UNIVERSITÉ PARIS 13

ÉCOLE DOCTORALE GALILÉE

THÈSE

Signaux et Images

DANG Thanh Trung

APPROCHE PERCEPTUELLE POUR LA RETOUCHE D'IMAGES

ALGORITHMES, ÉVALUATION ET DÉTECTION

Thèse dirigée par

Prof. A. BEGHDADI and Dr. M-C LARABI

Soutenue le 26-06-2014

JURY

M. Frederic DUFAUX	Rapporteur	Telecom ParisTech, France
Mme. Stefania COLONNESE	Rapporteur	Sapienza University of Rome, Italy
M. Faouzi ALAYA-CHEIKH	Examinateur	Gjøvik University College, Norway
Mme. Anissa MOKRAOUI	Examinateur	Université Paris 13, France
M. Mohamed-Chaker LARABI	Co-Directeur de thèse	Université de Poitiers, France
M. Azeddine BEGHDADI	Directeur de thèse	Université Paris 13, France

Acknowledgments

Three years of living away from home, away from family, away from friends to pursue a dream, *PhD course*. It is really not easy. The dream would not come true without support, advice, encouragement and sharing of many people during the research. In consonance with, there have been a lot of happiness, sadness and other emotions to be experienced. Finally, now is the moment for me to express my sincere gratitude to:

My supervisor, Prof. Azeddine BEGHDADI, for his generous support and intellectual guidance throughout my years as a graduate student; his insightful advice, clear vision, many suggestions, and endless efforts to be available for many educational discussions, were invaluable;

my co-supervisor, Dr. Mohamed-Chaker LARABI, whose friendship and encouragement were helpful; and whose profound and concrete comments and suggestions on drafts of this thesis have enabled my deepened understanding of my work;

my committee members, Dr. Frederic DUFAUX and Dr. Stefaina COLONNESE, for valuable insights and suggestions which have significantly improved this thesis in both structure and contents;

my external examiners, Dr. Faouzi ALAYA-CHEIKH and Prof. Anissa MOKRAOUI, for their critical comments and constructive suggestions on this thesis;

my beloved wife, PHAM Thi Mai Bao, and my small princess, DANG Mai Bao Linh, who shared with the pains and happiness during the course of this work; her endless support, sacrifice, and understanding kept me going through it all;

my parents, DANG Van Duong and NGUYEN Thi Song A, who born me and first taught me the importance of education as well as raised me with a love of science and supported me in all my pursuits;

all the other members of my family, parents in law, younger brothers and younger sisters who supports me in every possible ways at all times in life;

my dear friends in Vietnam, especially PHUONG-LE-CUN, KIEN-THOA-TUAT, ... who persistently encouraged me as well as aided my family during the most difficult period when I am away from home;

my dear friends in France, notably LOC-PHUONG, DUNG, NGHIA, TRANG BEO, NGUYEN-NGA, TUNG-DUNG, HOAN-LY, HOAN-NGA, BAO-CUC-LEO, BANG, PHAM

NHUNG, ... (it is impossible to list all here but you know who you are!) who made days spent together feel like seconds and help me defeat the obstacles in adapting to lonely life here;

and finally, my comrades, WALID, SAMEH, who shared together the ups and downs in mood in our ways of seeking the truth.

My gratitude is more than words that I can say.

Thank you all sincerely for everything which makes my dream come true.

Abstract

Image inpainting technique has emerged as one of the efficient solutions for restoring degraded images and even video and 3D visual contents. It is essential for restoration and conservation of archeological objects. Indeed, during the excavation of historic sites, many valuable artifacts such as statues, buildings, wall paintings, etc., have been unearthed and recovered. Unfortunately, most of them have been damaged or fractured in ancient times. With the aim to learn from these objects, digital techniques are used to restore them using some a priori knowledge provided by archeologists and artists. This helps to reconstruct, preserve and maintain these valuable artistic works. Recently, advances of computer technology and image acquisition systems made possible the transition from traditional manual retouching methods to digital techniques. It has opened up a very new and interesting research field within image processing, namely ***digital image inpainting***.

In this thesis, our work aims at analyzing and reviewing the state-of-the-art methods for image inpainting. Three groups of inpainting methods are identified. The advantages and drawbacks of each approach are analyzed using objective and subjective criteria. Based on this review, a new approach of image inpainting is introduced and compared to literature. The proposed method provides good performance in terms of quality and computation efficiency. Another contribution lies in the proposal of a new quality metric dedicated to image inpainting. To the best of our knowledge the inpainted images are very often evaluated subjectively or by means of some objective metrics far from being adapted to the peculiarities of image inpainting criteria. The last contribution of this work is a less investigated problem related to inpainting detection. This is motivated by the fact that, due to extensive research and rapid growth of technology, the output quality of inpainting algorithms became more than ever realistic and sophisticated. The inpainted regions are hard to detect by viewers, even for experts. As a result, these inpainted images could be used for different purposes, including digital tampering. Therefore, the last topic of the work is devoted to inpainting detection, also seen as an inverse problem of inpainting. Although, many papers have been introduced for forgery detection, there is almost no study about image inpainting forgery. Accordingly, a novel approach for inpainting detection is introduced based on the knowledge gathered in the previous steps of the thesis. Finally, the performance of proposed solutions is carefully evaluated with regards to human judgment as well as in comparison with the existing methods though a series of experimental studies.

Résumé

La technique d'inpainting (retouche d'images) est considérée comme l'une des solutions efficaces pour la restauration dimages dégradées et, même la vidéo ou les contenus 3D stéréoscopiques. Il s'agit d'un outil essentiel pour la restauration et la conservation des objets archéologiques. En effet, lors de fouilles de sites historiques, de nombreux objets de valeur tels que des statues, des bâtiments, des peintures murales, etc., sont découverts et récupérés. Malheureusement, la plupart d'entre eux se retrouvent endommagés ou fracturés. Dans l'optique d'apprendre de ces objets, les techniques numériques sont utilisées pour les restaurer en utilisant des connaissances a priori fournies par les archéologues et les artistes. Cela permet de reconstruire, préserver et maintenir ces oeuvres artistiques de grande valeur. En effet, les progrès des technologies et des systèmes d'acquisition d'images ont permis la transition des méthodes traditionnelles de retouche manuelle vers des techniques numériques, ouvrant un nouveau domaine de recherche appelé inpainting d'images.

Dans cette thèse, notre travail visait à analyser et examiner les méthodes de la littérature des techniques d'inpainting d'images qui se décomposent en trois groupes. Les avantages et les inconvénients de chaque approche sont analysés finement en utilisant des critères objectifs et subjectifs. Sur la base de cette étude, une nouvelle approche d'inpainting d'images est proposée et confrontée à la littérature. Elle offre de bonnes performances en termes de qualité de rendu et d'efficacité calculatoire. Une autre contribution réside dans la proposition d'une nouvelle métrique de qualité dédiée à l'évaluation des techniques d'inpainting. Au regard de l'état de l'art dans le domaine, les images ayant subi des retouches par inpainting sont très souvent évaluées subjectivement ou au moyen de quelques mesures objectives loin d'être adaptées aux particularités de cette technique. La métrique proposée offre à la fois de bonnes performances et une adaptabilité au domaine. La dernière contribution de ce travail est un problème moins étudié et lié à la détection d'inpainting. Ceci est motivé par les efforts importants de recherche sur la thématique ainsi que le développement rapide de la technologie. La conséquence est que la qualité des retouches par inpainting est devenue plus réaliste et sophistiquée que jamais. Les régions retouchées sont alors difficiles à détecter par les utilisateurs, même les experts. Ainsi, ces images pourraient être utilisées à différentes fins, légales comme illégales. Malgré l'importance du sujet, très peu de travaux l'ont adressés à cause de sa complexité. C'est sur cette base qu'une nouvelle approche pour la détection d'inpainting est proposée dans cette thèse, en

exploitant les connaissances acquises lors des étapes précédentes. Enfin, la performance des solutions proposées est soigneusement évaluée à travers plusieurs expérimentations, et ce au regard du jugement humain ainsi qu'en comparaison avec les méthodes existantes.

Contents

Chapter 1

Introduction

"All truths are easy to understand once they are discovered. The point is to discover them."

Galileo Galilei

Contents

1.1 Problem statement

Pompeii and *Herculaneum* were once thriving towns in the Bay of Naples. Both towns have rich histories influenced by Greeks, Oscans, Etruscans, Samnites and finally the Romans. However, they were destroyed during an eruption of the volcano Mount Vesuvius in 79 AD. The volcano erupted and covered completely the town and its inhabitants in many tons of pumice and volcanic ash.

The discovery of *Pompeii* is of great importance for our modern-day understanding of the ancient Roman-Italic world - partly because the more public and monumental ruins left behind by Imperial Rome have often been misleading. The excavations at *Pompeii* and *Herculaneum* offer an intact vision of daily life in a Roman society in all its aspects.

During the excavation, many valuable artifacts such as statues, buildings, wall paintings, etc., have been unearthed and recovered. Unfortunately, most of them have been damaged or fractured in ancient times. On way to preserve and maintain these valuable artistic works, they should be digitalized and restored by the art restoration artists, who could bring them back to the original or to close-to-original state.

In fact, it is impossible or unnecessary to restore these paintings to the original version because they are unique and there are no written documents to study. Thus, the main task of artists is filling in existent gaps in the paintings with visually pleasing content. On the other hand, it can be said that almost as old as art itself is the practice of making modifications to paintings, in such a way that if an observer would look at the modified work, without knowing the original one, he could not be able to perceive any alteration. This restoration is traditionally carried out by experts, such as museum art restorers, and it is commonly known as retouching or inpainting. Its desired outcome is to make a damaged artwork more discernible, while restoring its unity.

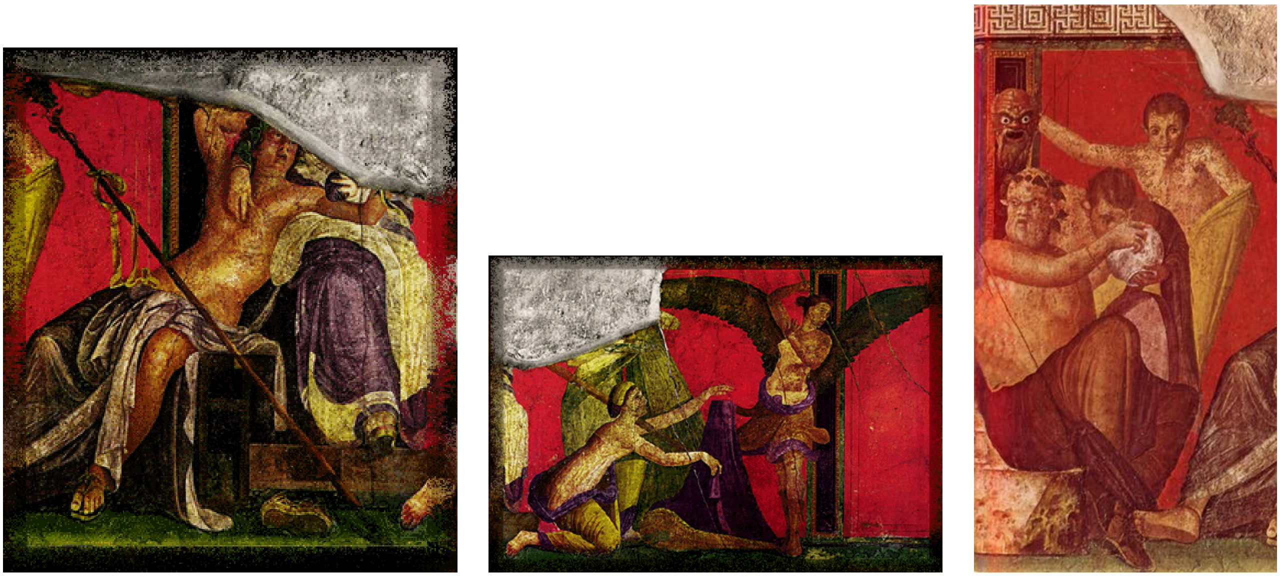


Figure 1.1. Some wall paintings in Villa of the Mysteries, Pompeii.

Progress in computer technology has changed conventional archaeology and its exhibition. Advances of technology made possible the transition from traditional manual retouching methods to digital techniques. It has opened up a very new and interesting research in the field of image processing, *digital image inpainting.*

Figure 1.2. Painting restoration work in the museum.

Researcher working on different applications of inpainting have adopted different names though their own individual characteristics: “error concealment” in telecommunications; “image disocclusion” or “image interpolation” in image processing.

The objectives of inpainting are numerous. They have been employed in various applications such as medical and astronomical imaging, film restoration, image and video coding and many others [1–3]. Meanwhile, in computer science, much research similar to the inpainting problem has also been carried out in the context of image replacement, image interpolation, error concealment, digital restoration of ancient paintings for conservation purposes, disocclusion in vision research [4, 5].

To illustrate the use of inpainting restoration in typical applications, an example of restoration is shown in Figure 1.3. In this figure, the original images with defects such as cracks and scratches are shown in Figure 1.3-a,c and the corresponding outputs are shown in Figure 1.3-b,d. The inpainting process is implemented in two different ways but with a common goal, i.e., an undetectable modification by viewers.

1.2 Thesis objectives

Firstly, our work aims to analysis and review state-of-the-art methods for inpainting problem. All three groups of inpainting are reviewed to analysis their advantages and disadvantages. Based on this background, a new proposal of inpainting is introduced with high-quality results and high performance computation.

In order to validate the performance of the proposed inpainting method, a comparison with existent approaches is implemented. Nevertheless, at our best knowledge the inpainted images are very often evaluated subjectively or by using some objective metrics not well adapted to the specificities of image inpainting criteria. In addition, subjective experiments are time consuming, complex and unpredictable due to some uncontrolled human factors such as fatigue, visual discomfort, background, etc. Thus, the second goal of our work is to propose an objective Image Inpainting Quality Assessment (IIQA) metric. The performance of the proposed metric is carefully assessed with regards to human judgement as well as comparison with some traditional image quality index and existing inpainting quality metrics.

Due to the extensive research and rapid technology advancements, the output quality of inpainting algorithms has become more realistic and sophisticated. The inpainted regions

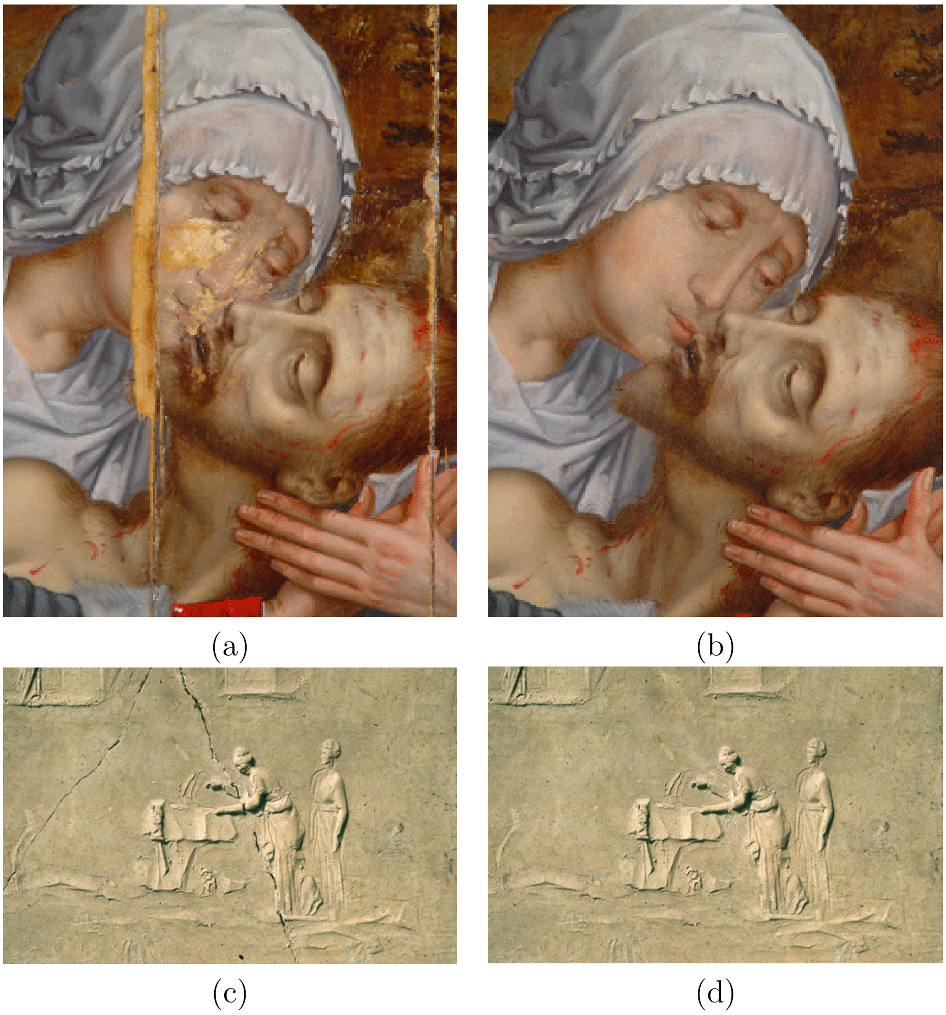


Figure 1.3. Examples of image restoration. (a), (c) The damaged image; (b) Manually restored image; (d) Automatically restored image using an inpainting algorithm.

are hard to detect by viewers, even for experts. As a result, the inpainted images could be used for many different purposes, including digital tampering. Therefore, the last objective in our work is devoted to inpainting detection, an inverse problem of inpainting. Although, many papers have been introduced for forgery detection, there is almost no study about image inpainting forgery. Consequently, a novel approach for inpainting detection is necessary and valuable in verifying the authenticity of the original image. The purpose of our work is then to introduce an efficient and reliable inpainting detection method.

