ENHANCED BLOCK-BASED COPY-MOVE IMAGE FORGERY DETECTION USING K-MEANS CLUSTERING TECHNIQUE

OSAMAH MOHAMMED ABDO AL-QERSHI

UNIVERSITI SAINS MALAYSIA

2018

ENHANCED BLOCK-BASED COPY-MOVE IMAGE FORGERY DETECTION USING K-MEANS CLUSTERING TECHNIQUE

by

OSAMAH MOHAMMED ABDO AL-QERSHI

Thesis submitted in fulfilment of
the requirements for the degree of
Doctor of Philosophy

June 2018

ACKNOWLEDGMENT

First of all, praise is due to Allah, the Almighty, for giving me the opportunity and strength to complete my thesis.

I would like to express my deep and sincere gratitude to my supervisor, Assc. Prof. Dr. Khoo Bee Ee. Her wide knowledge and her logical way of thinking have been of great value for me. Her understanding, encouraging and personal guidance have provided a good basis for the present thesis.

I would like to thank Professor Ashok Singh, University of Nevada, for doing the statistical analysis in chapter three of this thesis.

I owe my loving thanks to my wife Amal, for the unconditional love and patience that she gave me. Without her encouragement and understanding it would have been impossible for me to finish this work. My special gratitude is due to my parents, my brothers, my sisters and their families for their loving support and well-wishes/prays that kept me motivated all the time.

Also, I would like to express my sincere thanks to the School of Electrical and Electronic Engineering, Universiti Sains Malaysia, for providing the necessary facilities for this research.

Lastly, I offer my regards and blessings to all those who supported me in any aspect during the completion of this thesis.

TABLE OF CONTENTS

CHAPTER FOUR: RUSULTS OF THE PRELIMINARY INVESTIGATION

CHAPTER FIVE: EXPERIMENTAL RESULTS AND DISCUSSION

APPENDICES

LIST OF PUBLICATIONS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Acc	Overall Accuracy
AD test	Anderson-Darling test
ANOVA	Analysis of variance
AZS	Adaptive Zigzag Scanning
BBF	Best Bin First
CBF	Counting Bloom Filters
CCV	Color Coherence Vector
CLD	Colour Layout Descriptors
CMF	Copy-Move Forgery
CMFD	Copy-Move Forgery Detection
CMH	Copy-move Hard
CoMoFoD	Copy-Move Forgery Database
COVERAGE	COpy-moVe forgERy dAtabase with similar but Genuine objEcts
CRLBP	Completed Robust Local Binary Pattern
CSLBP	Center Symmetric Local Binary Pattern
DAR	Detection Accuracy Rate
DCT	Discrete Cosine Transform
DCT-QCD	Discrete Cosine Transform Quantization Coefficient Decomposition
DE	Differential Evolution
DoG	Difference of Gaussian
DP	Detection Precision
DWT	Discrete Wavelet Transform

ESS	Efficient Subwindow Search
FFT	Fast Fourier Transform
FLANN	Fast Library for Approximate Nearest Neighbors
FMT	Fourier-Mellin Transform
FN	False Negative
FNR	False Negative Rates
FP	False Positive
FPR	False Positive Ratio
FWHT	Fast Walsh-Hadamard Transform
g2NN	generalized 2NN
HIS	Histogram Intersection Similarity
HOG	Histogram of Orientated Gradients
KPCA	Kernel Principal Component Analysis
KW test	Kruskal–Wallis test
LBP	Local Binary Pattern
LFD	Local Fractal Dimension
LLE	Locally Linear Embedding
LPFFT	Log-Polar Fractional-Fourier Transform
LPT	Log-polar Transform
LS	Lexicographic sort
LSH	Locality Sensitive Hashing
MIFT	Mirror Reflection Invariant Feature
MROGH	Multisupport Region Order-based Gradient Histogram
NCH	Normalized Color Histogram

NMS	Non-Maxima Suppression
NN	Nearest Neighbor
P	Precision
PCA	Principal Component Analysis
PCET	Polar Complex Exponential Transform
PCT	Polar Cosine Transform
PDA	Pixel Detection Accuracy
PFP	Pixel False Positive
PHT	Polar Harmonic Transform
PST	Polar Sine Transform
R	Recall
RANSAC	Random sample consensus
SAD	Sum of Absolute Differences
SIFT	Scale-Invariant Feature Transform
SLIC	Simple linear Iterative Clustering
SPT	Steerable pyramid transform
SSD	Sum of Squared Differences
SURF	Speeded Up Robust Features
SVD	Singular Value Decomposition
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
TPR	True positive ratio
ULPF	Upsampled Log-Polar Fourier

WLD	Weber law descriptor
WP	Weber pattern
WPGMC	Weight Center of Mass Distance
ZMs	Zernike Moments

