

**IMAGE QUALITY AND FORGERY DETECTION COPULA-BASED  
ALGORITHMS**

by

**Radwa Abdelwahab Mohamed Hammad**

B.Sc., Cairo University, 2008

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF  
MASTER OF SCIENCE  
IN  
COMPUTER SCIENCE

UNIVERSITY OF NORTHERN BRITISH COLUMBIA

February 2016

© Radwa Hammad, 2016

## **Abstract**

Copula functions are important tools to investigate dependence structure between random variables. There are many copulas such as: Gaussian, Marshall-Olkin, Clayton, and Frank copulas. Although, copulas have been used in finance, oceanography, and hydrology, they have been applied in limited applications in the image processing field. In this thesis, copulas are applied to calculate the mutual information of two images, which in turn is used to measure image quality of a targeted image and also used to detect copy-move forgery in images. The proposed algorithms introduce new alternatives for existing image quality assessment and forgery detection methods. These algorithms are easy to use and highly accurate. The results for our image quality assessment algorithm are comparable or better than those of established methods in the literature, while the results for our image forgery detection algorithm are accurate even after applying different manipulation and post-processing techniques on the forged images.

## Table of Contents

<b>Abstract .....</b>	<b>ii</b>
<b>Table of Contents.....</b>	<b>iii</b>
<b>List of Figures .....</b>	<b>vii</b>
<b>List of Tables .....</b>	<b>x</b>
<b>List of Abbreviations .....</b>	<b>xi</b>
<b>Acknowledgments .....</b>	<b>xiv</b>
<b>1. Introduction .....</b>	<b>1</b>
<b>2. Background .....</b>	<b>3</b>
2.1 Image quality measure .....	3
2.2 Literature review on image quality measures .....	6
2.2.1 Statistical oriented methods .....	7
2.2.2 Human Vision System (HVS) oriented methods .....	8
2.2.2.1 Visual Information Fidelity (VIF) .....	8
2.2.2.2 Universal Quality Measure (UQI) .....	9
2.2.2.3 Structural Similarity (SSIM).....	10

2.2.2.4 Copula based image quality measure algorithm .....	12
2.3 Image forgery techniques .....	13
2.3.1 Copy-Move (Cloning) forgery .....	14
2.4 Literature review on copy-move forgery detection .....	16
<b>3. Image Quality and Dissimilarity Verification Copula-Based Algorithms .....</b>	<b>21</b>
3.1 Copula based mutual information .....	22
3.1.1 Copula functions overview .....	22
3.1.2 Mutual information and copula functions .....	23
3.1.3 Copula functions types .....	24
3.1.3.1 Gaussian copula .....	24
3.1.3.2 Marshall-Olkin copula .....	26
3.1.3.3 Clayton copula .....	27
3.1.3.4 Frank copula .....	28
3.1.3.5 Gumbel copula .....	29
3.2 Steerable pyramid .....	32
3.3 The proposed algorithm.....	35
3.4 Experimental results and discussion.....	40
3.5 Localizing unmatched regions between two images .....	50
3.6 Experimental results .....	54
3.7 Conclusions and future work .....	59
<b>4. Copula Based Copy-Move Forgery Detection Algorithm for Digital Images.....</b>	<b>60</b>
4.1 The proposed algorithm.....	61
4.2 Results and discussion .....	65

4.3 Conclusion .....	72
<b>5. Conclusion and Future Work.....</b>	<b>73</b>
<b>Bibliography.....</b>	<b>76</b>
<b>Appendix A .....</b>	<b>82</b>
A.1 Gaussian and Laplacian pyramids .....	82
A.2 Image retouching .....	84
A.3 Image splicing.....	85

## List of Figures

2.1	Image quality metrics.....	6
2.2	Bayard's self portrait as a drowned man (early 1840s).....	14
2.3	Sample images that show copy-move forgery technique .....	15
3.1	Probability density function for Gaussian copula.....	25
3.2	Probability density function for Marshall-Olkin copula.....	27
3.3	Probability density function for Clayton copula.....	28
3.4	Probability density function for Frank copula .....	29
3.5	Probability density function for Gumbel copula.....	30
3.6	Steerable pyramid decomposition diagram.....	33
3.7	Steerable pyramid decomposition on Lena image: (a) original image; (b) the resulted sub bands after applying the steerable pyramid technique on the original image .....	34
3.8	The flow chart of the proposed image quality measure algorithm .....	36
3.9	The steerable pyramid decomposition: (a) monarch image (LIVE database); (b) the 26 steerable pyramid sub bands of monarch image .....	38

3.10	Histogram of church and capitol image (LIVE database) .....	38
3.11	Selected sub bands for steerable pyramid (church and capitol original image) .....	39
3.12	Selected sub bands for steerable pyramid (church and capitol distorted image) .....	39
3.13	Image quality measure process using copula functions for church and capitol image .....	40
3.14	A sample image distortion (monarch) .....	42
3.15	Sample of original images (LIVE database) .....	42
3.16	Image segmentation technique: (a), (b), (c) the segmentation procedure for 128 x 128 pixels image; (d) all regions for 128 x 128 pixels image .....	52
3.17	The flow chart of the proposed algorithm .....	53
3.18	Test image: (a) fishbowl image; (b) tampered image .....	55
3.19	The results of applying Gaussian copula .....	56
3.20	The results of our proposed algorithm on test images using Gaussian copulas .....	58
4.1	Blind copy move forgery detection algorithm's flowchart .....	64
4.2	Results for different manipulation techniques (trees) .....	66
4.3	Results for different manipulation techniques (coins) .....	67
4.4	Results for different JPEG quality factors (QF) .....	68
4.5	Results for sample of test images (CoMoFoD database) .....	70

A.1	Gaussian pyramid technique on Lena image: (a) original image; (b) the resulted sub bands after applying Gaussian pyramid technique on the original image.....	83
A.2	Laplacian pyramid technique on Lena image: (a) original image; (b) the resulted sub bands after applying Laplacian pyramid technique on the original image.....	83
A.3	Image retouching technique.....	84
A.4	Example on image splicing.....	86



**List of Tables**

3.1 Image quality results for quality measure methods (monarch) ..... 44

3.2 Image quality rank for quality measure methods for monarch image. .... 45

3.3 Image quality rank for quality measure methods for building image..... 48

3.4 Image quality rank for quality measure methods for church and capitol image ..... 48

3.5 Image quality rank for quality measure methods for caps image ..... 49

3.6 Copula image quality results using different sub bands (monarch) ..... 49

4.1 Comparison results of the six approaches for a 512 x 512 image with 8 x 8 window  
block ..... 71

## **List of Abbreviations**

BSS	Blind Source Separation
CDF	Cumulative Distribution Function
CFA	Color Filter Array
CVIPG	Computer Vision and Image Processing Group
CMF	Copy-Move Forgery
DCT	Discrete Cosine Transform
DMOS	Difference Mean Opinion Score
DWT	Discrete Wavelet Transform
FR	Full Reference
FRIQM	Full Reference Image Quality Measure
GSM	Gaussian Scale Mixture
HVS	Human Visual System
ICA	Independent Component Analysis
MC	Markov Chain

MOS	Mean Opinion Score
MSE	Mean Squared Error
MS SSIM	Multi-Scale Structural Similarity
NSS	Natural Scene Statistics
NR	No Reference
PSNR	Peak Signal to Noise Ratio
PCA	Principal Component Analysis
PDF	Probability Density Function
QF	Quality Factors
QR	Quality Ruler
RR	Reduced Reference
RLRN	Run-Length Run-Number
SSIM	Structural Similarity
SS	Single Stimulus
SPN	Sensor Pattern Noise
TID2008	Tampere Image Database 2008
TPM	Transition Probability Matrix

UQI	Universal Quality Measure
VIF	Visual Information Fidelity

## **Acknowledgments**

"Imagination is more important than knowledge" ~ Albert Einstein

I would like to express the deepest appreciation to my supervisor, Professor Saif Al Zahir for his limitless guidance and help. Special thanks go to Professor Michael Rutherford and Professor Kumar for their support. Without their support and guidance this dissertation would not have been possible.

Above all, I am so grateful for being blessed with a wonderful child, Malak, and a wonderful husband Belal. In addition, this thesis would not be possible without the love and support of my mother, Sekeena, my father, Abdelwahab, my sister Rasha, and my brothers Karim and Mohamed. Not only have they encouraged me throughout my studies, but provided me with the work ethics and motivation to succeed.

Finally, I am very thankful to everybody who supported me during my master study.

## **Chapter 1**

### **Introduction**

Copula functions are significant tools for modeling dependence of random variables. Sklar 1959 [1] was the first work in which the term 'copula' was used. In copula theory, the cumulative distribution function (CDF) of a random vector can be represented by uniform marginal cumulative distribution functions and a copula that connects these marginal cumulative distribution functions [2]. Although, copula functions are used in various applications such as economics and finance, climate research, oceanography, hydrology, geodesy, evolutionary computation, they are used in limited image processing applications such as image change detection, and image registration [3]. Using copula functions has the advantage of joining pairs of data distributions regardless of their shape [4]. In this thesis, five copula functions are used to evaluate image quality and to detect the copy move forgery in digital images.

In chapter 2, we discuss the main concepts included in this work: image quality measures, image forgery detection techniques, to give a brief overview of the topic.

In chapter 3, copula functions are used to measure image quality of a distorted image and to detect the mismatched regions in the two images. In our algorithm, the original image and the distorted image should be available to measure the quality of the distorted image. By

calculating the copula based mutual information between the two images, the original and distorted images, the image quality can be measured. In this chapter, LIVE database is used to evaluate the proposed algorithm and to compare its results with the results of published image quality measures: visual information fidelity (VIF), universal quality measure (UQI), and structural similarity (SSIM).

In chapter 4, a new blind copy move forgery detection algorithm is proposed. In this algorithm, only the forged image is available. The mutual information is calculated between the  $16 \times 16$  blocks of the image to identify if there are any duplicate or forged regions in the image. The proposed algorithm is evaluated by applying it on CoMoFoD database. The experimental results show the effectiveness of the proposed algorithm even after applying some manipulation and post processing techniques, included in the database, on the forged image such as color adjustment, color reduction, JPEG compression, scaling, and rotation.

In chapter 5, we conclude our work and introduce the possible future work that can be conducted to improve our proposed algorithms.

## **Chapter 2**

### **Background**

In this chapter, we introduce an overview of the main topics included in this thesis. First, we discuss the image quality measures. In addition, we discuss the different techniques used in the literature to evaluate image quality subjectively and objectively. Further we introduce the copy-move forgery technique. Finally, we introduce the different techniques the researchers proposed to detect forgery in digital images.

#### **2.1 Image quality measure**

Measuring image quality is very significant for many image processing applications such as compression, retrieval, transmission, and recognition [5]. Image quality measures can be classified into two classes: subjective methods, and objective methods. In subjective methods human subjects are utilized to evaluate the visual quality of the images. Such methods are reliable and they give better understanding of image quality perception [6, 7]. There are three methods to assess the image quality subjectively: Single Stimulus (SS) method, Quality Ruler (QR) method, and Mean Opinion Score (MOS) method. In the Single Stimulus method, a set of stimuli, including the reference image, is taken one at a time.



Observers assess the quality in terms of a numerical category rating. The quality measure is the average score per stimulus. Because of the inconsistency of this method, another image quality assessment method was introduced. This quality measure is called the quality ruler method [7].

The Quality ruler (QR) method includes a number of reference images. The scale values of these reference images are already known and the reference images are spaced in quality. The observer identifies the reference image which is closest in quality to the test stimulus by visual matching. It is more consistent than the SS method. In addition it is strongly correlated to the objective measure of distortions than the SS scores [6, 7].

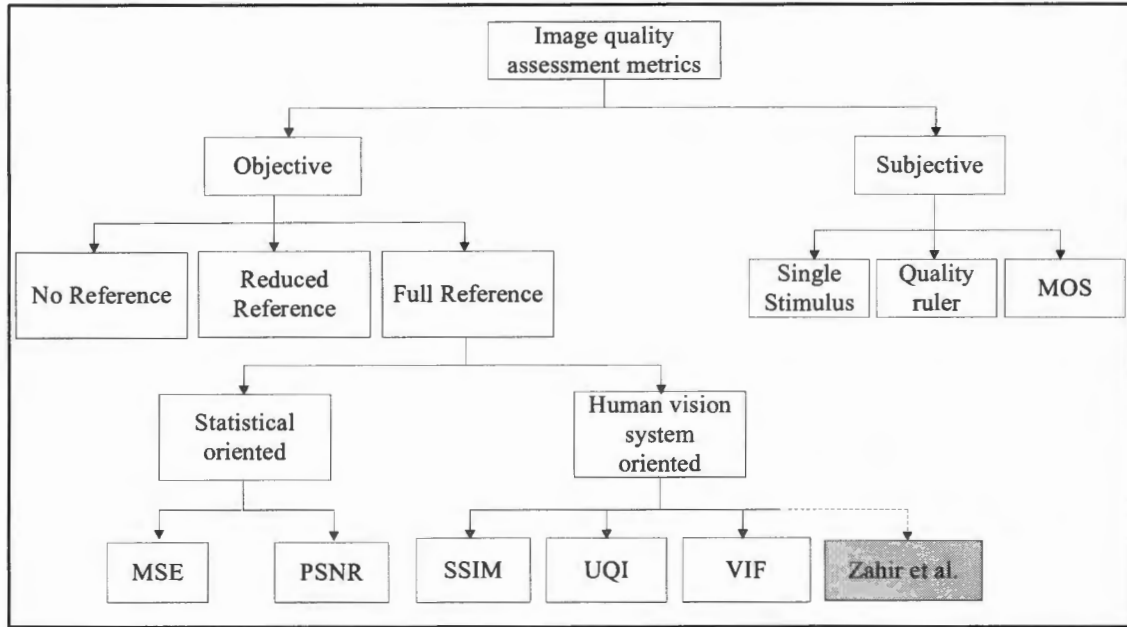
In mean opinion score (MOS), the scores are given by different individuals. Each observer scores the quality of the image by assigning an integer value between 1 and 5 (i.e. very poor (1), excellent (5)). By averaging the scores from different observers for the same image, the MOS can be calculated [8, 9].

The main disadvantages of subjective assessment are that they are time consuming and cannot be performed in real time [6]. Therefore, researchers started to develop objective image quality measures that estimate the quality automatically to avoid the issues of the subjective methods. Objective metrics can estimate the image quality automatically, without the need of the observers. Objective metrics can be categorized, depending on the availability of the reference image, into three categories: no reference (NR) image quality assessment metrics, reduced reference (RR) image quality assessment metrics, and full reference (FR) image quality assessment metrics.

In some applications, such as image denoising, the reference image is not available. Therefore, developing No Reference (NR) image quality assessment metrics is important. No reference (NR) image quality assessment metrics are more efficient than other image quality methods in some applications such as image denoising. In this case, it is required that the type of distortion be known prior to applying the no reference image quality assessment metrics. This is the limitation of these methods [6].

In reduced reference image quality assessment metrics, only partial information about the reference image is required to assess the image quality [6]. In these methods the researchers are seeking to reduce the amount of the data required from the reference image to measure the image quality.

In this work, we focus on full reference image quality assessment metrics in which the reference and the distorted images should be available to evaluate the quality of the distorted image. There are several developed full reference image quality assessment methods such as mean squared error (MSE), peak signal to noise ratio (PSNR), visual information fidelity (VIF) [10], universal quality measure (UQI) [11], structural similarity (SSIM) [12], and Zahir et al. algorithm [4, 13]. Fig. 2.1 shows the main methods of image quality assessment metrics.



**Fig. 2.1.** Image quality metrics

## 2.2 Literature review on image quality measures

In recent years, great efforts have been made to develop objective image quality metrics that correlate strongly with the human visual system (HVS). Objective image quality metrics, depending on the availability of the reference image, can be classified into no-reference (NR) image quality assessment metrics, reduced-reference (RR) image quality assessment metrics, and full-reference (FR) image quality assessment metrics. Full-reference image quality assessment methods require the availability of both the reference and distorted image, while no-reference image quality assessment methods do not require any access to the reference image. It only uses the distorted image to measure its quality. In reduced-reference image quality assessment metrics, only partial information extracted from the reference image is available to assess the quality of the distorted image [4, 14].

### 2.2.1 Statistics oriented methods

Gonzalez et al. in [15] presented Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) to present the signal quality of the distorted signal with respect to the original\reference image. MSE can be defined as follows:

$$\text{MSE}(x, y) = \frac{1}{N} \sum_{i=1}^N (x_i - y_i)^2 \quad (2.1)$$

where  $N$  is the number of samples,  $x$  is the original signal,  $y$  is the distorted version, and the error signal is  $e_i = x_i - y_i$ , where  $e$  is the difference between the original signal ( $x$ ) and distorted signal ( $y$ ). If one of the signals is an original signal of acceptable quality, and the other is a distorted version, then the MSE may also be used as a measure of signal quality (i.e. image quality).

Another method to calculate the image quality is the Peak signal-to-noise ratio (PSNR). PSNR uses the Mean squared error (MSE) to compute the distortion in the distortion version by applying the following formula:

$$\text{PSNR} = 10 \log_{10} \frac{L^2}{\text{MSE}} \quad (2.2)$$

where  $L$  is the dynamic range of allowable image pixel intensities. The PSNR is useful if the compared images have different dynamic ranges [15, 16]. MSE and PSNR are very simple and easy to implement due to their low computational complexities. But on the other hand, MSE and PSNR fail to estimate the image quality when they are used to measure across distortion types [17]. Therefore, the researchers started to develop image quality metrics that consider the human vision system.

## 2.2.2 Human Vision System (HVS) oriented methods

The aim of the HVS-based image quality assessment is to evaluate how strong the distorted information is identified by HVS. A number of image quality assessment metrics (IQA) based on HVS have been introduced to evaluate the perceptual quality [18]. In the following section, the popular HVS-based IQA methods are reviewed.

### 2.2.2.1 Visual Information Fidelity (VIF)

Sheikh et al. [10] developed visual information fidelity (VIF) criterion. VIF measures the Shannon information in the distorted image relative to the information in the reference image by using the Natural Scene Statistics (NSS) modeling. In this method the source and destination models are generated from the reference and the distorted scenes respectively. The natural scene model used in this method is Gaussian scale mixture (GSM) model in the wavelet domain. They used certain bands from reference and distorted images to calculate the quality of the distorted image. Therefore, VIF is considered to be a semi referenced image quality based algorithm. The mutual information between the reference and the distorted models will be calculated using variance of the internal neuron noise that is provided by the user. The visual information fidelity in their model is calculated as follows:

$$\text{VIF} = \frac{\sum_{j \in \text{subbands}} I(\bar{C}^{N,j}; \bar{F}^{N,j} | S^{N,j})}{\sum_{j \in \text{subbands}} I(\bar{C}^{N,j}; \bar{E}^{N,j} | S^{N,j})} \quad (2.3)$$

The numerator and denominator are the information extracted from the reference and distorted images respectively. They tested the performance of VIF algorithm using 779 images. The experimental results showed that the VIF method is comparable with the state-of-the-art methods.

### 2.2.2.2 Universal Quality Measure (UQI)

Wang et al. [11] developed a full referenced image quality assessment algorithm to calculate the image quality by using the reference and the distorted images. In this algorithm they used a sliding window of size 8x8 pixels which will slide the 8x8 pixels of distorted image on the corresponding areas of the reference image and calculate Q as follows:

$$Q = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \cdot \frac{2 \bar{x} \bar{y}}{(\bar{x})^2 + (\bar{y})^2} \cdot \frac{2 \sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2} \quad (2.4)$$

where

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$$

$$\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2, \quad \sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \bar{y})^2$$

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})$$

where x is the reference image and y is the distorted image. This will provide a map of Qs and the average value of this map will provide the quality measure as follows:

$$Q = \frac{1}{M} \sum_{j=1}^M Q_j \quad (2.5)$$

where M is the number of steps depending on the size of the image. The higher values of Q may indicate a better quality for the distorted image. The experimental results showed that this method performs better than the mean squared error metric.