1.3 Thesis contributions

Many inpainting algorithms have been reviewed and analysed in this thesis. It could be considered as the foundation for our work. Two other trends have been exploited such as *inpainting quality assessment* and *inpainting forgery detection.* For each issue, we have specific contribution for solving the corresponding problems. In summary, the significant contributions are summarized below:

- **Inpainting problem**: A novel framework for inpainting has been introduced. However, it has some limitations due to computational complexity and time running. Some solutions are proposed to improve the performance of the proposed algorithm.
- **Inpainting quality**: Once the inpainted images are obtained, the most common method for evaluating the quality is observing and scoring by human viewers. In this dissertation, we present a new metric to estimate objectively the quality of the restored images.
- **Inpainting detection**: A new model is developed to detect whether a given image is edited by inpainting algorithms or not. The idea is to consider inpainting as a tampering process. This contribution is necessary and valuable in verifying the authenticity of the original image.

As a result, the ideas developed in this thesis have been published in international conferences and journal. Below is the list of publications:

1. **T. T. Dang**, M. C. Larabi, A. Beghdadi, "*Multi-resolution patch and window-based priority for digital image inpainting problem*", 3rd International Conference on Image Processing Theory, Tools and Applications (IPTA2012), pp. 280-284, 2012, Istabul, Turkey.
2. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*Archaeological image inpainting*", the 4th European Workshop on Visual Information Processing (EUVIP2013), 2013, Paris, France.
3. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*Inpainted image quality assessment*", the 4th European Workshop on Visual Information Processing (EUVIP2013), pp. 76-81, 2013, Paris, France.

4. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*Perceptual quality assessment for color image inpainting*", IEEE International Conference on Image Processing (ICIP2013), 2013, Melbourne, Australia.

5. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*Perceptual evaluation of digital image completion quality*", 21^{st} European Signal Processing Conference (EUSIPCO2013), 2013, Marrakech, Morocco.

6. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*Visual coherence metric for evaluation of color image restoration*", the Colour and Visual Computing Symposium (CVCS2013), pp. 1-6, 2013, Gjvik, Norway.

7. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*A hierarchical approach for high-quality and fast image completion*", The Fifth International Conference on Knowledge and Systems Engineering (KSE2013), pp. 11-21, 2013, Hanoi, Vietnam.

8. **T. T. Dang**, A. Beghdadi, M. C. Larabi, "*A perceptual image completion approach based on a hierarchical optimization scheme*", Signal Processing, Volume 103, pp. 127-141, October 2014.

1.4 Thesis outline

This report consists of six chapters. After the introductory chapter, four main chapters corresponding to our contributions are presented. The last chapter is devoted to conclusion and perspectives. The content of the chapters is briefly summarized below:

- **Chapter 2** introduces the state-of-the-art inpainting algorithms. Starting from the one of the earliest work until the present one, a series of inpainting algorithms has been summarized to provide for reader a basic look on digital image inpainting methods that have been developed during the last decade.

- **Chapter 3** discusses relevant work on inpainting and introduces our proposal for this problem. First, a novel framework of image completion is introduced using multi-resolution representation of the image. Second, an improvement is proposed to enhance the algorithm performance. A series of experimental studies is carried out in order to evaluate and compare our approach with the existing methods.

- **Chapter 4** presents a new methodology for evaluating perceptual inpainting quality. Some image quality assessment metrics which can be applied for inpainting in some cases are investigated. A few existing inpainting quality metrics are analysed in details. From this analysis, some novel metrics for inpainting quality are proposed. A subjective study is then launched to evaluate the proposed metrics in comparison with both the traditional image quality assessment metrics and existing inpainting quality metrics.

- **Chapter 5** is related to a new trend of inpainting problem, inpainting forgery detection. Although, many current forgery detection techniques have been introduced, they cannot apply to detect inpainting forgery because of the fundamental difference between them. Based on the analysis of the current inpainting algorithms in conjunction with the survey of principles of existing forgery detection techniques, a novel forgery detection method is proposed for inpainting forgery in this chapter. The performance of the proposed detection method is evaluated both subjectively and objectively on a series of experiments.

- **Chapter 6** summarizes the results of the current research and points out several lines of future work.

The list of tables, list of figures and bibliography are given at last.

CHAPTER 2

Overview of Digital Image Inpainting

"To know that we know what we know, and to know that we do not know what we do not know, that is true knowledge."

Nicolaus Copernicus

Contents

This chapter aims to provide an introduction of inpainting problem as well as a classification for image inpainting techniques. In addition, a brief description for each method is also presented to help the readers have an overview of the current inpainting methods.

2.1 State-of-the-art

Image inpainting has become an active research topic in image processing field. It is not only art of modifying image but also a powerful technique for automatically restoring or removing undesired objects in an undetectable form to an observer not familiar with the original image and is as ancient as art itself. This work refers to the traditional practice of expert and professional artists in museums to restore manually the old pictures or paintings. The problem is extremely difficult and high time cost because they had to complete smoothly the damaged image as well as preserve the global content based on their skills and experiences. Digital image inpainting attempts to mimic the basic techniques of this process automatically. Since it does not refer to the original image and it is too hard to define mathematically "*global content*" term of image, the objective of digital image inapainting is only restoring the damage image so that it still looks natural, i.e., the restored parts are not easily detectable by viewers.

In the parlance of digital inpainting, the missing region is often referred to as *blotch* or *scratch* in the case of small or thin size regions; and referred to as *hole* or *gap* in case of large size regions. Very often, the blotch or hole of image is provided by the user in the form of binary mask or can be obtained by semi-automatic means.

Besides, the nomenclature is classified by this criterion. Namely, the removing blotch or scratch is called as inpainting but the filling in hole or gap is considered as image completion. The terminology of "inpainting" was invented by *Bertalmío et al.* [6] who were the first group to develop inpainting models with the motivation coming from the professional artists. This approach is suitable to remove blotches or scratches. When it is developed to filling in holes or gaps, it is called as image completion.

Despite the significant differences between the existent techniques, an image inpainting model is often composed of two stages:

1. *identify* the damaged regions (inpainting regions).
2. *fill in* the inpainting regions based on the known information in the same image.

Up to now, there has been no way or model to detect automatically the inpainting regions because of a lack of objective criteria or measures to identify and quantify the level of degradation. There are a few special applications in which the inpainting regions can be detected semi-automatically, for example blotch detection of old video. However, accuracy and performance of detection are still much debate. For the sake of simplicity, they are marked manually in form of binary mask by users. The digital inpainting focuses on the second stage in order to perform automatically the restoration using the input parameters. The digital image completion consists of mathematically modeling and performing the restoration under constraints of local smoothness preservation, but without taking into account the global semantic of image or artistic expertise.

Overview of inpainting methods

Digital image inpainting is considered as an ill-posed inverse problem which has no well-defined unique solution [7]. To solve the problem, only input image including the known and unknown parts is given and the requirement of output in terms of quality is that the recovered region looks natural to the human eye.

Nowadays, computer technology facilitates a rapid development of digital inpainting techniques. There have been numerous and very different automatic inpainting algorithms in recent years. In general, most of them are categorized into three main groups as listed below [7, 8]:

- *Geometry-oriented methods*
- *Texture-oriented methods*
- *Hybrid inpainting methods*

Basically, an overall assessment of existing methods is considered as an importance for next proposals in our research. In this chapter, some typical inpainting methods as shown in Table 2.1 will be introduced and analysed in the next section in order to build a foundation for our next research.

In Table 2.1 the first column, *Year*, indicates the year of the publication followed by *Authors*' names. *Type* column indicates the category to which the algorithm belongs. The last column *H-type*, short for 'hole type', determines the size of hole or damaged region for which the algorithm is more adapted. A detailed definition of the size of the hole will be given in the next section.

Notations

Before proceeding with a review of the inpainting methods , a note must be made on the notation that will be used. The image input is denoted by character I. The inpainting domain in the image is defined by the omega symbol, Ω, while $\delta\Omega$ stands for its boundary. The source region, representing the area not covered by the mask, or the area complementary to the gap, will be denoted by Φ ($\Phi = I - \Omega$). This notation was first used in [6] and

Table 2.1. Some typical inpainting methods.

Year	Authors	Group	H-type
1985	Ogden *et al.* [9]	Geometry	Small
2000	Bertalmío *et al.* [6, 10]	Geometry	Small
2001	Chan and Shen [5]	Geometry	Small
2002	Esedoglu and Shen [11]	Geometry	Small
2006	Tschumperlé [12]	Geometry	Small
2013	Li *et al.* [13]	Geometry	Small
1999	Efros and Leung [14]	Texture	Large
2001	Efros and Freeman [15]	Texture	Large
2001	Ashikhmin [16]	Texture	Large
2004	Criminisi *et al.* [17]	Texture	Large
2005	Cheng *et al.* [18]	Texture	Large
2005	Sun *et al.* [19]	Texture	Large
2006	Wu and Ruan [20]	Texture	Large
2012	Zhang and Lin [21]	Texture	Large
2008	Mairal *et al.* [22]	Texture	Small
2009	Mobahi *et al.* [23]	Texture	Small
2006	Komodakis *et al.* [24]	Texture	Large
2013	Le Meur *et al.* [25]	Texture	Large
2007	Wexler *et al.* [26]	Texture	Large
2009	Pritch *et al.* [27]	Texture	Large
2013	Liu and Caselles [28]	Texture	Large
2003	Bertalmío *et al.* [29]	Hybrid	Small
2009	Bugeau and Bertalmío [30]	Hybrid	Small
2013	Sairam *et al.* [31]	Hybrid	Small
2007	Hays and Efros [32]	Multi-source images	Large
2009	Whyte *et al.* [33]	Multi-source images	Large

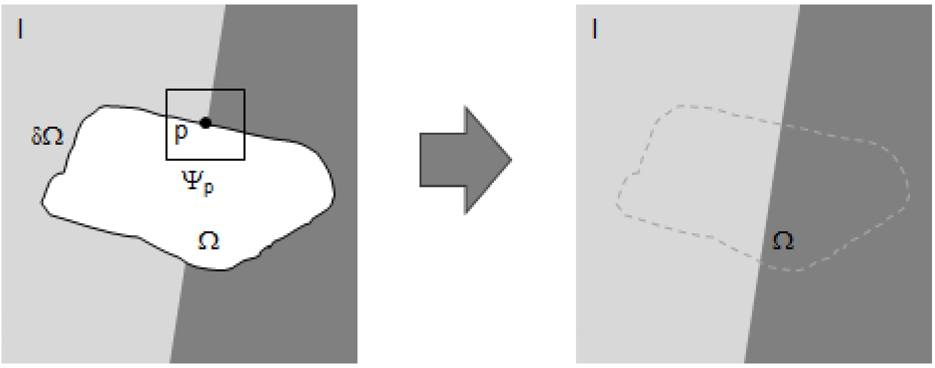


Figure 2.1. Inpainting problem.

adopted by the following works (see Figure 2.1). A window centred at pixel p is denoted as a patch, Ψ_p.

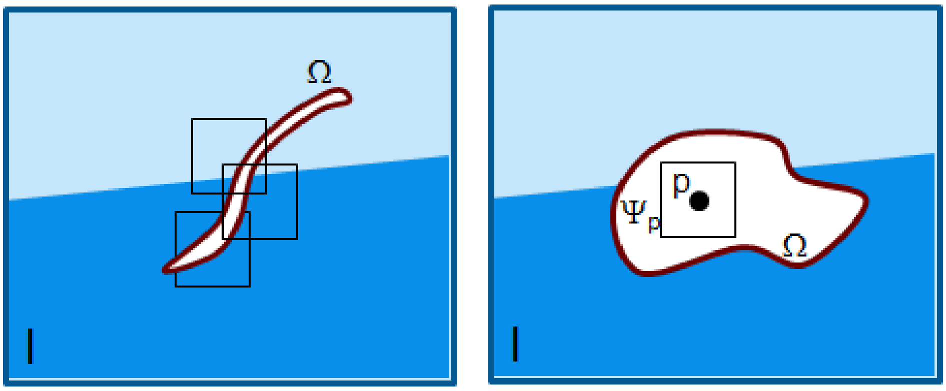
The distinction of the inpainting region

The area of the damaged region plays an important role in analysing and designing the recovery methods. In addition, it is also the key criterion for classification of inpainting methods. Nevertheless, we agree with the fact that the notion of large missing regions used in the inpainting community is not a well/quantitatively defined criterion. For the small inpainting regions, such as scratches and blotches of reduced size, is not a challenging issue. In contrast, in the case of large regions, it is uneasy to guarantee the spatial coherence and object continuity during the inpainting process.

The percentage of missing pixels cannot be the only criterion because an image may contain some thin scratches across the image. In that case the scratch could not be considered as large.

We can say that a missing region is large if it contains completely a square patch whose size length is greater than r pixels. In our experiments r is set to 5. It is worth noticing that this parameter does not depend on the image size but directly related to the inpainting approach. For example a size of 3×3 does not necessitate any sophisticated inpainting approach. A simple interpolation would work. A typical size of 5×5 represents the most adequate choice for the used inpainting approach. A large missing region, Ω, can be defined mathematically as follows:

$$\Omega \text{ is a large inpainting region} \Leftrightarrow \{\exists p \in \Omega : \Psi_p \cap \Omega \equiv \Psi_p\} \tag{2.1}$$



(a) (b)

Figure 2.2. An illustration of inpainting region classification based on the equation 2.1; (a) a small inpainting region; (b)a large inpainting region.

2.2 Geometry-oriented methods

The first group of inpainting approaches discussed here is *geometry-oriented* methods. In this group, the image is modelled as a function of smoothness and the restoration is solved by interpolating the geometric information within the adjacent regions into the target region. Several methods of this group have been proposed [5, 6, 12, 34]. Generally, partial differential equation (PDE) based methods are the most studied ones.

2.2.1 Linear interpolation for inpainting

An early interpolation method is introduced by *Ogden et al.* [9] based on a pyramidal representation of the input image. Starting from an initial image, a Gaussian filtering is applied to generate a pyramid representation with iterated convolution and subsampling. The inpainting process starts at lowest resolution where the unknown region is shrunk to only one pixel and filled in by extrapolation. Continuing this process, the higher levels can be completed by successive linear interpolations using the pyramid operators such as *downsampling* and *upsampling*.

The experimental results show that this approach could produce reasonable output in some simple occlusion situations. The result is so natural looking that the flaw would not be detected except by close examination.