PENAMBAHBAIKAN PENGESANAN PEMALSUAN IMEJ SALIN-GERAK BERDASARKAN BLOK MENGGUNAKAN TEKNIK PENGELOMPOKAN PURATA-K

ABSTRAK

Dalam tesis ini, kesan jenis ciri dan kaedah pemadanan telah dianalisis dengan membandingkan gabungan yang berbeza bagi kaedah pemadanan - jenis ciri untuk pengesanan pemalsuan imej salin-gerak. Hasilnya menunjukkan terdapat interaksi di antara beberapa ciri dan beberapa kaedah pemadanan. Oleh kerana pentingnya proses pemadanan, tesis ini memberi tumpuan kepada peningkatan proses pemadanan dengan mencadangkan penambahbaikan aliran proses pengesanan pemalsuan salin-gerak berasaskan blok. Aliran proses yang dicadangkan bergantung kepada teknik pengelompokan yang mengatur blok imej ke dalam kelompok yang berlainan, dan kemudian secara bebas melaksanakan pemadanan blok di dalam setiap kelompok yang akan mengurangkan masa yang diperlukan untuk pemadanan dan meningkatkan nisbah positif yang benar (TPR) juga. Untuk melaksanakan aliran proses yang dicadangkan, dua gabungan kaedah pemadanan - jenis ciri digunakan. Dalam kes pertama, Momen Zernike (ZMs) digabungkan dengan Cincangan Kepekaan Tempatan (LSH) dan diuji pada tiga set data. Keputusan ujikaji menunjukkan bahawa aliran proses yang dicadangkan mengurangkan masa pemprosesan sebanyak 73.05% hingga 84.70% dan meningkatkan ketepatan pengesanan sebanyak 5.56% hingga 25.43%. Dalam kes kedua, Jelmaan Kosinus Polar (PCT) telah digabungkan dengan Penyusunan Leksikografi (LS). Walaupun aliran proses yang dicadangkan tidak dapat

mengurangkan masa pemprosesan, ia meningkatkan ketepatan pengesanan sebanyak 32.46%. Hasil yang diperolehi dianalisis secara statistik, dan terbukti bahawa aliran proses yang dicadangkan dapat meningkatkan ketepatan pengesanan dengan ketara berdasarkan perbandingan dengan dua kaedah yang lain.

ENHANCED BLOCK-BASED COPY-MOVE IMAGE FORGERY DETECTION USING K-MEANS CLUSTERING TECHNIQUE

ABSTRACT

In this thesis, the effect of feature type and matching method has been analyzed by comparing different combinations of matching method – feature type for copy-move image forgery detection. The results showed an interaction between some of the features and some of the matching methods. Due to the importance of matching process, this thesis focused on improving the matching process by proposing an enhanced block-based copy-move forgery detection pipeline. The proposed pipeline relied on clustering the image blocks into clusters, and then independently performing the matching of the blocks within each cluster which will reduce the time required for matching and increase the true positive ratio (TPR) as well. In order to deploy the proposed pipeline, two combinations of matching method - feature type are considered. In the first case, Zernike Moments (ZMs) were combined with Locality Sensitive Hashing (LSH) and tested on three datasets. The experimental results showed that the proposed pipeline reduced the processing time by 73.05% to 84.70% and enhanced the accuracy of detection by 5.56% to 25.43%. In the second case, Polar Cosine Transform (PCT) was combined with Lexicographical Sort (LS). Although the proposed pipeline could not reduce the processing time, it enhanced the accuracy of detection by 32.46%. The obtained results were statistically analyzed, and it was proven that the proposed pipeline can enhance the accuracy of detection significantly based on the comparison with other two methods.

CHAPTER ONE

INTRODUCTION

1.1 Background

In our current world, digital images have become a very important source of information. However, the art of making an image forgery is as old as photography itself. Traditionally, an image implies the truth of what has happened, but photography lost its innocence many years ago (Wang, Dong and Tan, 2009). It has been said that ‘a picture is worth a thousand words’ and that ‘seeing is believing’. However, those sayings appear to not be completely acceptable considering the existence of simple and effective photo editing software. Popular and simple computer software can be used by average computer users to tamper with digital images in such a way that it does not leave a noticeable trace. People can share forged, or tampered, images for fun on social media. However, forged images can be used in many serious cases such as scientific publication and media (Mahdian and Saic, 2010).

During the Arab Spring in 2011, Yemen's state newspaper published a photo of a large pro-regime rally, while downplaying the significance of anti-regime protests. It was obvious that the image was photoshopped to make the rally look twice as big (Lubin, 2011). The photo and a close-up are shown in Figure 1.1. In another serious case, Dr. Hwang Woo-Suk, a Korean scientist, is one of the famous examples of employing forged images in scientific research (Normile, 2009). He managed to publish his remarkable results in pioneering stem cell research in the journal Science

using tampered images in 2004. After finding that he faked much of his stem cell research, he was charged with embezzlement and bioethics law violations. That scandal made other journals consider the importance of investigating the authenticity of images in submitted manuscripts. Since 2002, the Journal of Cell Biology has been testing images. The editors of the journal estimate that 25% of the accepted manuscripts have images that are modified beyond their standards, while 1% contain fraudulent images (Wade, 2006). There are many cases involving manipulated images with more serious implications have arisen in science and law. Forged images might be also exploited to tarnish the public opinion of a celebrity or a public figure. As a result, we cannot take the integrity and authenticity of digital images for granted any more.

The photoshopped photo A close-up

Figure 1.1: Photoshopped photo published in Yemen's state newspaper (Lubin 2011)

1.2 Motivation

Forged images are exploited to mislead people's opinion in many fields. Therefore, a powerful and user-friendly forensic tool for detecting image forgery is required. It

could be used not only by academic journals or forensics experts, but also by news agencies. Moreover, the detection tool could be used on the web as an important step, and it could work like an initial spam filter. Online trading community could also use such a tool to check the images of products and make sure that "what buyers see is what they get". The availability of the tool could minimize the impact of fake images and stop misleading the public opinions. Such a tool, will be built by making use of the current digital image authentication techniques.

Overall, there are two types of methods that can be used for authenticating digital images: active authentication and passive authentication (Lian and Kanellopoulos, 2009). Active authentication methods are those methods which require prior processing, such as embedding data or generating signature, to be able to authenticate the images. In contrast, passive authentication methods do not require any prior processing. The methods that belong to the active authentication category can be divided into two types. The first type is based on digital watermarking, which embeds a watermark into the image to be protected and extracts it when an authenticity check is made of the image. In the second type, a digital signature is generated at the acquirement end, and then another one is regenerated using the same method when authentication is required. By comparing the two signatures, the authenticity of the image can be judged.