Toet et al. [19] extended the universal gray scale image quality index to a newly developed perceptually de-correlated color space. The resulting color image fidelity metric

measures the distortion of a processed color image compared to its original version. In their paper they defined the color fidelity metric  $Q_{color}$  as follows:

$$Q_{color} = \sqrt{w_l(Q_l)^2 + w_\alpha(Q_\alpha)^2 + w_\beta(Q_\beta)^2} \quad (2.6)$$

where  $Q_l, Q_\alpha$ , and  $Q_\beta$  represent, respectively, the fidelity factors computed for each of the individual  $l\alpha\beta$  color channels,  $w_l, w_\alpha$ , and  $w_\beta$  are the corresponding weights attributed to the perceived distortions. They evaluated their method through observer experiments in which subjects ranked images according to perceived distortion. The results showed a strong correlation between their index and human perception. In addition, the metric is computationally simple, which makes that metric useful in the real-time implementation.

### 2.2.2.3 Structural Similarity (SSIM)

Wang et al. [12] developed a full referenced image quality assessment metric by constructing the Structural Similarity (SSIM) quality measure method. In this method, Wang et al. improved the universal image quality model (UQI). This algorithm uses 8x8 windows and will make a map of the SSIM values as shown in equation 2.7.

$$SSIM(x, y) = \frac{(2\mu_x \mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2.7)$$

where  $C_1 = (K_1 L)^2$ , and  $C_2 = (K_2 L)^2$

where  $L$  is the dynamic range of the pixel values (255 for 8-bit gray scale images), and  $K_1, K_2 > 1$  are constants. The average of the obtained SSIM map will provide the quality measure as shown in equation 2.8.

$$\text{MSSIM}(X, Y) = \frac{1}{M} \sum_{j=1}^M \text{SSIM}(x_j, y_j) \quad (2.8)$$

where  $X$  is the reference image and  $y$  is the distorted image. The experimental results on 344 JPEG and JPEG2000 compressed images showed that it is competitive with other quality measure indices.

Wang et al. in [20] proposed a multi-scale structural similarity (MS SSIM) approach for image quality assessment, which provides more flexibility than single scale approach. Experimental results showed that with appropriate parameter settings, the multi-scale method's results are better than the best single-scale SSIM model as well as state-of-the-art image quality metrics. The most challenging problem in structural similarity based algorithms is calibrating the model parameters. In their paper they used an image synthesis approach to calibrate the parameters. The improvement from single-scale to multi-scale methods observed in their tests suggests the usefulness of this approach.

Gu et al. [21] applied the SSIM function to compensate itself to develop an improved full-reference image quality assessment model based on structure compensation (SC). They examined the proposed method on Laboratory for Image and Video Engineering (LIVE) database and Tampere Image Database 2008 (TID2008). The experimental results confirmed that their introduced approach has better prediction performance as compared to state-of-art image quality metrics. It has an effective capability of image distortion classification. The results showed that, after applying different categories of distortions, the proposed SC-SSIM method gets higher prediction accuracy than SSIM, MS-SSIM, and VIF algorithms on LIVE and TID2008 databases. The authors claimed that their proposed method can be used as a



categorical indicator to effectively distinguish different image distortion types, and the SC-SSIM is fast and has low computational complexity.

#### 2.2.2.4 Copula based image quality measure algorithm

Al Zahir et al. [4, 13] developed new image quality indices by using the Gaussian, Marshall-Olkin, and Clayton copulas based mutual information, which refer to the measure of the dependence between the reference and the distorted images. In Al Zahir et al. [4, 13], they calculated the Gaussian and Marshall-Olkin copula based mutual information by the following simplified expressions [22, 23]:

$$MI_{Gau} = -\frac{1}{2}\ln(1 - \rho^2) \quad (2.9)$$

$$M(X1, X2) = 2\frac{1 - \theta}{2 - \theta}\log(1 - \theta) - \frac{\theta}{2 - \theta} + \frac{\theta^2}{(2 - \theta)^2} \quad (2.10)$$

where  $MI_{Gau}$  is Gaussian copula based mutual information,  $\rho$  is the Pearson correlation between the reference and the distorted images, and  $\theta$  is the Marshall-Olkin dependency parameter ( $0 \leq \theta < 1$ ).

In Al Zahir et al. [13], they calculated the image quality by applying the Clayton copula based mutual information. They used the following expression of Clayton copula probability density function (PDF) [3]:

$$c(x, y) = (1 + \theta)x^{-1-\theta}y^{-1-\theta}(-1 + x^{-\theta} + y^{-1-\theta})^{-2-\frac{1}{\theta}} \quad (2.11)$$

where  $x$  and  $y$  are random variables,  $\theta$  is Clayton dependency parameter where ( $0 < \theta < \infty$ ).

The relationship between Clayton copula dependency parameter  $\theta$  and Kendalls rank correlation  $\tau$  is given by:

$$\theta = \frac{2\tau}{1 - \tau} \quad (2.12)$$

where  $\tau$  is Kendall's tau ( $0 \leq \tau < 1$ ).

The main advantage of the work by Al Zahir et al. [4, 13] is the ability to measure the image quality by using only one sub band (i.e. sub band 4) instead of using different sub bands (i.e. 8 sub bands) as in Sheikh et al. work [10]. By comparing their results with three popular image quality measures, Visual Information Fidelity (VIF); Structural Similarity (SSIM); and Universal Quality Measure (UQI), they concluded that their proposed quality measure has a low computational complexity. Moreover, it obtained comparable results with the three methods.

### 2.3 Image forgery

Although, a picture may worth a thousand words, it may have scores of interpretations. Some proverbs like '*seeing is believing*', are not relevant nowadays. Images and videos can be easily forged with a variety of editing tools. Owing to such sophisticated digital editing software tools, the establishment of the authenticity of an image has become a challenging task, involving a variety of issues. [24]

Digital image forensics is a field that analyses images of a particular scenario to establish credibility and authenticity for the digital images. It is a fast growing field because of its potential applications in many fields, such as sports, legal services, news reporting, and medical imaging [24, 25, 26]. Image forgery started as early as 1840s, when Hippolyte

Bayrad introduced the first fake image as shown in Fig. 2.2, in which he was shown committing a suicide [24, 27].



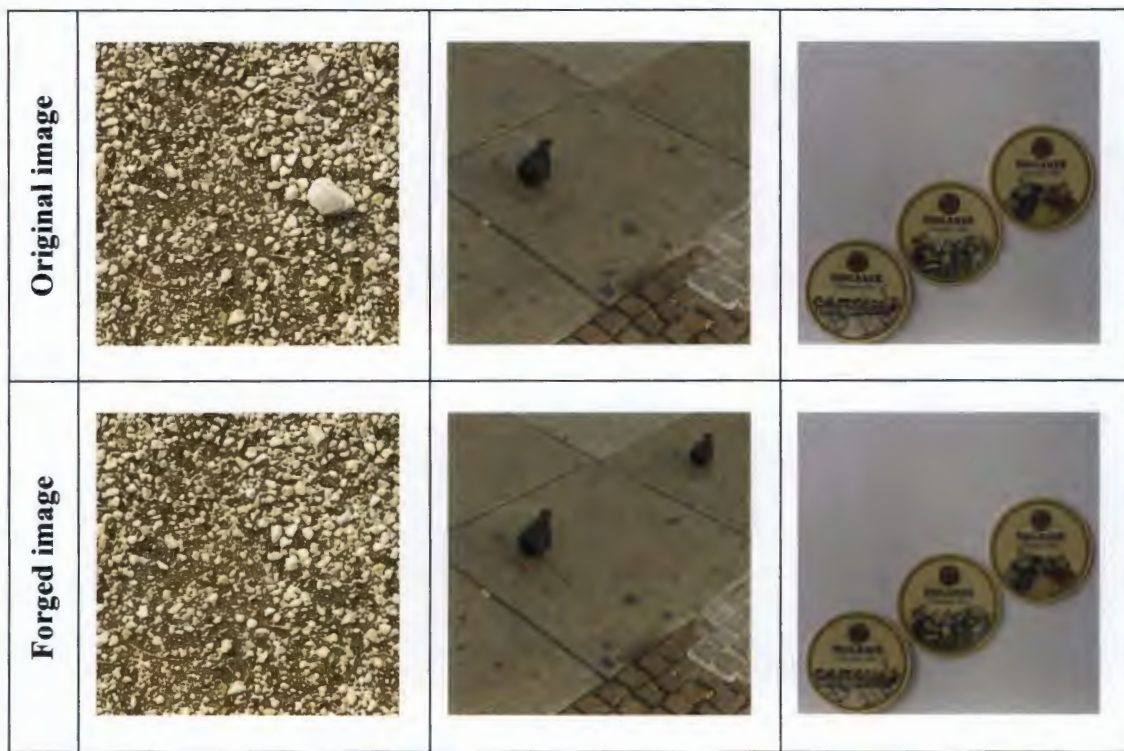
**Fig. 2.2.** Bayard's self portrait as a drowned man (early 1840s)

There are three main categories of image forgery (a) copy-move forgery, (b) image splicing, and (c) image retouching. In this thesis, we focus on copy-move forgery. For further information about image splicing and image retouching techniques see [24, 28, 29, 30, 31, 32, 33, 34] (i.e. Appendix A).

### **2.3.1 Copy-Move (Cloning) forgery**

Copy-move or cloning forgery (CMF) is one type of the techniques used for tampering images where a region of an image is copied from one part and is moved to another part in the same image. Different manipulation techniques like rotation, scaling, etc. can be applied on the copied region to be suited with the entire image. The tampered image

may be manipulated using compression, noise, rotation, etc. to make it hard for the human eyes to discover the forgery. This makes CMF detection task hard and challenging [24, 32]. CMF has remained a growing field and more research is required to be performed [35]. The researchers have introduced different methods to detect CMF. Fig. 2.3 shows three original images and their forged versions after applying the copy-move forgery technique.



**Fig. 2.3.** Sample images that show copy-move forgery technique

## 2.4 Literature review on copy-move forgery detection

Copy-move forgery detection methods may either be brute force, or block based [24]. The 'brute force' techniques involve an exhaustive search that covers a given image with circularly shifted versions to examine matching segments [36]. The disadvantages of applying brute force techniques is that exhaustive search is not that effective when some post-processing techniques are applied to the copied area before pasting it [36]. In contrast, block-based matching techniques give better results than the exhaustive search. Block-based matching techniques include two categories. The first category is the exact matching techniques and the second one is the approximate matching category. In the exact matching techniques, the matched regions should be identical (i.e. 100% matching) to identify forged region. Although this approach is easy to implement, it fails to identify many forged regions given that the forged areas are usually post-processed and may not maintain the original values. In contrast, approximate block matching can be a better option to identify the forged regions even if the forged region is post-processed. A typical approximate block matching technique divides the image into overlapping blocks and applies a suitable technique to extract features to determine similarity [24]. Great effort has been done on this topic, and several researchers discussed and introduced new techniques to detect the copy-move forgery in digital images. In this section we review some copy - move forgery detection techniques.

Y. Cao et al in [37] proposed a robust method relied on improved Discrete Cosine Transform technique (DCT) to locate the duplicated regions in a given image. The proposed method used fewer numbers of features for representing every block to detect copy-move forgery. The method used the circle block for representing DCT coefficient's array; circle block represented most of the coefficients of the array and discarded a few of them. The

circle block was divided into four parts ( $Cb1, Cb2, Cb3, \text{ and } Cb4$ ). The feature vector for each block was  $V = [vb1, vb2, vb3, \text{ and } vb4]$ .  $Cbi$  is represented by DCT coefficients. The feature vectors for all blocks are arranged into an array  $A$ , and it was lexicographically sorted. After that, they compute the Euclidian distance ( $E_D$ ) between the adjacent vector pairs of the array  $A$ . If it was smaller than predefined threshold ( $S_T$ ), so the adjacent vector pairs were similar and candidates for the forgery, then a block map  $B$  was initialized. Also actual distance ( $D$ ) between the two similar blocks was calculated as follows:  $(Vi, +j) = (ai - ai + j)^2 + (bi - bi + j)^2$  where  $(a, b)$  was the circle center of the corresponding block. If  $E_D < S_T$  and  $D > N$  where  $N$  was the threshold distance, then, a color map was marked for the actual region. The proposed method was tested on three databases: DVMM dataset, different uncompressed PNG color images with  $768 \times 512$  size, and collected images from the internet with large resolution ( $1600 \times 1000$ ). The proposed method could identify further than 80% of the regions in copy-move areas with fewer than 13% of false-positive rate.

Zhao et al. in [38] used chrominance spaces with run-length run-number (RLRN) for copy-move forgery detection. The input color image was transformed into the YCbCr color mode. Then run-length run-number was used to extract the features from the de-correlation of the chrominance channels. First fifteen run-length run-numbers of each de-correlated image were extracted as a feature vector. Support vector machine was used for classification purpose. This method obtained 94% accuracy of the chrominance components on CASIA dataset while it obtained 85% with Cr component and 82.1% with Cb component on DVMM dataset.

Wei Wang et al. [39] used the chrominance components of the color image to detect image tampering. Their method used edge information to extract feature vector which was obtained by applying a filter on the image chrominance components using convolution operation. The edge image is thresholded with  $Th$  threshold value. Then the Markov chain (MC) is used to model the thresholded edge image and one-step transition probability matrix (TPM) is used to characterize the finite-state Markov chain. The stationary-distribution for a finite-state Markov chain (MC) is used as a feature vector with the low dimension  $Th+1$ . Support vector machine (SVM) was used for classification. The authors used CASIA TIDE database for the detection method. They used 5,123 tampered and 5,123 authentic images. The accuracy reached up 95.6% and 95.5% with Cb chrominance component and with Cr when  $Th$  equal to 15 respectively. This method used feature vector with a low dimension. It depended on the statistical dependency between two adjacent pixels that could be characterized using one-step transition probability matrix of Markov chain.

Sridevi et al. [40] proposed a parallel block matching algorithm to detect the forged regions for the copy – move forgery. The method uses overlapping blocks and lexicographical sorting to detect the copy-move forgery. The results showed that the proposed parallel version detects the forged region faster. Their proposed algorithm reduced the execution time, so that it is best suited for real time applications. The false detection rate enables them to decide the correct size of overlapping blocks for accurate detection.

Christlein et al. [41] introduced a common pipeline for copy-move forgery detection by extracting Fourier-Mellin features. In addition they performed a comparative study on 10 proposed copy-move features, and finally they introduced a new benchmark database for copy-move forgery detection. The dataset contains images and copy-moved regions of



varying size and texture. Experimental results showed that their method performs exceptionally if no geometric transformations are applied to the copied region. Furthermore, their experiments strongly support the use of kd-trees for the matching of similar blocks instead of lexicographic sorting.

Popescu et al. [42] presented an efficient technique that automatically detects duplicated regions in a digital image. Their technique works by applying a principal component analysis (PCA) to small fixed size image blocks to represent the image by reduced dimensions. Duplicated regions are then detected by lexicographically sorting all of the image blocks. This representation is robust to minor variations in the image due to additive noise or lossy compression. The experimental results showed the effectiveness of this technique on reasonable forgeries.

Sarode et al. [43] proposed a copy move forgery detection algorithm. Their algorithm generates a Hybrid Wavelet Transform (DCT-Walsh Hybrid Wavelet Transform) from discrete cosine transform (DCT) and Walsh transforms and used to detect copy-move forgery. The image is divided into overlapping blocks. DCT-Walsh Hybrid wavelet transform is applied on each block. Then some features are extracted from each block. These feature vectors are then lexicographically sorted and finally, block matching step is applied to detect duplicated blocks. Experimental results show that these transforms are able to detect duplicated regions with more accuracy than using basic orthogonal transformations. This method also detects forged regions even when some post-processing techniques, such as blurring and edge sharpening operations, are applied on the image. The main disadvantage of their proposed method is that it does not detect duplicated regions when copied part in an image is rotated or scaled.



Khan et al. [44] introduced a blind forensics approach for detecting Copy-Move forgery. First they applied Discrete Wavelet Transform (DWT) to the input image to yield a reduced dimension representation [45]. After that, the compressed image is split into overlapping blocks. These blocks are then sorted. Finally they detected the duplicated blocks by using Phase Correlation. Due to DWT usage, detection is first carried out on lowest level image representation. Their proposed approach reduces the time needed for the detection process. Their algorithm has low computational complexity. Moreover, this algorithm succeeded in detecting the forgery even for the images where the attacker has made detection more difficult by applying noise and JPEG quality level changes. Experiments and analysis proved that their proposed method have acceptable robustness to common post processing operations. The main problem with their algorithm is that it failed to detect duplicated regions with rotation and scaling.

Warbhe et al. [46] presented a method based on statistical techniques, Independent Component Analysis (ICA), and Blind Source Separation (BSS), to detect the copy-move forgery in digital images. The results of this method showed the effectiveness of using ICA for image forgery detection in digital image. Although, the performance of their proposed method is very good at detecting the forgery, the main limitation of this method is that it needs both the original and the forged.

## **Chapter 3**

### **Image Quality and Dissimilarity Verification Copula-Based Algorithms**

Image quality measurement is widely used in different image processing applications such as transmission, recognition, retrieval, classification, and compression [5]. As mentioned above, image quality measurements can be classified into two different classes namely: subjective methods, and objective methods.

In this chapter, we propose objective full reference image quality measure (FRIQM) that uses a group of statistical functions called copulas. Our proposed algorithm applies the steerable pyramid technique to decompose the original image and the distorted version. After the decomposition stage of the original and distorted images, we use copula functions (Gaussian, Marshall-Olkin, Clayton, Frank, and Gumbel copulas) to evaluate the quality of the distorted version. We used 'LIVE' database to test our proposed algorithm. Moreover, we presented an algorithm to detect unmatched regions between the images. The experimental results showed that our algorithm provide comparable results with popular image quality measures such as Visual Information fidelity (VIF), Universal image Quality Index (UQI), Structural Similarity image quality index (SSIM), and the most used subjective method based on HVS. In the next section, before proposing our algorithms, we briefly introduce copula

functions, mutual information, Pearson coefficient, Kendall's coefficient, and the steerable pyramid technique.

### **3.1 Copula based mutual information**

#### **3.1.1 Copula functions overview**

Copula is derived from the Latin word 'copulare', which means join, connect, or tie. Copula is widely known as a family of distribution functions. Copula functions are used in various applications such as economics and finance, climate research, oceanography, hydrology, geodesy, evolutionary computation, and image processing applications such as image registration [3, 47].

The main idea behind copula theory is that the cumulative distribution function (CDF) of a random vector can be represented by uniform marginal cumulative distribution functions, and a copula that connects these marginal cumulative distribution functions [2]. Sklar's theorem (1959) [1] shows that an n-dimensional joint distribution function can be decomposed into its n marginal distributions, and a copula, which completely describes the dependence between the n variables. The Sklar theorem is given by:

For any two random variables X and Y with their joint probability distribution  $F(x, y)$ , there exists a unique copula C such that

$$F(x, y) = C(F_x(x), F_y(y)) = C(u, v) \quad (3.1)$$

where  $C(u, v)$  is the copula function,  $u=F_x(x)$ ,  $v=F_y(y)$  are marginal probability distributions,  $F(x, y)$  is the joint distribution [3]. The copula density function,  $c(u, v)$ , is given by:

$$c(u, v) = \frac{\partial^2 C(u, v)}{\partial u \partial v} \quad (3.2)$$

There are limited publications on the use of copula functions in image quality assessment. There are only two publications in the field of copula based image quality measure [4, 13]. Zahir et al. applied Gaussian, Marshall-Olkin, and Clayton copula functions to measure the image quality of distorted images and then rank these distortion versions according to their quality measure.

In this chapter, we estimate the image quality of the distorted image by computing copula based mutual information between the reference and distorted images.