2.2.2 PDE based inpainting method

In context of image processing, the terminology *inpainting* has been coined first by *Bertalmío et al.* in [6]. This approach is inspired from real painting restoration in the museums. The underlying idea of this algorithm is to propagate the geometric and photometric information arriving at the boundary of the occluded area, $\delta\Omega$, into the area Ω itself. The inpainting information is updated iteratively until a steady state is attained using the following equation:

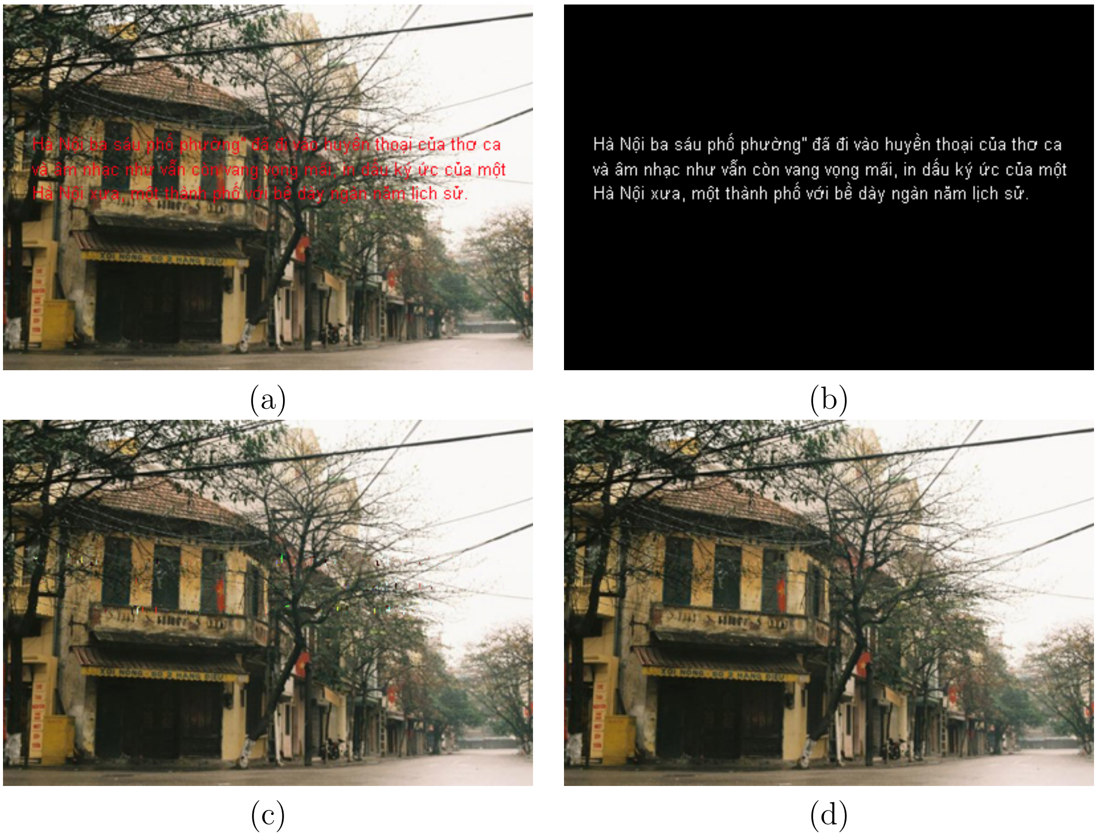
$$I^{n+1}(i,j) = I^{n}(i,j) + \Delta t I_t^n(i,j) \tag{2.2}$$

where I^{n+1} is the restored image at the iteration $(n+1)$th and Δt is a user-defined value and called as the rate of change and I_t^n is the improvement of the image I^n. The multiplication of the latter two terms gives the improvement observed in image I^{n+1} as compared to the previous version of the image, I^n. To preserve both the propagation information and direction, the improvement is defined by the following equation:

$$I_t^n(i,j) = \overrightarrow{\delta L^n}(i,j)\overrightarrow{N^n}(i,j) \tag{2.3}$$

where $\overrightarrow{\delta L^n}(i,j)$ represents a measure of the change in the propagated information at iteration n and $\overrightarrow{N^n} = \nabla^{\perp}.I^n$. Here, the gradient vector at iteration n, noted ∇I^n, is called isophote direction and $\nabla^{\perp}.I^n$ defines the orthogonal direction of gradient.

As mention before, this algorithm tries to complete the damaged area with a smooth propagation from source region, thus selecting smoothness estimator is very important. In [6], authors proposed the use of a simple discrete implementation of the Laplacian for propagation: $L^n(i,j) = I_{xx}^n + I_{yy}^n$. The subscripts in the latter equation represent second derivatives. The main idea behind the calculation of the isophote direction is mainly based on obtaining the gradient vector at pixel location (i,j). The discretized gradient vector, ∇I^n, gives the direction of largest spatial change, while its 90 degrees rotation is the direction of smallest spatial change, so the vector $\nabla^{\perp} I^n$ gives the isophote direction.


Hà Nội ba sáu phố phường" đã đi vào huyền thoại của thơ ca
và âm nhạc như vẫn còn vang vọng mãi, in dấu ký ức của một
Hà Nội xưa, một thành phố với bề dày ngàn năm lịch sử.

(a) (b) (c) (d)

Figure 2.3. An illustration for approaches of Bertalmío *et al.*. (a) An overimposed text image; (b) The corresponding masked image; The removed text images using approaches in (c) [6] and (d) [10].

In subsequent works, *Bertalmío et al.* [10] has introduced an improvement of the previous work by using the classical fluid dynamics and Navier-Stokes equations (describing the motion of fluid substances). The authors established a connection between inpainting problem and fluid dynamics theory. In this context, the image intensity is equivalent to the fluid's stream, whereas the level lines in the image define the stream lines of the flow. Furthermore, the isophote direction is an equivalent of the fluid's velocity, while the smoothness of the image is comparable, in fluid dynamics terms, to the curl of the fluid's velocity, called vorticity. Following this approach, inpainting resumes to solving a vorticity transport equation, instead of the transport equation given in (2.3). Although the anisotropic diffusion equation is not the exact counterpart of the viscous diffusion term used in the Navier-Stokes model for incompressible and Newtonian flows, yet a lot of the numerical knowledge on fluid mechanics seem to be adaptable to design stable and efficient schemes for inpainting. For more detailed specification on the implementation, the reader is referred to [10]. Some outputs of these approaches are shown in Figure 2.3. In the first approach, the climb rate Δt is set to 0.05, and the inpainting steps is set to 20.

2.2.3 Total variation based inpainting method

Another perspective on this group is proposed by *Chan and Shen* in [5, 35], where a denoising/inpainting first-order model based on the joint minimization of a quadratic fidelity term outside Ω and a total variation criterion in Ω is proposed. In this approach, the inpainting problem is considered as the minimization problem as following:

$$u^* = argmin_u(\int_I |\nabla u| dx + \frac{\lambda}{2}\int_\Phi |u - u_0|^2 dx) \tag{2.4}$$

where $u(x)$ is the intensity of the grey level at point x and λ is a Lagrange multiplier. The first term in equation (2.4) is known as the *regularising term* or *regulariser*. The minimization of this term is responsible for the filling-in process inside the inpainting domain. The second term of the energy function is a measure of *fidelity*. This is non-negative and has the property that it is zero when $u(x) = u_0(x)$ for every $x \in \Phi$. Since we desire the value of u to be close to u_0, we would like to keep value of the fidelity term as small as possible. In [35], the fidelity term is chosen to be square of the L^2 norm of $u - u_0$ because this is convex and differentiable.

The existence of solutions to this problem follows easily from the properties of functions of bounded variation. As for the implementation, a Gauss-Jacobi iteration scheme for the linear system associated to an approximation of the Euler-Lagrange equation by finite differences is applied to calculate the critical points of the energy function. The advantage of total variation over harmonic inpainting is that it is able to preserve discontinuities inside the inpainting area. More details could be found in [35].

In the framework of total variation in [11], *Esedoglu and Shen* introduced an improved model, combining the celebrated Mumford-Shah segmentation model for images and the Euler's elastica model for curves. The elastica model is used instead of the fidelity term. The energy function is modified as follows:

$$u^* = argmin_u(\int_\Gamma (\alpha + \beta\kappa^2) ds + \frac{\lambda}{2}\int_{I\backslash\Gamma} |\nabla|^2 dx) \tag{2.5}$$

Two numerical implementations to the minimization of such criterion are discussed fully in [11]. The first one is based on level set approach where Γ is considered as the zero-level set of a sequence of smooth functions, and the explicit derivation using finite differences of the Euler-Lagrange equations associated with the criterion. The second implementation addressed by *Esedoglu and Shen* is a $\Gamma - convergence$ approach based on a result originally conjectured by *De Giorgi* [36] and recently proved by *Schatzle* [37]. A comparison of efficiency between the two approaches is illustrated in Figure 2.4. In both cases, the final system of discrete equations is of order four, facing again difficult issues of convergence and stability.

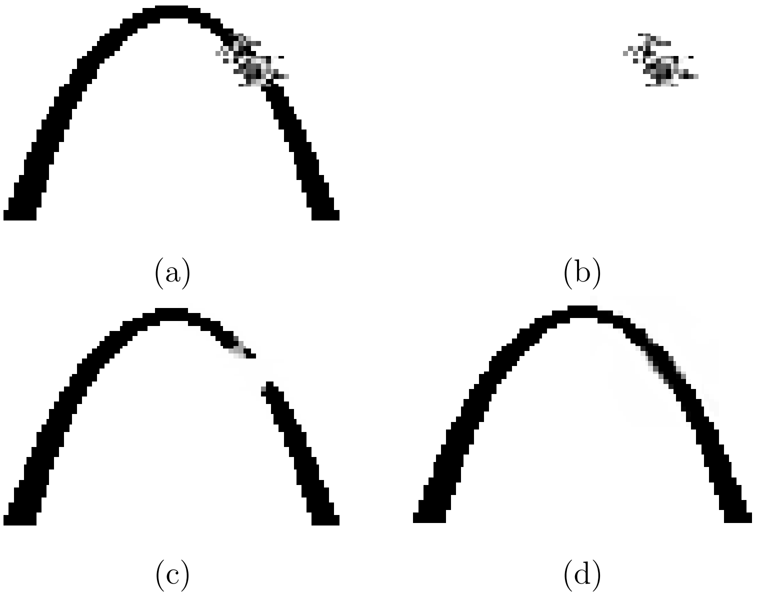


Figure 2.4. An illustration of comparison between elastica and total variation inpainting. In the case of large aspect ratios, the TV inpainting model fails to comply to the Connectivity Principle. (a) original image; (b) inpainting domain; the output when using (c) total variation and (d) elastica inpaintings

2.2.4 Vector-valued image regularization with PDEs

In [12], *Tschumperlé et al.* introduced an efficient second-order anisotropic diffusion model for multi-valued image regularization and inpainting. The regularization of an image is the process through which specific image is simplified in such a way that interesting features are preserved and unimportant data is removed. Given a R^N-valued image, I, with the damaged region, Ω, and starting from an initial rough inpainting obtained by straightforward advection of boundary values, the pixels in the inpainting area are iteratively updated according to a finite difference approximation of the equation:

$$\frac{\partial I_i}{\partial t} = trace(\nabla^2 I_i) \tag{2.6}$$

Here, T is the tensor field defined as:

$$T = \frac{1}{(1+\lambda^+ + \lambda^-)^{p_1}}\theta^-\theta^{-T} + \frac{1}{(1+\lambda^+ + \lambda^-)^{p_2}}\theta^+\theta^{+T} \tag{2.7}$$

with $0 < p_1 \ll p_2$, and $\lambda^-, \lambda^+, \theta^-, \theta^+$ are the eigenvalues and eigenvetors, respectively, of the smoothed structure tensor field, $\mathbf{G}_\sigma$:

$$\mathbf{G}_\sigma = G_\sigma * \sum_{i=1}^{n} \begin{pmatrix} \left(\frac{\partial I_i}{\partial x}\right)^2 & \left(\frac{\partial I_i}{\partial x}\right)\left(\frac{\partial I_i}{\partial y}\right) \\ \left(\frac{\partial I_i}{\partial x}\right)\left(\frac{\partial I_i}{\partial y}\right) & \left(\frac{\partial I_i}{\partial y}\right)^2 \end{pmatrix} \tag{2.8}$$

where σ depends on the noise scale. Based on image regularization, this method successfully fills in the gap by diffusing the boundary pixels until completion of the missing areas, in a structure preserving way. The main drawback of this problem is the introduction of blur effect which results in unsharp edges inherent to PDE-based methods. Thus, it can be inferred that this technique will fail in reproducing textured region, but will successfully fill in small and narrow scratches.

2.2.5 DCT induced wavelet regularization for inpainting

With the advent of sparse representation and compressed sensing, sparse priors have also been considered for solving the inpainting problem. In [13], *Li et al.* assumed the input image is sparse in a given basis, discrete cosine transform (DCT) where the known and unknown parts of the image share the same sparse representation. The inpainting problem is solved based on an optimization model whose objective function is a smoothed ℓ_1 norm of the coefficient of the underlying image under a given redundant system generated from the DCT matrix of second type [38]. Then, the DCT-Haar wavelet system is used as a filter and yields a redundant system.

The inpainting model, in this case, is formulated as an optimization problem in which the variational objective function has a regularization term formed by a sparse representation of the underlying image. Let f_o the original image to be defined on the domain

$\Omega = 1, 2, ..., n$ and a non-empty proper subset D of Ω be given. The observed image g is modeled as

$$g[k] = \begin{cases} f_o[k] & k \in \Omega \backslash D \\ h[k] & k \in \backslash D \end{cases} \tag{2.9}$$

where $h[k]$ with $k \in D$ could represent any types of degradations to the original image. Then, the inpainting problem with a redendant system, W, generated from the DCT-II is as follows:

$$\min_{f,d} \left\{ \frac{1}{2} ||Wf - d||_2^2 + ||\Gamma d||_1 : P_D f = P_D g \right\} \tag{2.10}$$

where Γ is a diagonal matrix with non-negative diagonal components and d is an auxiliary vector. P_D denotes the $n \times n$ diagonal matrix whose k-th diagonal entry is 1 if $k \in \Omega \backslash D$ and 0 if $k \in D$.

The solution is obtained based on an adaptive algorithm with some input parameters. Some experimental results could be found in [13] show that the proposed approach is suitable for inpainting problems such as impulsive noise removal and filling missing information over regions with small sizes.

2.3 Texture-oriented methods

The second class of inpainting algorithms corresponds to *texture-oriented* (or *exemplar-based*) methods. These approaches stem from the texture synthesis techniques where texture is modeled through probability distribution of the pixel brightness values. The exemplar-based algorithms tend to be less computationally efficient than interpolation and they also overcome the problems associated with interpolation. In fact, the actual pixel values of the known regions are used instead of average values as in the interpolation. The blurring effect is generally removed completely. In addition, several of these algorithms favor edges when choosing pixels to replace, this helps in preserving the edges within the missing regions and make it continuous in the output.

This group could be further subdivided into two sub-groups: greedy strategy and global optimization strategy. In the former group, the authors in [14] proposed a pixel by pixel completion based on the matching patch comparison. An extension of this version was presented in [15] where textures were synthesized patch by patch. However, these approaches are too slow or the input image contains only pure textures. Some sophisticated methods have been introduced such as fragment-based [39], user guidance based [40], probabilistic graphical model based [24, 41], different priorities of filling [17, 18, 20, 21], etc. Notwithstanding, these approaches suffer from problems such as local optimization, patch section, etc. In the later one, the inpainting problem is considered as a global optimization problem and it can be solved by optimizing discrete Markov Random Fields, as in belief propagation (BP) [24, 41] and graph cuts [27, 42] or minimizing the coherence measure as in [26]. Nevertheless, the global optimization strategies are often more computationally expensive and its complexity is too high. Compared with the first class, the approaches in the second class achieve impressive results in recovering the large damaged regions but the output results may be distorted when the number of patches is insufficient.