The main disadvantage of watermarks is that they must be embedded in the image either at the time of capturing the image or later by an authorized person (Lin *et al.*, 2013). Embedding the watermark requires cameras that have a watermarking facility or subsequent processing of the original image. Moreover, watermarks can degrade the visual quality of the watermarked image. The same thing can be said about digital signatures because they share the same drawbacks of watermarking. On

the other hand, passive authentication is performed without any help from the additional information (Birajdar and Mankar, 2013).

There are two categories of forgery type-dependent passive authentication techniques: copy-move detection techniques and image-splicing techniques (Qureshi and Deriche, 2015). Image splicing is achieved simply by cutting a region from one or more images and pasting it, or them, into another image (Bakiah *et al.*, 2016). This technique can cause inconsistencies in many features, such as an abnormally sharp transient at the splicing edges (Zhen, Shuozhong and Xinpeng, 2010), and these inconsistencies are used to detect the forgery.

Due to the need to more than one image to do image splicing, such a forgery is relatively harder to achieve. That is because the chosen images should share some characteristics such as noise, illumination, color tone, direction of the shadow, etc. However, even when much care is taken during doing the image splicing, there must be some inconsistencies caused, which makes detecting such a forgery relatively easier especially with the help of machine learning methods (Xiao, 2014).

In contrast, copy-move forgery is easy and common because it can be accomplished using only one image by means of a simple copy-and-paste operation in which a region of the image is cloned and then pasted in somewhere else at the same image. Such a process can be achieved even using a smart phone by any regular user. Because the cloned region belongs to the same image, the essential characteristics, such as the color palette and the noise of the cloned region and the remainder of the image, are almost the same, which makes it harder to be detected. The ease and effectiveness of copy-move forgery make it the most common way to

create fake images (Ardizzone, Bruno and Mazzola, 2010a; Redi, Taktak and Dugelay, 2011).

1.3 Problem Statement

Compared to image splicing, Copy-Move Forgery (CMF) is more common, easier to achieve, but harder to be detected. There are two categories of copy-move forgery detection algorithms; block-based category and keypoint-based category. Compared to block-based category, the algorithms which belong to keypoint-based category have a higher computational efficiency (Fan, Zhu and Liu, 2016). However, the major drawback of keypoint-based methods is their inability to detect cloned regions that have highly homogeneous texture where salient keypoints remain undetected (Amerini *et al.*, 2011). In addition, the keypoints can be removed by whoever created the forgery using copy-move counter-forensic methods (Amerini, Barni, *et al.*, 2013).

The majority of the block-based copy-move detection schemes comply to a general pipeline (Ryu *et al.*, 2013). In that pipeline, the image is subdivided into overlapping blocks, and some feature, as a vector, is extracted from every single block. All feature vectors are matched, and blocks with highly similar feature vectors are paired. The corresponding blocks to the paired vectors are considered as duplicated regions. Two stages of the pipeline have the highest impact on the general effectiveness of the copy-move detection methods and they are: feature extraction step (type of feature) and matching step (method of matching) (Christlein, Riess and Angelopoulou, 2010b).

A wide spectrum of features is used in the existing copy-move detection algorithms. Nevertheless, only a few matching methods have been exploited in those algorithms. Researchers always justify employing certain types of features, which are

invariant to geometrical operations or noise, but not all of them do the same thing when they choose matching methods. They only focus on the advantages of the feature type that they use and neglect the effect of the matching method. It is not known if the two stages have the same impact on the performance of Copy-Move Forgery Detection (CMFD) algorithms because this topic has not been studied before. Also, it is not known if there is some interaction between these two stages/factors. In addition, the main drawback of block-based methods is the huge number of blocks produced by dividing the image to be tested into overlapping blocks spaced be 1 pixel in most cases. Such a huge number of blocks requires a very long time to finish the matching step. The CMFD pipeline needs enhancement to overcome the time consumption issue. Moreover, the pipeline should take into consideration the interaction between the feature type and matching method which may affect the overall performance.

There is another issue related to CMFD, which is the datasets that have been used for evaluation. The existing CMFD datasets have some disadvantages in the case of in-depth evaluation of CMFD methods, such as size of images or applied intermediate/postprocessing attacks. Those disadvantages limit the usability and reliability of those datasets. There is still a need for a reliable and standard dataset that can be used widely for evaluating the CMFD algorithms. Such a dataset will make it easy for researchers to compare the performance of their work with the others.

1.4 Research Objectives

There are many block-based CMFD algorithms that have been proposed so far. However, all of them share the same problem of time-consuming matching step.

Therefore, the main objective of this research is to enhance the performance of CMFD in terms of speed and accuracy by means of enhancing the general pipeline of block-based CMFD. Following the problems stated in the previous section, this thesis has the following specific objectives:

1- To investigate the effect of feature type and matching method on the performance of CMFD algorithms.
2- To develop a new block-based CMFD pipeline/matching method that can enhance the performance of CMFD in terms of accuracy and processing time based on the results of objective 1.
3- To deploy the proposed pipeline/matching method in a block-based CMFD algorithm, and evaluate the proposed algorithm using three benchmark datasets to study the effect of the proposed pipeline
4- To validate the impact of the proposed pipeline by deploying it in another CMFD algorithm with a different feature type and a different matching method. The second proposed algorithm is to be evaluated as well.

1.5 Research Scope

The main goal of this research is to enhance the performance of CMFD. It was mentioned previously that keypoint-based methods have critical drawbacks which make them unreliable. Therefore, this thesis focuses on block-based methods.