### 3.1.2 Mutual information and copula functions

Two random variables  $X$  and  $Y$  are said to be independent if, and only if, their joint probability density function (PDF) equals the product of their marginal PDFs. On the other hand,  $X$  and  $Y$  are dependent if  $F(x, y) \neq f_x(x)f_y(y)$ , where  $f_x(x)$  and  $f_y(y)$  are marginal densities and  $F(x, y)$  is the joint probability density function. Estimating mutual information is a convenient way to quantify statistical dependencies. Mutual information can be calculated as follows [2, 48]:

$$I(X, Y) = \int dx dy F(x, y) \log \left[ \frac{F(x, y)}{f_x(x)f_y(y)} \right] \quad (3.3)$$

where  $I(X, Y)$  is the mutual information between the two random variables  $X$  and  $Y$ .

Mutual information and copula densities can be connected by merging equations 3.1, 3.2, and 3.3, to conclude that the formula of the mutual information can be written [2, 9] as follows:

$$I(X, Y) = \iint_{[0,1]^2} du dv c(u, v) \log[c(u, v)] \quad (3.4)$$

The base of logarithm in the previous equation (i.e. equation 3.4) does not matter (i.e. we can choose any base). The different units of information, for example bits for  $\log_2$ , nats for  $\ln$ , are just constant multiples of each other. Therefore equation 3.4 can be written in a different way as follows:

$$I(X, Y) = \iint_{[0,1]^2} du dv c(u, v) \ln[c(u, v)] \quad (3.5)$$

### 3.1.3 Copula function types

There are several copula functions. In this thesis we focus on five of them, namely: Gaussian, Marshall-Olkin, Clayton, Frank, and Gumbel copulas.

#### 3.1.3.1 Gaussian copula

Gaussian copula cumulative distribution function (CDF) [49] is given by:

$$C(u, v) = \int_{-\infty}^{\Phi^{-1}(u)} \int_{-\infty}^{\Phi^{-1}(v)} \frac{e^{-\frac{1}{2}t'\Sigma^{-1}t}}{2\pi|\Sigma|^{1/2}} dt_1 dt_2 ; \theta \in (-1, 1) \quad (3.6)$$

$$\tau = \frac{2}{\pi} \sin^{-1}(\theta) \quad (3.7)$$

where  $u$  and  $v$  are random variables,  $\Phi^{-1}(\cdot)$  is the quantile function of the standard normal cumulative distribution,  $\theta$  is the dependence parameter of Gaussian copula function,  $\tau$  is the Kendall's tau, and  $\Sigma$  is  $2 \times 2$  correlation matrix with 1 on the diagonal and  $\theta$  otherwise. The Gaussian copula probability density function (PDF) [49], by applying equation 3.2, is given by:

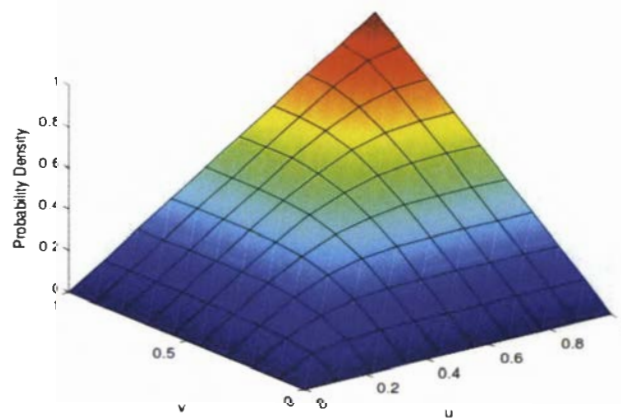
$$c(u, v) = (1 - \theta^2)^{1/2} \exp\left(-\frac{(x^2 + y^2 - 2\theta xy)}{2(1 - \theta^2)} + \frac{(x^2 + y^2)}{2}\right) \quad (3.8)$$

where  $x = \Phi^{-1}(u)$  and  $y = \Phi^{-1}(v)$ .

In this work we use the simplified version of Gaussian copula based mutual information which is calculated by the following equation (i.e. equation 2.9) [22, 48, 50]:

$$MI_{Gau} = -\frac{1}{2} \ln(1 - \rho^2)$$

where  $MI_{Gau}$  is the mutual information, and  $\rho$  is the Pearson correlation between the reference and distorted images. Fig. 3.1 shows probability density function (PDF) for Gaussian copula. The colors on the surface plot (i.e. from blue to red) represent different probability density values for each pair  $u$  and  $v$ .



**Fig. 3.1.** Probability density function for Gaussian copula

### 3.1.3.2 Marshall-Olkin copula

The following equation (i.e. equation 2.10) calculates the Marshall-Olkin copula based mutual information which represents the image quality of the distorted image [13, 23] in our work:

$$M(X1, X2) = 2 \frac{1 - \theta}{2 - \theta} \log(1 - \theta) - \frac{\theta}{2 - \theta} + \frac{\theta^2}{(2 - \theta)^2}$$

where  $\theta$  is the Marshall-Olkin copula dependency parameter ( $0 \leq \theta < 1$ ). The relationship between Marshall-Olkin copula dependency parameter  $\theta$  and Kendall's rank correlation  $\tau$  is given by:

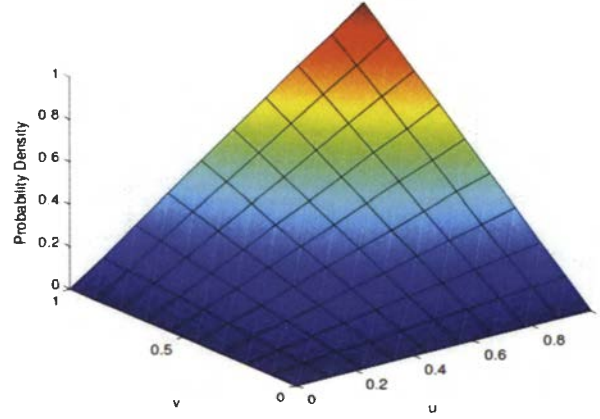
$$\theta = \frac{2\tau}{1 + \tau} \quad (3.9)$$

tau ( $\tau$ ) can be written as a function of copula function as follows:

$$\tau = 4 \int_{[0,1]^2} C(u, v) dC(u, v) - 1 \quad (3.10)$$

$$C(u, v) = \min(u^{1-\theta} v, uv^{1-\theta}) \quad (3.11)$$

where  $\tau$  is Kendall's tau ( $0 \leq \tau < 1$ ),  $C(u, v)$  is the Marshall-Olkin copula cumulative distribution function (CDF). Fig. 3.2 shows the probability density function (PDF) for Marshall- Olkin copula. The colors on the surface plot (i.e. from blue to red) represent different probability density values for each pair  $u$  and  $v$ .



**Fig. 3.2.** Probability density function for Marshall-Olkin copula

### 3.1.3.3 Clayton copula

Clayton copula cumulative distribution function (CDF) [49] is given by:

$$C(u, v) = \max \left\{ (u^{-\theta} + v^{-\theta} - 1)^{-1/\theta}, 0 \right\}; \quad \theta \in [-1, \infty) \setminus \{0\} \quad (3.12)$$

By applying  $c(u, v) = \frac{\partial C(u, v)}{\partial u \partial v}$ , the Clayton copula probability density function (PDF) as mentioned in chapter two (i.e. equation 2.11) is given by [3, 13, 49]:

$$c(u, v) = (1 + \theta) u^{-1-\theta} v^{-1-\theta} (-1 + u^{-\theta} + v^{-1-\theta})^{-2-\frac{1}{\theta}}$$

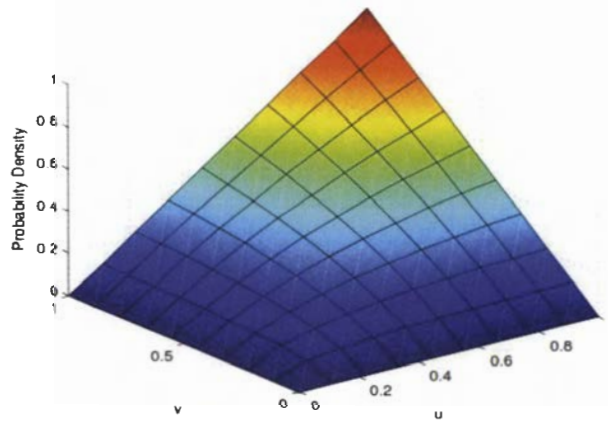
where  $u$  and  $v$  are random variables,  $\theta$  is Clayton copula dependency parameter where  $(0 < \theta < \infty)$ . The relationship between Clayton copula dependency parameter  $\theta$  and Kendall's rank correlation ( $\tau$ ) (i.e. equation 2.12) [49] is given by:

$$\theta = \frac{2\tau}{1 - \tau}$$

where  $\tau$  is Kendall's tau ( $0 \leq \tau < 1$ ). By applying equation 3.4, we can calculate Clayton copula based mutual information. Fig. 3.3 shows the probability density function (PDF) for



Clayton copula. The colors on the surface plot (i.e. from blue to red) represent different probability density values for each pair  $u$  and  $v$ .



**Fig. 3.3.** Probability density function for Clayton copula

#### 3.1.3.4 Frank copula

Frank copula cumulative distribution function (CDF) and Frank copula probability density function (PDF) [49] are given by:

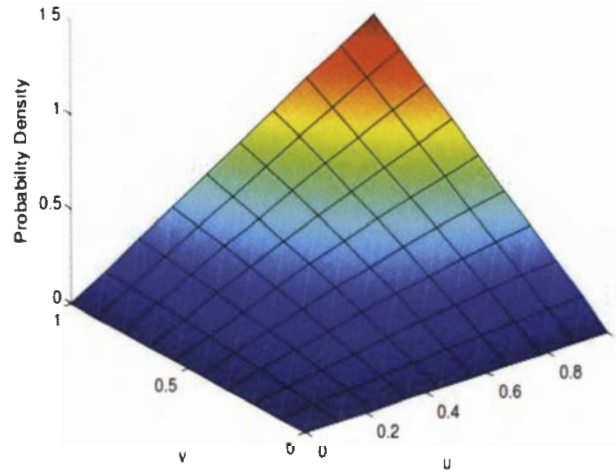
$$C(u, v) = \frac{-1}{\theta} \ln \left( 1 + \frac{(e^{-\theta u} - 1)(e^{-\theta v} - 1)}{(e^{-\theta} - 1)} \right) ; \theta \in (-\infty, \infty) \setminus \{0\} \quad (3.13)$$

$$c(u, v) = \frac{-\theta(e^{-\theta} - 1)e^{-\theta(u+v)}}{((e^{-\theta u} - 1)(e^{-\theta v} - 1) + (e^{-\theta} - 1))^2} \quad (3.14)$$

where  $u$  and  $v$  are random variables,  $\theta$  is Frank copula dependency parameter. The relationship between the Frank copula dependency parameter  $\theta$  and Kendall's rank correlation ( $\tau$ ) [49] is given by:

$$\tau = 1 - \frac{4}{\theta} \left[ 1 - \frac{1}{\theta} \int_0^\theta \frac{t}{e^t - 1} dt \right] \quad (3.15)$$

Fig. 3.4 shows the probability density function (PDF) for Frank copula. The colors on the surface plot (i.e. from blue to red) represent different probability density values for each pair  $u$  and  $v$ .



**Fig. 3.4.** Probability density function for Frank copula

### 3.1.3.5 Gumbel copula

Gumbel copula cumulative distribution function (CDF) and Gumbel copula probability density function (PDF) [49] are respectively given by:

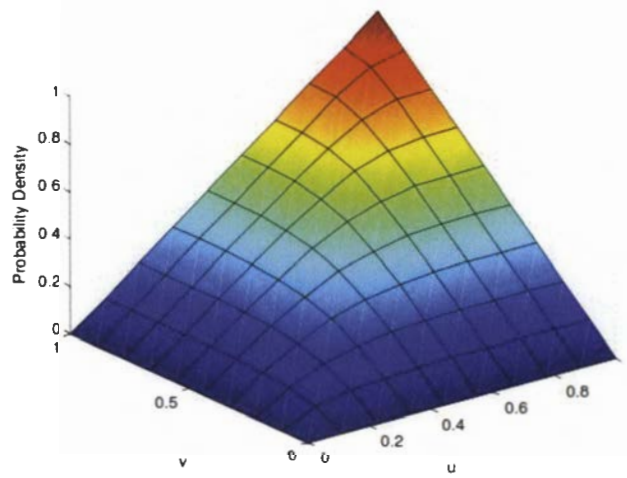
$$C(u, v) = \exp\left(-(\tilde{u}^\theta + \tilde{v}^\theta)^{1/\theta}\right); \quad \theta \in [1, \infty) \quad (3.16)$$

$$c(u, v) = \frac{C(u, v)}{uv} \frac{(\tilde{u}\tilde{v})^{\theta-1}}{(\tilde{u}^\theta + \tilde{v}^\theta)^{2-1/\theta}} \left( (\tilde{u}^\theta + \tilde{v}^\theta)^{1/\theta} + \theta - 1 \right) \quad (3.17)$$

where  $\tilde{u} = -\ln(u)$  and  $\tilde{v} = -\ln(v)$ , and  $u$  and  $v$  are random variables,  $\theta$  is Gumbel copula dependency parameter. The relationship between Frank's copula dependency parameter  $\theta$  and Kendall's rank correlation ( $\tau$ ) is given by:

$$\tau = 1 - \frac{1}{\theta} \quad (3.18)$$

Fig. 3.5 shows probability density function (PDF) for Gumbel copula. The colors on the surface plot (i.e. from blue to red) represent different probability density values for each pair  $u$  and  $v$ .



**Fig. 3.5.** Probability density function for Gumbel copula

### **Pearson coefficient versus Kendall's coefficient**

Pearson coefficient,  $\rho$  (rho), was developed by Karl Pearson from a related idea introduced by Francis Galton in the 1880s [51, 52, 53, 54]. Pearson coefficient is a measure of the linear correlation (i.e. dependence) between two variables  $X$  and  $Y$ . Pearson coefficient value is between  $+1$  and  $-1$  (i.e.  $+1 \geq \rho \geq -1$ ), where  $1$  refers to total positive correlation,  $0$  refers to no correlation, and  $-1$  refers to total negative correlation.

For random variables  $X$  and  $Y$ , the linear correlation coefficient is defined by:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \quad (3.19)$$

where  $cov$  is the covariance, and  $\sigma_X, \sigma_Y$  are the standard deviations of  $X$  and  $Y$  respectively.

On the other hand, Maurice Kendall [55], developed Kendall's tau coefficient ( $\tau$ ) in 1938. Kendall's tau coefficient ( $\tau$ ) is used to measure the similarity of the orderings of the data when ranked by each of the quantities (i.e. rank correlation).

For two random variables  $X$  and  $Y$ , assume that there are  $n$  pairs of observations of the joint random variables  $X$  and  $Y$  respectively (i.e.  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ ), such that the values of  $(x_i)$  and  $(y_i)$  are unique. In this case, any pair of observations  $(x_i, y_i)$  and  $(x_j, y_j)$  are said to be concordant if the ranks for both elements agree: with different way, if both  $x_i > x_j$  and  $y_i > y_j$  or if both  $x_i < x_j$  and  $y_i < y_j$ . On contrast, they are said to be discordant, if  $x_i > x_j$  and  $y_i < y_j$  or if  $x_i < x_j$  and  $y_i > y_j$ . However, If  $x_i = x_j$  or  $y_i = y_j$ , then the pair is neither concordant nor discordant [56, 57].

Kendal tau is given by [56]:

$$\tau_K = \frac{(C - D)}{n(n-1)/2} \quad (3.20)$$

where  $C$  is the number of concordant pairs and  $D$  is the number of discordant pairs and  $n$  is the number of observations. The Kendal tau value is in the range  $-1 \leq \tau \leq 1$ .

- If the two rankings are the same, then the coefficient will be 1.
- If one ranking is the reverse of the other, then the coefficient will be  $-1$ .

- If X and Y are independent, then the coefficient will be approximately zero.

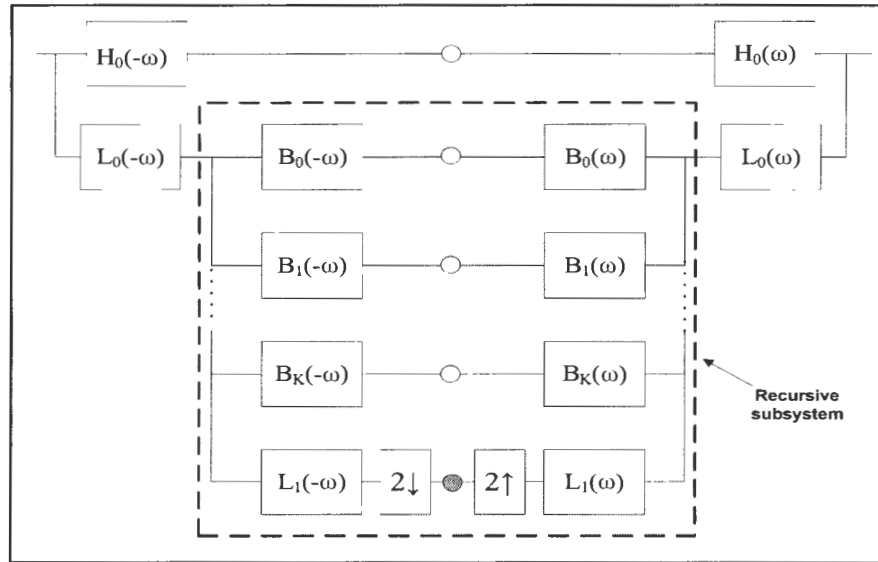
### 3.2 Steerable pyramid

Steerable pyramid is one of image pyramid techniques. The goal of applying image pyramids is to represent the image at different resolutions. The idea behind image pyramids is to generate a number of homogeneous coefficients. These coefficients represent the response of a bank of filters on the image at different scales and in sometimes different orientations. There are various pyramid techniques in image processing such as; Gaussian, Laplacian, and Steerable pyramids. In this work, we use the steerable pyramid technique. For further information about Gaussian pyramid and Laplacian pyramid see [58, 59, 60, 61] (i.e. Appendix A).

The steerable pyramid is a linear multi-scale, multi-orientation image decomposition technique that provides a useful front-end for image-processing and computer vision applications. Steerable pyramids were developed in 1990. The basis functions of the steerable pyramid are  $K^{\text{th}}$  order directional derivative operators that come in different sizes and orientations ( $K+1$  orientations). As directional derivatives, they span a rotation-invariant subspace, and they are designed such that the whole transform form a tight frame.