2.3.1 Greedy strategy

2.3.1.1 From texture synthesis to patch-based inpainting

One of the early works in this group is introduced by *Efros and Leung* [14, 15]. The initial goal of this approach was intended for texture synthesis and it results in good outputs in some cases of inpainting. The main idea is to fill-in the unknown pixel with known pixels based on their known neighborhood. The hole is completed recursively, inwards from the inpainting boundary. Start with an unknown pixel $p \in \delta\Omega$, a known pixel, $q \in \Phi$ is used to fill-in pixel p if the neighborhood Ψ_q of q is the most similar one to the neighborhood Ψ_p of p. The similarity is considered to only known pixels in both Ψ_p and Ψ_q and is defined using the sum of squared differences (SSD) as follows:

$$d(\Psi_p, \Psi_q) = \sum_i \sum_j ||\Psi_p(i,j) - \Psi_q(i,j)||^2 \tag{2.11}$$

This process is repeated randomly for every unknown pixel in the inpainting boundary until no pixel is found, so it is also known as *onion peel*. The main shortcoming of this algorithm is its computational time. In fact, this approach is quite slow since it works on pixel by pixel basis. In addition, since the filling order is random, it may fail to smooth boundaries of some objects. The patch size defined as a global parameter in original paper may make the output slip into wrong part or get locked onto one place and produce verbatim copies. In fact, this approach produces poor results for image perspective because it was intended to synthesize frontal textures but not inpainting problem. Figure 2.5 gives an overview of the algorithm principle. The pixel in red patch will be filled-in by one of centers of candidate patches (blue patches).

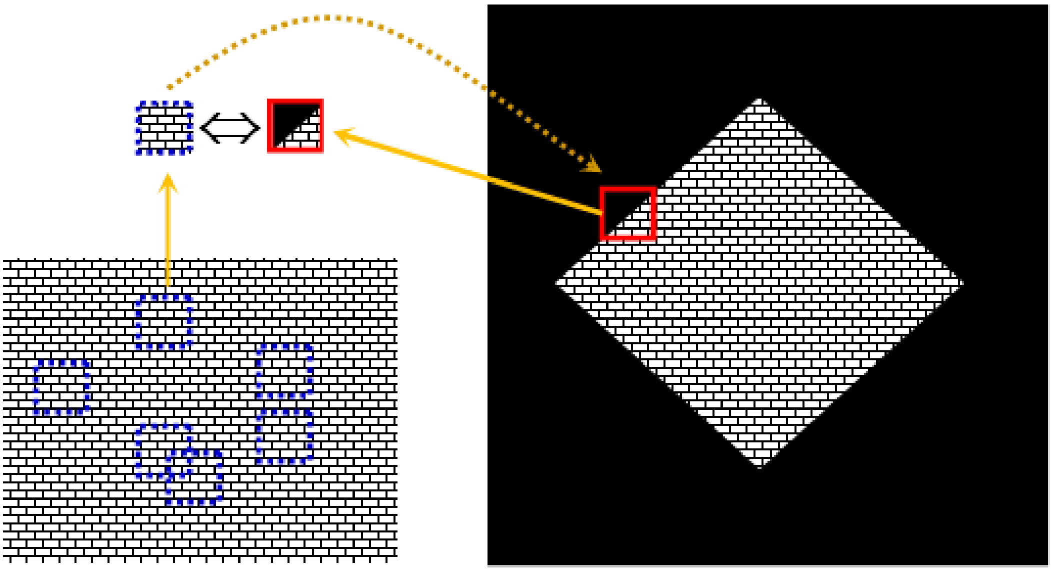


Figure 2.5. Overview of Efros' algorithm

Many ideas have been proposed to improve the computational time of synthesis such as to reduce the search space, organize the image patches in tree structure or trim the dimensionality of the patches with techniques like Principal Component Analysis (PCA) or using randomized approaches. A remarkable contribution is introduced by *Ashikhmin* [16] based on the correlation between unknown pixels and its neighbors to generate the shifted candidates for searching. Instead of searching on the whole image, only some candidates of the neighbors of p in the source region are considered. The speed-up achieved with this simple technique is considerable, and also there is a very positive effect regarding the visual quality of the output. Figure 2.6 illustrates the main idea of this approach.

2.3.1.2 Exemplar-based inpainting

From the limitation of *Efros'* work [14], *Criminisi et al.* [17] improved on this work with two remarks. Firstly, the filling order is changed from the original "onion peel" into a priority scheme where unknown pixels at boundary of a hole have higher priority than one on flat

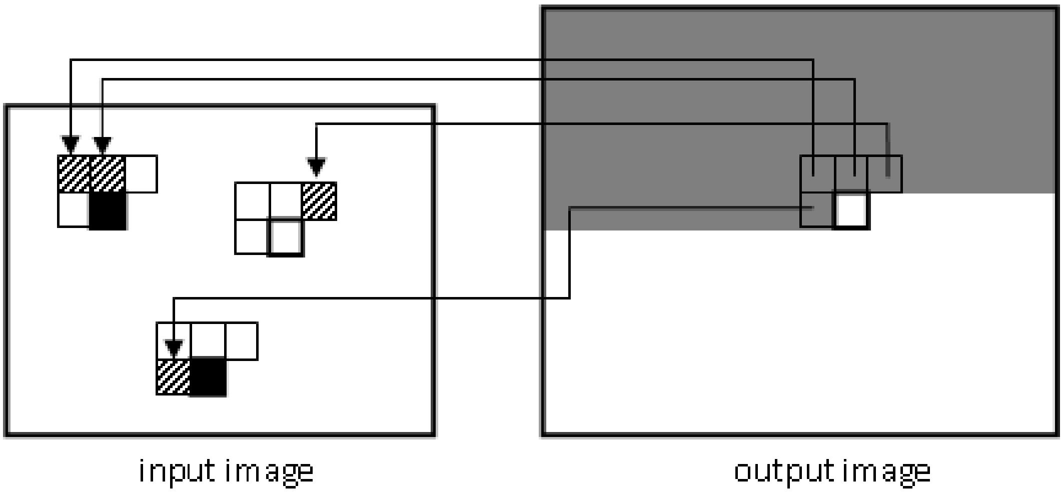


Figure 2.6. An illustration of Ahikhmin's texture synthesis method [16]. The candidate patches are considered based on the correlation between unknown pixel and its neighbors.

regions. Thus, the completion is able to correctly inpaint straight object boundaries which could have otherwise ended up disconnected with the original formulation. Secondly, the filling process is implemented by copying entire patches (or exemplars) instead of single pixels, so this method is considerably faster. The algorithm is briefly described as the following steps:

1. *Initialization*: Identify the target region, Ω, and its boundary, $\delta\Omega$. If there is no pixel on the boundary, i.e., $\delta\Omega = \emptyset$, the algorithm is terminated.
2. *Priority estimation*: Compute the priority, $P(p)$, for all pixels on boundary, $p \in \Omega$. Select randomly a pixel p with the highest priority and define its patch, Ψ_p.
3. *Patch match*: Find the patch Ψ_q that is most similar to Ψ_p, with mean squared error of the known pixels and select randomly one if there are more than two patches.
4. *Inpaint*: Fill the missing information in patch Ψ_p by copying the corresponding pixels from patch Ψ_q.
5. *Update*: Update the status information and return to the step 1 for next iteration.

As an example, Figure 2.7 gives an illustration of the steps in *Ciminisi*'s approach. Figure 2.7-a shows a highest priority pixel p, belonging patch Ψ_p, represented as a square surrounding it and lying on the gap's boundary contour $\delta\Omega$. After finding the pixel need to be filled with highest priority, the similar patches in the source region Φ is searched. Figure 2.7-b shows a candidate for filling in the patch determined by the pixel p and the inpaint step is exemplified in Figure 2.7-c.

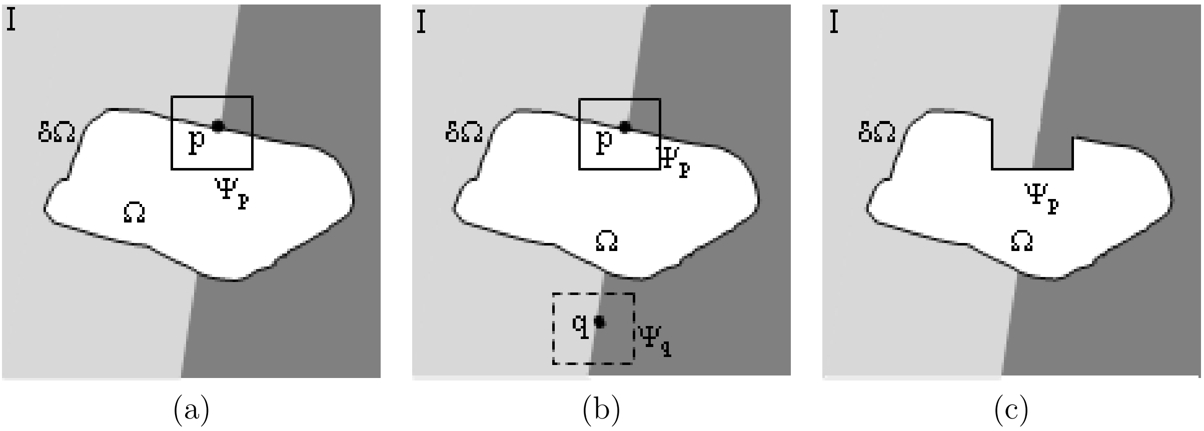


Figure 2.7. A visualisation of the exemplar-based inpainting process. (a) Estimate the priority; (b) Search the best similar patch; (c) Perform inpainting.

The most important contribution in *Criminisi*'s work is designing the filling priority $P(p)$ for each pixel in the hole boundary. In the original paper, the priority is built based on the isophote direction as in [6] and the known information of the current patch. Equation (2.12) expresses the priority measure.

$$P(p) = C(p)D(p) \tag{2.12}$$

Here, $C(p)$ and $D(p)$ are called *confidence* and *data* terms, respectively and are defined by:

$$C(p) = \frac{\sum_{q\in\Psi_p\cap\Phi} C(q)}{|\Psi_p|}; \quad D(p) = \frac{|\nabla I_p^{\perp}.n_p|}{\alpha} \tag{2.13}$$

where $|\Psi_p|$ denotes the area of Ψ_p and α is a normalization factor (e.g., $\alpha = 255$ for a typical grey-level image), and n_p is a unit vector orthogonal to the front $\delta\Omega$ at the pixel p. Figure 2.8 shows a result using *Criminisi*'s approach [17]. Two maps of confidence and data terms in inpainting process are shown in Figure 2.9.

It is worth noticing that the patch-based methods using priority has proven to be efficient in solving inpainting problem with large damaged region in image. Many extensions

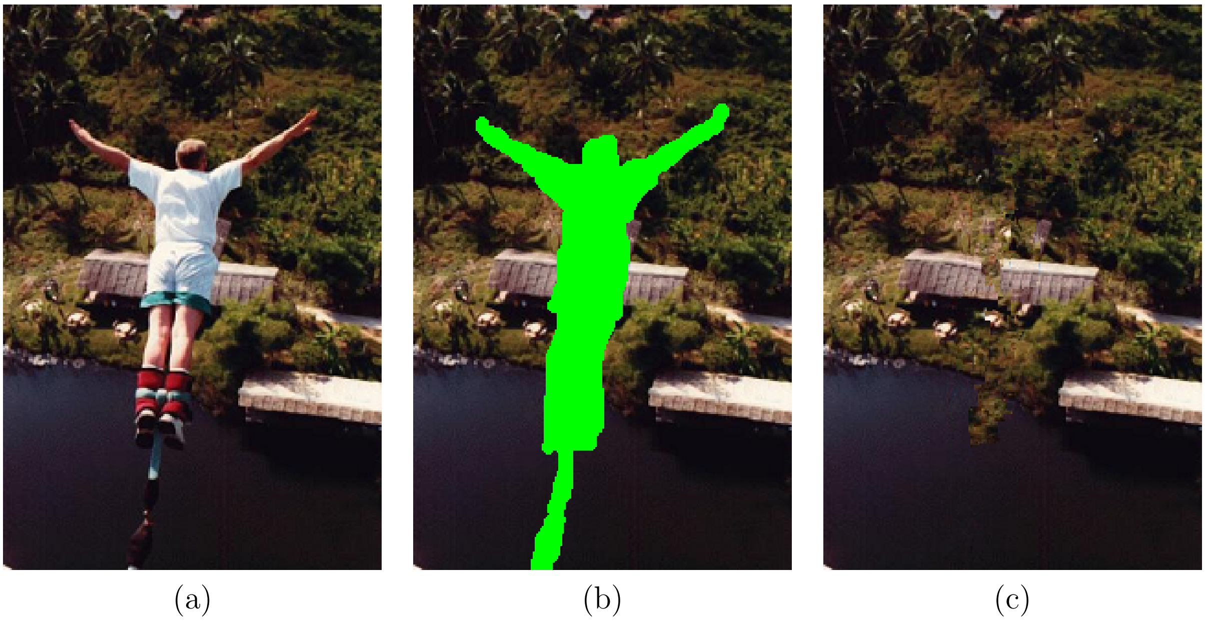


Figure 2.8. An output of Criminisi's approach [17]. (a) The original image; (b) The mask image; (c) The inpainted image.

have been published to improve the quality as well as performance of the algorithm. In the work of *Wu et al.* [20], the priority is built based on cross-isophotes. In [21], authors defined the priority based on an analysis of color distribution. In other way, *Cheng et al.* [18] recombined the priority using a regularizing factor. They observed that the confidence term drops rapidly to zero as the filling process proceeds, which makes the computed priority values indistinguishable, and in turn, the results in incorrect filling orders. Thus the priority function is modified to an additive rather than multiplicative forms. For more detailed specifications on the analysis, the reader is referred to [18].

While most image inpainting methods attempt to be fully automatic (aside from the manual setting of some parameters), they are user-assisted methods that provide remarkable results with just a little input from user. One of the known methods in this group has been introduced by *Sun et al.* [40]. The user must specify curves in the unknown region before inpainting. The curves will guide the recovery process. Firstly, patch synthesis is performed along these curves inside the image hole. Then, structure propagation will restore the rest. Figure 2.10 shows some interesting results of this approach. As can be seen, two different outputs are obtained using two different user-defined curves.