In order to achieve the goal of this research, the factors, feature type and matching method that have an impact on the performance of CMFD should be investigated first. To do so, a group of features and matching methods are investigated. The selected features are Zernike Moments (ZMs), Polar Cosine Transform (PCT), and Local Binary Pattern (LBP). The selected matching methods

are: Lexicographic sort (LS), LS with grouping, *k*-d tree, and Locality Sensitive Hashing (LSH). Those features and methods have been selected based on their popularity among researchers and based on their performances which have been reported in the literature. All possible combinations of feature type-matching method are implemented and tested using the self-prepared benchmark dataset, CMH dataset (Silva *et al.*, 2015), and CoMoFoD dataset (Tralic *et al.*, 2013). The performance is evaluated based on the accuracy of detecting of the copy-moved regions. Based on the results of the investigation, a single feature and a single matching method will be chosen to develop a CMFD algorithm based on the proposed pipeline.

1.6 Thesis organization

This thesis is organized in order to reflect the importance of CMFD leading to realization of the research objectives, as follows.

Chapter One presents a general introduction to research work, and the background and the motivation of this research are discussed. Also, the objectives, scope and approach are identified in this chapter.

Chapter Two presents a literature review on the field of CMFD, which is related to this thesis. This chapter covers the current and past studies that have been carried out and found in the literature. These studies are categorized into different approaches discovering their advantages and limitations.

Chapter Three introduces the methods and the techniques which are used to achieve the objectives of this thesis. It also presents the process of preparing the dataset and the metrics which are used in the evaluation process.

Chapter Four presents a comparison and analysis of the effect of feature type and matching method on the performance of CMFD in terms of accuracy of detection. Then, it describes the development of the new CMFD pipeline.

Chapter Five presents the testing and evaluating results of the proposed CMFD pipeline in terms of detection accuracy and processing time. The results are compared with those which are obtained from the existing CMFD method of the same category. Finally, all obtained results are discussed.

Chapter Six concludes this research, and presents the contribution of this research. Some ideas for future work are also suggested.

CHAPTER TWO

LITRATURE REVIEW

2.1 Introduction

There are many powerful and easy-to-use computer software programs that can be used to make digital image forgery while leaving no visual clues from the tampering process. This circumstance makes the authenticity of images very questionable, especially in fields such as legal, medical, journalism, and criminal, where digital images can play an important role. Overall, there are two types of methods that can be used for authenticating digital images: active authentication and passive authentication. Also, passive authentication techniques can be categorized based on the way of creating the forgery. This chapter starts with a brief overview of digital image forgery detection. Next, the state-of-the-art studies in CMFD with different approaches are discussed in details.

2.2 Image Authentication

The issues of multimedia security have led to the development of several approaches to tampering detection. In general, there are two types of techniques that can be used for image tampering detection, active authentication and passive authentication (Lian and Kanellopoulos, 2009), as shown in Figure 2.1. Active authentication methods are classified into two categories. The first category is based on digital watermarking, which conceals a watermark into the image at the capturing end and extracts it at the authentication end to examine whether the image has been tampered with (Kundur and Hatzinakos, 1999; Rey and Dugelay, 2002). Inserting the watermark either at the time of capturing the image using a specially equipped

camera or later by an authorized person is the main drawback of watermarking (Ho *et al.*, 2009). In addition, the subsequent processing of the original image could degrade the image visual quality.

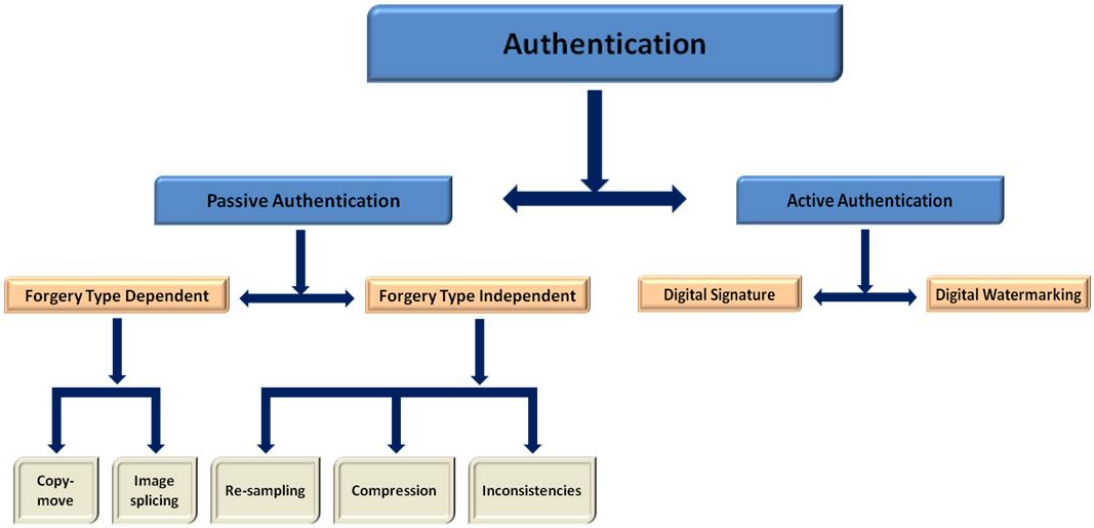

Figure 2.1: Classification of image authentication methods (Redi, Taktak and Dugelay, 2011)

The second category of methods is based on digital signatures. Those methods extract unique features from the image as a signature at the image capturing end. At the authentication end, the signature is regenerated using the same method, and the authenticity of the image can be identified through comparison. Digital signatures have similar disadvantages to the watermarking category.

2.2.1 Passive Image Authentication

Passive image authentication, which is also called digital image forensics, is the process of authenticating digital images without using any additional information aside from the pictures themselves (Zhou *et al.*, 2007). Passive techniques are classified into two categories: forgery-type dependent or independent (Redi, Taktak and Dugelay, 2011). The independent techniques are designed to detect general or

global tampering, such as traces of re-sampling, compression and inconsistencies. On the other hand, the forgery type-dependent techniques can by divided into two categories: copy-move-detection techniques (single image-based forgery) and image-splicing techniques (multiple image-based forgery) (Singh and Kaur, 2016).

