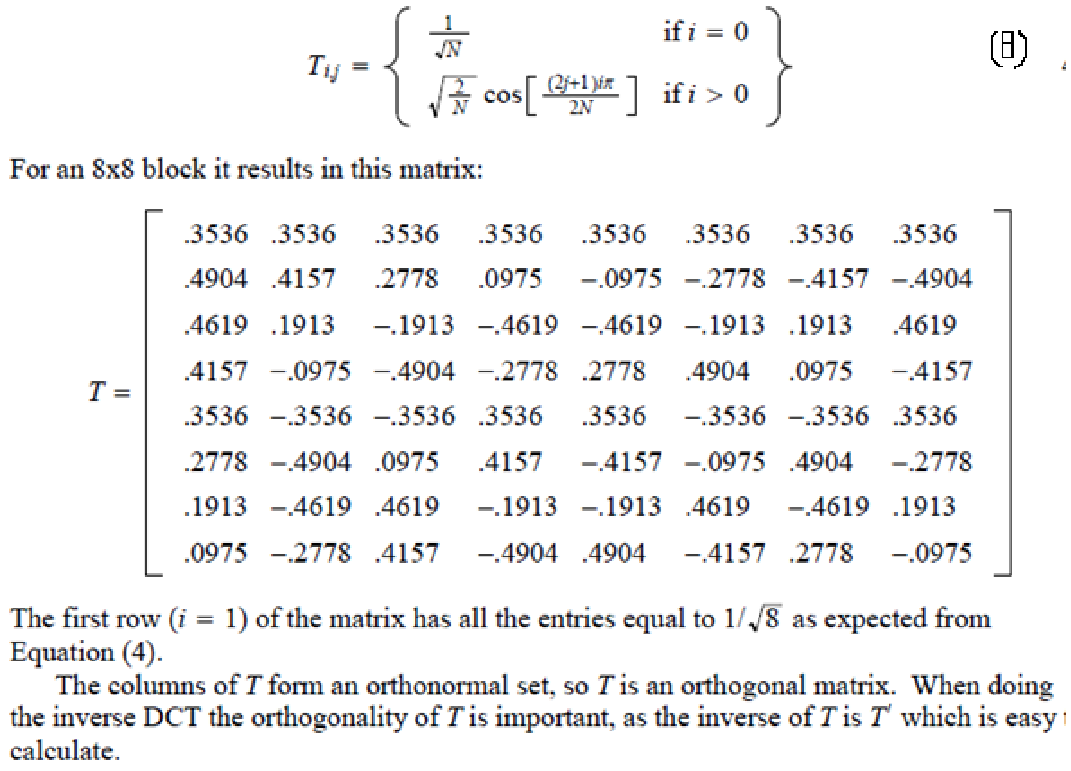
In order to extend ideas presented in the preceding section, let us denote the inverse transformation of (6) as

$$f = A^{-1} T A^{-1}. \tag{7}$$

As discussed previously, DCT basis functions are orthogonal . Thus, the inverse transformation matrix of *A* is equal to its transpose i.e. $A^{-1} = A^{T}$. Therefore, and in addition to its decorrelation characteristics, this property renders some reduction in the pre-computation complexity.

3.5 Discrete Cosine Transform Matrix

To transform equation (1) into a matrix form, we can use the equation (8).



$$T_{i,j} = \begin{cases} \frac{1}{\sqrt{N}} & \text{if } i = 0 \\ \sqrt{\frac{2}{N}} \cos\left[\frac{(2j+1)i\pi}{2N} \right] & \text{if } i > 0 \end{cases} \tag{8}$$

For an 8x8 block it results in this matrix:

$$T = \begin{bmatrix} .3536 & .3536 & .3536 & .3536 & .3536 & .3536 & .3536 & .3536 \\ .4904 & .4157 & .2778 & .0975 & -.0975 & -.2778 & -.4157 & -.4904 \\ .4619 & .1913 & -.1913 & -.4619 & -.4619 & -.1913 & .1913 & .4619 \\ .4157 & -.0975 & -.4904 & -.2778 & .2778 & .4904 & .0975 & -.4157 \\ .3536 & -.3536 & -.3536 & .3536 & .3536 & -.3536 & -.3536 & .3536 \\ .2778 & -.4904 & .0975 & .4157 & -.4157 & -.0975 & .4904 & -.2778 \\ .1913 & -.4619 & .4619 & -.1913 & -.1913 & .4619 & -.4619 & .1913 \\ .0975 & -.2778 & .4157 & -.4904 & .4904 & -.4157 & .2778 & -.0975 \end{bmatrix}$$

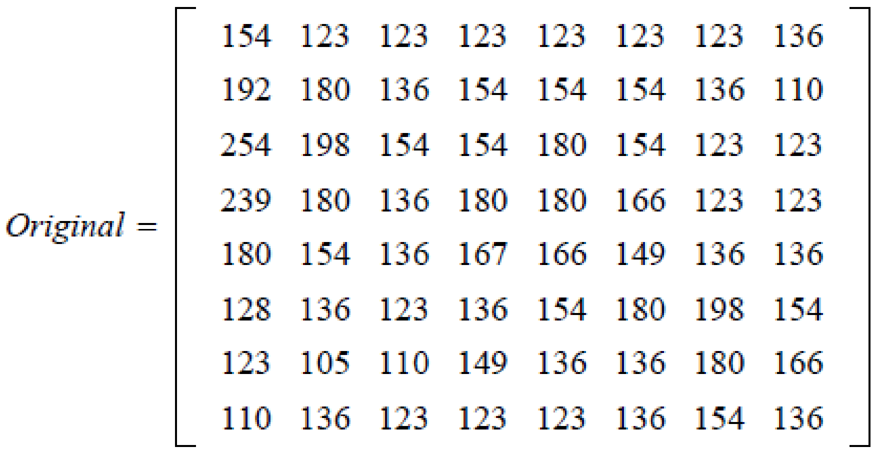
The first row ($i = 1$) of the matrix has all the entries equal to $1/\sqrt{8}$ as expected from Equation (4).