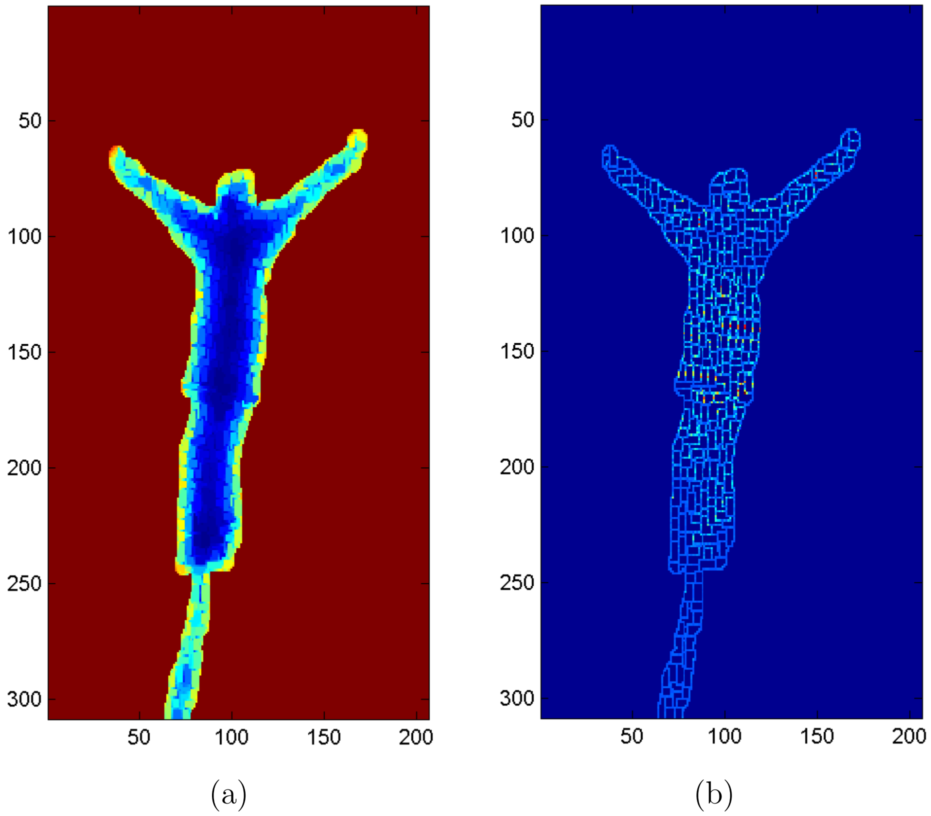
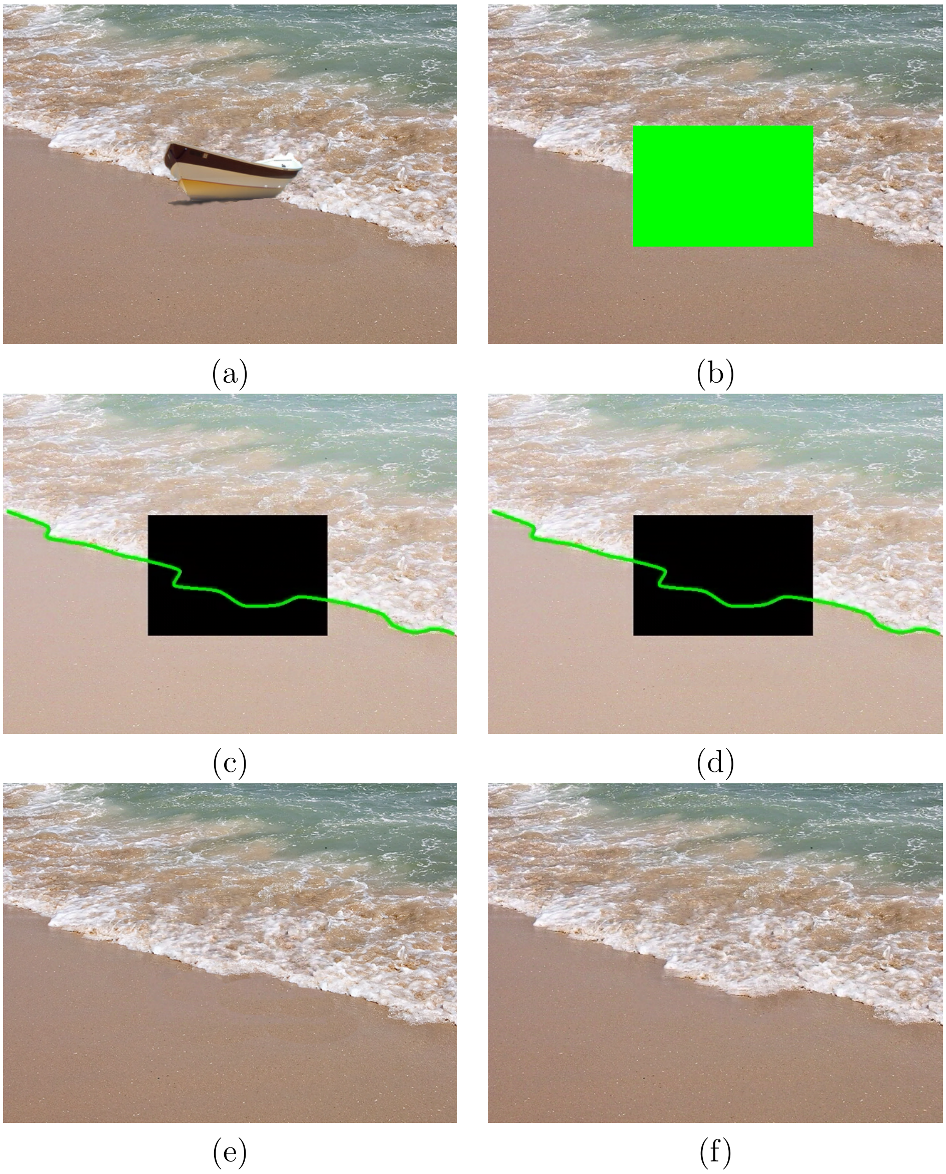


Figure 2.9. The maps of confidence and data terms in Criminisi's priority. (a) Confidence map; and (b) Data map.

2.3.1.3 Data-driven inpainting

Following the success of patch-based methods, it becomes clear that the patches of an image provide a good resource to image editing. Most previous methods are strongly model-driven and have little adaption data. Rather than building mathematical models from scratch, the input image is considered as a dictionary of patches which can be used to express other parts of the image. This idea has been successfully applied to other areas of image processing, e.g. denoising and segmentation [22, 43, 44].

Following this idea, *Mairal et al.* [22] developed a framework using dictionaries of patches to restore the unknown region. The input image is broken into a set of patches with size $n = \sqrt{n} \times \sqrt{n}$, called dictionary D, and they are vectorized into columns of D, represented by a matrix of size $n \times k$ (k is number of patches extracted from input image). If $\alpha \in R^k$ is a vector of coefficients, then $D\alpha$ represents the patch obtained by linear combination of the column of D. Given an image v, the purpose is to find a dictionary $\widehat{D}$,


(a)

(b)

(c)

(d)

(e)

(f)

Figure 2.10. The curve-based outputs using Sun's method [40]. (a) The original image; (b) The mask image; two different curves are pre-defined in (c) and (d); (e) and (f) two corresponding outputs.

an image $\widehat{u}$ and coefficients' vector $\widehat{\alpha}$ which minimize the energy:

$$\begin{aligned}\{\widehat{\alpha}, \widehat{D}, \widehat{u}\} &= \arg\min_{\alpha,D,u} \lambda||v-u||_2 + \sum_{ij} \mu_{ij}||\alpha_{ij}||_0 \\ &+ \sum_{ij} ||D\alpha_{ij} - \Psi_{ij}u||_2 \end{aligned} \tag{2.14}$$

In this equation, $\Psi_{ij}u$ denotes the patch of u and μ_{ij} are positive weights. The inpainting problem can be considered as a case of non-homogeneous noise. Each pixel to be inpainted is defined by a coefficient $\beta_{ij} = 0$ in the mask image. Then, the inpainting problem is expressed as follows:

$$\begin{aligned}\{\widehat{\alpha}, \widehat{D}, \widehat{u}\} &= \underset{\alpha,D,u}{\arg\min}\, \lambda||\beta \otimes (v-u)||_2 + \sum_{ij} \mu_{ij}||\alpha_{ij}||_0 \\ &+ \sum_{ij} ||(\Psi_{ij}\beta) \otimes (D\alpha_{ij} - \Psi_{ij}u)||_2 \end{aligned} \tag{2.15}$$

With suitable adaptation, this model has been applied to inpainting problem of relatively small gaps.

In [23], *Mobahi et al.* extended this approach with the degenerate Gaussian assumption which results in more accurate and robust completion. The hole is filled-in using novel patches synthesize from a limited dictionary of real patches. The novel patches are linear combination of the real patches in the original image. Thus, even if no patch in the dictionary matches a given missing region, it is still possible to fill in the region with an appropriate linear interpolation of patches. To avoid problem of over fitting, the novel patch has sparse representation. To find more detailed implementation of this approach, the reader can refer to [23].

2.3.2 Global optimization strategy

Exemplar-based methods have proven their effectiveness to image completion. But, in this case, a major drawback of related approaches stems from their greedy fashion of filling image, which can often lead to visual inconsistencies. In order to avoid the occurrence of visually inconsistent results, some other methods are preferred. A more global approach where the problem is formulated in a way that a deterministic EM-like optimization scheme has to be used for image completion [26, 45] or discrete global optimization via Belief Propagation (BP) optimization [24, 46] or graph-cuts algorithm [27, 28, 47] is considered. In general, the global optimization strategies often provide better results than greedy strategies, but they have main shortcomings of computational complexity and time running.

2.3.2.1 Space-time video completion

In [26], *Wexler et al.* considered the completion as a global optimization and the inpainting problem is extended to the restoration of video sequence. To allow for a uniform treatment of dynamic and static information, the video sequence is treated as space-time volumes, i.e., each pixel (x, y) in frame t will be regarded as a space-time point $p(x, y, t)$ and each patch Ψ_p is defined as a fixed-size window around p both in space and in time. The main idea of this approach is preserving the *global visual coherence.* A video S has *global visual coherence* with some other video D if every local space-time in S can be found somewhere in the video D. In other words, a video S can be restored with small windows in D.

Let S be an input sequence with a hole $H \subseteq S$. The inpainting problem is to complete H such that the result video S^* will have as much *global visual coherence* with the rest of the input sequence, $T = S \setminus H$. Mathematically, this could be formulated as searching procedure for the sequence S^* maximizes the following objective function:

$$Coherence(S^*|T) = \sum_{p \in S^*} \max_{q \in T} sim(\Psi_p, \Psi_q) \tag{2.16}$$

where p, q run over all space-time points in their respective sequences, and Ψ_p, Ψ_q denote small space-time patches centered at p, q, respectively; sim is a local similarity measure between two space-time patches and defined through distance between two patches, $d(\Psi_p, \Psi_q)$, as follows:

$$sim(\Psi_p, \Psi_q) = e^{\frac{-d(\Psi_p, \Psi_q)}{2*\sigma}} \tag{2.17}$$

The choice of σ parameter is important as it controls the smoothness of the induced error surface. Very often, the sum of square differences (SSD) of color information is widely used for image completion but it does not suffice in this case. The main reason for this is that the human visual system is very sensitive to motion. Maintaining motion continuity is more important than finding the exact spatial pattern match within an image of the video. Therefore, the motion information must be included in the similarity measurement. Indeed, each space-time point is extended into 5D representation $(R, G, B, \alpha u, \alpha v)$ $(\alpha = 5$ as in the original paper). Two added components, $u = Y_t/Y_x$ and $v = Y_t/Y_y$, represent the instantaneous motion in the x and y directions, respectively. Here, Y_x, Y_y, Y_t are the spatial and temporal derivatives of gray level at each space-time point (x, y, t).

The solution of equation (2.16) is obtained based on an iterative scheme using the Mean-Shift algorithm [48]. A pseudo-code of this approach is described particularly in [26].

To further enforce global consistency and to speed up convergence, the iterative process is performed in multiple scales using spatio-temporal pyramids. Each pyramid level contains half the resolution in the spatial and in the temporal dimensions. The optimization starts at the coarsest pyramid level and the solution is propagated to finer levels for further refinement.

2.3.2.2 Belief propagation for image completion

Markov Random Field (MRF) models provide a robust and unified framework for early vision problems such as stereo and image restoration. Inference algorithms based on belief propagation (BP) have been found to yield accurate results. These methods are good both in the sense that the local minima they find are minima over “large neighborhoods”, and in the sense that they produce highly accurate results in practice.

The general framework of this approach considers the inpainting problem in a global optimization framework where the optimal function is obtained by minimizing an energy function. Developing this idea, *Komodakis et al.* [24] has developed a novel optimization scheme to complete image based on belief propagation algorithm. Given an input image I as well as a target region T and source region S ($S = I - T$). The goal of inpainting problem is to fill in T in a visually plausible way by simply copying patches from S. The inpainting problem is turned into a discrete optimization problem and is modeled as discrete MRF. A set of labels L of the MRF consists of all patches from the source region S. Each label represents a patch, while each node n_i is a lattice point whose patch intersects the target region. The edges $\mathcal{E}$ of the MRF will make up a 4-neighborhood system on that lattice. Based on this information, the inpainting problem becomes a labeling problem in which find a set of labels $\widehat{l_i} \in L$ for each node n_i so that the total energy $E(\widehat{l_i})$ of the MRF is minimized:

$$E(\widehat{l_i}) = \sum_{i}^{N} V_i(\widehat{l_i}) + \sum_{(i,j)\in\mathcal{E}} V_{i,j}(\widehat{l_i}, \widehat{l_j}) \tag{2.18}$$

The single node potential $V_i(l)$, called label cost hereafter, for placing patch l over node n_i, encodes how well that patch agrees with the source region around n_i and equal the sum of squared differences (SSD) like equation (2.19).

$$V_i(l) = \sum_{p \in \Psi_{n_i}} M(n_i + p)(I(n_i + p) - I(l + p))^2 \tag{2.19}$$

where M is a binary mask. In a similar fashion, the pairwise potential $V_{ij}(l_i, l_j)$, due to placing patches l_i, l_j over neighbors n_i, n_j, measures how well these patches agree at the resulting region of overlap and is again given by the SSD over that region.

To minimize the energy function, belief propagation algorithm is applied [49]. However, the main drawback of this approach is computational cost of belief propagation. Therefore, two major improvements over standard belief propagation are introduced such as "*dynamic label pruning*" and "*priority-based message scheduling*". Together, they bring a dramatic reduction in the overall computational cost of BP which would otherwise be intolerable due to the huge number of existing labels. The detailed scheme of "Priority-BP" algorithm for image completion is presented in [24].

A different setting has been proposed by *Le Meur et al.* [25]. Authors have applied Belief Propagation algorithm more effectively to restore the damaged region. The main novelty of proposed method is the combination of multiple inpainted version of input image. Figure 2.11 illustrates the main concept underlying of the proposed algorithm.

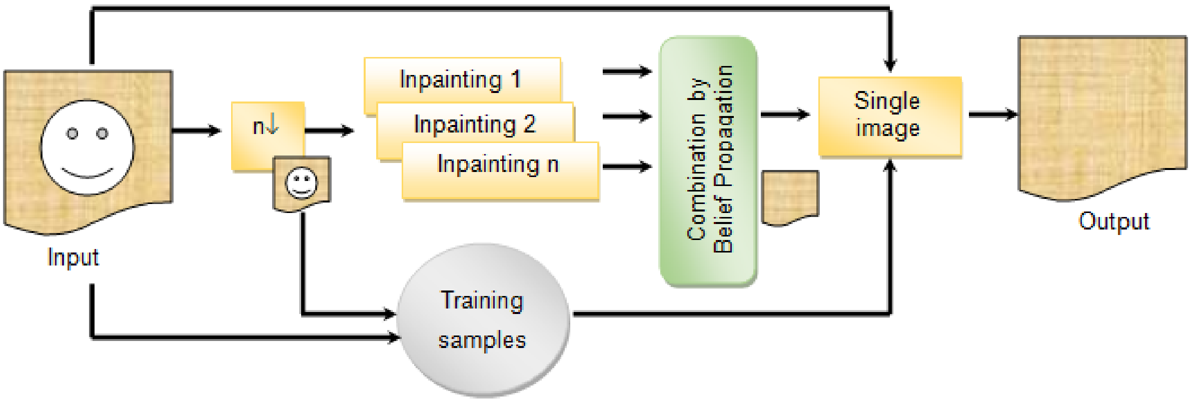
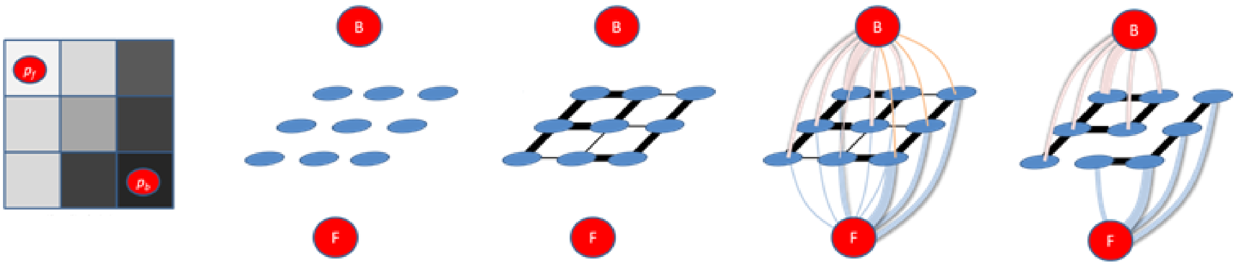


Figure 2.11. The framework of Le Meur's approach.

First, a non-parametric patch sampling methods used to fill in missing regions. The inpainting algorithm is preferably applied on a coarse version of the input picture. Regarding priority for filling order, two data terms have been used: tensor-based [50] and sparsity-based [51]. For texture synthesis, a similarity measurement as in [46] is used under a coherence constraint in [26].