Image splicing is a common process that is used to produce a digital image forgery (Ng and Chang, 2004); image splicing is achieved simply by cutting a region from one or more images and pasting it, or them, into another image (Dong *et al.*, 2009). This technique can cause inconsistencies in many features, such as an abnormally sharp transient at the splicing edges (Zhen, Shuozhong and Xinpeng, 2010), and these inconsistencies are used to detect the forgery. Copy-move is one of the most popular methods for manipulating a semantics image (Redi, Taktak and Dugelay, 2011). Copy-move is achieved by copying a region of an image and pasting it into the same image with the intent of hiding undesired objects or replicating objects. In copy-move forgery, the tampered region still shares most of its inherent characteristics, such as the color palette or pattern noise, with the remainder of the image. Moreover, a structural analysis of image regions might reveal a high level of similarity between the duplicated regions. Based on this concept, a first attempt at detecting copy-move forgery was described by Fridrich *et al.* 2003(Fridrich, Soukal and Lukáš, 2003). Since then, a substantial amount of work has been performed in the field of copy-move forgery detection, and the copy-move domain has been attracting growing interest from researchers.

2.3 Copy-Move Forgery Detection (CMFD)

The ease and effectiveness of copy-move forgery makes it the most common forgery that is used to alter the content of an image (Ardizzone, Bruno and Mazzola, 2010a).

Because the copied regions come from the same image, their most important characteristics, such as the color palette and dynamic range, will be compatible with the remainder of the image (Kang and Cheng, 2010). However, in practical situations, forgery may involve more than a simple duplicating operation. Several image processing operations, attacks, could be involved in practical copy-move forgery. These attacks can be divided into two groups: intermediate attacks and preprocessing attacks. Intermediate attacks are used to provide a type of spatial synchronization and homogeneity between the copied region and its neighbors (Ryu, Lee and Lee, 2010). They could be rotation, scaling, mirroring, illumination modifying, or chrominance modifying. In a practical situation, the intermediate attacks could be a combination of two or more operations. The post-processing attacks, such as the additive noise, JPEG compression or blurring, are used to remove any detectable traces of the copy-move operation, such as sharp edges (Liu *et al.*, 2010).

The algorithms, which are used for Copy-Move forgery detection, are categorized into two folds: keypoint-based and block-based. The methods that belong to the keypoint-based category operate on the entire image. The features are extracted only for the detected keypoints in the image, which increases the computational efficiency. Those methods employ features such as SIFT (Scale-Invariant Feature Transform) (Amerini, Ballan, *et al.*, 2013) and SURF (Speeded Up Robust Features) (Bo *et al.*, 2010). In contrast, the block-based algorithms work on all image areas. They divide the image into overlapping blocks and extract some features from the blocks (Li *et al.*, 2013; Ustubioglu *et al.*, 2016a). The feature vectors are then compared and matched to find the identical corresponding blocks. In the next sections, the algorithms that belong to both categories are presented and discussed.

2.4 Keypoint-based CMFD

The keypoint-based algorithms extract the distinctive local features such as corners, blobs, and edge from the image (Bakiah *et al.*, 2016). Each feature is presented with a set of descriptor produced within a region around the features. The descriptor helps to increase the reliability of the features to the affine transformation. Then, both features and descriptors in the image are classified and matched to each other to find the duplicated regions in the copy-move forgery.

The keypoint-based CMFD algorithms comply with a general and straightforward pipeline shown in Figure 2.2. First, the image is optionally preprocessed; resized to minimize the size and/or converted to a grayscale to reduce the number of color channels. Then, the robust and invariant keypoints are localized and their associated features extracted for each keypoint. For this stage, most of the exiting keypoint-based algorithms employ SIFT (Amerini *et al.*, 2011; Kudke and Gawande, 2013; Ustubioglu *et al.*, 2015) or SURF (Shivakumar and Baboo, 2011b; Zhang and Wang, 2012; Manu and Mehtre, 2016). Then, the feature vectors are matched, and keypoints with highly similar feature vectors are paired. Matching feature vectors is achieved using Nearest Neighbor (NN) methods such as Best Bin First (BBF) (Jaberi *et al.*, 2013a), 2NN (Hashmi, Anand and Keskar, 2014) and generalized 2NN (g2NN) (Amerini, Ballan, *et al.*, 2013). To eliminate the outliers, which resemble false matches, typically RANSAC is used (Amerini *et al.*, 2011) at the last stage, a verification process is used to pair the keypoints that represent the identical regions. Also, this stage is responsible for removing the outliers which are the isolated keypoints. Some algorithms have an additional stage in which the detected identical regions are localized and visualized (Pan and Lyu, 2010b).