Fig. 3.6 shows the filtering and the sampling operations, and the recursive construction of the steerable pyramid. Initially, by using low pass filter (L0) and high pass filter (H0), the image is separated into low and high pass sub bands. Then the low pass sub band is divided into a set of oriented band pass sub bands and a low(er)-pass sub band. This

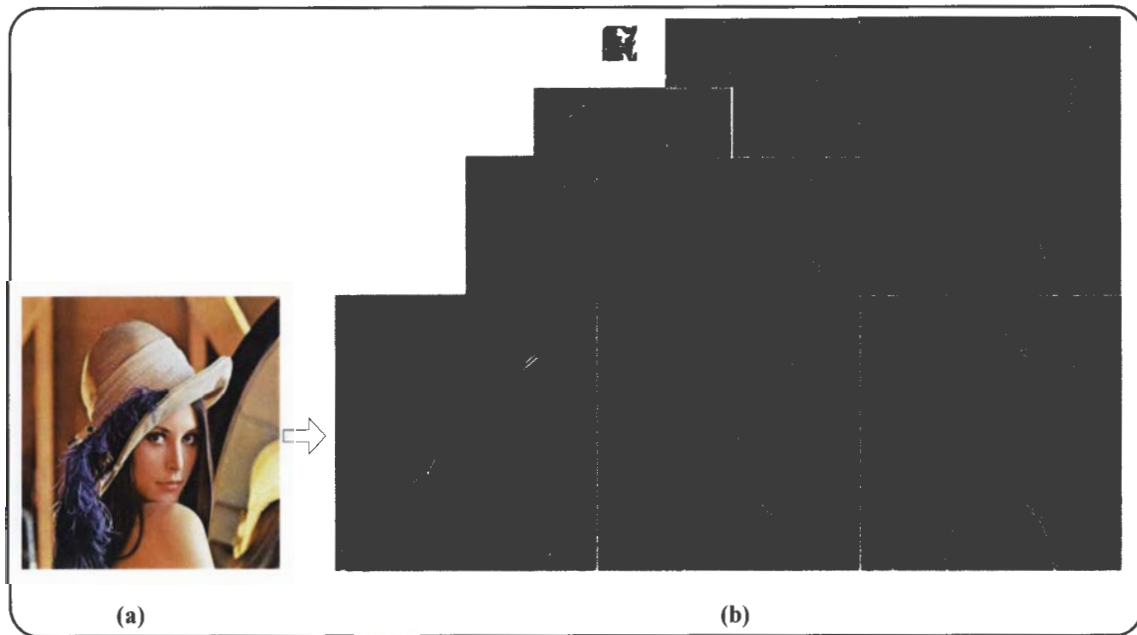
low(er)-pass sub band is then sub-sampled by a factor of 2 in both X and Y directions. The recursive construction of a pyramid is achieved by inserting a copy of the shaded portion of the diagram at the location of the solid circle (i.e., the low pass filter branch) [62]. The others circle refers to the resulted sub bands. The blocks  $2\downarrow$  and  $2\uparrow$  refers to down-sampling and up-sampling respectively. As we can notice from Fig. 3.6 that steerable pyramid design apply  $H_0(-\omega)$  and its space-reversed version  $H_0(\omega)$  as well. Also for the low pass filter, it uses  $L_0(-\omega)$  and its space-reversed version  $L_0(\omega)$ , and finally for the band pass filters  $B_K(-\omega)$  and its space-reversed version  $B_K(\omega)$ . The use of these space-reversed versions creates a self-inverted pyramid (i.e. steerable pyramid is self-inverting).



**Fig. 3.6.** Steerable pyramid decomposition diagram

Steerable pyramid allows independent representation of scale and orientation. Moreover, the steerable pyramid representation is translation-invariant and rotation-invariant,

which can make a significant difference in applications that include the position or the orientation representations of the image. The main disadvantage of steerable pyramid is that it is over-complete by a factor of  $4k/3$ , where  $k$  is the number of orientation bands. Applications of steerable pyramids include: orientation analysis, noise removal and enhancement, transient detection, texture representation, and edge detection [62, 63, 64]. An example of steerable pyramid decomposition of "Lena" image is shown in Fig. 3.7. As shown in Fig. 3.7 steerable pyramid decomposition consists of 4 orientation sub bands, at 3 scales. The smallest sub band is the residual low pass information. The residual high pass sub band is not shown.

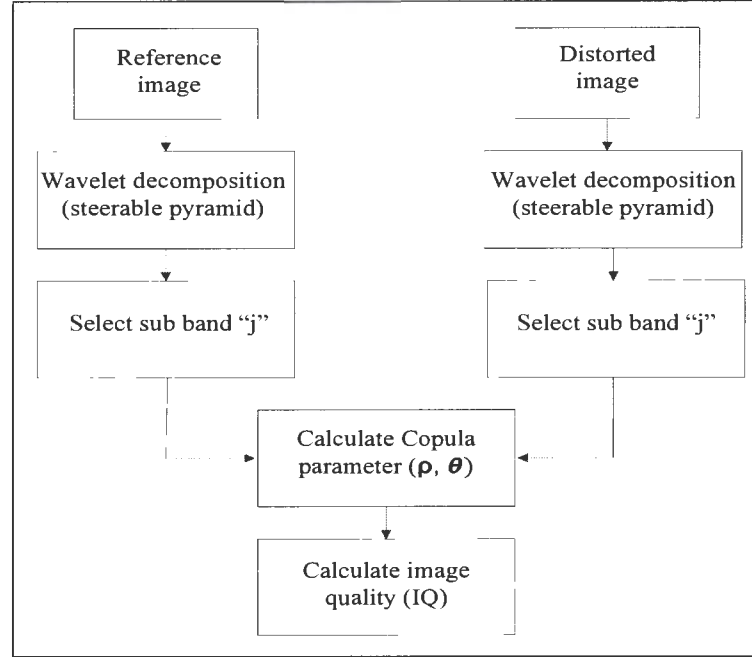


**Fig. 3.7.** Steerable pyramid decomposition on Lena image: (a) original image; (b) the resulted sub bands after applying the steerable pyramid technique on the original image

### 3.3 The proposed algorithm

In our proposed algorithm, we use two versions of images. The first image is the reference image and the second one is the distorted version. Fig. 3.8 shows our proposed algorithm's procedure. First step in our proposed algorithm is reading the two images (i.e. the original image and the distorted version). Then we decompose each image by using steerable pyramid decomposition. The outputs of steerable pyramid are the pyramid sub bands. After obtaining the sub bands, we choose only one sub band (i.e. sub band 4) to calculate the Pearson correlation ( $\rho$ ) and copula dependency parameter ( $\theta$ ). We use the calculated Pearson correlation ( $\rho$ ) to calculate Gaussian copula based mutual information (i.e. equation 2.9) and copula dependency parameter ( $\theta$ ) to calculate Marshall-Olkin copula (i.e. equation 2.10 & 3.9 & 3.10 & 3.11), Clayton copula (i.e. equations 2.11 & 2.12 & 3.4), Frank copula (i.e. equations 3.4 & 3.14), and Gumbel copula (i.e. equations 3.4 & 3.17) based mutual information respectively. The values of mutual information represent the image quality of the distorted image. Higher values of mutual information refer to better quality images.





**Fig. 3.8.** The flow chart of the proposed image quality measure algorithm

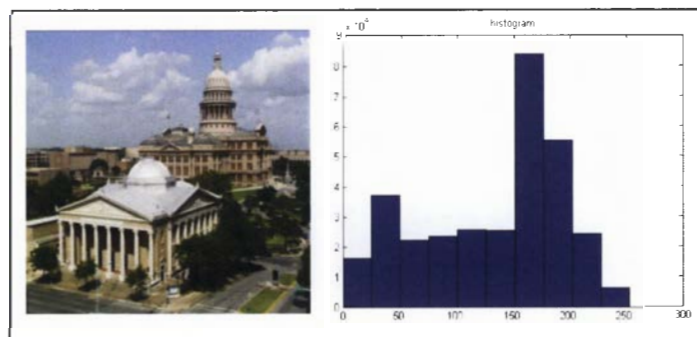
Fig. 3.9 represents one of the original test images (monarch) with its six orientations using the steerable pyramid (note: The first row is the results of a high pass filter (size: 512 x 768); the second row is the steerable pyramids of sub bands 25 to 20 from right to left (size: 512 x 768); the third row is the steerable pyramids of sub bands 19 to 14 from right to left (size: 256 x 384); the fourth row is the steerable pyramids of sub bands 13 to 8 from right to left (size: 128 x 192); the fifth row is the steerable pyramids of sub bands 7 to 2 from right to left (size: 64 x 96) and finally, the sixth row is the result of a low pass filter (size: 32 x 48)). In our work, we apply five copula functions namely: Gaussian, Marshall-Olkin, Clayton, Frank, and Gumbel copulas on one sub band (i.e. sub band 4) of steerable pyramid for both the reference image and the distorted image to measure the quality of the distorted image.

Moreover, we measure the image quality, of the distorted image, on different sub bands to examine the effect of choosing different sub bands (i.e. other than sub band 4) on our results.

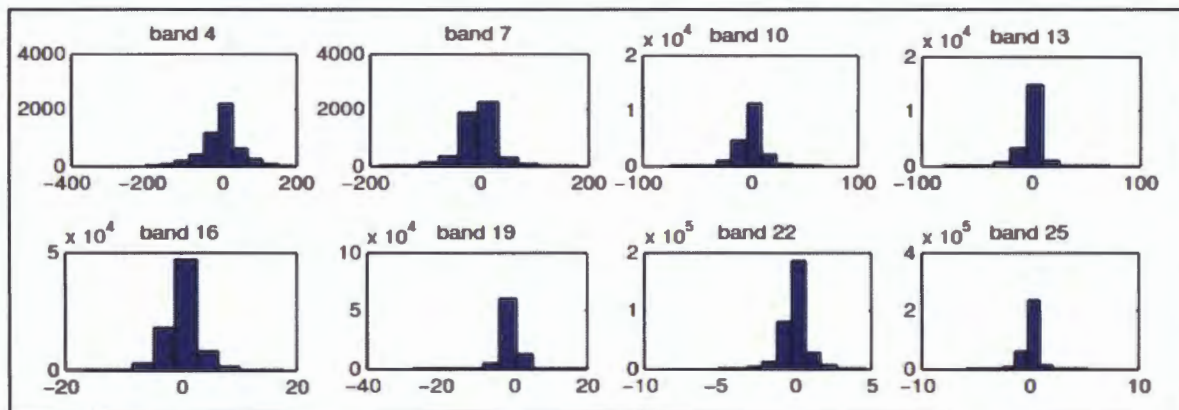
Fig. 3.10 shows the histogram of church and capitol image (LIVE database). Fig. 3.11 and Fig. 3.12 show the histogram of the sub bands 4, 7, 10, 13, 16, 19, 22, 25 of the original and the distorted version of "church and capitol" respectively after applying the steerable pyramid algorithm. In Sheikh et al. work [10] all eight sub bands have been used to calculate the image quality, However in our work, we used sub band number 4 only to calculate the image quality. Fig. 3.13 shows the procedure and the results of measuring image quality for distorted image (church and capitol) in LIVE database by applying the five copula functions. As we can see from Fig. 3.13, we read the original and the distorted images. After reading the images we decompose each one by applying steerable pyramid algorithm to get the sub bands. By choosing only one sub band (sub band 4) we can calculate the Pearson correlation and the Kendall's tau (we use tau ( $\tau$ ) to get the dependence parameter theta ( $\theta$ )). From the value of Pearson correlation ( $\rho$ ) and theta ( $\theta$ ), we can calculate the image quality for our proposed algorithm (each copula function has its own equation).



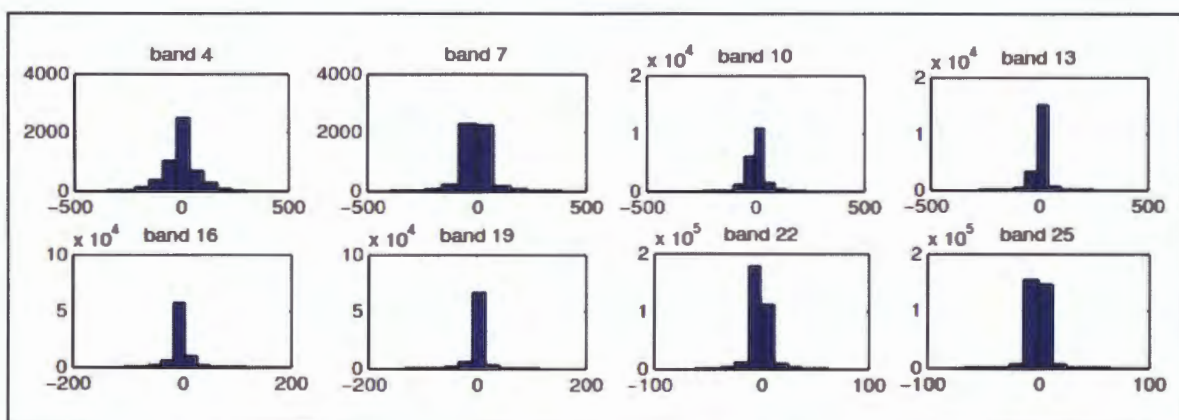
**Fig. 3.9.** Steerable pyramid decomposition: (a) monarch image (LIVE database); (b) the 26 steerable pyramid sub bands of monarch image



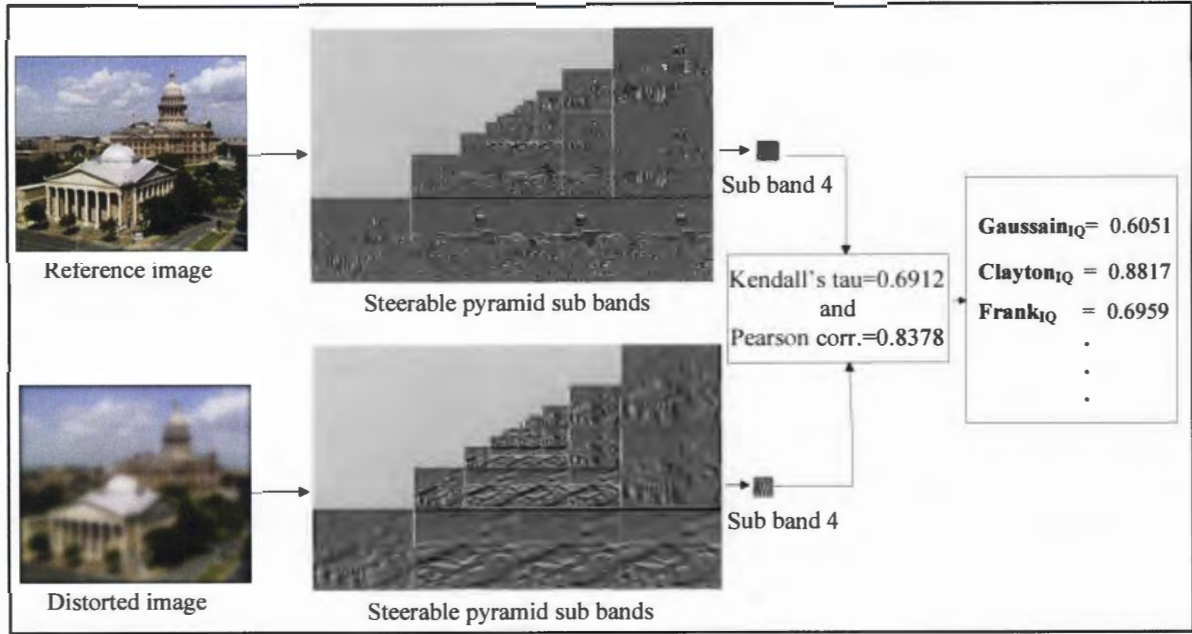
**Fig. 3.10.** Histogram of church and capitol image (LIVE database)



**Fig. 3.11.** Selected sub bands for steerable pyramid (church and capitol original image)



**Fig. 3.12.** Selected sub bands for steerable pyramid (church and capitol distorted image)



**Fig. 3.13.** Image quality measure process using copula functions for church and capitol image

### 3.4 Experimental results and discussion

In this chapter, we used LIVE image database release 2 [65] to testify the performance of the Gaussian, Marshall-Olkin, Clayton, Frank, and Gumbel copulas image quality measures and to compare our results with the popular three image quality measure methods (VIF, UQI, SSIM),. Also, we compared our algorithm with the well known subjective method (HVS). LIVE database is one of the most commonly used databases for evaluation of image quality metrics. LIVE image database release 2 includes 29 original images. For each image there are 5 distortion types. Four distortion types of them (i.e. Gaussian Blur, Fast Fading, JPEG, and White Noise) have 5 levels of distortion and only one type (i.e. JPEG 2000) has 6 levels of distortion (i.e.  $29 * 4 * 5 + 29 * 6 = 754$  images). The 29

original test images were distorted using the following distortion types: JPEG 2000, JPEG, White noise, Gaussian blur, and fast fading distortion. Twenty observers were asked to provide their perception of the quality of the distorted images on a continuous linear scale [66]. The scale was divided into five equal regions "Bad", "Poor", "Fair", "Good", and "Excellent". For all of twenty observers, the images display device configurations were identical [66]. LIVE database includes the subjective scores based on HVS of all distorted versions. The images are originally colored images and we converted them to their gray scale version.

In Fig. 3.14, we have one of the 29 original images in LIVE image database release 2 named as "monarch" and the five distortion types are presented in five rows with their distortion levels on the columns. These results are displayed to show their level of distortion from left to right (i.e. the best to worst quality according to the LIVE HVS reference results). The first row displays the Gaussian Blur distortion (images 5,1,3,4,2 in LIVE database), second row displays the Fast Fading distortion (images 5,4,3,2,1), third row displays JPEG distortion (images 5,2,3,1,4), the fourth row displays the JPEG 2000 distortion (images: 1,5,4,3,2), and the fifth row displays the White Noise distortion (images 5,1,3,2,4). Fig. 3.15 shows a sample of 4 images (monarch, buildings, church and capitol, and caps) of LIVE database. In this work we provide all results of these 4 images. In the next section we provide our results compared with the most used subjective method (i.e. HVS is the reference for comparing the results of any objective method).



<b>Gaussian Blur</b>	 Image 5	 Image 1	 Image 3	 Image 4	 Image 2
<b>Fast Fading</b>	 Image 5	 Image 4	 Image 3	 Image 2	 Image 1
<b>JPEG</b>	 Image 5	 Image 2	 Image 3	 Image 1	 Image 4
<b>JPEG 2000</b>	 Image 1	 Image 5	 Image 4	 Image 3	 Image 2
<b>White Noise</b>	 Image 5	 Image 1	 Image 3	 Image 2	 Image 4

**Fig. 3.14.** A sample image distortion (monarch)



**Fig. 3.15.** Sample of original images (LIVE database)

Table 3.1 shows the image quality rank results for the “monarch” image in LIVE image database. The table shows the results of our algorithm using three copula functions (Gaussian, Clayton, and Frank) compared with three known image quality measure methods

VIF, SSIM, and UQI and also compared with the Human Visual System (HVS) method. The rows of Table 3.1 consist of the five different distortions added to the image. For each distortion type (row), the rank (R), and the measure value (M) are included for all 7 algorithms. For our proposed algorithm, the larger values of the quality measure (M) represent a better image quality.

As we can see in Table 3.1, the rank (R) of our proposed algorithm for Gaussian blur, JPEG, JPEG 2000, and white noise are exactly the same as the HVS, VIF, UQI, and SSIM results. For example for JPEG (the third row) we can see that the rank of the distorted images, from the best quality to the worst quality, is 5, 2, 3, 1, 4 with measure values of 2.648, 0.775, 0.601, 0.321, and 0.181 respectively, which means that image 5 has the best quality with quality score equals 2.648 and image 4 has the worst quality with quality score equals 0.181. By comparing the resulting rank obtained by the three copula functions with the other four popular image quality measure methods, we can notice that we got the same exact results of the other popular methods. For Gaussian blur, the resulting rank of the images by applying our algorithm using Gaussian, Clayton, and Frank copulas is 5, 1, 3, 4, 2 which is the exact rank of the other popular methods. In JPEG 2000 case, our resulting rank of the distorted images is 1, 5, 4, 3, 2 which is also the same rank of the other popular methods. In addition, we can also notice that the rank of our proposed algorithm for white noise distortion (5, 3, 1, 2, 4) is the same as all other four popular methods. Finally, in fast fading distortion type, only a slight difference was found in the resulting rank of our algorithm compared with HVS (LIVE database results). In addition, we can notice from Table 3.1 that the recent most popular image quality measures (VIF, UQI, and SSIM) have also a slight difference in their results for fast fading compared with HVS as well. For



example, HVS resulting rank is 5, 4, 3, 2, 1, However, Gaussian copula based image quality algorithm's result is 3, 4, 5, 2, 1 which means that there are two wrong ranks in Gaussian copula based image quality algorithm's result compared with HVS. For Clayton and Frank copulas based image quality algorithms' result, the rank result is 4, 3, 5, 2, 1 which means that there are three wrong ranks compared with HVS. Also we can notice that the resulting rank by applying the popular three methods (VIF, UQI, and SSIM) is 4, 5, 3, 2, 1 which means that there are two wrong ranks in VIF, UQI, and SSIM methods compared to HVS results. In this work, we consider the ranks (i.e. not the value) of the image quality to compare different versions of distortion. Table 3.2 shows the corresponding rank for our proposed algorithm using five copula functions for the "monarch" image.