Second, the output is enhanced the resolution and the subjective quality of the inpainted areas. The high-resolution is recovered using a single-image super-resolution approach. To combine multiple inpainted images, a Loopy Belief Propagation is applied with the same energy function as equation 2.19. Experimental results on a wide variety of images have demonstrated the effectiveness of the proposed methods.

2.3.2.3 Graph cuts for image completion



(a) (b) (c) (d) e

Figure 2.12. How to apply graph cuts for computer vision. (a) a given image; (b) nodes of graph; (c) edges of graph (n-link); (d) nodes with t-link and n-link; (e) a cut of graph.

As applied in the field of computer vision, graph cuts can be employed to efficiently solve a wide variety of low-level computer vision problems such as image smoothing, the stereo correspondence problem, and many other computer vision problems that can be formulated in terms of energy minimization.

In [27, 42], *Pritch et al.* considered and solved the inpainting problem in a different form of graph cuts. The main idea of inpainting problem is to copy pixels from the known region into the unknown regions. Thus, it is stated as a graph labeling problem which defines a relationship between them. Based on this observation, a shift-map, $sm(x, y) = (\nabla_x, \nabla_y)$, is defined mathematically for the relationship and the output $O(x, y)$ can be derived from the input pixel $I(x + \nabla_x, y + \nabla_y)$. Then, the optimal shift-map sm minimizes the following energy function:

$$E(sm) = \alpha \sum_{p \in I} E_d(sm(p)) + \sum_{(p,q) \in N} E_s(sm(p), sm(q)) \tag{2.20}$$

where α is a user defined weight balancing the two terms. The energy function has two terms such as data term E_d which indicates constraints such as the change in image size, object rearrangement, or a possible saliency map, etc., while the smoothness term, E_s minimizing the new discontinuities in the output image caused by discontinuities in the shift-map. They are defined by following equations:

$$E_d(sm(p)) = \begin{cases} 0 & (p_i + \nabla i, p_j + \nabla j) \in \Omega \\ \infty & (p_i + \nabla i, p_j + \nabla j) \in \bar{\Omega} \end{cases} \tag{2.21}$$

$$E_s(sm(p), sm(q)) = \begin{cases} 0 & sm(p) = sm(q) \\ \delta I(sm(p)) + \beta\delta\nabla I(sm(p)) & otherwise \end{cases} \tag{2.22}$$

where I and ∇I are the magnitude and gradient. δI and $\delta\nabla I$ denote their differences. Parameter β is a weight to combine these two terms. After defining a global objective function and its constraints, the graph-cuts algorithm [52, 53] is applied to solve it. However, finding the optimal graph labeling can be computationally infeasible, due to the very large number of nodes and labels. To overcome this limitation, a hierarchical approach is designed to reduce the memory and computational requirements of graph-cuts algorithm as in [26]. Although the hierarchical approach is not guaranteed to give the global optimum, the results are very good as can be seen in the examples (Figure 2.13).

The proposed approach may have unexpected bias, and their visual quality is very often unsatisfactory. In [28], authors improve the data term by comparing the patches Ψ_p and $\Psi_{p+sm(p)}$ for the pixels near boundary $\partial\Omega$. The patches are compared only using their known pixels. This term ensures the continuity of the reconstruction on the boundary.

$$E_d(sm(p)) = \sum_{p \in \partial\Omega} D(\Psi_p, \Psi_{p+sm(p)}) \tag{2.23}$$

where $D(\Psi_p, \Psi_{p+sm(p)})$ measures the similarity between two patches based on the Euclidean distance. To further enforce global consistency and to speed up convergence, a multiscale Gaussian pyramid is integrated to solve the energy function. Moreover, the use of a feature vector representation allows to compensate the loss information at low resolution levels. A series of experimental results and comparisons with existing inpainting methods show a good performance of the proposed algorithm.

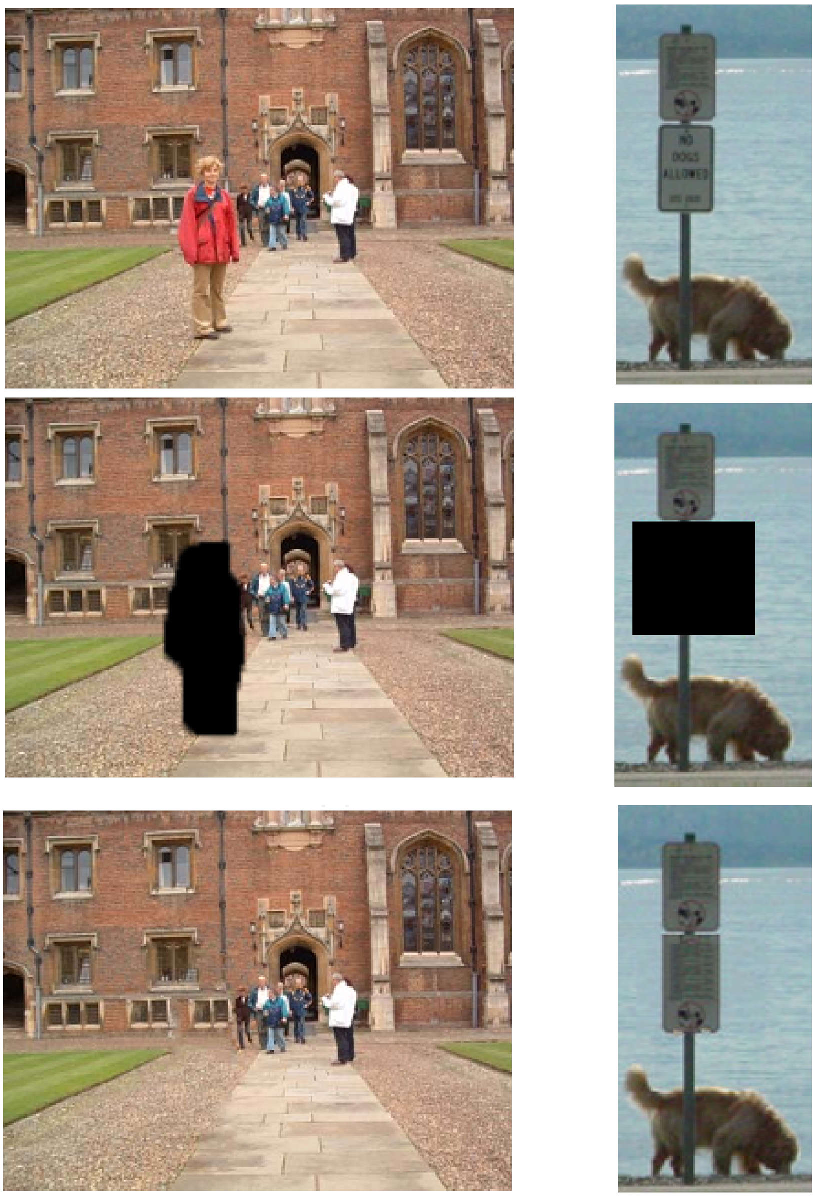


Figure 2.13. Inpainting based on the shift-map optimization [52]. From top to bottom: original image, black pixels need to be removed, and output.

An extension of this approach has been developed for video inpainting in [47]. A 3D offset, spatio-temporal displacement $sm(x, y, t)$, including temporal component is replaced for 2D offset. For further study, please refer to [47].

2.4 Hybrid inpainting methods

In general speaking, both texture synthesis and diffusion have their own advantages and drawbacks for image inpainting. Most inpainting methods in the first group presented so far perform well in case of small or thin damaged regions. However, they are unable to restore texture properly, and they introduce blur artifacts on large inpainting regions. Whereas, most inpainting methods in the second group are not able to handle sparse inpainting domains where no valid squared patch can be found that does not reduce to a point.

Indeed, natural images are composed of structures and textures. Structures refer to edges or contours and textures are image regions with homogeneous patterns or feature statistics. This is why pure texture synthesis techniques cannot be efficiently applied to missing regions/objects with composite textures and structures. To overcome the weakness of these methods, there have been several attempts to explicitly combine texture-based and geometry-based methods for completing the damaged regions, called *hybrid methods*.

The work of *Bertamío et al.* [29] used an additive decomposition of the image to be inpainted into a geometric component that contains all edges information, and a texture component. Then, the geometric image is interpolated with the method proposed in [6], while the texture image is restored using the method introduced by *Efros and Leung* [14]. The final image is obtained by addition of the restored texture and geometric components. In a few situations where the additive decomposition makes sense, this approach does indeed improve the result and extends the applications domain of inpainting. A basic framework of such an approach is shown in Figure 2.14.

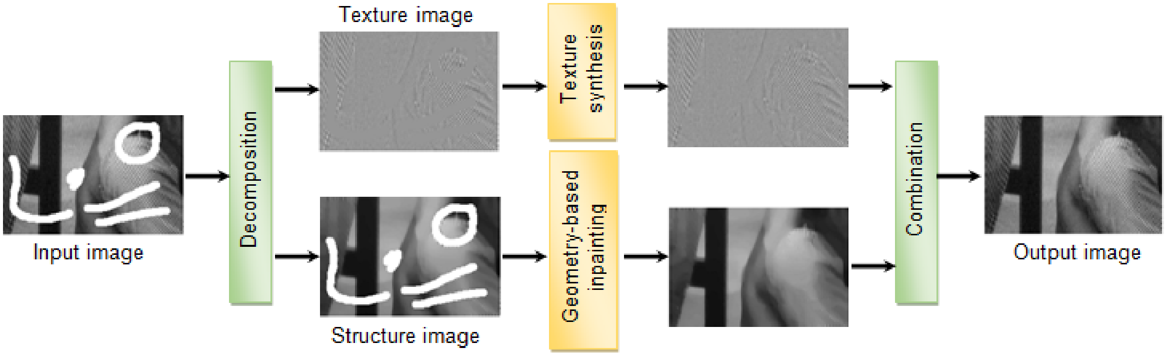


Figure 2.14. Basic scheme of the method in [29]. The input image is decomposed into the structure and texture images. These two images are reconstructed via inpainting [6] and texture synthesis [14]. The output is obtained by adding back two reconstructed images.

In [30], *Bugeau et al.* introduced a different combination of two kinds of inpainting methods. The original image is first inpainted with the anisotropic smoothing method from *Tschumperlé* [12] before decomposing into two parts: a structure and a texture image using the texture-structure image decomposition in [54]. Because the diffusion algorithm is unable to restore the texture parts, the texture image is then reconstructed using the algorithm in [17]. For each pixel in the damaged regions, a texture synthesis is carried out if that pixel does not belong to a strong structure. On the other hand with diffusion method, we should keep the current intensity values on the important structures and only apply texture synthesis on the other pixels. The notion of strong structure depends on the force of the gradient which can be characterized by the tensor of structure eigenvalues as equation (2.24).

$$\lambda^{+}(p) - \lambda^{-}(p) < \beta \tag{2.24}$$

where β is a threshold equal to mean:

$$\beta = \frac{\sum_{q \in \Phi} \lambda^{+}(p) - \lambda^{-}(p)}{|\Omega|} \tag{2.25}$$

Finally, the restored image is obtained by adding back the restored texture image into the diffused image.

Authors in [31] introduced a different process of unifying the geometric and exemplar approaches for image inpainting based on graph regularization. The advantages of using graph is that they can model well the interactions between data. The local or non-local regularization can be derived by the way the connections are made in the graph. Moreover, a novel data term for filling-order has been proposed based on the gradient of the current pixel as follows:

$$D(p) = |\nabla I(p)| + log(1 + |\nabla I(p)|) \tag{2.26}$$

where $\nabla I(p)$ is the gradient of the image at pixel p. The detailed unification algorithm and illustrations are given in [31]. Unlike other works, the decision regarding exemplar or diffusion is made dynamically at each pixel based on texture information. Gabor filter responses at different orientations and scales are applied to automatically decide if diffusion or exemplar method to be used in the course of inpainting.

2.5 Beyond single image inpainting

All the methods mentioned above involve just a single image. On the other hand, the missing information is filled in based on the known information from the image itself. Since the development of internet as well as the image retrieval techniques, searching a set of images similar to query image becomes easier. In this case, inpainting a single image using information from several images could be a valuable direction.

An idea of using large regions of other images for image completion was explored by *Hays and Efros* [32] who performed inpainting of a single image using information from a database with several millions of photographs. They used the gist scene descriptor [55, 56] which has been shown to perform well at grouping semantically similar scenes (eg. city, tall buildings, office, fields, forest, beach) and for place recognition to reduce the search space from two million to two hundred images, those images from the database which are semantically closer to the image the user wants to inpaint. Using template matching, they align the two hundred best matching scenes to the local image around the region to inpaint. Then, they composite into the target image each matching scene into the target using seam finding and image blending.

Whyte et al. [33] tackled the inpainting problem using images of the same scene retrieved from Internet using viewpoint-invariant image search. Only top 30 search results are used for completing the original image. The retrieved images are registered to the query image with multiple homographs allowing registration of multiple scene planes or images taken from the same viewpoint. Each registered image is applied to propose a solution, by combining it into the target region using Poisson blending, to minimize the effect of differences in lighting. In the final step, a MRF formulation allows the multiple proposals to be combined into a single result, selecting reliable regions of different proposals and concealing the boundaries between regions. Figure 2.15 shows an example of *Whyte*'s system based on OxfordBuildings Database.

The common shortcoming of these methods is that it relies on managing and operating a huge image database. When the algorithm fails, it can be due to a lack of good scene matches (if the target image is atypical), or because of semantic violations (e.g. failure to recognize people hence copying only part of them), or in the case of uniformly textured backgrounds (where this algorithm might not find the precise same texture in another picture of the database).

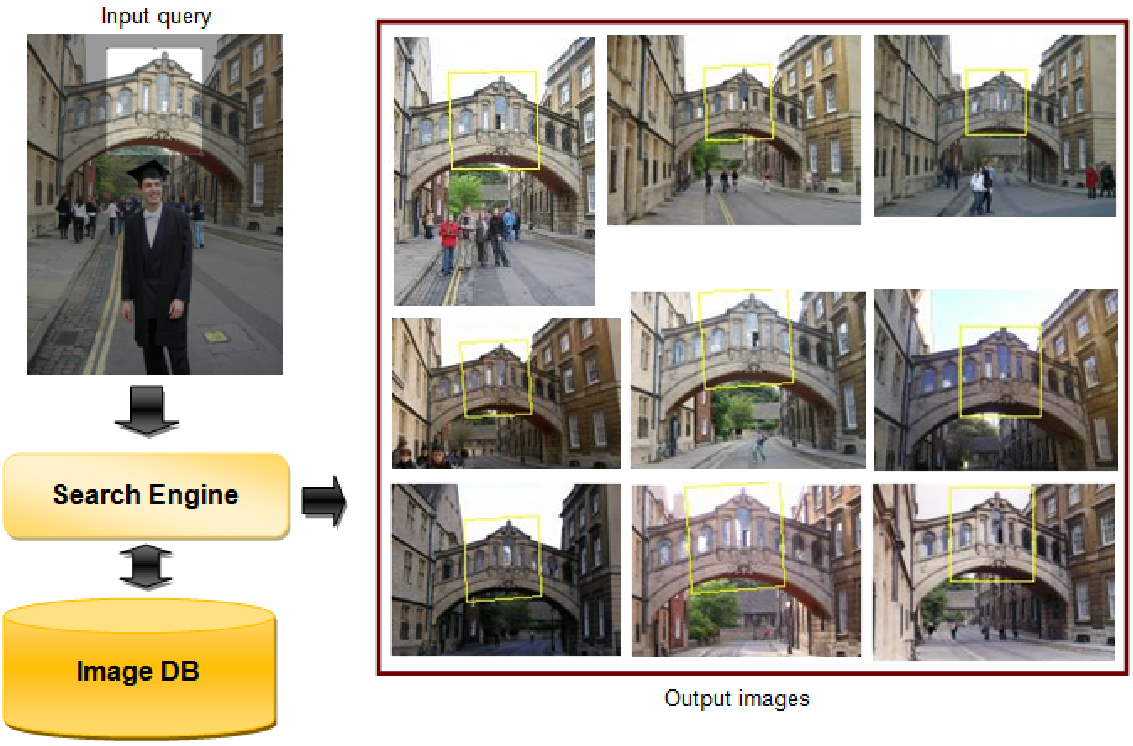


Figure 2.15. An example query (left image) and the first 9 results returned by the viewpoint invariant image search engine.