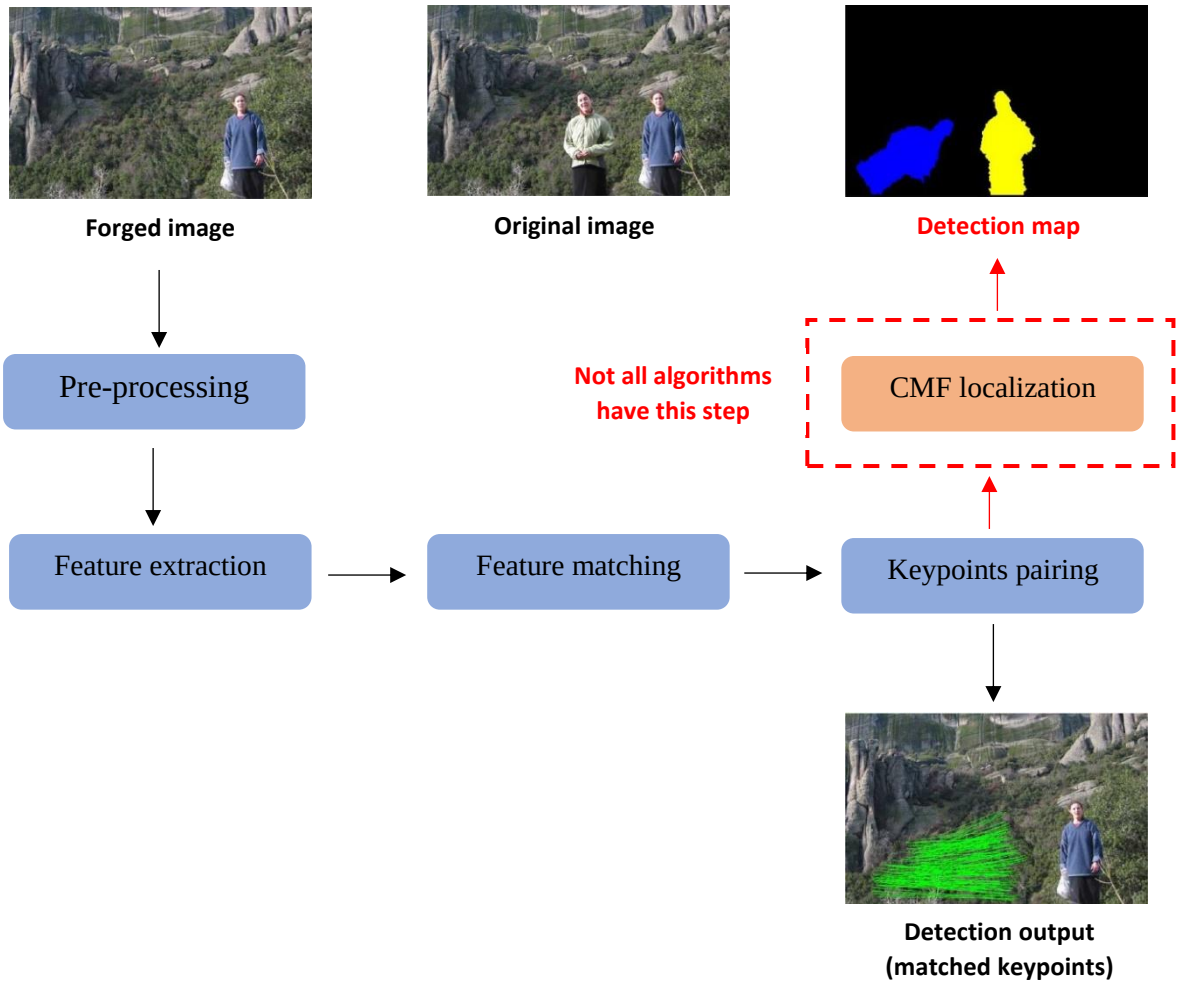

Figure 2.2: Keypoint-based CMFD pipeline. Images were taken from Pan and Lyu, (2010b)

Despite numerous keypoint-based algorithms have been proposed over the past few years, the major differences between those algorithms are the types of features and the matching methods. This is due to the significance of these two stages of the keypoint-based CMFD pipeline. Therefore, these two stages are presented in details in the next section.

2.4.1 Feature Types in Keypoint-based CMFD Algorithms

Keypoint-based CMFD algorithms depend on extracting distinctive local features in the image and to produce keypoint descriptors that present those features. Those feature vectors/descriptors are invariant to rotation, translation, and scaling, are

partially invariant to illumination changes and are robust to local geometric distortion (Lowe, 1999; Bay, Tuytelaars and Van Gool, 2006).

2.4.1.a SIFT

SIFT is the most popular keypoint feature type in keypoint-based CMFD algorithms (Bakiah *et al.*, 2016). It was proposed by Lowe *et al.* for feature matching between two images. SIFT detects salient and robust keypoints of an image at different scales from Difference of Gaussian (DoG) pyramid in scale-space representation to improve the computational speed during the extraction process (Juan and Gwun, 2009). Subsequently, the 128-d SIFT descriptor is built from the gradient orientation histogram in each SIFT keypoint. SIFT features are extracted only for keypoints, which increases computational efficiency. The robustness of SIFT against intermediate and postprocessing attacks made it very popular among researchers who adopted it in CMFD (Ardizzone, Bruno and Mazzola, 2010b). However, exploiting SIFT in CMFD algorithms has some drawbacks. First, the size of the SIFT descriptor/feature vector is relatively high compared to other types of keypoint descriptors, which increases the computational complexity. Therefore, Principal Component Analysis (PCA) can be applied to reduce the dimensionality of the SIFT descriptor (He, Huang and Kuang, 2013). Secondly, SIFT is unable to detect the duplicated regions with a highly uniform texture. Such a limitation may be minimized by incorporating another feature such as Zernike moments (Ouyang, Coatrieux and Shu, 2015). Thirdly, utilizing SIFT make the CMFD algorithm unable to define the shape of the copy-move regions, and an extra step is needed to connect the boundaries of keypoints in the final stage using methods such as J-Linkage (Amerini, Ballan, *et al.*, 2013).

The first attempt to utilize the SIFT was made by Huang, Guo and Zhang, (2008). In their algorithm, only the matching of SIFT key-points can be performed, by means of the best-bin-first nearest-neighbor identification. Ardizzone, Bruno and Mazzola, (2010b) adopted SIFT to detect multiple copies in forged images. Usually, a SIFT descriptor is used to detect copy-move regions by matching key-points instead of blocks, but their algorithm is based on matching objects rather than points.