The columns of *T* form an orthonormal set, so *T* is an orthogonal matrix. When doing the inverse DCT the orthogonality of *T* is important, as the inverse of *T* is T' which is easy t calculate.

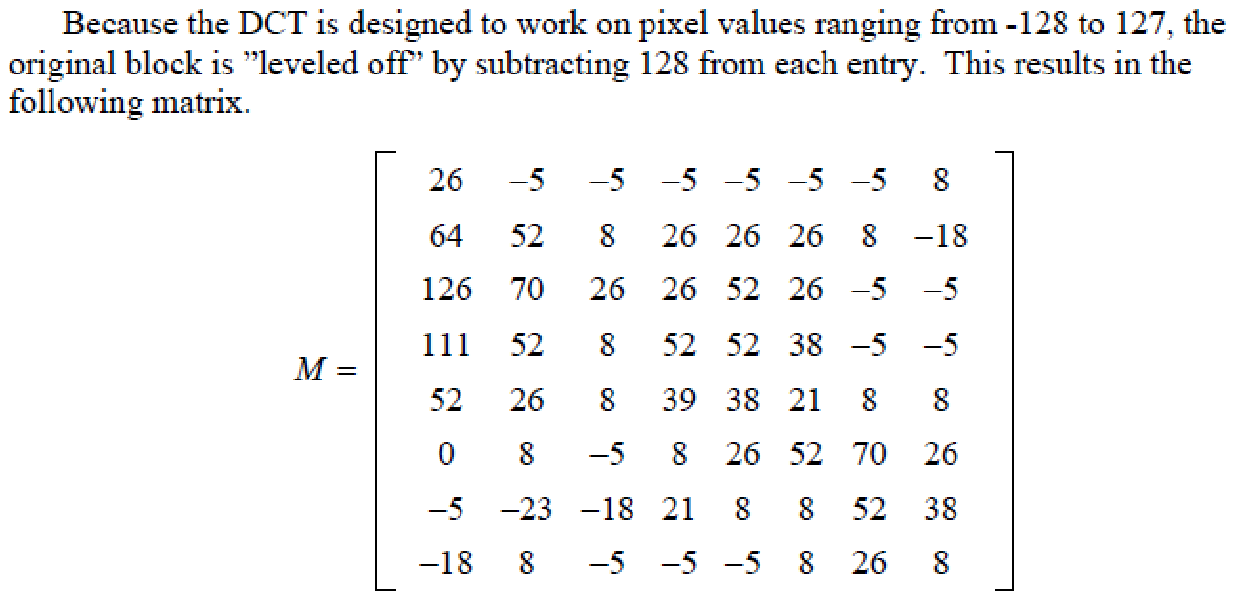
3.6Applying the Discrete Cosine Transform on an 8x8 block

Before starting we should make note that in a greyscale picture the pixel values lie between 0 to 255. In that 255 symbolize plain white and 0 represents plain black. Thus these 256 shades of grey accurately represent a photo or illustrations.

Since an image may contain a number of overlapping blocks, below is the description of how it works on a single 8x8 matrix. It is applied on the other blocks in a similar manner.



$$Original = \begin{bmatrix} 154 & 123 & 123 & 123 & 123 & 123 & 123 & 136 \\ 192 & 180 & 136 & 154 & 154 & 154 & 136 & 110 \\ 254 & 198 & 154 & 154 & 180 & 154 & 123 & 123 \\ 239 & 180 & 136 & 180 & 180 & 166 & 123 & 123 \\ 180 & 154 & 136 & 167 & 166 & 149 & 136 & 136 \\ 128 & 136 & 123 & 136 & 154 & 180 & 198 & 154 \\ 123 & 105 & 110 & 149 & 136 & 136 & 180 & 166 \\ 110 & 136 & 123 & 123 & 123 & 136 & 154 & 136 \end{bmatrix}$$



Because the DCT is designed to work on pixel values ranging from -128 to 127, the original block is ”leveled off” by subtracting 128 from each entry. This results in the following matrix.

$$M = \begin{bmatrix} 26 & -5 & -5 & -5 & -5 & -5 & -5 & 8 \\ 64 & 52 & 8 & 26 & 26 & 26 & 8 & -18 \\ 126 & 70 & 26 & 26 & 52 & 26 & -5 & -5 \\ 111 & 52 & 8 & 52 & 52 & 38 & -5 & -5 \\ 52 & 26 & 8 & 39 & 38 & 21 & 8 & 8 \\ 0 & 8 & -5 & 8 & 26 & 52 & 70 & 26 \\ -5 & -23 & -18 & 21 & 8 & 8 & 52 & 38 \\ -18 & 8 & -5 & -5 & -5 & 8 & 26 & 8 \end{bmatrix}$$

We can now perform DCT by multiplying matrices

$$D = TMT` \qquad (9)$$

The above equation (9) multiplies the matrix M with matrix T and its transpose T’. When M is multiplied by matrix T on the left, it transforms the rows of M and when the product is multiplied by T’ on the right, it transforms the columns as well.

$$D = \begin{bmatrix} 162.3 & 40.6 & 20.0 & 72.3 & 30.3 & 12.5 & -19.7 & -11.5 \\ 30.5 & 108.4 & 10.5 & 32.3 & 27.7 & -15.5 & 18.4 & -2.0 \\ -94.1 & -60.1 & 12.3 & -43.4 & -31.3 & 6.1 & -3.3 & 7.1 \\ -38.6 & -83.4 & -5.4 & -22.2 & -13.5 & 15.5 & -1.3 & 3.5 \\ -31.3 & 17.9 & -5.5 & -12.4 & 14.3 & -6.0 & 11.5 & -6.0 \\ -0.9 & -11.8 & 12.8 & 0.2 & 28.1 & 12.6 & 8.4 & 2.9 \\ 4.6 & -2.4 & 12.2 & 6.6 & -18.7 & -12.8 & 7.7 & 12.0 \\ -10.0 & 11.2 & 7.8 & -16.3 & 21.5 & 0.0 & 5.9 & 10.7 \end{bmatrix}$$