2.6 Conclusion

In this chapter, an overview of current inpainting algorithms is presented. Starting from the earliest work until the present, a series of inpainting algorithms has been summarized to provide for reader a basic look of digital image inpainting developed during the last decade.

While many PDE/variational methods have been developed and considered as the first group of inpainting techniques, the experimental results indicated that most methods in the first group were particularly effective for the synthesis of long, thin regions. Since these methods focused on maintaining the structure of the inpainting area, they are less suitable for synthesize semantic textures or structures. First, the interpolation tends to produce a blurring effect on missing spots in the image. For scratches and thin markings, the blurring can produce a very acceptable output image. Second, the interpolation does not distinguish the structures and edges in the image. Namely, if a missing region intersects an edge in the image, typically the output will not connect the edge through the missing region and this can lead to very noticeable defects in the output image.

The second group is texture-oriented methods inspired from texture synthesis techniques. These methods tend to be less computationally efficient than interpolation and overcome the problem associated with interpolation. Since the actual pixel values of the known regions are used instead of average as in interpolation, the blurring effect is generally removed completely. This group could be further subdivided into two sub-groups: greedy strategy and global optimization strategy. The greedy strategy have acceptable computation time and take into account human perception features (priority is designed based on the salient structures considered as important for human perception), but they show some common problems such as local optimization, patch priority, patch selections, etc. Whereas, global optimization strategies often provide better results. Nevertheless, they are computationally expensive. This is mainly due to the fact that time complexity increases linearly both with the number of source pixels and unknown pixels.

Some other methods are categorized as the third group, hybrid methods which combine texture-based and geometry-based methods in order to complete properly the damaged regions. The original image is decomposed into two parts: a texture image and a geometry image. Each part is restored using methods in the corresponding group. The final result is reconstructed by adding back two restored parts. In a few situations where the additive decomposition makes sense, this approach does indeed improve result and extends the applications domain of inpainting.

Recently, beyond single image inpainting, a novel direction of inpainting problem has been introduced that is inpainting a single image using information from several similar images instead of that image itself. These approaches also have drawbacks of managing and operating a huge image database. Indeed, the completion is meaningful if it is performed blindly, i.e. without reference to original images or other images.

Although hundreds of publications have been introduced during the last decade, there is still room for developing other approaches to overcome the limitation of the existing methods as well as obtain better outputs. One promising direction for improving them is use of perceptual models which can result in more plausible outputs. In summary, we believe that perceptual-based approaches are more potential and promising.

CHAPTER 3

A Hierarchical Approach for High-Quality Fast Image Completion

“Nothing takes place in the world whose meaning is not that of some maximum or minimum.”

Leonhard Euler

Contents

Image completion in the case of images with large missing regions is a very challenging problem. As summarized in the Chapter 2, several solutions have been developed to tackle this problem. In this chapter, we introduce a novel framework based on hierarchical representation which deals effectively with the case of filling in large missing areas. The following sections describe in detail the main stages of the proposed method. The performance of proposed methods is evaluated and discussed based on a series of experimental results.

3.1 A hierarchical approach for image completion

The exemplar-based methods consist of two key steps: the determination of filling order and the selection of the best matching patch. The crucial part of this model is the filling priority which is designed to preserve linear structure. The isophote-driven priority proposed in [17] generates pleasant outputs in many cases, but it still has many drawbacks. Many modified versions have been developed [18, 20, 21]. However, the regions with high isophote strength, which may be certain high frequency components such as noise and complex textures, are not equivalent to the structural features e.g. edges, contours, etc.; therefore, it may lead to undesired results.

The second part in this model is patch selection. By using only the best match sample, the method runs the risk of choosing a sample that is corrupted, or not a perfect match. Therefore, choosing the suitable patch for piecing also significantly affects to the inpainting quality of an image with redundant content where several samples could be combined to form a more robust estimate of the missing information. Some techniques [19, 57] have been introduced to improve for patch selection, but there still remain some issues such as introduction of blur artifacts because of interpolation or missing correct information for filling in a region when it is located far away from that region.

Our framework is designed to overcome the limitations of the current exemplar-based methods in both these steps. The proposed technique is inspired by the observation that the human visual system is sensitive to salient structures which are stable and repetitive at different scales. In other words, one can still see the main structural features when the resolution of the image is reduced to a given factor. Therefore, a hierarchical image inpainting scheme is developed in order to control and preserve salient features during the completion process. This scheme allows restoring the missing regions in a visually plausible way.

3.1.1 Algorithm overview

The proposed approach is composed of two main successive operations based on a greedy strategy. Firstly, a single resolution is restored as in the framework [17]. At low resolution the inpainting is less sensitive to local singularities and noise effect, but it may be affected by blur artifacts due to the reduction operation. Thus, in order to classify more logically image components, a novel priority definition is introduced. Secondly, the damaged region on the higher resolution image is completed by exploiting spatial information contained in the lower resolution image provided at previous step. At higher resolution, multi-resolution patches are the best way to preserve the main features in the restoration process. By applying window-based priority, the decision is more suitable to maintain the high frequency components in a visually plausible way. By estimating patches at different scales, the selected patch would be more consistent and faithful to the source patch than one in a single resolution. Furthermore, it is much less computationally demanding than when processing the original full resolution, and hence it copes well with the problem of large region inpainting.

At the beginning of the proposed algorithm, a Gaussian pyramid is constructed from the image to be inpainted. This step consists of low-pass filtering and downsampling images of the preceding level of the pyramid. According to the above idea, a set of images $\{G_0, G_1, ..., G_N\}$ with various levels of details is generated, where $G_0 = I$ is the input or original image [58]. The number of pyramid levels is linked to the original size of the image and the smallest allowed resolution. The minimum size should be adjusted according to the width and height of the original image. In our experiments, to avoid missing relevant details, the following condition is used: $min(width(G_N); height(G_N)) \geq 32$. The inpainting regions are also reduced level by level. Then, an exemplar-based method for image completion with some modifications is applied at each level. An iterative process is completed gradually from the lowest resolution, G_N, to the highest resolution, G_0. The algorithm is fully described in Algorithm 3.1. Although different criteria for stopping the process can be used depending on application, we opted for a fixed number of iterations. All the results shown in our thesis were computed with only 2 or 3 iterations.

Algorithm 3.1. The proposed framework.

```
$G_0 = I$;
$\{G_1, G_2, ..., G_N\} = buildPyramid(G_0)$;
Complete $G_N$ with the scheme in [17] using window-based priority in section 3.1.2.1;
while (stop=false) do
    for detail level $i = N - 1$ downto 0 do
        Complete $G_i$ with the scheme in [17] using using multi-resolution patch in
        section 3.1.2.2 and the window-based priority in section 3.1.2.1;
    end
    $\{G_1, G_2, ..., G_N\} = buildPyramid(G_0)$;
end
Output = $G_0$;
```

3.1.2 Inpainting the lowest resolution image

In order to integrate some low-level features of the HVS and simulate its hierarchical perceptual properties, an exemplar-based method is applied to the lowest resolution image, G_N. In the proposed scheme, an extension of [17] is developed to complete the coarse resolution. Generally, there are two main steps in this scheme: i) determination of the filling order and ii) selection of the best matching patch. Both steps are analyzed in details in the succeeding two subsections.

3.1.2.1 Window-based priority for filling order

Naturally, the exemplar-based algorithm is a greedy strategy, an acceptable computation load; however, it suffers from the common problems of the greedy algorithms, in which the filling order (namely priority) is very critical. An appropriate definition of the priority is essential since the decision, taken with it, is irreversible in the following stages. Otherwise, errors may accumulate continuously because no improvement is achieved on previous stages. Many formulations have been proposed for priority selection [17, 18, 20, 21]. In this work, we would like to introduce a new priority, called *window-based priority*, which is considered as more robust and efficient than others.

Classically, a priority, $P(p)$, is composed of two terms: i) confidence term and ii) data term [17]:

$$P(p) = C(p)D(p) \tag{3.1}$$

In [18], the authors discovered that the confidence term in [17] decreased exponentially and proposed an additive form of priority which used weights to maintain a balance between the confidence term and the data term.

$$RP(p) = \alpha((1-\omega)C(p) + \omega) + \beta D(p) \tag{3.2}$$

where $\omega \in [0,1]$ is the regularizing factor for controlling the curve smoothness (in experiment, ω is often set to 0.7), the value range of $C(p)$ is regularized to $[\omega, 1]$, and α, β are positive parameters with $\alpha + \beta = 1$.

Here, we concentrate on the analysis of the data term which distinguishes the structures from the textures. The confidence term is not mentioned here since it does not provide any additional improvement. A data term with a high value indicates the presence of a structure or contour, while a low value introduces the presence of texture or flat region. In [17, 20], authors introduced a pixel-based data term, $D(p)$, depending proportionally on the isophote direction or gradient of the known region. Thus, if the gradient at pixel p is large, the priority will be high. By the way of explanation, when the gradient values of the texture components are greater than those of the structure components or when the regions are affected by noise, the isophote-driven priority method may violate the requirement of an appropriate priority rule and yields bad results. To overcome this difficulty, a patch-based data term is introduced as in [21, 51]. This solves to some extent the problem but at the cost of an increased running time and complexity.

Reasonably, the data term should be estimated from all neighborhood pixels in an exemplar centered on the current pixel, because the human vision is good at seeing the group of adjacent pixels rather than single pixel. Our first improvement focuses on a better definition of data term based on the local changes of pixel intensities in each window, W_p, centered at pixel $p(x, y)$ with shifted windows in different directions. The local change of intensity at each pixel $p(x, y)$ is characterized by the following second-moment matrix or structure tensor [59]:

$$M(p) = \sum_{W_p} G_{W_p}(x, y) \begin{pmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} \\ \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} & \left(\frac{\partial I}{\partial y}\right)^2 \end{pmatrix} \tag{3.3}$$

where G_{W_p} is a Gaussian window function. This structure tensor is a 2×2 symmetric and semi-positive matrix which captures the intensity structure of the local neighborhood. The 2D structure tensor and its eigenvalues, $\lambda_1 \geq \lambda_2 \geq 0$, summarize the distribution of the gradient within the defined window. The two corresponding eigenvectors represent two orthogonal directions along the local maximum and minimum variation of image intensities. Whereas, the eigenvalues measure the effective variations (strength of contours) of image intensities along these vectors. Consequently, our data term is defined as follows:

$$D(p) = \frac{\lambda_1}{\lambda_2 + \epsilon} \tag{3.4}$$

where ϵ is a very small positive value introduced to ensure computation stability (in our experiment, $\epsilon = 10^{-10}$). This data term is related not only to geometric features, such as contours or edges but also texture features. There are three cases to be considered for each window as those defined by *Beaudet* [60] and illustrated in Figure 3.1:

- If the data term is much greater than one i.e. one eigenvalue is high (λ_1 is high) and the other is low (λ_2 is low), the local shifts in one direction cause little change and significant change in the orthogonal direction; the window contains strong edges;
- If the data term is close to one, i.e λ_1 is approximate to λ_2, there are two possible cases:
  - If both eigenvalues are high, the shift in any direction will result in a significant change. This indicates a texture or complex structure.
  - If both eigenvalues are small, the shift in any direction will cause a little change. The patch is of approximately constant intensity (flat region).

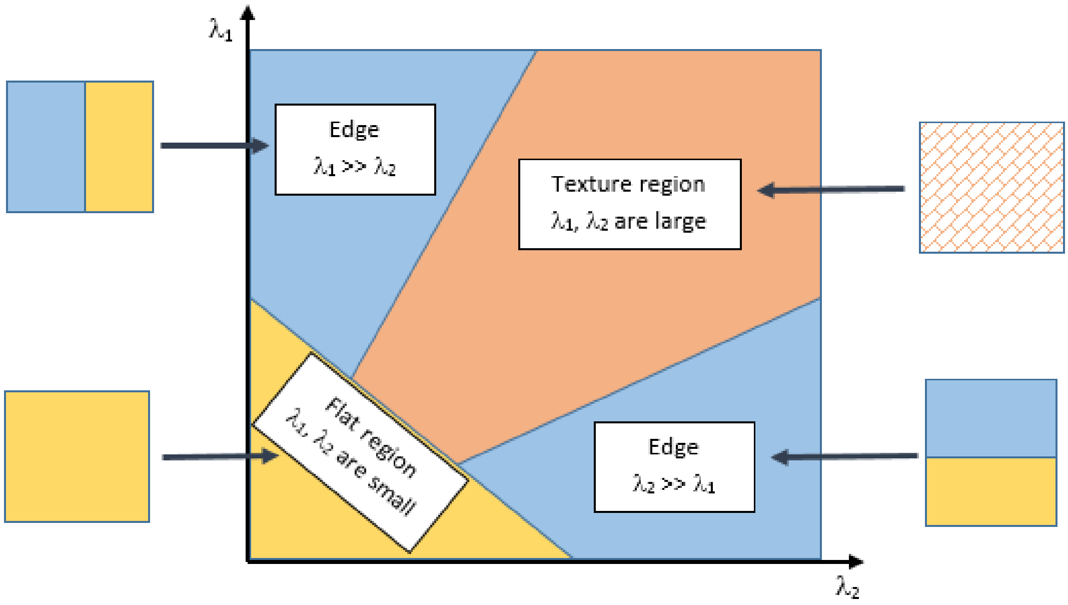


Figure 3.1. Classification of patches via eigenvalues

Because both confidence and data terms are evaluated by contribution of all pixels in a window, our priority is called a *window-based priority* [61]. With this priority, the patches are classified in a more robust way and the computation time is acceptable. Table 3.1 illustrates some numerical values corresponding to subimages in Figure 3.2. In our opinion, the contours should be preserved, thus the priority of pixels on the contour should be higher than others. Obviously, the values in Table 3.1 demonstrate that our priority is more reasonable than the ones proposed in [17] and [20].

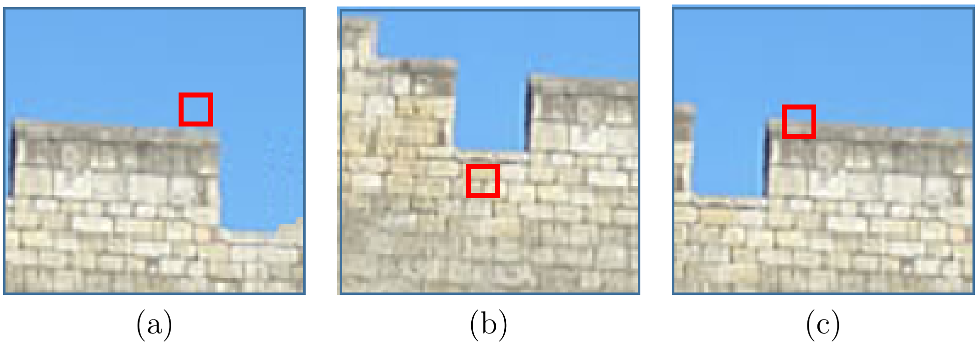


Figure 3.2. The window-based priority of different regions. (a) Flat region; (b) Texture region; (c) Edge or contour.

Table 3.1. Priority of the different regions.