Zhang, Guo and Cao, (2010) extracted SIFT keypoints of an image and then, matched them to obtain keypoint pairs. To distinguish between the source and the target, these authors used a voting strategy that was based on the match vectors' orientations. Then, they used the Efficient Subwindow Search algorithm (ESS) twice to localize the duplicated regions with bounding boxes. Finally, pixel-wise segmentation is obtained by planar homographic constraint and homogeneity census.

Amerini *et al.* proposed a SIFT-based detection algorithm that can detect and then estimate the geometric transformations that are used in the copy-move forgery (Amerini *et al.*, 2011). The detection process has three steps: in the first step, SIFT features are extracted, and the keypoints are matched; the second step is devoted to keypoint clustering and forgery detection, and the third step estimates the geometric transformations that occurred, if any. The experimental results showed very good performance in terms of a high true positive ratio (TPR) and a low false positive ratio (FPR), even with JPEG compression and additive noise. Moreover, the results showed a high degree of precision in the estimate of the various parameters of the affine transformation.

Similar to Amerini's algorithm, Pan and Lyu proposed another SIFT-based detection algorithm that has the ability to estimate the geometric transformations

used in the copy-move (Pan and Lyu, 2010a, 2010b). This algorithm starts with converting the suspect image into grayscale. SIFT is then used to find image key-points and to collect image features at the detected key-points. The detected SIFT key-points are then initially matched based on their feature vectors using the best-bin-first algorithm. Based on the key-point matching, the possible geometric distortions of the duplicated regions are estimated using RANSAC.

In their SIFT-based algorithm, Chihaoui, Bourouis and Hamrouni, (2014) used two techniques in parallel to match SIFT descriptors. The first one is based on Euclidian distance and the second one is based on Singular Value Decomposition (SVD) factorization proximity matrix. In the last stage, the results of the two matching techniques are fused to identify the duplicated regions. The algorithm reduced the false matching but it became impractical for images with a large number of features.

2.4.1.b SURF

In 2006, Bay *et al.* introduced the SURF technique to tackle the problem of point and line segment correspondences between two images of the same scene or object (Bay, Tuytelaars and Van Gool, 2006). The SURF descriptor has been exploited in CMFD to overcome geometric operations such as rotation and scaling. The advantage of SURF over SIFT is that the size of SURF descriptor is 64-d and it is much faster than SIFT while keeping a good performance as SIFT.

Bo *et al.*, (2010) adopted the SURF descriptor to overcome geometric attacks. After extracting the SURF descriptors from each keypoint, the keypoints are divided randomly into two subsets. For each subset, the nearest neighbors are determined for

each keypoint and the matching records are saved. Then, the previous two steps are repeated until only one keypoint is left in each subset.

Lin and Wu, (2011) proposed an integrated technique for splicing and copy-move forgery detection. The copy-move part is based on extracting keypoints using SURF from both the image blocks and undivided image. The algorithm has the same drawbacks of Bo's algorithm.

To enhance the matching step, Shivakumar and Baboo, (2011b) adopted *k*-d tree technique in their SURF-based CMFD algorithm. Experimental results showed that the proposed method can detect copy-move forgery with a minimum of false matches, even with scaling, rotation, and Gaussian noise. However, only matched keypoints can be detected.

Neamtu *et al.*, (2013) aimed at enhancing the performance of SURF-based CMFD by adopting a fast approximation to the nearest neighbor method during matching step. For that purpose they used the Fast Library for Approximate Nearest Neighbors (FLANN). The experimental results were presented visually and not using quantitative measures. Similar work was done by Amtullah and Koul, (2014) where keypoint descriptors are matched with each other to identify the matching keypoints. The best match of the keypoint is found by identifying its nearest neighbor based on a predefined threshold. To enhance the localization of the CMFD using SURF, Manu and Mehtre, (2016) utilized segmentation based on simple linear iterative clustering (SLIC). After segmenting and labeling image regions, the algorithm matched the SURF keypoints. Then, the algorithm searched for regions that contain the maximum number of matched keypoints.

2.4.1.c Other Keypoint-based Feature Types

Besides SIFT and SURF, researcher exploited other types of keypoints-based features in order to enhance the performance CMFD even further. Jaberi *et al.*, (2013b, 2013a) exploited Mirror Reflection Invariant Feature (MIFT) features instead of SIFT features to find similar regions in images. MIFT features, which was proposed by Guo *et al.*, (2010), shares all properties of SIFT features but are also invariant to mirror reflection transformations. Although their MIFT-based algorithm outperformed the algorithm of Pan and Lyu, (2010a), but it has the same disadvantage of the other keypoint-based algorithms which do not work well if the duplicated region corresponds to a flat surface where no interest points can be detected. To overcome the problem of undetectable flat surfaces, Zheng, Hao and Zhu, (2012) proposed a new feature based on keypoint position relationship. However, neither quantitative results nor comparisons were reported in their paper.

2.4.1.d Multiple Keypoint-based Feature Types

In this category, more than one technique is incorporated in the CMFD algorithm. Generally, the keypoints are detected using one technique while the feature descriptors of the keypoint are generated using another technique. A clear example of this combination is employing the Harris corner detector to detect the keypoints with different types of feature descriptors. The Harris corner detector is a point-feature extraction operator that is proposed by Harris and Stephens, (1988) as an improved version of Moravec corner detector. The detector computes an autocorrelation matrix for each pixel of the image, and then it computes the corner response function value for each pixel. Finally, it obtains the corner points using non-maxima suppression (NMS) algorithm. The Harris corner detector is one of the most popular

algorithms for extracting keypoints invariant to translation, rotation, and partially to illumination (Moreels and Perona, 2007).

The first algorithm of this category was proposed by Shivakumar and Baboo, (2011a), where SIFT features were extracted from at keypoints and *k*-d tree was used to match the keypoints. They tested their algorithm only on 4 images and they did not compare it with other algorithms.