The derived matrix contains 64 DCT coefficients. The coefficient Coo corresponds to lowest frequency in the actual block. When we traverse in the D matrix in zig-zag order, the coefficients corresponds to higher frequencies of the block, thus C_{77} corresponds to highest frequency. It should be noted here that human eye is most sensitive to low frequency

3.7 Quantization

Now the derived block containing DCT coefficients can be compressed by Quantization. One of the finest features in the DCT is that we can obtain varying level of compression and image quality by choosing from a range of quantization matrix. We obtain the Quantization by dividing transformed image DCT matrix by the quantization matrix used. Values of the resultant matrix are then rounded off. The quantized coefficient is defined in (10), and the reverse process can be achieved by the (11).

These quantization matrixes are derived from a standard quantization matrix and the standard quantization matrix is a result of various subjective experiments on human visual system. To generate varying quantization matrix, we use the scalar multiple of the standard quantization matrix.

$$f(u,v)Quantization = round[\frac{f(u,v)}{Q(u,v)}] \qquad (10)$$

$$f(u,v)deQ = f(u,v)Quantization \times \quad Q \quad (u, \quad v)$$
(11)

Quantization aims at reducing most of the less important high frequency DCT coefficients to Zero, the more zeros the better the image will compress. Lower frequencies are used to reconstruct the image because human eye is more sensitive to them and higher frequencies are Discarded. Matrix Q_y and Q_c defines the Q matrix for luminance and chrominance components.

$$Q_Y = \begin{pmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{pmatrix}$$

$$Q_C = \begin{pmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{pmatrix}$$

After quantization, the "zig-zag" sequence orders all of the quantized coefficients as shown in Figure(12). In the "zig-zag" sequence, firstly it encodes the coefficients with lower frequencies(typically with higher values) and then the higher frequencies (typically zero or almost zero). The result is an extended sequence of similar data bytes, permitting efficient entropy encoding.

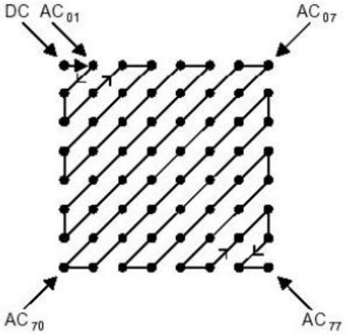


Figure (12). Zigzag Sequencing

3.8Nature of Discrete Cosine Transform

The inherent nature of DCT is such that when we apply DCT on image the energy focuses only on the coefficients which correspond to low frequency, which means that all elements are not equally important but only the low frequency coefficients play a major role.

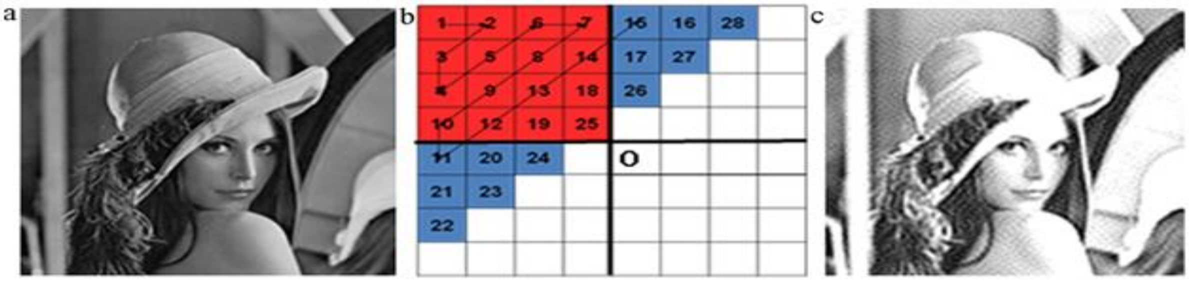


Fig. [a) the Lena image (b) Zigzag order scanning (c) the reconstruction image of Lena by using 1/4 DCT coefficients.

To prove the above statement, we made an illustration by using the Lena image of size 256x256 pixels. Fig (a) is the Lena image. We applied the discrete cosine transform to Fig (a) and extract a matrix of DCT coefficients. Subsequently, low frequency DCT coefficients are extracted in zig-zag sequence as shown in Fig (b) where red area is the part that corresponds to low frequency, which accounts to 1/4 of the entire DCT coefficients. Now with only these DCT coefficients we reconstruct the image using inverse of discrete cosine transform and generate Fig(c). Thus, through the analysis of the given image we came to know that with the help of only 1/4th of DCT coefficients, one can retrieve the information without any significant loss.

Chapter 4

The Proposed Method

We propose an efficient and effective methodology to detect the cloning forgery within same image using the block based and radix sort approach which is less computation complex than our basic method. The following figure illustrates the architecture of our approach.