**Table 3.1.** Image quality results for quality measure methods (monarch)

	HVS		VIF		UQI		SSIM		Gaussian		Clayton		Frank	
	R	M	R	M	R	M	R	M	R	M	R	M	R	M
<b>G. Blur</b>	5	0.906	5	0.6149	5	0.8070	5	0.9577	5	3.6105	5	2.911	5	2.726
	1	1.708	1	0.3653	1	0.6604	1	0.8899	1	2.6240	1	2.552	1	2.346
	3	1.851	3	0.3384	3	0.6418	3	0.8784	3	2.5010	3	2.487	3	2.278
	4	2.854	4	0.2175	4	0.5307	4	0.8084	4	1.8701	4	2.048	4	1.821
	2	11.33	2	0.0226	2	0.1744	2	0.6317	2	0.1616	2	0.396	2	0.287
<b>Fast Fading</b>	5	26.1	4	0.8969	4	0.8398	4	0.9801	3	5.1198	4	3.089	4	2.915
	4	23.7	5	0.8594	5	0.8227	5	0.9791	4	4.9973	3	3.084	3	2.911
	3	21.3	3	0.6370	3	0.7963	3	0.9548	5	4.9580	5	3.075	5	2.901
	2	17.9	2	0.3501	2	0.6573	2	0.8824	2	3.7345	2	2.866	2	2.678
	1	15.5	1	0.1365	1	0.4545	1	0.7600	1	1.4775	1	1.283	1	1.063
<b>JPEG</b>	5	2.648	5	0.9796	5	0.8899	5	0.9851	5	6.0702	5	3.163	5	2.995
	2	0.775	2	0.7077	2	0.7118	2	0.9527	2	4.2308	2	2.887	2	2.701
	3	0.601	3	0.6084	3	0.6707	3	0.9390	3	3.7582	3	2.715	3	2.518
	1	0.321	1	0.3553	1	0.5159	1	0.8701	1	2.6484	1	2.043	1	1.817
	4	0.181	4	0.1381	4	0.2796	4	0.7197	4	1.4358	4	1.192	4	0.977
<b>JPEG 2000</b>	1	2.789	1	0.9527	1	0.9100	1	0.9898	1	5.0713	1	3.066	1	2.891
	5	0.427	5	0.5302	5	0.6769	5	0.9443	5	3.4684	5	2.587	5	2.383
	4	0.200	4	0.3445	4	0.6165	4	0.9019	4	2.8655	4	2.273	4	2.054
	3	0.102	3	0.2183	3	0.5306	3	0.8452	3	2.1978	3	1.842	3	1.612
	2	0.042	2	0.1200	2	0.4178	2	0.7668	2	1.4837	2	1.274	2	1.055
<b>White Noise</b>	5	0.027	5	0.7539	5	0.6223	5	0.8455	5	4.1042	5	2.845	5	2.656
	1	0.035	1	0.6793	1	0.5636	1	0.7764	1	3.8980	1	2.778	1	2.585
	3	0.117	3	0.3375	3	0.3168	3	0.3504	3	2.7332	3	2.091	3	1.866
	2	0.187	2	0.2392	2	0.2350	2	0.2317	2	2.2458	2	1.769	2	1.538
	4	1	4	0.0523	4	0.0483	4	0.0442	4	0.6100	4	0.518	4	0.386

**Table 3.2.** Image quality rank for quality measure methods for monarch image

	HVS	VIF	UQI	SSIM	Gaussian	M-Olkin	Clayton	Frank	Gumbel
<b>Gaussian Blur</b>	5	5	5	5	5	5	5	5	5
	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3
	4	4	4	4	4	4	4	4	4
	2	2	2	2	2	2	2	2	2
<b>Fast Fading</b>	5	4	4	4	3	4	4	4	4
	4	5	5	5	4	3	3	3	3
	3	3	3	3	5	5	5	5	5
	2	2	2	2	2	2	2	2	2
	1	1	1	1	1	1	1	1	1
<b>JPEG</b>	5	5	5	5	5	5	5	5	5
	2	2	2	2	2	2	2	2	2
	3	3	3	3	3	3	3	3	3
	1	1	1	1	1	1	1	1	1
	4	4	4	4	4	4	4	4	4
<b>JPEG 2000</b>	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5
	4	4	4	4	4	4	4	4	4
	3	3	3	3	3	3	3	3	3
	2	2	2	2	2	2	2	2	2
<b>White Noise</b>	5	5	5	5	5	5	5	5	5
	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3
	2	2	2	2	2	2	2	2	2
	4	4	4	4	4	4	4	4	4

Tables (3.3-3.5) show the rank (R) of the distorted versions of three images shown in Fig. 3.15 (buildings, church and capitol, and caps). Although, in the buildings case (see Table 3.3), Gaussian copula based image quality algorithm obtained the same exact results of VIF, UQI, and SSIM, in the fast fading distortion type, Gaussian copula algorithm obtained rank 5, 1, 3, 2, 4 which is different than HVS (HVS rank is 5, 4, 3, 2, 1) in two ranks. Marshall-Olkin algorithm obtained the same results as the four popular methods (VIF, UQI, SSIM, and HVS) except in few ranks in fast fading distortion type. In that distortion type, Marshall-Olkin algorithm obtained 5,2,1,3,4 rank, However HVS obtained 5,4,3,2,1 rank which means that Marshall-Olkin got four wrong ranks. Finally, Clayton, Frank, and Gumbel obtained the same rank as HVS, VIF, UQI, and SSIM in four distortion types of (Gaussian blur, JPEG, JPEG2000, and white noise), however they obtained different results for the fast fading distortion type in three ranks.

For church and capitol image (see Table 3.4), Gaussian copula algorithm obtained better results than the VIF, UQI, and SSIM methods. As we can see that in the four distortion types, Gaussian blur, JPEG, JPEG 2000, and white noise the results of the Gaussian copula algorithm obtained the same results as HVS method. In fast fading distortion case, we can see that the rank of the Gaussian copula algorithm is 4,5,3,2,1 and the resulting rank of HVS is 5,4,3,2,1 which means that they are different in two ranks, However, VIF, UQI, and SSIM obtained the following rank: 4,3,5,2,1 which means that these three methods gave different results than HVS results in three ranks (Gaussian copula obtained better results). Our algorithm using Marshall-Olkin, Clayton, Frank, and Gumbel copulas obtained 100% same results as HVS for all distortion types (i.e. including fast fading distortion). For caps images (see Table 3.5), our proposed algorithm using all five copula functions obtained 100% same results as HVS method, However VIF, UQI, and SSIM obtained different results in fast fading distortion. Tables (3.1- 3.5) show how our results are comparable with and even better, in some cases (see fast fading distortion results in Table 3.4 and Table 3.5), than the three popular image quality measure methods (VIF, UQI, and SSIM).

From Tables (3.1-3.5) we can notice that all of the unmatched results for all objective methods (Gaussian, Marshall-Olkin, Clayton, Frank, Gumbel, VIF, UQI, and SSIM) are in the fast fading distortion type. These results confirm the work done by Wajid et al. in [66]. Wajid et al. studied the human perception similarity across geographically distant population samples. They compared the Difference Mean Opinion Score (DMOS) reported on LIVE with their own experimentation. In their work they focused on three distortion types Gaussian blur, white noise, and fast fading. In their work the quality scores of the images in LIVE database were recalculated by utilizing larger number of human subjects (50 subjects) at

Computer Vision and Image Processing Group (CVIPG), The results of their study showed that, although, their experimentation results demonstrate similar pattern to LIVE results, in fast fading distortion images with the same distortion levels got different results for both models (LIVE database results and CVIPG results). Their results showed that modeling of human perception in the case of fast fading distortion is much more complex than modeling the human perception in case of white noise and Gaussian blur due to the changeable trend of DMOS for fast fading distortion.

In our work after examining the results for the 754 images of the LIVE database, we observed that the three image quality measures (i.e. VIF, UQI, SSIM) are comparable with HVS (LIVE database results). In addition, we found that our proposed algorithm obtained comparable results with the most popular three methods (i.e. VIF, SSIM, UQI) and in some cases our proposed algorithm obtained better results. Moreover, our algorithm uses only one arbitrary sub band, however VIF method uses 8 sub bands to estimate the image quality. Such results are impressive and put the proposed copula based image quality measures at equal footing with the other image quality measures.

As we mentioned earlier, we used sub band 4 to calculate the image quality of the distorted images. In our work we test the effect of using different sub bands on the resulting rank. Table 3.6 shows the rank of the distorted versions of image (monarch) after using different sub bands (4, 7, 10, 13) of the steerable pyramid. As we can notice that the results are almost the same except in few distorted images. For example, for Gaussian blur (first row) the rank of our proposed algorithm using all five copula functions is 5, 1, 3, 4, and 2, which means that using different sub bands will not have a big effect on the results.

**Table 3.3.** Image quality rank for quality measure methods for building image

	HVS	VIF	UQI	SSIM	Gaussian	M-Olkin	Clayton	Frank	Gumbel
<b>Gaussian Blur</b>	2	2	2	2	2	2	2	2	2
	4	4	4	4	4	4	4	4	4
	3	3	3	3	3	3	3	3	3
	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5
<b>Fast Fading</b>	5	5	5	5	5	5	5	5	5
	4	1	1	1	1	2	2	2	2
	3	3	3	3	3	1	3	3	3
	2	2	2	2	2	3	1	1	1
	1	4	4	4	4	4	4	4	4
<b>JPEG</b>	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3
	5	5	5	5	5	5	5	5	5
	2	2	2	2	2	2	2	2	2
	4	4	4	4	4	4	4	4	4
<b>JPEG 2000</b>	3	3	3	3	3	3	3	3	3
	4	4	4	4	4	4	4	4	4
	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5
	2	2	2	2	2	2	2	2	2
<b>White Noise</b>	1	1	1	1	1	1	1	1	1
	4	4	4	4	4	4	4	4	4
	2	2	2	2	2	2	2	2	2
	5	5	5	5	5	5	5	5	5
	3	3	3	3	3	3	3	3	3

**Table 3.4.** Image quality rank for quality measure methods for church and capitol image

	HVS	VIF	UQI	SSIM	Gaussian	M-Olkin	Clayton	Frank	Gumbel
<b>Gaussian Blur</b>	2	2	2	2	2	2	2	2	2
	3	3	3	3	3	3	3	3	3
	5	5	5	5	5	5	5	5	5
	4	4	4	4	4	4	4	4	4
	1	1	1	1	1	1	1	1	1
<b>Fast Fading</b>	5	4	4	4	4	5	5	5	5
	4	3	3	3	5	4	4	4	4
	3	5	5	5	3	3	3	3	3
	2	2	2	2	2	2	2	2	2
	1	1	1	1	1	1	1	1	1
<b>JPEG</b>	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3
	4	4	4	4	4	4	4	4	4
	5	5	5	5	5	5	5	5	5
	2	2	2	2	2	2	2	2	2
<b>JPEG 2000</b>	3	3	3	3	3	3	3	3	3
	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5
	4	4	4	4	4	4	4	4	4
	2	2	2	2	2	2	2	2	2
<b>White Noise</b>	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2
	4	4	4	4	4	4	4	4	4
	5	5	5	5	5	5	5	5	5
	3	3	3	3	3	3	3	3	3

**Table 3.5.** Image quality rank for quality measure methods for caps image

	HVS	VIF	UQI	SSIM	Gaussian	M-Olkin	Clayton	Frank	Gumbel
Gaussian Blur	3	3	3	3	3	3	3	3	3
	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2
	5	5	5	5	5	5	5	5	5
Fast Fading	4	4	4	4	4	4	4	4	4
	5	4	4	4	5	5	5	5	5
	4	5	5	5	4	4	4	4	4
	3	3	3	3	3	3	3	3	3
JPEG	2	2	2	2	2	2	2	2	2
	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5
	4	4	4	4	4	4	4	4	4
JPEG 2000	3	3	3	3	3	3	3	3	3
	4	4	4	4	4	4	4	4	4
	5	5	5	5	5	5	5	5	5
	1	1	1	1	1	1	1	1	1
White Noise	2	2	2	2	2	2	2	2	2
	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3
	5	5	5	5	5	5	5	5	5

**Table 3.6.** Copula image quality results using different sub bands (monarch)

	Gaussian				Marshall-Olkin				Clayton				Frank				Gumbel			
	S <sub>4</sub>	S <sub>7</sub>	S <sub>10</sub>	S <sub>13</sub>	S <sub>4</sub>	S <sub>7</sub>	S <sub>10</sub>	S <sub>13</sub>	S <sub>4</sub>	S <sub>7</sub>	S <sub>10</sub>	S <sub>13</sub>	S <sub>4</sub>	S <sub>7</sub>	S <sub>10</sub>	S <sub>13</sub>	S <sub>4</sub>	S <sub>7</sub>	S <sub>10</sub>	S <sub>13</sub>
G. Blur	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
Fast Fading	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	3	4	5	4	4	4	3	4	4	4	3	4	4	4	3	4	4	4	3	4
	4	2	3	2	3	2	5	2	3	2	5	2	3	2	5	2	3	2	5	2
	5	3	4	3	5	3	4	3	5	3	4	3	5	3	4	3	5	3	4	3
JPEG	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5	2	5
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
JPEG 2000	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
White Noise	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

### 3.5 Localizing unmatched regions between two images

In this section, we present the algorithm of localizing the unmatched regions between two images. We apply copula functions to indicate the unmatched regions. We apply the five copula functions discussed above. Given two images,  $image_1$  and  $image_2$ , our objective is to identify if these two images are identical or there are unmatched regions. Let  $H$  refer to the image similarity value (i.e. image quality) between two identical images, and  $IQ_i$  represents the actual measured image quality value (i.e. image similarity) using copula functions, where  $i$  is the number of sub-regions in the image ( $i = 1, 2, 3, \dots$ ). For example,  $IQ_1$  refers to the measured image similarity between the images before segmenting the tested regions (i.e. number of sub-regions=1 in the tested/suspected region). The value of  $IQ_1$  varies depending on the tested (i.e. suspected) region. To calculate the value of  $IQ_1$  we should place all other regions from the  $image_1$  into  $image_2$  and calculate the corresponding image quality of  $image_2$ . Fig. 3.16 shows the sequence of segmenting 128 x 128 image into smaller regions. The segmentation technique is not affected by the size of the images. in this work, we assume that the size dimensions of the images are even, for example 28 x 128 pixels, however if the images have odd size, for example 129 x 129 we choose to have the bottom and the left side to be bigger than the top and the right side by one pixel (arbitrary choice). That will not affect the performance of our algorithm because we segment both images (original and distorted images) with the same technique.

Fig. 3.17 shows the flow chart of our algorithm. In this algorithm, we first read the images. After that, we apply steerable pyramid technique to decompose both images. We choose only one sub band from the resulted steerable pyramid sub bands to calculate the image quality (i.e. similarity),  $IQ_1$ , by applying copula functions. In this case the whole

image is the tested\suspected region (i.e. number of sub-regions =1). If the resulted image similarity,  $IQ_1$ , does not equal the value of  $H$ , then we know that there is a difference (i.e. unmatched regions) between the images. Therefore we segment (i.e. number of regions,  $i$ , increments by 1) both images, into smaller regions, to test smaller regions as shown in Fig. 3.16 (a-d) and copy all regions from  $image_1$  into  $image_2$  except the last region ( $region_i$ ), where  $i$  refers to the total number of segmented regions in the image). After that we calculate new image similarity  $IQ_i$ . We continue segmenting both images and in each time we copy all regions from  $image_1$  into  $image_2$  except  $region_i$  and calculate new image similarity (i.e. image quality), until  $IQ_i = H$ . We repeat the same procedure for each unmatched region until we get 16 x 16 unmatched regions. We should note that for each unmatched region, we recalculate  $IQ_1$  for that region by copying all another region from the  $image_1$  into  $image_2$  except that unmatched region (see Fig. 3.17). As we can notice that the value of the resulting image similarity ( $IQ_i$ ) is the key factor for tracking unmatched regions between  $image_1$  and  $image_2$ . Each time we calculate new image similarity ( $IQ_i$ ), there are three possibilities as follows:

**Case 1:** The resulting image similarity ( $IQ_i$ ) equals  $H$

If the value of  $IQ_i = H$ . This case indicates that  $image_2$  became identical to the  $image_1$  after copying  $region_{i-1}$  from  $image_1$  into  $image_2$ . Which means that  $region_{i-1}$  is unmatched region between the two images. In this case if  $region_{i-1}$  is 16 x 16 pixels then we identified the unmatched region, otherwise we segment  $region_{i-1}$  by the same procedure shown in Fig. 3.16 and replace the corresponding regions to indicate the 16 x 16 pixels unmatched regions. We stop segmenting the unmatched region and replacing the corresponding regions when  $IQ_i = H$  and repeat the same procedure for each unmatched region until we reach 16 x 16 regions.

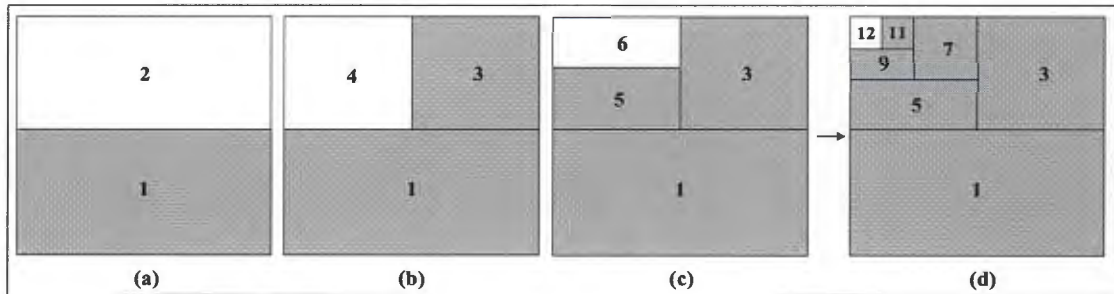


**Case 2:** The resulting image similarity ( $IQ_i$ ) equals the image similarity ( $IQ_1$ )

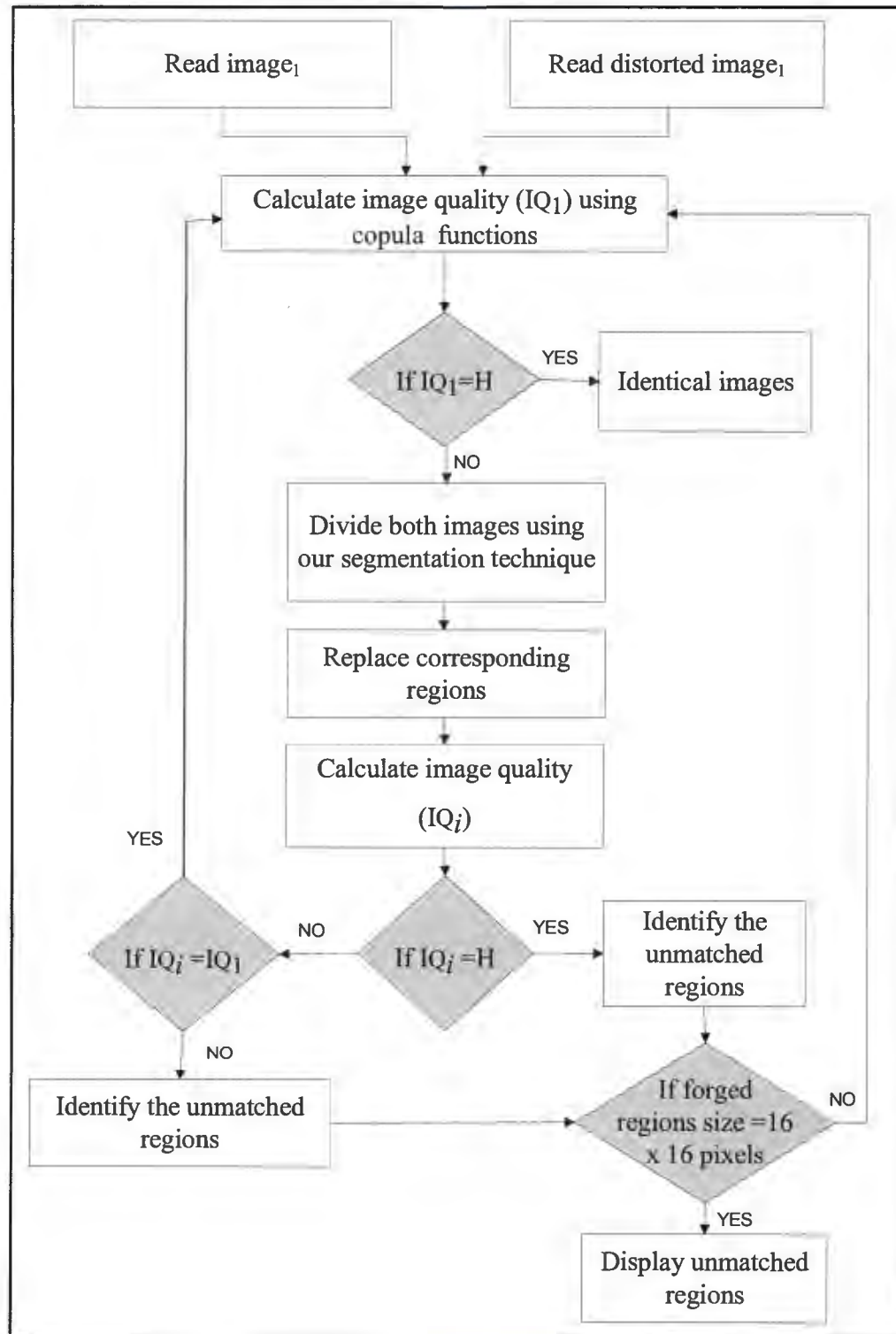
In this case, the difference is in other regions except  $region_{i-1}$  (because the image quality did not change by copying  $region_{i-1}$ ). In this case we do not segment that  $region_{i-1}$  (i.e. skip that region). However, we continue segmenting the whole image as shown in Fig. 3.16 (a-d) and replacing the corresponding regions until  $IQ_i = H$  and repeat the same procedure until we reach 16 x 16 unmatched regions.