To increase the robustness of CMFD against scaling and rotation attacks, Kakar and Sudha, (2011, 2012) opted to use MPEG-7 image signature tools, which was designed for robust and fast image and video retrieval After extracting keypoints using the Harris corner detector, a signature is extracted from the circular region around each keypoint to form feature descriptors.

Zhao and Zhao, (2013) attempted to tackle the rotation attack by employing Local Binary Pattern (LBP). Three variants of rotation invariant uniform LBP, were applied to the circle patch around each feature point to extract the features. For a given circle patch around the keypoint, three histograms of rotation invariant uniform LBP were used as feature vectors. Then the keypoints were matched based on their representation feature vectors using the (BBF) algorithm. The experimental results showed that their algorithm outperformed the algorithm of Pan and Lyu, (2010a), but not when rotation was involved in creating the forgery.

Aiming at improving the robustness of CMFD against rotation attacks, Chen *et al.*, (2013) used step sector statistics as a descriptor to represent the small circle image region around each Harris points. The experimental results showed that their algorithm outperformed Pan's SIFT algorithm even with several geometrical attacks

(including rotation, scaling and flipping) and image degradations (including JPEG compression and Gaussian noises).

To enhance the robustness of CMFD, Guo, Liu and Wu, (2013) used a modified version of the DAISY descriptor. The DAISY descriptor is insensitivity to contrast variation and scale changes, and they modified it to become rotation-invariant. The experimental results showed that the algorithm could detect if any region had been duplicated by any diverse types of transformation, such as rotation, scaling, JPEG compression, and Gaussian noise. In addition, their algorithm outperformed SIFT-based algorithms by Huang *et al.*, (2008) and Zhao and Zhao, (2013).

Zheng and Chang, (2014) extracted SURF descriptors from Harris points and then used nearest neighbor search to match the keypoints. The experimental results showed that their algorithm can effectively detect copy-move forgery for images that have subjected to various forms of attacks, including scaling, rotating, white Gaussian noise, and lossy JEPG compression etc.

Yu, Han and Niu, (2014) employed another rotation invariant, which is Multisupport Region Order-based Gradient Histogram (MROGH), to represent the Harris keypoints. The performance of their algorithm outperformed the performance of Amerini's SIFT-based and Shivakumar's SURF-based method in terms of robustness against rotation attacks. However, they did not test the robustness of their algorithm against other types of attacks.

2.4.2 Matching Methods in Keypoint-based CMFD Algorithms

Nearest neighbor search can be used to match the detected keypoints of a suspect image. Basically, it examines the similarity between keypoints by calculating the

distance of each point in vector space. The keypoints are considered as match if the distances satisfy a predefined threshold. However, such a method has a high computation complexity, which encouraged researchers to adopt improved versions of the nearest neighbor search to suit CMFD algorithms (Bakiah *et al.*, 2016). One of the methods that have been used to limit the amount of computation in high dimensional space is the Best Bin First (BBF) (Beis and Lowe, 1997). BBF is a variant of the *k*-d tree search algorithm that finds the nearest neighbor for a large fraction of the queries, and a very close neighbor in the remaining cases. Good examples of the algorithms that exploited BBF can be found in the works that have done by Huang, Guo and Zhang, (2008), Kakar and Sudha, (2012), Chen et al., (2013), Jaberi et al., (2013a), and Zhao and Zhao, (2013).

To eliminate any the unavoidable false match that can be caused by BNN, the researchers adopted the 2NN method to match the keypoints. The 2NN accepts a keypoint as a match if the ratio of distance between the closest and second-closest neighbors is less than a predefined threshold as it can be seen in the works by Guo, Liu and Wu, (2013) and Hashmi, Anand and Keskar, (2014).

Amerini *et al.*, (2011) enhanced the 2NN by proposing the g2NN to produce the highest match, especially the multiple copy-move forgeries in an image (Amerini *et al.*, 2011). In contrast to 2NN, the g2NN accepts a keypoint as a match if the ratio of distance between the closest to the *k*-closest neighbors is less than a predefined threshold. Hence, the g2NN can detect the *k* matches of a keypoint as it can be found in the works of Amerini *et al.*, (2011) and Mohamadian and Pouyan, (2013).

Another nearest neighbor search technique that has been used to match keypoints is the *k*-d tree. The *k*-d tree, which produces reliable results and a lower

false negative rate, preprocesses data into a data structure allowing users to make efficient range queries (Shivakumar and Baboo, 2011b). The *k*-d tree has been used in the following keypoint-based CMFD algorithms (Shivakumar and Baboo, 2011b, 2011a; Jaberi *et al.*, 2013a; Yu, Han and Niu, 2014).

In a different scenario, Ardizzone, Bruno and Mazzola, (2010b) used an agglomerative hierarchical-tree clustering method to group keypoints, followed by Weight Center of Mass Distance (WPGMC) linkage to obtain the object's region represented by clusters. After that, the algorithm matches clusters of keypoints, rather than single keypoints, Other clustering methods have been in CMFD used such as *k*-means clustering (Pan and Lyu, 2010b; Anantharaj, 2014) or based on a Gaussian model (Yadav and Kapdi, 2015).

2.4.3 Advantages and Disadvantages of Keypoints-based CMFD

Exploiting keypoints in CMFD algorithms has gained a lot of attention by researchers because they aimed at enhancing the performance of detection in terms of accuracy and execution time. Therefore, the keypoints-based CMFD algorithms have two advantages in general which are:

1- **Low computational complexity**: Because the number of keypoints to be extracted and matched is way smaller than the number of blocks in block-based CMFD algorithms. In general, the number of keypoints that can be extracted is a few hundreds to a few thousands depending on the contents of the image. In contrast, in block-based CMFD, the number of features to be extracted and matched may be hundreds of thousands depending on the size of the image.