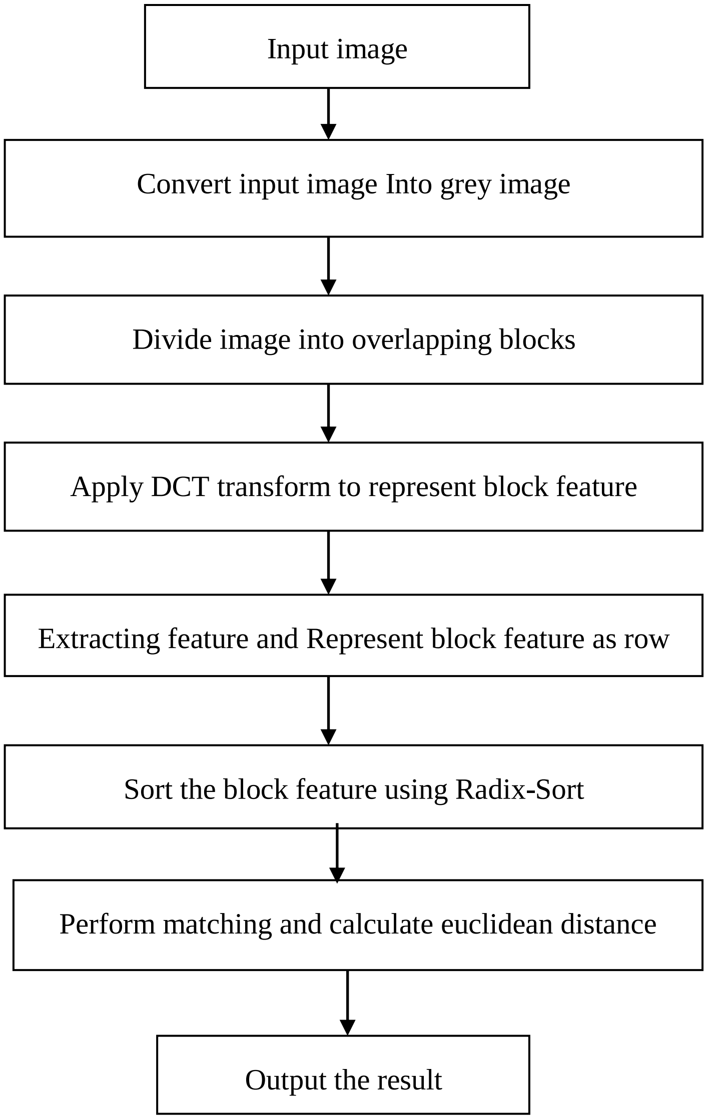

Input image
Convert input image Into grey image
Divide image into overlapping blocks
Apply DCT transform to represent block feature
Extracting feature and Represent block feature as row
Sort the block feature using Radix-Sort
Perform matching and calculate euclidean distance
Output the result

Figure (14) Flow diagram of praposed method

In a Copy-Move forgery, a part of the image itself is copied and pasted into another part of the Same image. This is usually performed with the intention to make an object "disappear" from the image by covering it with a segment copied from another part of the image. Textured areas, such as grass, foliage, gravel, or fabric with irregular patterns, are ideal for this purpose because the Copied areas will likely blend with the background and the human eye cannot easily discern any Suspicious artifacts. Because the copied parts come from the same image, its noise component, Color palette, dynamic range, and most other important properties will be *compatible* with the rest of the image and thus will not be detectable using methods that look for incompatibilities in Statistical measures in different parts of the image. To make the forgery even harder to detect, one can use the feathered crop or the retouch tool to further mask any traces of the copied-and-moved segments.

Steps

Going by the discussion above, the framework is given as below:

(1) Pre-processing of Input Image

Applying Rescaling, and RGB to grey conversion.

(2) Dividing the image into Blocks

(3) Apply DCT Transform of each block

(4) Extracting feature and Represent block feature as row

(5) Radix- Sorting of Feature Vector Matrix

(6) Detection, Decision and display forged part.

Further we shall see each step explained in detail:

4.1 Pre-processing of Input Image:

The aim of pre-processing is the improvement of image data that suppresses unwanted distortions or enhances some image features important for further detection. The given image is converted into grey-scale (color conversion) when applicable (except for algorithms that require color channels). Other pre-processing techniques includes, dimension reduction, image resizing, low-pass filtering etc. In both block-based and key-point based methods necessary pre-processing can be applied.

The second sub-step is to convert the image so obtained into its grayscale equivalent. For this, one can either utilize the formula that is “I = 0.228R + 0.587G + 0.114B” or alternatively matlab has a dedicated function for this purpose.

The third sub-step is to resize the image into the dimensions of our choice. Conventionally we chose to resize it into a square matrix. However the scale of resizing depends upon various factors such as processing and memory capabilities of the system. The algorithm works better without any downscaling.

4.2 dividing the image into overlapping blocks:

The image so obtained after step 1 is divided into overlapping blocks of size *bxb* pixels in such a way that it replicates a sliding window of size *bxb* pixel. At each iteration, the window slides to its right by 1 pixel and when it cannot move any further to its right, the window slides down by 1pixel and begin from leftmost end. Thus there is a difference of only one row or column between two adjacent blocks.

Suppose we have an image of size M x N and we divide it into overlapping blocks of size b. Then number of blocks is NB = (M – b + 1) x (N – b + 1)

Bij is used to denote each block, where i and j indicates the starting location of B_{ij}.

$$B_{ij}(x,y) = f(x+j, y+i),$$

where $x,\ y \in \{0,\ \ldots,\ B-1\}$, $i \in \{1,\ \ldots,\ M-B+1\}$, and $j \in \{1,\ \ldots,$ $N-B+1\}$

Concurrently we fill a matrix B of size (NB x 2) where row number indicates the number of block and it corresponding column elements indicates the starting position of the block that is, i and j respectively. This matrix shall come into use in the later stages after lexicographic sorting of the feature vector matrix, when we will be needing to find out the starting locations of the similar blocks in order to take a decision about the forgery.

4.3 Apply DCT Transform of each block:

For block-based methods, feature vectors are extracted for each block. While for key-point based methods, feature vectors are computed only key-points in the image such as regions with entropy etc. Onto every block, DCT is applied so as to generate the quantized discrete coefficients transform matrix of size b x b, which is same as that of the block’s size. Assuming our block size to be 8x8, the coefficients matrix that we get is also of the same size and can be used to represent the corresponding block. The nature of DCT is such that it’s energy gets focused on lower frequency coefficients which mean that each component is not equally significant and the coefficients with low frequency play the relatively vital part. This is illustrated in Chapter 2 where DCT is explained.