**Case 3:** The resulting image similarity ( $IQ_i$ ) does not equal both  $H$  and  $IQ_1$

In this case,  $region_{i-1}$  is unmatched region, however there are other unmatched regions in the other untested regions. In this case we segment the  $region_{i-1}$  by applying the same procedure and replace the corresponding regions to test smaller regions until we reach 16 x 16 pixels regions. In addition, we segment the whole image as shown in Fig. 3.16 (a-d) and replace the corresponding regions. We stop segmenting the region and replacing the corresponding regions when  $IQ_i = H$  and repeat the same procedure until we get 16 x 16 unmatched regions. We display the unmatched regions when two conditions are satisfied: 1)  $IQ_i = H$  and 2) all unmatched regions are 16 x 16 pixels.



**Fig. 3.16.** Image segmentation technique: (a), (b), (c) the segmentation procedure for 128 x 128 pixels image; (d) all regions for 128 x 128 pixels image

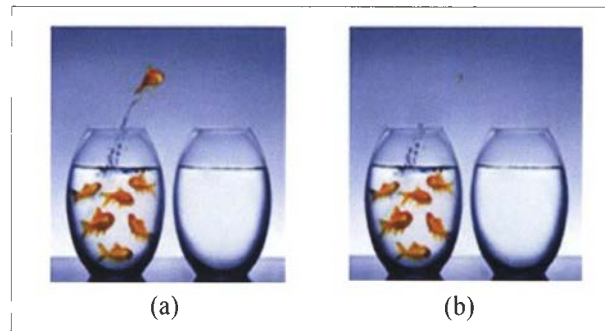


**Fig. 3.17.** The flow chart of the proposed algorithm

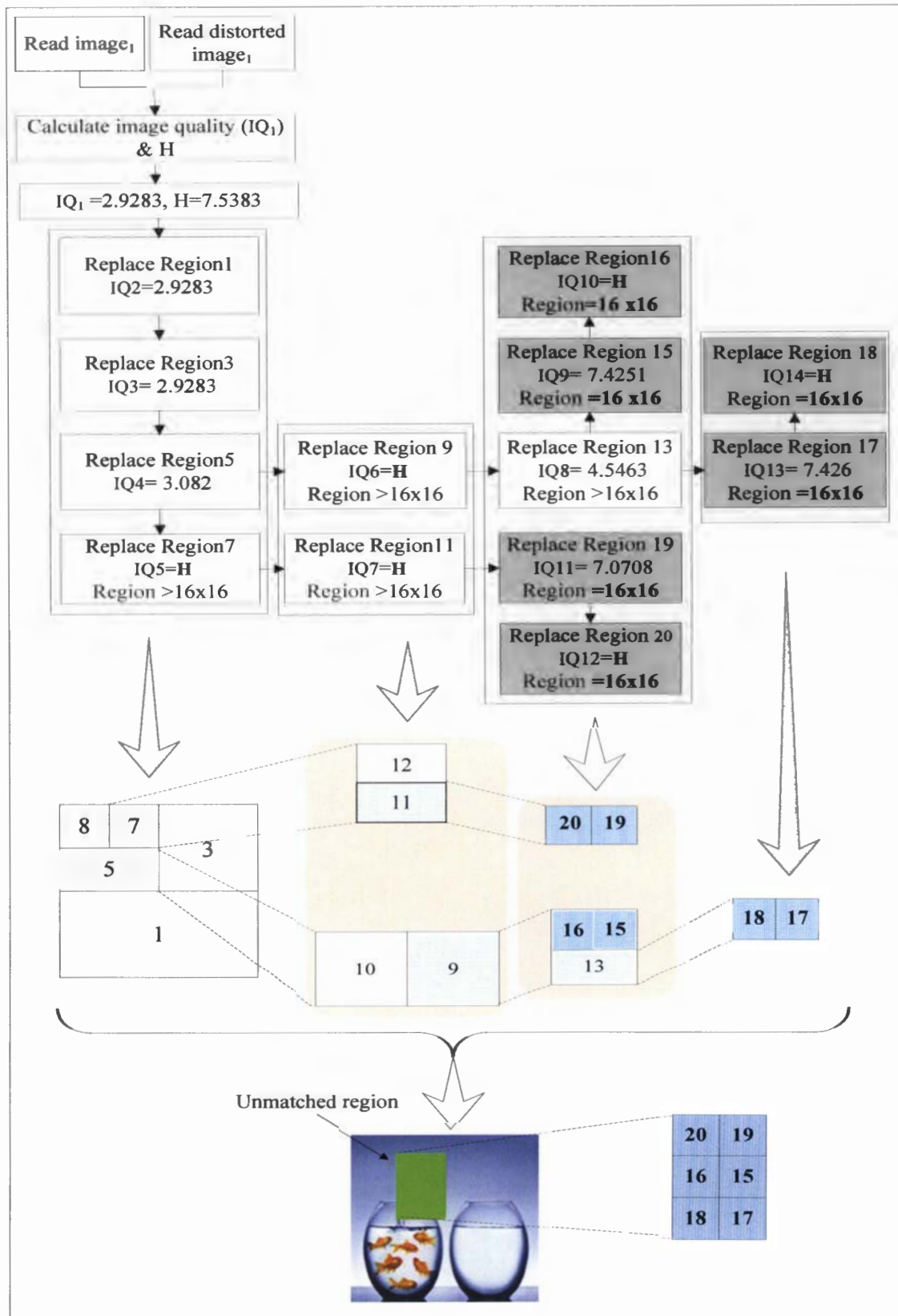
### 3.6 Experimental results

To examine the performance of the proposed algorithm, we apply it on a set of 250 images downloaded from two databases [67, 68]. The first database [67] includes images of colored Birds: with different sizes and the second Database [68] includes different types of gray images and colored images (Animals, Food, Nature, Miscellaneous) with different sizes. We first resized the images into 128 x 128 pixels. The objective of the resizing step is to decrease the number of iterations to get 16 x 16 regions. After resizing the tested original images, we tamper them by adding\removing some objects. Fig. 3.18 shows a simple example we use to apply our procedure of identifying unmatched regions between two images by applying copula functions. Fig. 3.19 illustrates the results of applying our algorithm by using Gaussian copula. The value of  $H$  for the fishbowl example is 7.5383 (i.e. the image quality between two identical images of fishbowl image). First step is calculating the image similarity (i.e. image quality) between the images. The resulting image similarity ( $IQ_1$ ) is 2.9283. After that we segment the two images into smaller regions until the condition  $IQ_i = H$ . The results are 2.9283, 2.9283, 3.0814, and  $H$  after copying region<sub>1</sub>, region<sub>3</sub>, region<sub>5</sub>, and region<sub>7</sub> respectively. These results mean that region<sub>5</sub> and region<sub>7</sub> are unmatched regions. We repeat the same procedure for region<sub>5</sub> and region<sub>7</sub>. For region<sub>5</sub>, we first recalculate  $IQ_1$ , as explained in our proposed algorithm, by copying all regions from image<sub>1</sub> into image<sub>2</sub> except region<sub>5</sub> (i.e. region<sub>5</sub> is suspected region).  $IQ_1$  in that case equals 4.5028 and the result is  $H$  after copying region<sub>9</sub> but the size of the unmatched region is  $> 16 \times 16$  pixels which means that we should recalculate  $IQ_1$  and segment region<sub>9</sub> in both images into smaller regions.  $IQ_1$  in this case is also 4.5028, and the results are 4.5463, 7.426, and  $H$  after copying region<sub>13</sub>, region<sub>17</sub>, and region<sub>18</sub> respectively. The later results indicate that there is difference between the two images in region<sub>13</sub>, region<sub>17</sub>, and region<sub>18</sub>. As we can see from Fig. 3.19 that region<sub>17</sub>

and region<sub>18</sub> are 16 x 16 pixels, However region<sub>13</sub> is greater than 16 x 16 pixels. Therefore we recalculate  $IQ_1$  and segment region<sub>13</sub> with the same procedure.  $IQ_1$  in this case is 5.1731, and the results of segmenting region<sub>13</sub> are 7.4251 and H after copying region<sub>15</sub> and region<sub>16</sub> respectively, which means that there is difference between the two images in these two regions (16 x 16 pixels). On the other hand for region<sub>7</sub>,  $IQ_1=3.0814$ , and the image similarity is H after copying region<sub>11</sub>, but the size of the region<sub>11</sub> is greater than 16 x 16 pixels, therefore we should recalculate  $IQ_1$  for region<sub>11</sub> and segment it into smaller regions.  $IQ_1$  in this case is also 3.0814, and the results of segmenting region<sub>11</sub> are 7.0708 and H after copying region<sub>19</sub> and region<sub>20</sub>. We can see in Fig. 3.19 that region<sub>19</sub> and region<sub>20</sub> are 16 x 16 pixels. Therefore we do not segment the images anymore. The results of our proposed algorithm indicate that region<sub>15</sub>, region<sub>16</sub>, region<sub>17</sub>, region<sub>18</sub>, region<sub>19</sub>, and region<sub>20</sub> are unmatched regions.





















**Fig. 3.18.** Test image: (a) fishbowl image; (b) tampered image



**Fig. 3.19.** The results of applying Gaussian copula

Fig. 3.20 shows a sample of original images and tampered images and the resulted unmatched regions by applying our algorithm using Gaussian copula. We applied five copula functions namely: Gaussian; Marshall-Olkin; Clayton; Frank; and Gumbel copulas. The results of identifying unmatched regions between the images, by applying all five copulas were exactly the same. In this proposed algorithm, we identified 16 x 16 block to be the smallest unmatched region, which means that if there is only one unmatched pixel between two images, then the resulted unmatched region after applying our algorithm will be 16 x 16 pixel.

Original image	Tampered image	Our result
		
		
		
		
		
		

**Fig. 3.20.** The results of our proposed algorithm on test images using Gaussian copula



### 3.7 Conclusions and future work

In this work, we employed five copula functions as image quality measures. By applying the proposed algorithm on the tested images from the LIVE image database and comparing the results with HVS (LIVE database result) and three popular image quality measures VIF, UQI, and SSIM, we obtained comparable results with all of them. We also showed how and why the results were different in all methods including all objective methods in the case of fast fading distortion type. We also tested the effect of the sub band choice on the results. Moreover we presented an algorithm to indicate the unmatched regions in two images. The results showed how accurate our algorithm is (i.e. it detects the difference in one pixel). The advantage of using copula based image quality measure is that the value of it changes with any slight difference between the images which enables us to detect a very slight difference that cannot be notice by the naked eye. The results showed that the choice of the sub band does not affect the results for four types of distortion (Gaussian blur, JPEG, JPEG 2000, and white noise). We can also notice that the most popular image quality measures such as VIF, SSIM, and UQI have different results than the HVS results in few distorted versions as well. Moreover, copulas' simple calculations and the fact that they are not dependent on the distribution of the used data are the observed advantages in this work. We are encouraged to investigate other copula functions that can be employed to measure image quality. Also we are working on using our proposed algorithm in blind image forgery detection.



## **Chapter 4**

### **Copula Based Copy-Move Forgery Detection Algorithm for Digital Images**

Over the past few years, many techniques have been introduced to tamper images or videos. These techniques can be classified into three general categories namely: copy-move forgery; image splicing; and image retouching [27, 35]. The passive blind techniques, where the analyser has just the final image, provide a solution to identify image modifications without the need of inserting data or digital signatures for the image authentication. Blind passive forgery detection methods are classified as being (a) visual and (b) statistical. Visual methods are based on visual clues that may not require any hardware or software tools, for example, irregularity in images and light distortion on an object within an image. On the other hand, the statistical methods are considered more robust as they analyse the pixel values of the image. [24]

In this chapter, we focus on copy-move forgery, in which a part from the image is copied and pasted into another part of the same image. Because the copied part comes from the same image, the properties, such as noise, color palette and texture, will be similar to the rest of the image and that will make it more difficult to detect these forged parts [44].

In this chapter, we present a blind copy-move forgery detection algorithm. We use only the forged image to detect and localize the forgery. The objective of copy-move forgery detection is detecting image areas that are same or extremely similar [24, 69]. In this proposed algorithm, we first divide the forged image into  $16 \times 16$  overlapping blocks by sliding a  $16 \times 16$  window with one pixel step. We decompose each sub image (block) by applying steerable pyramids technique as discussed in chapter 3. Then, we apply the five copula functions to detect the similar objects in the image. In the next section, we introduce our copy-move forgery detection algorithm.

#### **4.1 Proposed algorithm**

Given a suspected image with size  $M \times N$ , our goal is to identify if there are duplicated regions (i.e. copy move forgery) in this single image, or there is no forged regions. Let  $H$  refer to the value of the copula based mutual information between two identical images. Fig. 4.1 shows the flow chart of our proposed algorithm. In this section we explain the procedure of our proposed algorithm.

**Step 1:** Read the suspected image.

**Step 2:** Convert the image into gray-scale level. By converting the colored images into gray scale images, we reduce the image dimension and also we will be able to apply the steerable pyramid on it.

- Step 3:** Create a  $16 \times 16$  sliding window with one pixel step. An image with size of  $M \times N$  will generate, by sliding the window,  $(M - S + 1) \times (N - S + 1)$  overlapping  $16 \times 16$  blocks, where  $S$  represents the block size.
- Step 4:** Use steerable pyramid technique as discussed in chapter 3 to decompose each block into its sub bands. As discussed before in [13] the choice of the sub band does not affect the results. That is why we can choose any sub band.
- Step 5:** Choose sub band number 1 with size  $4 \times 4$  to decrease the number of coefficients that represent the blocks.
- Step 6:** Transform each selected sub band into a row in a matrix. The row length is  $1 \times 16$  and the Matrix size is  $B \times 16$ , where  $B = (M - S + 1) \times (N - S + 1)$ , is the number of overlapping blocks in the image created by the sliding window. Now we have one matrix that has all the information we need to detect the forged region in the image.
- Step 7:** Quantize the matrix (i.e. sub bands) to get better representation of the blocks. The objective of applying the quantization process is reducing the number of discrete levels that represents the matrix (i.e. sub bands). Quantizing the matrix enables us to overcome the problem of not detecting the forged regions (i.e. matched regions) after applying some image processing techniques such as compression, scaling, and rotation. We have experimented with different quantization levels and we found that using 16 levels for quantization is sufficient and appropriate. Although, the quantization process helps us to identify forged regions even after reasonable compression, scaling, and rotation, but it increases the possibility of identifying wrong matched regions. That is why we avoid this issue in steps 10 & 12 in this algorithm

**Step 8:** Sort the matrix rows in ascending order to simplify the process of indicating the matched regions in the image and to reduce the time taken to indicate the matched rows.

**Step 9:** Calculate copula based mutual information between the rows in the matrix as discussed in chapter 3. If the result of calculating the mutual information between any two rows equals  $H$ , that means that these two rows (i.e. blocks) are matched, otherwise we check other rows. We repeat that for each row in the matrix. For all matched rows, we save their original indices (i.e. the index refers to the original 16 x 16 block position in the image) in the matrix before sorting. From these indices we can return the original coordinates of the matched blocks. We repeat this procedure until we get all matched blocks.

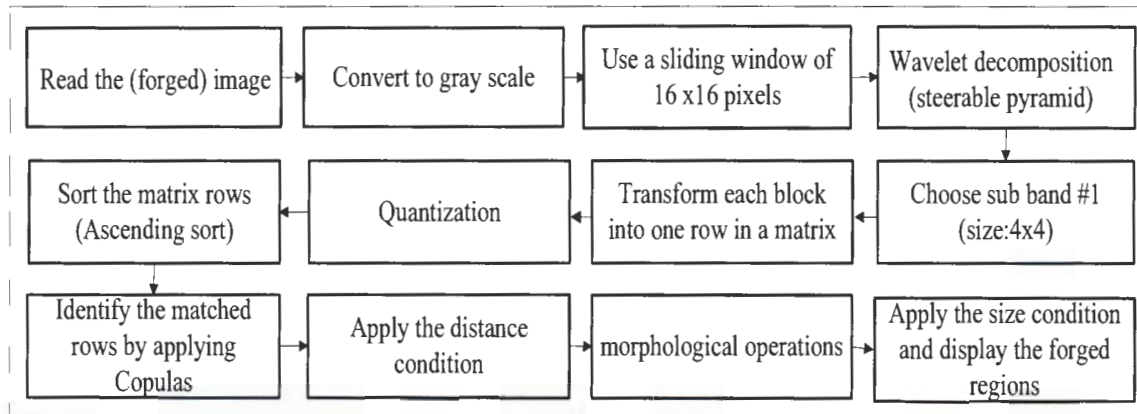
**Step 10:** Identify the matched regions that satisfy that the distance between matched regions  $\geq 100$  pixels. This condition enables us to avoid the similarity between the blocks of the same region. For example, if there is a large square block in the image, the algorithm might indicate that the upper part of the square is a matched/forged region of the lower part of the same square. That is why many researchers, in the literature, preferred to choose a threshold value to the distance between the acceptable matched regions to avoid this issue.

**Step 11:** Apply some morphological operations on the resulted matched regions. These operations are: 1) filling the small holes in the matched region to assure that we have solid regions and 2) eroding the regions to avoid merging two close regions into one region. Although, applying the morphological operations is not mandatory, but it ensures getting acceptable forged regions.

**Step 12:** Identify the final acceptable forged regions by applying second condition on them.

We identify 200 blocks to be the minimum number of blocks in the acceptable matched regions. Therefore if we get less than 200 matched blocks in two regions, we do not consider them as matched ones. This condition enables us to avoid matching very small regions. For example, if the image includes a sky scene, small parts from the sky might be matched by mistake (i.e. they might have the same pixels' intensities). As we can see in steps 10 & 12, that we check two conditions to overcome the issue of identifying unmatched region as matched ones mistakenly due to the quantization step (step 7).

**Step 13:** Display the forged regions.



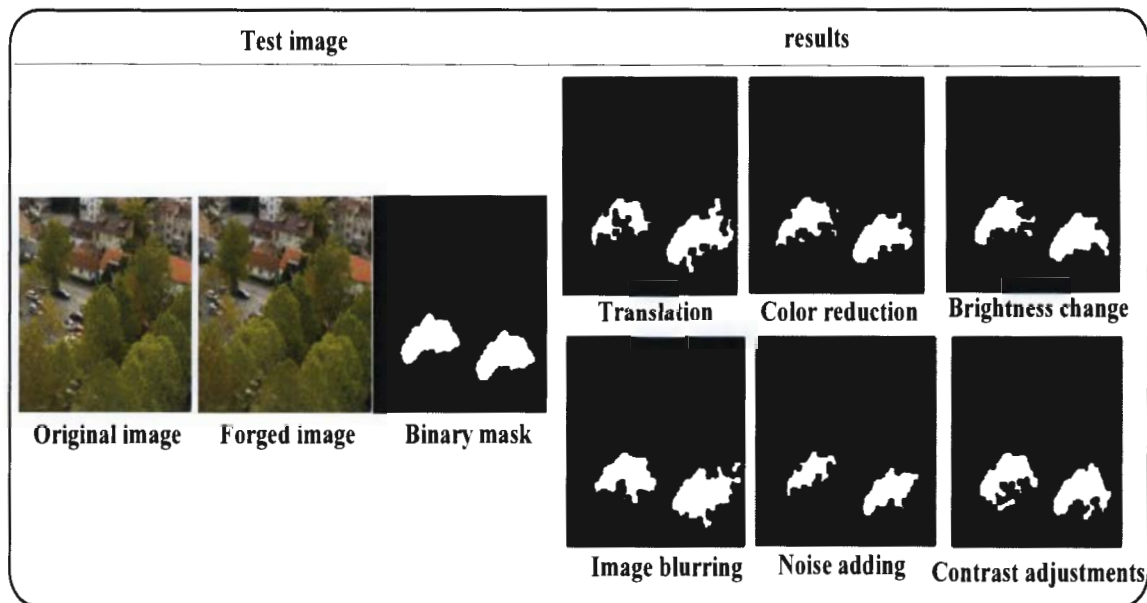
**Fig. 4.1.** Blind copy move forgery detection algorithm's flowchart

## 4.2 Results and discussion

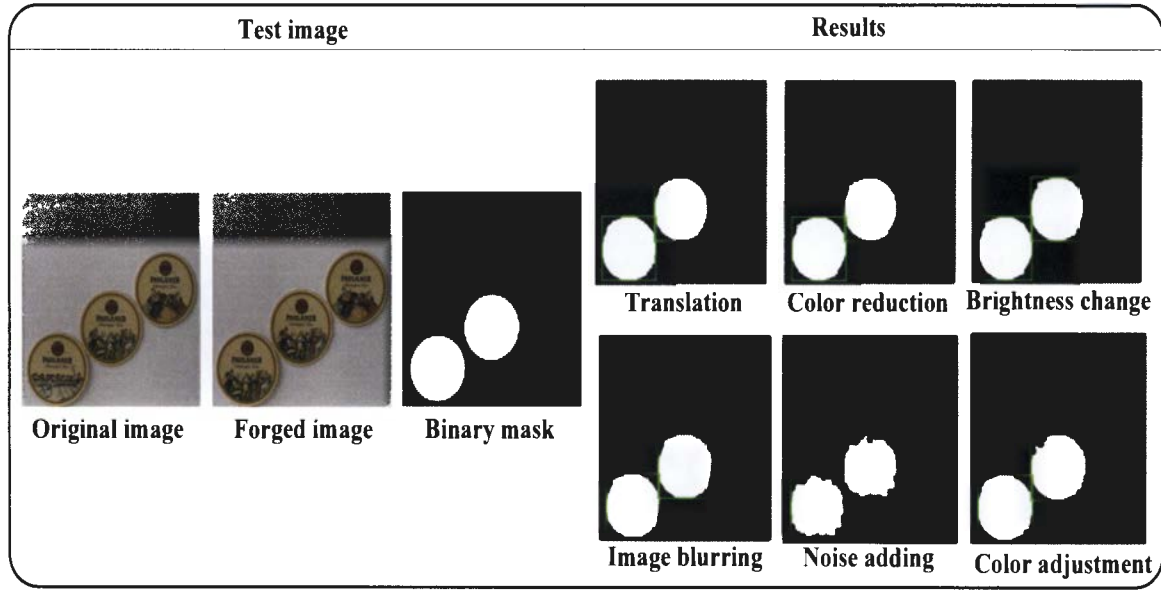
To evaluate our proposed algorithm, we tested it on Copy Move Forgery detection (CoMoFoD) database [70]. There are several databases that can be used to evaluate copy move forgery detection techniques, However most of them do not include any post processing methods (i.e. blurring, noise adding, color reduction, etc). The advantage of CoMoFoD database over other databases is that it can be used to evaluate the forgery detection method for different manipulations and different post processing techniques as well. CoMoFoD database consists of 260 forged image sets in two categories (small 512 x 512, and large 3000 x 2000). The 512 x 512 images group has 200 images and the other 60 images are in the 3000 x 3000 images categories. In this work we used the 512 x 512 images to test the performance of our algorithm. The 512 x 512 images database is grouped in five groups, each group includes 40 images, according to applied manipulation: translation, rotation, scaling, distortion, and combination (i.e. more than one manipulation). For each group in the database, different types of post-processing methods, such as JPEG compression, blurring, noise adding, and color reduction are applied to all forged and original images. Each image in the five groups has 25 post-processed versions, which means that we have 5000 versions of forged images (i.e.  $200 \text{ images} \times 25 \text{ versions} = 5000$ ). We tested our algorithm on three groups of the database. These groups are the translation, scaling, and rotation groups. The total number of the tested forged images we used from the database is:  $3 \text{ (groups)} \times 40 \text{ (images)} \times 25 \text{ (post-processed versions)} = 3000 \text{ forged versions}$ . In this database, every image set consisted of original image without any transformation, colored mask that indicates original and forged regions, binary mask for detection evaluation, and the forged images.

We applied our algorithm on different images in CoMoFoD database. These images were created by applying different types of post processing techniques on these images. These post processing techniques include: image blurring (IB), noise adding (NA), and color reduction (CR), brightness change (BC), and contrast adjustments (CA).

Fig. 4.2 and Fig. 4.3 show an example of test images. The original images, the forged images, and the binary masks are from the database. By comparing our results as shown in Fig. 4.2 and Fig. 4.3 with the binary masks (i.e. the exact forged region), we can notice the accuracy of our algorithm not only after applying the translation technique but also after applying different post processing techniques. The white region in our results indicates the forged region and the black region indicates the non forged region.



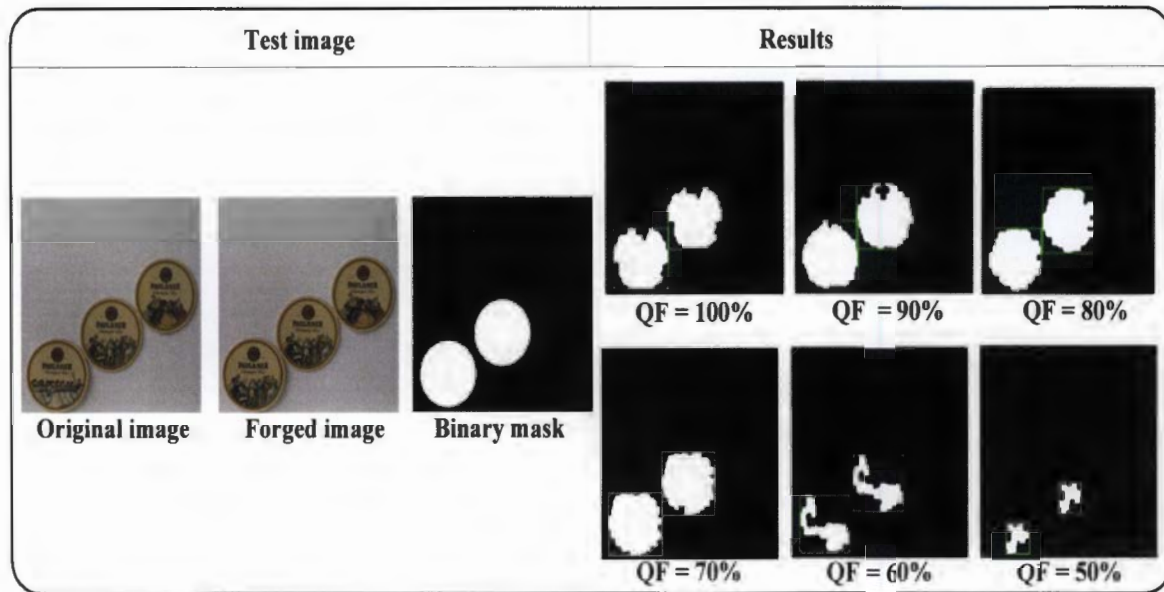
**Fig. 4.2.** Results for different manipulation techniques (trees)



**Fig. 4.3.** Results for different manipulation techniques (coins)

Fig. 4.4 illustrates the performance of our algorithm for the compressed forged images in the database (i.e. JPEG compression). We tested the performance of our proposed algorithm on different quality factors from the database (QF): 100%, 90%, 80%, 70%, 60%, and 50%. The value of quality factors determines the degree of loss in the compression process. Low quality factors means high compression level which lead to small file size (i.e. image size). High quality factors means low compression level which leads to large file size. As we can see in Fig. 4.4, for compressed images with quality factors of 70% to 100% the results are very accurate. Although decreasing the quality factor (QF) below 70% affects the accuracy of our results as we can see in Fig. 4.4, the results maintain indicating part of the forged region and it does not include any wrong forged objects.




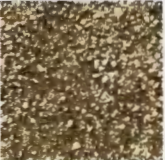






















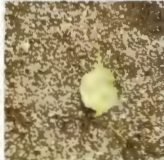
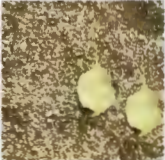








**Fig. 4.4.** Results for different JPEG quality factors (QF)

Fig. 4.5 illustrates the performance of our proposed algorithm for a sample of test images after applying different manipulation techniques namely: translation, rotation, scaling. The first row in Fig. 4.5 shows the original image, the forged version, the forged regions from the database, and our result for the translation type. As we can notice from first row in Fig. 4.5 that our algorithm indicates the forged region perfectly compared to the database. As we can notice from the results, our algorithm does not detect any unmatched regions. For the rotation technique, Fig. 4.5 shows the results of applying our algorithm after applying different rotation angles on three test images. The rotation angles in the bird images are 2, 4, -4 degrees respectively, and the rotation in the tree image is 2 degrees. Also the rotation angle in the label image is -3 degrees. As we can notice that our algorithm succeeds to indicate the forged regions even after rotating the forged region. Fig. 4.5 also shows the performance of our proposed algorithm after scaling the forged regions. For the balcony

image, tree leaves, roof, the scale factors are 102%, 96%, and 115% respectively. As we can notice in the tree leaves example, our algorithm succeeded to indicate the forged leave without the upper small part. That is because we use 16 x 16 blocks, therefore we neglect any region with smaller size. From Fig. 4.2, 4.3, 4.4, and 4.5 we can notice that our proposed algorithm does not indicate any wrong forged objects.

Table 4.1 shows a comparison between our proposed algorithm and five other methods. We compared all methods according to four main factors. These factors are; image representation technique, number of blocks, extracted feature dimension, and finally, the effectiveness against different distorted operations (i.e. JPEG compression, blurring, etc.) and geometrical transformation (i.e. scale, rotation). As we can see from Table 4.1, our algorithm needs shorter feature dimension (i.e. 16-dimension), to identify the forged regions, compared to Fridrich et al. [36], Popescu et al. [42], and Jing et al. [71] methods that need 64-dimension, 32-dimension, 128-dimension extracted features respectively. Furthermore, Fridrich et al. and Popescu et al. methods cannot detect the forgery after applying some post-processing techniques (i.e. rotation, scaling, and color reduction, etc.) Also we can notice that although, Li et al. [72] and Cao et al. [37] methods need shorter feature dimension than our algorithm (i.e. 8-dimension and 4-dimension extracted features respectively), but their methods cannot detect the forgery if some post-processing techniques are applied on the forged region such as; scaling, rotation, color reduction, and brightness change. As a result we can conclude that our proposed algorithm can detect the forgery even after applying different distortion operations and geometrical transformation on the forged region with reasonable extracted feature dimension.

Translation				
	Original image	Forged image	Binary mask	Our result
Rotation				
	Original image	Forged image	Binary mask 1	Our result 1
				
	Binary mask 2	The result 2	Binary mask 3	Our result 3
				
	Original image	Forged image	Binary mask	Our result
Scaling				
	Original image	Forged image	Binary mask	Our result
				
	Original image	Forged image	Binary mask	Our result
				
	Original image	Forged image	Binary mask	Our result
				
	Original image	Forged image	Binary mask	Our result

**Fig. 4.5.** Results for sample of test images (CoMoFoD database)

**Table 4.1.** Comparison results of the six approaches for a 512 x 512 image with 8 x 8 window block.

Algorithm	Fridrich	Popescu	Li	Cao	Jing	Proposed algorithm
Image representation	DCT	PCA	DWT & SVD	Block representing	SIFT	Steerable pyramid
Block number (8 x 8)	255,025	255,025	62001	255,025	>2000	255,025
Feature dimension	64	32	8	4	128	16
Translation	✓	✓	✓	✓	✓	✓
Scale	×	×	×	×	✓	✓
Rotation	×	×	×	×	✓	✓
JPEG compression	✓	✓	✓	✓	✓	✓
Noise addition	×	✓	×	✓	✓	✓
Colour adjustment	×	×	×	×	×	✓
Blurring	×	×	×	✓	✓	✓
Color reduction	×	×	×	×	×	✓
Brightness change	×	×	×	×	×	✓

### 4.3 Conclusion

In this chapter, we presented a new blind forgery detection algorithm for copy move forgery. We applied copula based mutual information on the forged image only to identify whether there are any forged regions in the image or not. In this work we used five copula functions namely: Gaussian, Marshall-Olkin, Clayton, Frank, and Gumbel copulas. Our work showed how copulas can be used not only to measure the image quality, but also to localize the copy move forgery in the image with high accuracy. Our proposed algorithm was able to identify the forged region in the images even after applying scaling process on the forged region by a factor  $\approx \pm 20$ , and also after applying rotation angle  $\approx \pm 5$  degrees on the forged region, and by any translation distance. The proposed forgery detection algorithm can be applied on security documents, insurance documents, and even medical documents. The results of testing the proposed algorithm on CoMoFoD database showed the accuracy of it even after applying different types of manipulations (i.e. translation, rotation, scaling) and also after applying reasonable post-processing techniques such as JPEG compression, blurring, noise adding, and color reduction.



## **Chapter 5**

### **Conclusion and Future Work**

#### **5.1 Conclusion**

In this thesis, five copula functions were employed to measure the image quality and to detect the copy move forgery in the digital images as well. For the proposed image quality measure (i.e. chapter 3), copulas were applied on the tested images from the LIVE image database to compare our image quality measure algorithm with HVS (LIVE database result) and other three popular image quality measures VIF, UQI, and SSIM. Our proposed algorithm obtained comparable results with all of them. We also showed how and why the results were different in all methods including all objective methods in the case of fast fading distortion type. We also investigated the effect of the sub band choice on the performance of our proposed algorithm. The results showed the flexibility of our algorithm to choose any sub band to measure the image quality. In addition, we presented a full reference image quality assessment algorithm to indicate the unmatched regions in two images. The results showed how accurate our algorithm is (i.e. it detects the difference in one pixel). The advantage of using copula based image quality measure is that the value of it changes with any slight difference between the images which enables us to detect a very slight difference that cannot be notice by the naked eye. The results showed that the choice of the sub band does not affect

the results for four types of distortion (Gaussian blur, JPEG, JPEG 2000, and white noise). We can also notice that the most popular image quality measures such as VIF, SSIM, and UQI have different results than the HVS results in few distorted versions as well. Moreover, copulas' simple calculations and the fact that they are not dependent on the distribution of the used data are the observed advantages in this work.

Finally, a new blind forgery detection algorithm for copy- move forgery was proposed in chapter 4. We applied copula based mutual information on the forged image only to identify whether there are any forged regions in the image or not. In this work we used five copula functions namely: Gaussian, Matshall-Olkin, Clayton, Frank, and Gumbel copulas. Our work showed how copulas can be used not only to measure the image quality, but also to localize the copy move forgery in the image with high accuracy. The proposed forgery detection algorithm can be applied on security documents, insurance documents, and even medical documents. The results of testing the proposed algorithm on CoMoFoD database showed the accuracy of it even after applying different types of manipulations (i.e. translation, rotation, scaling) and reasonable post-processing techniques such as JPEG compression, blurring, noise adding, and color reduction.

## **5.2 Future work**

From the results presented in this thesis, we can notice the effectiveness and the accuracy of the results by applying copula functions. Therefore, researchers, in the future, can investigate the use of copula functions in image processing applications such as indexing, retrieving, registration, classification, and forgery detection. Moreover, the researchers can

improve our blind forgery detection algorithm to be able to detect the forgery region even after applying other manipulation techniques and post-processing methods other than those we addressed in this thesis. Finally, in the future work, researchers can improve our proposed algorithm to be able to detect the forgery even after applying rotation with a large rotation angles and large scale factors.