Another key aspect to the JPEG algorithm is the decision to perform these operations on 8x8 blocks. These dimensions may seem somewhat arbitrary, and that is in part because they are. However, there are also a few reasons to support this decision. First, if the patch sizes were larger, then it is possible that the image would have larger color gradients between these blocks. It's helpful to think about the masking step as basically averaging together the values in the block before it's returned back to the viewer. If there are finer differences in an image between smaller blocks of pixels, this process won't capture those differences well, which will make your compression's result worse. However, this would seem to indicate that an even smaller block size, such as 4x4 or 2x2 should be used. This is not the case because that would make the compression more arduous to perform. For each block, you have to take the DCT, multiply by the mask, and take the inverse DCT. With more blocks, this process would take longer. As we'll see with experimental results later, it is also harder to compress an image to the same accuracy while achieving smaller _le sizes when you're using smaller blocks. Therefore, the 8x8 block size approach has emerged as the dominant way to split up the image.

4.4 extracting F ature and Represent block f ature as row:

In this step a set of sensitive features are extracted for each part of the image. These features are mainly used to distinguish each part from all others. DCT methods are used for feature extraction: After extraction these features are stored in a feature vector. One of the desirable characteristics of selected features and constructed feature vector should be with low dimension, which will reduce the computational complexity.

Input image is divided in Square blocks of dimension B×B. A square size window slides from top left corner to bottom right corner. DCT coefficients are calculated corresponding to each block also known as feature vector.

4.5 Radix- Sorting of Feature Vector Matrix:

Radix-sort is a non-comparative integer sorting algorithm that sorts value with integer keys by grouping keys by individual digits which share the same significant position and value. A positional notation is required, but because integers can represent strings of characters (e.g., names or dates) and specially formatted floating point numbers, radix sort is not limited to integers.

A feature vector matrix of size *NB* x *4* is constructed in which each row F_i correspond to a block B_i and each row contains 4 feature vectors corresponding to the block B_i. Here *NB* is the no. of blocks i.e. (M – b + 1)(N – b + 1)

Radix-sort (A, d)

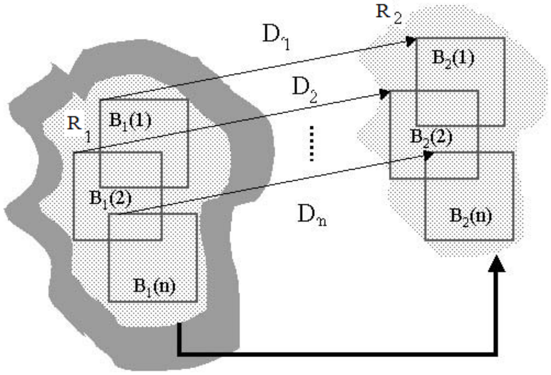
```
for $i$ ← 1 to d do
    use a stable sort to sort $A$ on digit $i$
    // counting sort will do the job
```

$$\text{FVM or F} = \begin{bmatrix} V_1 \\ \vdots \\ V_{(M-B+1)(N-B+1)} \end{bmatrix}$$

From above we $V1, V2\ldots\ldots V_{(M-B+2)(N-B+1)}$ denote sorted list of the feature vectors of blocks B1, B2,…, Bi, respectively.The running time depends on the stable used as an intermediate sorting algorithm. When each digits is in the range 1 to *k*, and *k* is not too large, COUNTING_SORT is the obvious choice. In case of counting sort, each pass over *n d*-digit numbers takes O(*n* + *k*) time. There are *d* passes, so the total time for Radix sort is $\Theta(n+k)$ time. There are *d* passes, so the total time for Radix sort is $\Theta(dn+kd)$. When *d* is constant and $k = \Theta(n)$, the Radix sort runs in linear time.

4.6 Detection, Optimization, Decision and Display forged part:

The FVM obtained in the previous step contain the vectors of the blocks in such a way that the more similar two blocks are, nearer would be there feature vector in the FVM. Thus the vectors of same blocks would appear adjacent to each other in the FVM. However this is not enough since in an image there can be many blocks which are similar to each other. Secondly the blocks adjacent to each other in an image would naturally have similar feature vectors since only a row or column is different in two adjacent blocks. Thirdly in natural images of blue sky or desert or grass, the blocks tend to appear similar over a large area. Because we deliberately kept our block size smaller than any viable duplicated section, we make out that any duplicated region will have a number of blocks in it. The judgment about tampering can only be taken, if there exist a sufficient number of similar blocks that are equidistant from each other and that are connected as well. In other words each of these blocks are equal distant apart from there source. This can be evidently understood from fig. below.



Figure(15) A diagram which shows the copied regions and alike blocks in it.

From the diagram, it is clear that in case of cloned region the distance between similar blocks is identical and the blocks are linked to each other.

Now the each distance are optimized through genetic algorithm. When we perform the implementation of proposed method than we take the absolute integer values of distance calculation but output is if decimal and we eliminate the decimal values but its not provide a accurate result. for accurate and precious result we apply the genetic algorithm on distance. as follows

Genetic algorithms are a part of evolutionary computing, which is a rapidly growing area of artificial intelligence. algorithm of genetic describe given below:

1. **[Start]** Generate random population of *n* chromosomes (suitable solutions for the problem)
2. **[Fitness]** Evaluate the fitness *f(x)* of each chromosome *x* in the population
3. **[New population]** Create a new population by repeating following steps until the new population is complete
    1. **[Selection]** Select two parent chromosomes from a population according to their fitness (the better fitness, the bigger chance to be selected)
    2. **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
    3. **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
    4. **[Accepting]** Place new offspring in a new population
4. **[Replace]** Use new generated population for a further run of algorithm
5. **[Test]** If the end condition is satisfied, **stop**, and return the best solution in current population
6. **[Loop]** Go to step **2**

Encoding of a Chromosome

The chromosome should in some way contain information about solution which it represents. The most used way of encoding is a binary string. The chromosome then could look like this:

Chromosome 1 1101100100110110

Chromosome 2 1101111000011110

Each chromosome has one binary string. Each bit in this string can represent some characteristic of the solution. Or the whole string can represent a number - this has been used in the basic GA.