## Bibliography

- [1] A. Sklar, "Fonctions de répartition à  $n$  dimensions et leurs marges," Publications de l'Institut de Statistique de L'Université de Paris, vol. 8, pp. 229–231, 1959
- [2] R. S. Calsaverini and R. Vicente, "An information-theoretic approach to statistical dependence: copula information," A Letters Journal Exploring the Frontiers of Physics, epl, vol. 88, pp. 1-6, December 2009
- [3] T. S. Durrani and X. Zeng, "Copula based divergence measures and their use in image registration," 17th European Signal Processing Conference, pp. 1309-1313, August 2009, Glasgow, Scotland
- [4] S. AlZahir and F. Kashanchi, "A new image quality measure," 26th IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), May 2013, Regina, SK, Canada
- [5] A. George, S. J. Livingston, "A survey on full reference image quality assessment algorithms," International Journal of Research in Engineering and Technology, vol. 2, no. 12, pp. 303-307, December 2013
- [6] A. G. George, A. K. Prabavathy, "A survey on different approaches used in image quality assessment," International Journal of Emerging Technology and Advanced Engineering, vol. 3, no. 2, pp. 197-203, February 2013
- [7] J. Redi, H. Liu, H. Alers, R. Zunino, and I. Heynderickx, "Comparing subjective image quality measurement methods for the creation of public databases," Proceedings of SPIE-IS & T Electronic Imaging, SPIE, vol. 7529, pp. 752903-1 -752903-11, 2010, San Jose, CA, USA.
- [8] D. M. Rouse, R. P'epion, P. Le Callet, and S. S. Hemami, "Tradeoffs in subjective testing methods for image and video quality assessment," in Human Vision and Electronic Imaging XV, Proceedings of SPIE, vol. 7527, San Jose, California, USA, January 2010
- [9] A. Lahouhou, E. Viennet, and A. Beghdadi, "Selecting low-level features for image quality assessment by statistical methods," Journal of Computing and Information Technology, vol. 2, pp. 183–189, 2010

- [10] H. R. Sheikh, and A. C. Bovik, "Image information and visual quality," IEEE Transactions on Image Processing, vol. 15, no. 2, pp. 430- 444, Feb. 2006
- [11] Z. Wang and A. C. Bovik, "A universal image quality index," IEEE Signal Processing Letters, vol. 9, pp. 81–84, March 2002
- [12] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transaction on Image Processing, vol. 13, no. 4, pp. 600–612, 2004
- [13] S. Al Zahir, and R. Hammad, "New gage for measuring image quality," 28th IEEE Canadian Conference on Electrical and Computer Engineering, CCECE, May 2015, Halifax, Nova Scotia, Canada
- [14] Z. Wang, "Applications of objective image quality assessment methods," IEEE Signal Processing Magazine, vol 137, pp. 137-142, November 2011
- [15] R. C. Gonzalez and R. E. Woods, "Digital image processing," Addison-Wesley, New York, 1992
- [16] Z. Wang and A. C. Bovik, "Mean squared error: love it or leave it?: A new look at signal fidelity measures," IEEE Signal Processing Magazine, pp. 98-117, January 2009
- [17] K. Silpa and S. Aruna Mastani, Comparison of image quality metrics, International Journal of Engineering Research & Technology, IJERT, vol. 1, no. 4, pp. 1-5, June 2012
- [18] X. Gao, W. Lu, D. Tao, X. Li, "Image quality assessment and human visual system," Visual Communications and Image Processing, Proceedings of SPIE, vol. 7744 77440Z-1, 2010
- [19] A. Toet and M. P. Lucassen, "A new universal colour image fidelity metric," Displays, Elsevier, vol. 24, no. 4-5, pp. 197 – 207, 2003
- [20] Z. Wang, E. P. Simoncelli and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," Proceedings of the 37th IEEE Asilomar Conference on Signals, Systems and Computers, Nov. 19-12, 2003, Pacific Grove, CA
- [21] K. Gu, G. Zhai, X. Yang, and W. Zhang "An improved full reference image quality metric based on structure compensation", APSIPA ASC, pp. 1 - 6, 2012
- [22] C. J. Cellucci, A. M. Albano, and P. E. Rapp, "Statistical validation of mutual information calculations: comparison of alternative numerical algorithms," Physical Review. The American Physical Society, vol. 71, no. 6, pp. 1-14, 2005

- [23] G. Mercier, S. Derrode, W. Pieczynski, J. M. Nicolas, A. J. Chardin, and J. Inglada, "Copula-based stochastic kernels for abrupt change detection," in Proceedings of International Geo-science and Remote Sensing Symposium conference, IGARSS, pp. 204–207, 2006, Denver, Colorado, USA
- [24] T. Qazi, K. Hayat, S. U. Khan, S. A. Madani, I. A. Khan, J. Kolodziej, H. Li, W. Lin, K. C. Yow, and C. Z. Xu, "Survey on blind image forgery detection," IET Image Processing, the Institution of Engineering and Technology, vol. 7, no. 7, pp. 660–670
- [25] B. Mahdian, S. Saic, "Blind methods for detecting image fakery," IEEE Aerospace, Electronics System Magazine, vol. 4, no. 25, pp. 18-24, 2010
- [26] B. L. Shivakumar, S. S. Baboo, "Detecting copy-move forgery in digital images: a survey and analysis of current methods," Global Journal of Computer Science Technology, vol. 10, pp. 61-65, 2010
- [27] S. Z. M. Shaid, "Estimating optimal block size of copy-move attack detection on highly textured image," Thesis Submitted to the University of Technology, Malaysia, 2009.  
Available at:  
[http://www.csc.fsksm.utm.my/syed/images/files/publications/thesis/estimating\\_optimal\\_block\\_size\\_for\\_copy-move\\_attack\\_detection\\_on\\_highly\\_textured\\_image.pdf](http://www.csc.fsksm.utm.my/syed/images/files/publications/thesis/estimating_optimal_block_size_for_copy-move_attack_detection_on_highly_textured_image.pdf)
- [28] E. Kee and H. Farid "A perceptual metric for photo retouching," PNAS Proceeding of the National academy of science of the United State of America, November 28, 2011
- [29] <http://www.flickr.com/photos/manekineko>, accessed October 26, 2011
- [30] G. Boato, F. G. B. D. Natale, P. Zontone, "How digital forensics may help assessing the perceptual impact of image formation and manipulation," Proceedings of 5th International Workshop on Video Processing and Quality Metrics for Consumer Electronics, VPQM, 2010.
- [31] X. F. Li, X. J. Shen, H. P. Chen, "Blind identification algorithm for the retouched images based on bi-Laplacian," Journal of Computer Applications, vol. 31, pp. 239–242, 2011
- [32] S. Q. Saleh, "Image splicing and copy-move forgery detection," King Saud University College of Computer & Information Sciences Department of Computer Science, Thesis, December 2012
- [33] B. Wang and X. Kong, "Image splicing localization based on re-demosaiicing," Advances in Information Technology and Industry Applications, pp. 725–732, 2012
- [34] F. Peng, Y. Nie, M. Long, "A complete passive blind image copy-move forensics scheme based on compound statistics features," Forensic Science International, vol. 212, pp. e21– e25, 2012

- [35] B. L. Shivakumar and L. D. S. S. Baboo, "Detecting copy-move forgery in digital images: a survey and analysis of current methods," *Global Journal of Computer Science and Technology*, vol. 10, no. 7, pp. 61-65, 2010
- [36] J. Fridrich, D. Soukal, and J. Lukas, "Detection of copy-move forgery in digital images," *Proceedings of the Digital Forensic Research Workshop*, 2003, Cleveland, OH, USA
- [37] Y. Cao, T. Gao, L. Fan, and Q. Yang, "A robust detection algorithm for copy-move forgery in digital images," *Forensic Science International* 214, Elsevier, pp. 33-43, 2012
- [38] X. Zhao, J. Li, S. Li, and S. Wang, "Detecting digital image splicing in chroma spaces," *Digital Watermarking*, pp. 12-22, 2011
- [39] W. Wang, J. Dong, and T. Tan, "Image tampering detection based on stationary distribution of Markov chain," *17th IEEE International Conference on Image Processing (ICIP)*, pp. 2101-2104, 2010
- [40] M. Sridevi, C. Mala and S. Sandeep, "Copy move image forgery detection in a parallel environment," *Computer Science & Information Technology, CS&IT*, pp. 19-29, 2012
- [41] V. Christlein, C. Riess, and E. Angelopoulou, "A study on features for the detection of copy-move forgeries," in *GI SICHERHEIT*, pp. 105-116, October 2010
- [42] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," *Dartmouth College, Tech. Rep. TR2004-515*, August, 2004
- [43] T. K. Sarode, N. Vaswani, "Region duplication forgery detection using hybrid wavelet transforms," *Proceedings of International Journal of Computer Applications*, vol. 90, no 11, March 2014
- [44] E. S. Khan, E. A. Kulkarni, "An efficient method for detection of copy-move forgery using discrete wavelet transform," *International Journal on Computer Science and Engineering, IJCSE*, vol. 02, no. 05, pp. 1801-1806, 2010
- [45] A. N. Myna, M. G. Venkateshmurthy, C. G. Patil, "Detection of region duplication forgery in digital images using wavelets and log-polar mapping", of *International Conference on Computational Intelligence and Multimedia Applications*, Volume 3, no. 13-15, pp. 371-377, December 2007, Sivakasi, India.
- [46] A. D. Warbhe, R. V. Dharaskar "Blind method for image forgery detection: a tool for digital image forensics", *National Conference on Innovative Paradigms in Engineering & Technology, NCIPET*, Proceedings published by *International Journal of Computer Applications, IJCA*, no. 11, pp. 37-40, 2012
- [47] F. Kashanchi, "Copula based rigid-body image registration," Thesis Submitted to University of Northern British Columbia, 2012

- [48] X. Zeng and T. S. Durrani, "Estimation of mutual information using copula density function", *Electronics Letters*, vol. 47, no. 8, 14th April 2011
- [49] R. Salinas-Gutiérrez, A. Hernández-Aguirre, and E. R. Villa-Diharce, "Dependence trees with copula selection for continuous estimation of distribution algorithms," *Proceedings of the 13th annual conference on Genetic and Evolutionary Computation, GECCO*, July 2011 Dublin, Ireland
- [50] C. M. Thomas and T. A. Joy, "Elements of information theory," John Wiley & Sons, Inc., 1991, New York, USA
- [51] F. Galton, "The British association: section ii, anthropology: opening address by Francis Galton, F.R.S., etc., president of the Anthropological Institute, president of the section," Vol. 32, no. 830, pp. 507–510, September, 1885
- [52] F. Galton, "Regression towards mediocrity in hereditary stature," *Journal of the Anthropological Institute of Great Britain and Ireland*, vol. 15, pp. 246–263, 1886
- [53] K. Pearson, "Notes on regression and inheritance in the case of two parents," *Proceedings of the Royal Society of London*, vol. 58, pp. 240-242, June 20, 1895
- [54] S. M Stigler, "Francis Galton's Account of the Invention of Correlation". *Statistical Science*, vol. 4, no. 2, pp. 73-79, 1989
- [55] M. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 2, pp. 81–89, 1938
- [56] R. B., Nelsen, "Kendall tau metric," Hazewinkel, Michiel, *Encyclopedia of Mathematics*, Springer, 2001
- [57] U. Cherubini., E. Luciano, and W. Vecchiato, 2004, "Copula methods in finance," John Wiley & Sons, 2004, England
- [58] P. J. Burt and E. H. Shashua, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31 no. 4, April, 1983
- [59] H. Greenspan, S. Belongie, P. Perona, R. Goodman, S. Rakshit, and C. Anderson, "Overcomplete steerable pyramid filters and rotation invariance," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 222-228, 1994
- [60] J. Tom, K. N. Nair, " Novel method for contrast enhancement of digital images using image fusion," *International Journal of Engineering Research & Technology, IJERT*, vol. 3, no. 10, pp. 249-253, October 2014



- [61] S. Singh, N. S. Grewal, H. Singh, "Multi-resolution representation of multifocus image fusion using Gaussian and Laplacian pyramids," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 11, pp. 1639-1642, November 2013
- [62] E. P. Simoncelli, W.T. Freeman, "The steerable pyramid: a flexible architecture for multi-scale derivative computation," *2nd IEEE International Conference on Image Processing*, Washington, DC, vol. III, pp. 444-447. October, 1995
- [63] M. Unser, N. Chenouard, and D. Van De Ville, "Steerable pyramids and tight wavelet frames in  $L_2(R^d)$ ," *IEEE Trans. Image Processing*, vol. 20, no. 10, pp. 2705-2721, October 2011
- [64] A. Karasiridis and E. Simoncelli, "A filter design technique for steerable pyramid image transforms," *International Conference Acoustics Speech and Signal Processing*, May 1996, Atlanta.
- [65] H. R. Sheikh, Z. Wang, L. Cormack and A. C. Bovik, "LIVE image quality assessment database release 2", <http://live.ece.utexas.edu/research/quality>
- [66] R. Wajid, A. B. Mansoor, and M. Pedersen, "A study of human perception similarity for image quality assessment," *Colour and Visual Computing Symposium (CVCS)*, IEEE, September 2013, Gjøvik, Norway
- [67] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, "The Caltech-UCSD birds-200-2011 dataset." *Computation & Neural Systems Technical Report*, CNS-TR-2011-001, Available at: <http://www.vision.caltech.edu/visipedia/CUB-200-2011.html>
- [68] <http://www.abunawaf.com/post-7883.html>
- [69] S. Bayram, H. T. Sencar, and N. Memon, "A survey of copy-move forgery detection techniques," *IEEE Western New York Image Processing Workshop*, 2008
- [70] D. Tralic, I. Zupancic, S. Grgic, M. Grgic, "CoMoFoD - new database for copy-move forgery detection", *Proceedings of 55th International Symposium, ELMAR-2013*, pp. 49-54, September 2013.
- [71] Li Jing, and Chao Shao, "Image copy-move forgery detecting based on local invariant feature," *Journal of Multimedia*, vol. 7, no. 1, pp. 90-97, February 2012
- [72] G. H. Li, Q. Wu, D. Tu, and S. Sun, "A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD," *IEEE International Conference on Multimedia and Expo*, pp. 1750-1753, July 2007, Beijing, China

## **Appendix A**

### **A.1 Gaussian and Laplacian pyramids**

The Gaussian Pyramid decomposes the image into a set of low pass filtered images, which when piled one on top of the other build the Gaussian pyramid. Fig. A.1 shows the result of applying Gaussian pyramid technique on "Lena" image. In contrast, the Laplacian Pyramid decomposes the image into a set of band pass filtered images [58]. The Laplacian pyramid is obtained by calculating the difference between successive Gaussian levels [58, 59]. The Laplacian pyramid images are like edge images, which means that most of its elements are zeros [60].

It is frequently used in image processing and pattern recognition tasks because of its ease of computation [61]. Fig. A.2 shows the result of applying Laplacian pyramid technique on "Lena" image.



Fig. A.1. Gaussian pyramid technique on Lena image: (a) original image; (b) the resulted sub bands after applying Gaussian pyramid technique on the original image

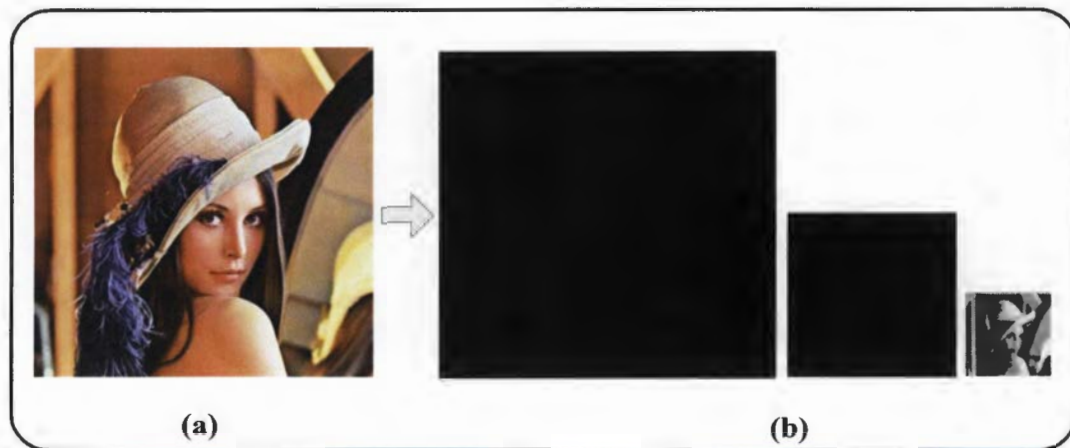


Fig. A.2. Laplacian pyramid technique on Lena image: (a) original image; (b) the resulted sub bands after applying Laplacian pyramid technique on the original image



## A.2 Image retouching

Image retouching is a class of forensic methods that make a slight change in the image for various artistic and commercial purposes. The retouching is mostly used to enhance or reduce the image features. Usually image retouching is realised by altering the color or texture of the objects. Fig. A.3 shows original images and the retouched versions of them. This type of forgery is also known as the image enhancement for its use to improve the facial features.



Fig. A.3. Image retouching technique [28, 29]

Forgery detection in case of image retouching, involves finding the enhancements, blurring, illumination and color changing. Image retouching detection may be an easy task only if the original image is available. Otherwise, the task may be very challenging [24]. Boato et al. [30], a non-blind detection technique is presented that takes into account the global modifications in the image. This detection technique detects the negative or positive changes within the image on the basis of image quality measures (IQMs) and the binary similarity measures. This technique is quite effective and produces accurate results in cases, when the image is highly modified. In [32] a blind identification algorithm, for the retouching forgery is based on the bi-Laplacian filtering is introduced. This technique searches for each block of the image on the basis of a KD tree and derives the adjacent matching blocks. This technique is applicable to the uncompressed images and compressed high resolution images.

### **A.3 Image splicing**

Image splicing is one of the most common image tampering techniques to create forged images. In image splicing, a region from one image is copied and pasted into another image which produces composite image called spliced image. This type of forgery is a challenging issue from tamper detection point of view [32]. As shown in Fig. A.4, by copying a spliced portion from the source image into a target image, one can create composite scenery to cheat others. The availability of the current state-of-the-art image editing software enabled even non-professional users to perform splicing without much difficulty. It is a challenging issue to detect the spliced image in a fully automated way. In

many cases, humans can detect such change easily. However, developing fully automatic algorithms to do the same task is extremely difficult. The difficulties come from the fact that a natural image usually consists of complicated edges of arbitrary magnitudes, orientations. Therefore hard to design an edge detector which can distinguish the changes caused by the forgery splicing and the changes that are parts of the image signal [32].

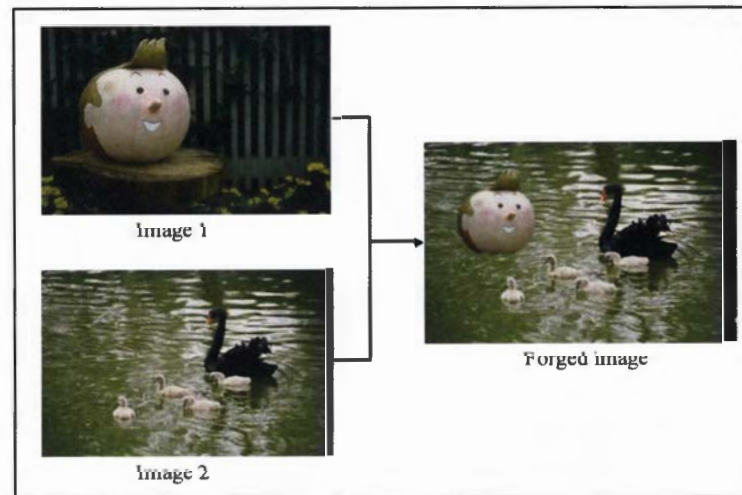


Fig. A.4. Example on image splicing

B. Wang et al. in [33] introduced a technique to localize image splicing that depended on re-demosaicing. The demosaicing was used in order to estimate the spliced image natural counterpart. Then a comparison between the natural counterpart of the tested image and the image itself was applied for classifying every pixel as forged or authentic pixel. According to the results of classification, a binary image with the same size of the input test image was generated. This binary image referred to the spliced contour after post-processing operation. The suspect image natural counterpart is estimated by Color Filter Array (CFA). After that the distance is calculated by taking the absolute value of the difference between counterpart

of the tested image and the original image. The proposed technique was examined on DVMM dataset that contains 180 tampered and 183 authentic images. The detection results of spliced tampering showed clear contour for the pasted region in white points.

F. Peng et al. in [34] introduced a blind method to detect image splicing using Sensor pattern noise (SPN). First, the input color image was transferred into the gray scale image then wavelet transform which based on de-noise operation was used and Sensor pattern noise (SPN) was calculated. Then, feature vector  $f$  was constructed from variance Sensor pattern noise (SPN), signal noise ratio (SNR), information entropy (H), and average energy gradient (AEG). After that, the original image  $I$ , de-noised image  $I_d$ , and difference image (i.e.  $N = I - I_d$ ) were divided into non-overlapped sub-blocks using sliding window with size of  $w \times w$ . The feature vector  $f_i$  was calculated for each block  $B(i, j)$ . Then, Euclidean distance was calculated between  $f_i$  and  $f$ . And the similarity  $S(f_i, f)$  between the block feature vector and whole image feature vector was computed and compared with a pre-identified threshold  $T$ . If  $S(f_i, f) < T$ , then the block  $B(i, j)$  considered as tampered region and it is added into the tampered region set. Finally, the previous steps were repeated for all blocks. The proposed method was tested on 100 faked images. The accuracy was 92.26% and the average of false detection rate was 8.17%.