Crossover

After we have decided what encoding we will use, we can make a step to crossover. Crossover selects genes from parent chromosomes and creates a new offspring. The simplest way how to do this is to choose randomly some crossover point and everything before this point copy from a first parent and then everything after a crossover point copy from the second parent.

Crossover can then look like this (is the crossover point):

Chromosome 1 11011 | 00100110110

Chromosome 2 11011 | 11000011110

Offspring 1 11011 | 11000011110

Offspring 2 11011 | 00100110110

Mutation

After a crossover is performed, mutation take place. This is to prevent falling all solutions in population into a local optimum of solved problem. Mutation changes randomly the new offspring. For binary encoding we can switch a few randomly chosen bits from 1 to 0 or from 0 to 1. Mutation can then be following:

Original offspring 1	1101111000011110
Original offspring 2	1101100100110110
Mutated offspring 1	1100111000011110
Mutated offspring 2	1101101100110110

The mutation depends on the encoding as well as the crossover.

Keeping the above factors in mind we need to have certain thresholds based on the type of image we are looking onto. These thresholds would help us determine if a match between the two feature vectors is an ideal match i.e.

i> The similarity threshold – which determines whether two feature vectors of blocks are similar to each other. In other words the normalized difference between two blocks should be less than similarity threshold.

ii> The distance threshold – Since many adjacent blocks would have similar feature vectors, they must not be included in our final result. Hence two similar blocks should be a certain distance apart in order.

Chapter 5

RESULT AND ANALYSIS

The method proposed in the previous section is implemented in the Matlab software. Several experiments are conducted on different images to gauze the efficacy of our approach. For the experiments, some were self doctored and a part of the image is copied on the other part to create a copy-move forgery. Besides, some tampered images made by researchers are also taken from the internet.

The experiments are designed carefully so that we can ascertain the behaviour of our proposed method. For this we take similar images of varying resolutions and sizes. Other than that we also used noisy and jpeg compressed image to gauze the detection ability and behaviour of the method. To measure the performance we observed four measures that are

5.1 Performance Measures

1. Accuracy of the detection: This means how precisely the algorithm is able to detect the forged region. One way of doing is to do the forgery by ourselves so that we know where exactly the duplicated region and source region lies in the image.
2. Effectiveness: This tells us about how well our method can detect when the forgeries are quite sophisticated or done over only a small portion image. To observe this, we used different images in which the duplicated regions were of varying sizes. In some there were multiple regions which were copied and pasted onto other region in the image.
3. Efficiency: Only detecting a forgery is not enough. A good method should be able to detect the forgery in reasonable amount of time. To measure this we used the timing clocks in our algorithm so that we can know the exact amount of time our method is taking.
4. Robustness: This performance measure tells us about the robustness or tolerability of a method. Sometimes forger uses a variety of techniques to make the detection difficult.

These techniques may be addition of noise or blur, compression, scaling, rotation and other similar methods. A robust method is one which is invariant to these.

5.2 The Parameters

We did our experiments on 32 bit AMD Radeon 2.50 GHz processor with 2GB memory. The code is executed using Matlab R2015a. For the sake of simplicity the images are converted to greyscale, since it is easier to handle. The test perform on 100 images which are selected from columbia dataset and internet images”

Before beginning with detection process, one needs to specify a number of parameters. The very first being the image size itself. The selection of image size depends majorly on the computational and memory capacities of the system on which detection is to be carried out. For our experiments, we took two sets of images. In the first set, we resized the image to a size of 128 x 128 pixels and in the second we choose a size of 256 x 256 pixels. The second important parameters is the block size which determines the overall number of blocks in picture and hence size of feature matrix. It should be noted here that the size of block must be less that size of duplicated region. We choose varying block size ranging from 8 to 32 for our experiments depending on the size of image and size of duplicated region. The third parameter is the size of the forgery that is the approximate size of the duplicated portion. For most of the experiment we preferred it to be 32 x 32 pixels or 64 x64 pixels depending on the size of the image.

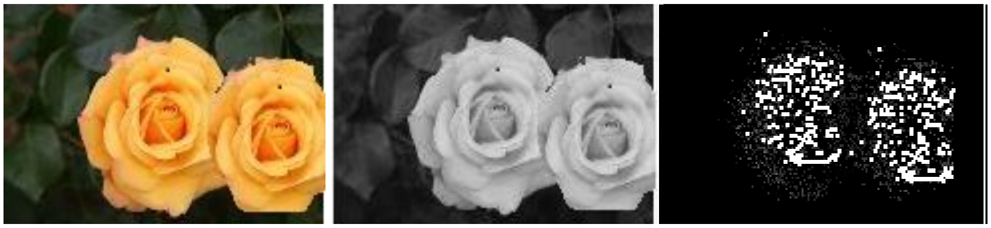
Other than the parameters described above there are certain thresholds that need to be specified. The first one is the N_D that is the minimum distance that should be there between two similar blocks so as to remove similar neighbouring blocks. For practical purposes we take it to be 30 pixels when block is 8 pixels and 60 pixels when block size is 32 pixels. The second threshold is S_{msr} that is the measure of similarity between two blocks. Normally the root mean square or the difference two blocks vectors is calculated and matched against $S_{msr.}$ For two blocks to be similar it should be less than $S_{msr.}$ The next threshold is shift threshold. When feature vector matrix is radix sorted, vector corresponding to similar features come near to each other. The shift threshold gives us a window size. For example if shift threshold is 2, then in the FVM we compare each block's vector to its 2 nearby vectors. In our experiment we mostly kept it 1. The last important threshold is N_c which tells us that a minimum of N_c similar and connected blocks must be there in order to take a decision about the forgery.

5.3 Experimental Results

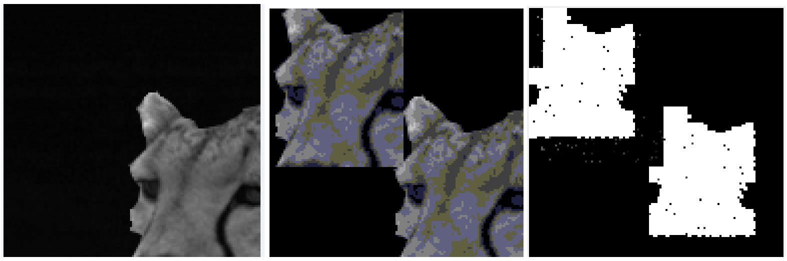
Several images are taken from various sources, some of them are self forged to accurately calculate the detection effectiveness and efficiency. Some other images are also taken from the internet on which other researchers have also worked so that comparison can be done between both. On each image, detection is carried out several times by changing its parameters such as image size and block size.

5.3.1 Efficiency

In figure(16) and figure(17) two images are shown for illustration purpose. The images are resized and detection is carried out by varying the block size as can be seen in the figure.


(a) (b) (c)

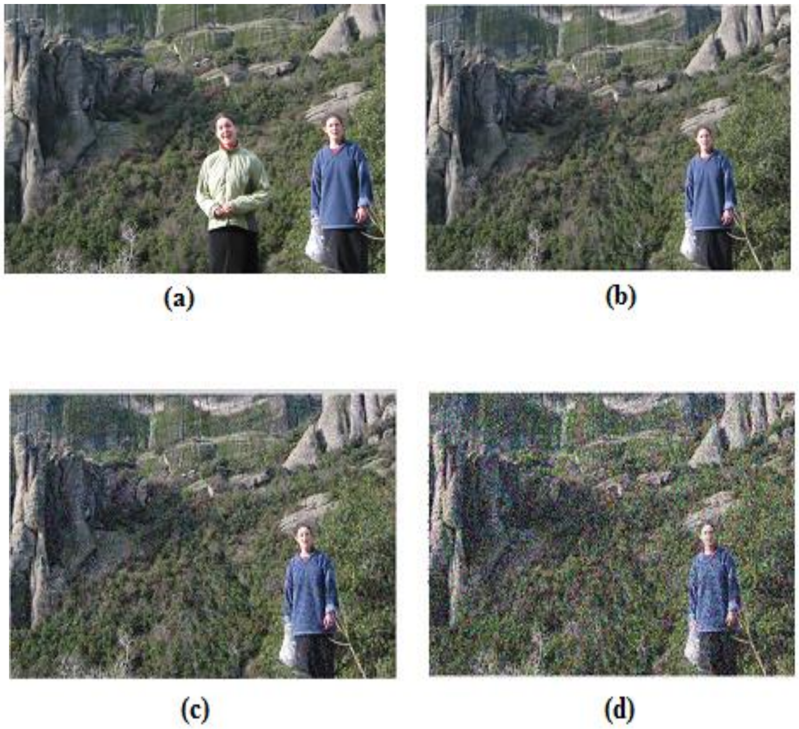
Figure(16): (a) Novel image taken from internet(b) Tampered image (c) The output image after applying the detection schema


(a) (b) (c)

Figure(17) (a) The original image of tiger taken from Database; (b) The portion of its mouth is taken and pasted onto upper left corner of the image; (c) The output image after applying the detection scheme.

5.3.2 Effectiveness

In the second set of our experiments, we tested our method on some images from Casia database [39] and some images which were tested by other authors. In this set of experiments, to test the effectiveness of our method and its robustness towards additive Gaussian noise, we first tested the method on a forged image without any noise. Our method was able to accurately detect the duplicated region in the region duplication map. Subsequently we added Gaussian noise with 0.01 variance; to check the robustness of the method. The proposed method was able to detect duplicated region although detection rate decreased and accuracy suffered a little bit. Further we raised the variance of noise to 0.05 to see if it could detect the regions. The method could detect the duplicated region but not in all cases and the accuracy decreased significantly. The illustration cab be seen in Figure(18).


(a)

(b)

(c)

(d)

Figure (18) (a)The original image; (b)forged image; (c) forged image with additive Gaussian noise (variance =0.01); (d) forged image with additive Gaussian noise (variance = .05)

5.4 Comparative Study and Result Analysis

The running time depends on the stable used as an intermediate sorting algorithm. When each digits is in the range 1 to k, and k is not too large, COUNTING_SORT is the obvious choice. In case of counting sort, each pass over n d-digit numbers takes $O(n + k)$ time. There are d passes, so the total time for Radix sort is $\Theta(n+k)$ time. There are d passes, so the total time for Radix sort is $\Theta(dn+kd)$. When d is constant and $k = \Theta(n)$, the Radix sort runs in linear time.

Now we compare the DCT with another existing method such as PCA,FMT. performance table is given below.

Manipulation Type	FMT	DCT	PCA
JPEG	20	40	50
Scaling	10^0	5^0	0^0
Rotation	10%	10%	0

Table 1. Performance Results

As expected, PCA values are changed when there is re-sampling but they are robust until JPEG 50.FMT work on 10% rotation, DCT work on 10% rotation and PCA not work on rotated forged image. When we consider on scaling of image than 10^0,$5^0$and 0^0,in FMT,DCT and PCA respectively.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The copy-move forgery detection is one of the emerging problems in the field of digital image forensics. Many techniques have been proposed to address this problem. One of the biggest issue these techniques had to deal with was, being able to detect the duplicated image regions without getting affected by the common image processing operations, e.g. compression, noise addition, rotation. The other challenge was computational time, which becomes important considering the large databases, these techniques word be used on. We presented an improved approach to detect copy- move forgery using DCT coefficients and then truncating the less important coefficients. We further used averaging to reduce the dimensionality of our feature vectors. We perform the radix sort and genetic algorithm for better performance which provide the satisfactory result.

6.2 Future Work

There are some challenges that need to be addressed. The first one being the computational complexity that is the method runs fine on images up to 256 x256 but the computational time increases significantly after that. Though in our approach we resized the image but to our preferred size, but it can be enhanced to work fine with larger images as well. The second challenge is method's robustness. Although the method runs on unmodified image, its detection rate and accuracy decreases when manipulations like rotation or scaling is applied on the duplicated region. It is the major challenge that is faced by all the copy move forgery detection algorithms and need to be further investigated.

Another issue is that our method is not able to detect the forgery when the copied area is sufficiently small as compared to image size. The block size that we kept in most of the experiments is 8x 8 and 16x 16 and it could typically discover the forgery when the duplicated region was larger than 32 x 32. Additionally, sometimes the accuracy of the detection decreases when the background of the image and duplicated region matches significantly, for example in case of uniform blue sky or grass.

REFERENCES

[1] Al-Hammadi M. (2013), Copy Move Forgery Detection In Digital Images Based On Multiresolution Techniques, Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh

[2] Bayram, S. Sencar, T. Memon, N. (2009), An Efficient And Robust Method For Detecting Copy-Move Forgery," in Proceeding of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09), pp. 1053–1056, Taipei, Taiwan.

[3] Fridrich, J. Soukal, D. Luk, J. (2003), Detection of copy-move forgery in digital images, Proc. Digital Forensic Research Workshop, Cleveland, OH, USA.

[4] Amerini, I. Ballan, L. Caldelli, R. Bimbo, A. Serra, G. (2011), A SIFT-based forensic method for copy-move attack detection and transformation recovery, IEEE Transactions on Information Forensics and Security, vol. 6, issue 3, pp. 1099-1110.

[5] Athitsos, V. Swain, M. Frankel, C. (1997). Distinguishing photographs and graphics on the World Wide Web. In IEEE Workshop on Content-Based Access of Image and Video Libraries, University of Chicago, pp. 10-17.

[6] Christlein, V. Riess, C. Jordan, J. Riess, C. Angelopoulou, E. (2012), An Evaluation of popular Copy-Move Forgery Detection Approaches, IEEE Transactions on Information Forensics and Security, Vol.7, pp.1841-1854.

[7] Hsu, H. and Wang, M. (2012), Detection Of Copy-Move Forgery Image Using Gabor Descriptor, in Proceedings of the International Conference on Anti-Counterfeiting, Security and Identification (ASID '12), IEEE, August 2012, pp. 1–4.

[8] Kaur, H. and Kaur, K. (2015), A Brief Survey of Different Techniques for Detecting Copy- Move Forgery**,** International Journal of Advanced Research in Computer Science and Software Engineering, vol. 5, issue: 4, pp. 875-882.

[9] Liu, Chun-Lin. (2010), A Tutorial of the Wavelet Transform, University of Colorado, pp. 72. Luo, W. Qu, Z. Pan, F. Huang, J. (2007), A Survey of Passive Technology For Digital Image Forensics, Frontiers of Computer Science in China, Vol. 1, pp. 166−179.

[10] Popescu, A. and Farid, H. (2004), Exposing Digital Forgeries By Detecting Duplicated Image Regions, Tech. Rep. TR2004-515, Dartmouth College, Computer Science, Hanover, Conn, USA.

[11] Sridevi, M. Mala, C. Sandeep, S. (2012), Copy–Move Image Forgery Detection In A Parallel Environment, Computer Science and Information Technology, vol. 2, pp. 19-29.

[12] Zhong, L. and . Xu, W. (2013), A Robust Image Copy-Move Forgery Detection Based On Mixed Moments, in Proceedings of the 4th IEEE International Conference on Software Engineering and Service Science (ICSESS '13), pp. 381–384, IEEE.

[13] Zhao, J. and Guo, J. (2013), Passive Forensics For Copy-Move Image Forgery Using A Method Based On DCT And SVD, Forensic Science International, Vol. 233, pp. 158–166.

[14] Robust Detection of Region-Duplication Forgery in Digital Image Weiqi Luo, Jiwu Huang GuangDong Key Lab. of Information Security Sun Yat-Sen University,Guangzhou,China,51027

[15] Al-Sawadi, M. (2013**),** Automatic Detection of Copy-Move Image Forgery Based on Clustering Technique, Master Thesis, Department of Computer Engineering, College of Computer and Information Sciences, King Saud University, Riyadh.

[16] Tao Jing Xinghua li, Feifei Zhang, Image Tamper Detection Algorithm Based on Radon and fourier-Mellin Transform",pp 212-215 IEEE 2010.

[17] B. L. Shivakumar, Lt. Dr. S. Santhosh Babu,"Detecting copy-Move Forgery in Digital Images: A Survey and Anaysis of Current Method", Global journal of computing science and Technology, vol. 10 issue 7 ver. 1.0 sep 2010.

[18] Leida Li, Shushang Li, Jun Wang, "Copy-Move Forgery Detection Based on PHT", IEEE conference on 2012.

[19] Er. Saika Khan, Er. Arun Kulkarni, "An Efficient Method For Detection of Copy-Move Forgery Using Discrete Wavelet Transform", IJCSE vol. 2, No. 05, 2010, 1801-1806.

[20] Judith A. Redi, Weim Taktak, Jean-Luc Dugelay, "Digital Image Forensics: a booklet for Beginners", Multiledia tools