Patch	Priority in [17]	Priority in [20]	Proposed Priority (λ_1, λ_2)
Flat region (Fig. 3.2-a)	0.001	3.269	0 (*0, 0*)
Texture region (Fig. 3.2-b)	0.032	3.916	1.374 (*0.067, 0.049*)
Edge or contour (Fig. 3.2-c)	0.024	3.130	4.059 (*0.025, 0.006*)

3.1.2.2 Patch selection

The second step of the proposed algorithm consists in finding the suitable patch for the filling process. The similarity measurement, computed based on all known pixels in the patch, should be consistent with human perception. When using only color information, it is insufficient to propagate accurate linear structures into the target region and leads to uncontrolled and incoherent increase. This is mainly due to the fact that the perceptual color appearance depends not only on the color of the observed patch but also on the surrounding and the context in which the patch is perceived. Consequently, to solve this problem, a similarity measure which takes into account the difference in colors and gradients is proposed below:

$$d(\Psi_p, \Psi_q) = \sum_i (\theta(I_p^i - I_q^i)^2 + (1-\theta)(\nabla I_p^i - \nabla I_q^i)^2) \tag{3.5}$$

where I_p and I_q are the corresponding RGB vectors in patches Ψ_p, Ψ_q; ∇I_p and ∇I_q represent the image gradient vectors; θ is a user defined weight balancing the two terms, here we set $\theta = 2/3$ in the experiment. The image gradient is added as an additional weight to maintain the color intensity variation from the known region to unknown region. The target patch with the minimal distance to the source one, Ψ_p, is the one that should be selected. It is given by:

$$\Psi_{\widehat{p}} = \arg\min_{\Psi_q \in \Phi} \{d(\Psi_p, \Psi_q)\} \tag{3.6}$$

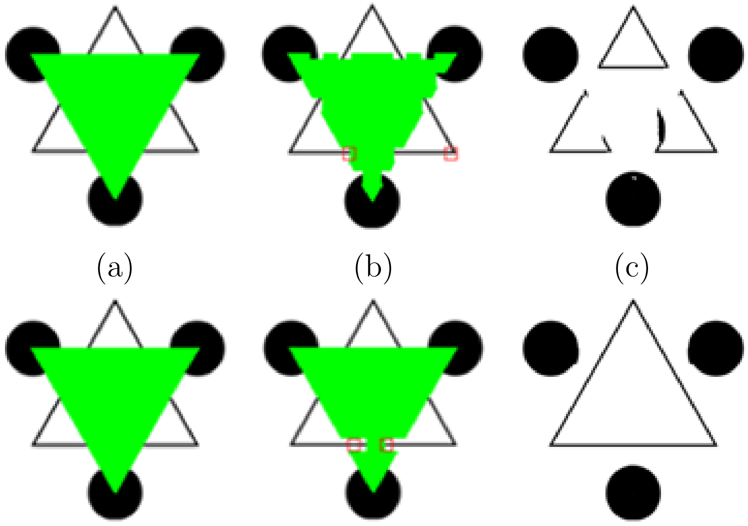
As noticed in [19], a major problem of local neighborhood search is its tendency to get stuck at a particular place in the same image and to produce verbatim copying. This may also generate blocking artifacts around the inpainted regions. In order to address this problem, we propose an improvement for the patch selection step. The idea is based on the fact that patches in the neighborhood in the input image should remain neighbors in the output image. First, K most similar patches obtained by the local neighborhood search patches are used as candidates. Second, for each patch, Ψ_p, a standard deviation describing the variability of neighboring source patches is formulated as follows:

$$V(\Psi_p) = \sum_{E \in \{R,G,B\}} \sqrt{\frac{\sum (E(\Psi_p) - \bar{E}(\Psi_{N(p)}))^2}{|N(p)|}} \tag{3.7}$$

where $N(p)$ is the neighborhood centered at p, $E(\Psi_p)$ is the variance of pixel values at the neighboring patch Ψ_p in each of the RGB channels, and $\bar{E}(\Psi_p)$ is the mean variance of $|N(p)|$ neighboring patches in RGB channels. The size of $N(p)$ is a global parameter that should be chosen larger than the patch size. Consequently, the chosen patch must satisfy the following equation:

$$\Psi_{\widehat{p}} = \arg\min_{\Psi_q \in \Phi} \{|V(\Psi_p) - V(\Psi_q)|\} \tag{3.8}$$

In order to illustrate the effect of patch selection, an interesting example is illustrated in Figure 3.3, a completion of the Kanizsa triangle. One can notice that unsuitable patch selection, as shown in Subfigure 3.3-b, may lead to unexpected results or artifacts (see subfigure 3.3-c). On the contrary, an appropriate selection, as shown in Subfigure 3.3-e, produces satisfactory results (see Subfigure 3.3-f). Finally, the missing pixels are copied from the corresponding pixels in the selected patch.

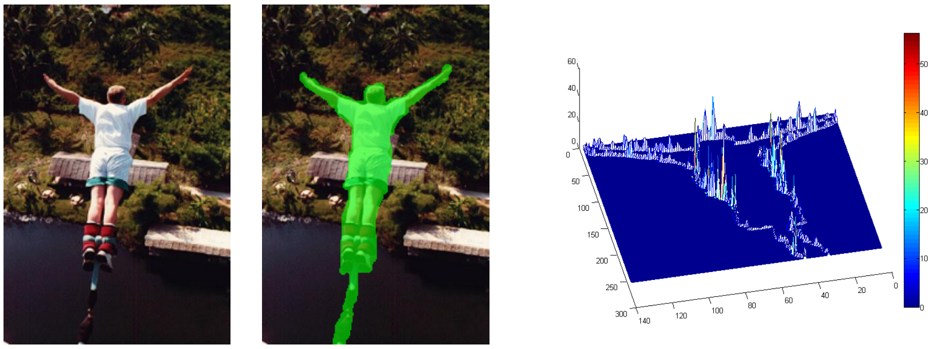


(a) (b) (c)

(d) (e) (f)

Figure 3.3. A restoration of the *Kanizsa* triangle using inappropriate and appropriate patch selection. (a) & (d) image to be inpainted; (b) & (e) a patch selection respectively with and without improvement; (c) & (f) Final results.

Figure 3.4 illustrates priority of pixels on the boundary of inpainting regions where the red color refers to highest value and the blue color refers to lowest one. One can notice that higher values can be observed on contours as explained previously.



(a) Original image (b) Mask image (c) Our priority

Figure 3.4. Illustration of the proposed priority focusing on edge preservation.

Figure 3.5 illustrates the performance of the proposed method on various low-resolution images. These images are downsampled versions of the original ones (the down sampling factor is set to 4 in both directions). Four inpainting methods, namely *Criminisi et al.* [17], *Wu et al.* [20], *Zhang et al.* [21] and *Cheng et al.* [18], belonging to the second group of approaches, are implemented with the same patch size for a fair comparison. Visual inspection of the results given in Figure 3.5 shows that the proposed approach achieves good results in most cases compared to the others.

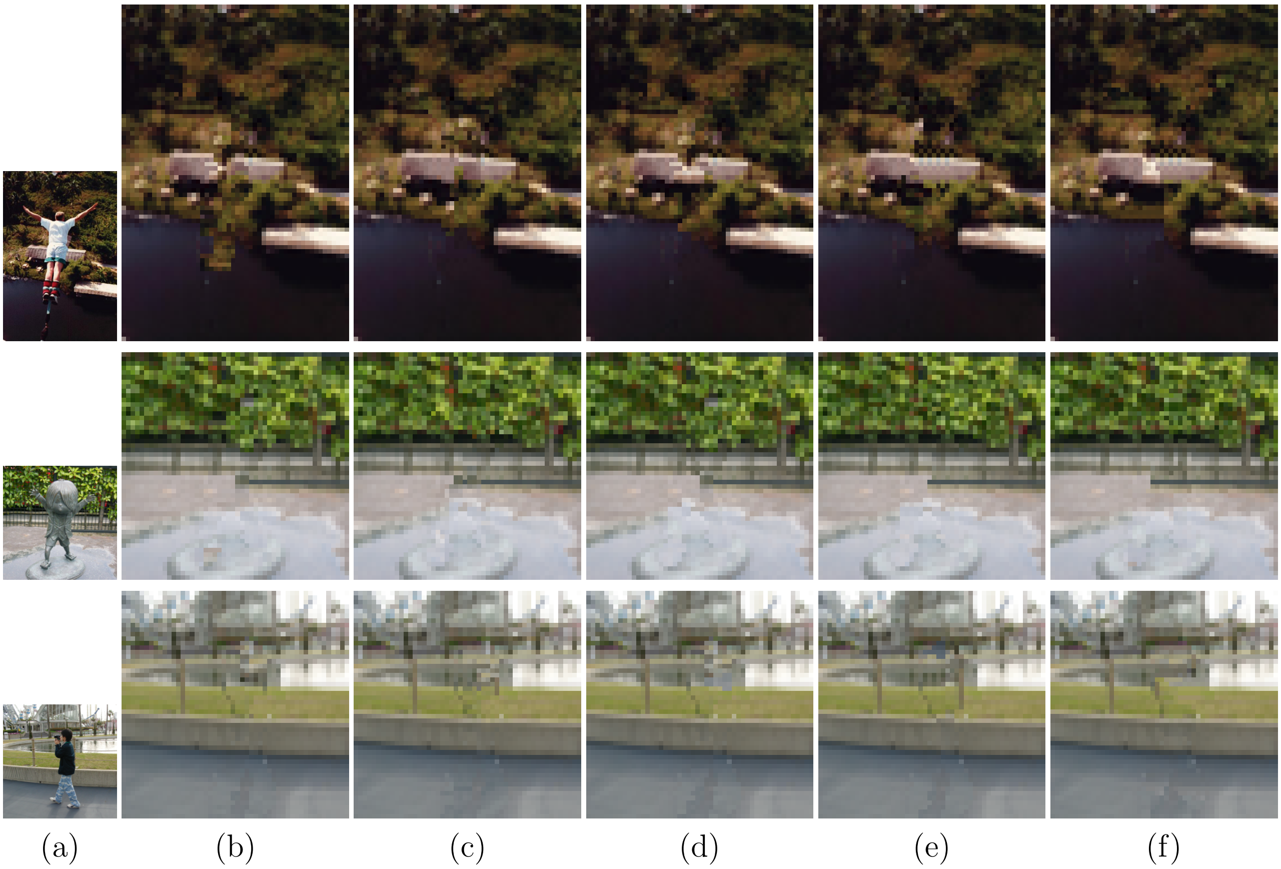


Figure 3.5. Completion of low resolution images. (a) image to be inpainted. Output when using priority adopted by (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18] and (f) our proposal.

3.1.3 Multi-resolution patch for higher resolution image completion

The second advantage of our method is the use of a multi-resolution patch-based scheme on the image pyramid [58]. A multi-resolution patch, Ψ_p, consists of two consecutive scales: one at the current level l, Ψ_p^l, and one at its parent level $l+1$, Ψ_p^{l+1}. Figure 3.6 illustrates an example of multi-resolution patch Ψ_p, that contains various single resolution patches with different sizes. The sizes of patches are global parameters and denoted as $\{(w_1 \times w_1); (w_2 \times w_2)\}$, where the former is for current level and the later is for the higher level. w_1 is often greater than w_2 because of the image resolution. In our work, we set $w_1 = 2w_2$ because the ratio between two consecutive images in the pyramid representation is 2. Consequently, the similarity between two multi-resolution patches at higher resolution is measured by the sum of the squared distance of all known pixels in both adjacent levels and is reformulated as follows:

$$d(\Psi_p, \Psi_q) = d_1(\Psi_p^l, \Psi_q^l) + d_2(\Psi_p^{l+1}, \Psi_q^{l+1}) \tag{3.9}$$

where d_1, d_2 are similarity of the known pixels in the current and higher level, respectively; they are computed by Equation (3.5) within the corresponding patches.

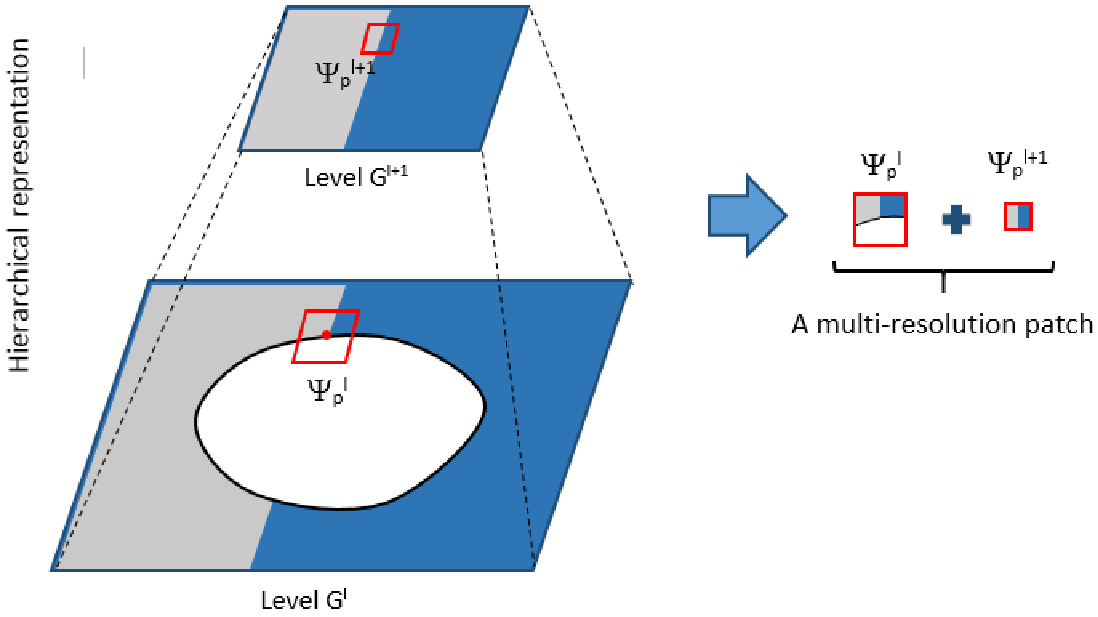


Figure 3.6. A multi-resolution patch composed of pixels on different scales

Figure 3.6 exhibits an example of multi-resolution patch which is being applied for image completion using multi-resolution representation. All pixels of multi-resolution patch at higher level $l+1$ have been restored completely, while a part of current level l marked by white is needed to be restored. This example shows a relationship between the restoration of the current and higher level patches.

3.1.4 Experimental results

The proposed image completion framework was tested on a series of different natural images and the results are presented in this section. To compare the effectiveness of our algorithm and to do it in an equal footing, we apply our technique on both data provided by [17] and other real images. The performance is compared with the existing exemplar-based methods in terms of the inpainting quality.

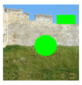
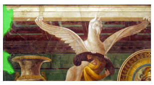
In most experiments, all of the algorithms are implemented in the same programming framework, C/C++ language and Matlab. The parameters are fixed at constant values for all tested methods so as to avoid any bias in the comparison. Specifically, the patch sizes are set to $\{(9\times 9);(5\times 5)\}$ for our approach and (9×9) for other exemplar-based methods. The size of window priority is set to (7×7). The missing regions are marked by the green color (R=0, G=255, B=0).

Table 3.2. Tested inpainting algorithms

Year	Authors	Group	H-type
2004	Criminisi *et al.* [17]	Texture	Large
2005	Cheng *et al.* [18]	Texture	Large
2006	Wu and Ruan [20]	Texture	Large
2011	Zhang and Lin [21]	Texture	Large

Since our proposal belongs to the second group, we selected the methods corresponding to the approaches of *Criminisi et al.* [17], *Wu et al.* [20], *Zhang et al.* [21] and *Cheng et al.* [18] for a fair comparison. Table 3.2 lists the details of inpainting techniques used in this section.

Table 3.3. Selected testing images

Image	Name	Size	Hole size	Type
	boat	300 × 225	10.73%	natural
	horse	300 × 225	9.29%	natural
	wall	256 × 256	14.8%	natural
	angle	300 × 252	5.83%	archaeological
	affresco	726 × 373	3.43%	archaeological

In some applications, the visual quality of the outputs is the most important consideration. Therefore, in this work, we concentrate on only evaluating and comparing the outputs in terms of the visual quality. Five images, including three natural images and two real archaeological images (listed in Table 3.3), were presented to validate our algorithm in comparison with other approaches. The damage areas occupy 10.73%, 9.29%, 14.8%, 5.83% and 3.43% of the whole image, respectively. Although areas of holes account for small percentage of the area, they are still considered as large areas with respect to our definition in the section 1 of chapter 2.

The visual results in Figures 3.7, 3.8 and 3.9 confirm the effectiveness of our algorithm in compare with existing methods in case of natural images, while the visual results in Figures 3.10 and 3.11 give an outstanding quality of our algorithm in case of archaeological images.

As we can observe from these figures, our method produces a better visual quality than the other methods. We should mention that for performance evaluation, we use only subjective criteria, i.e., visual appearance of the output because there is no efficient numerical estimation for outputs of inpainting methods.

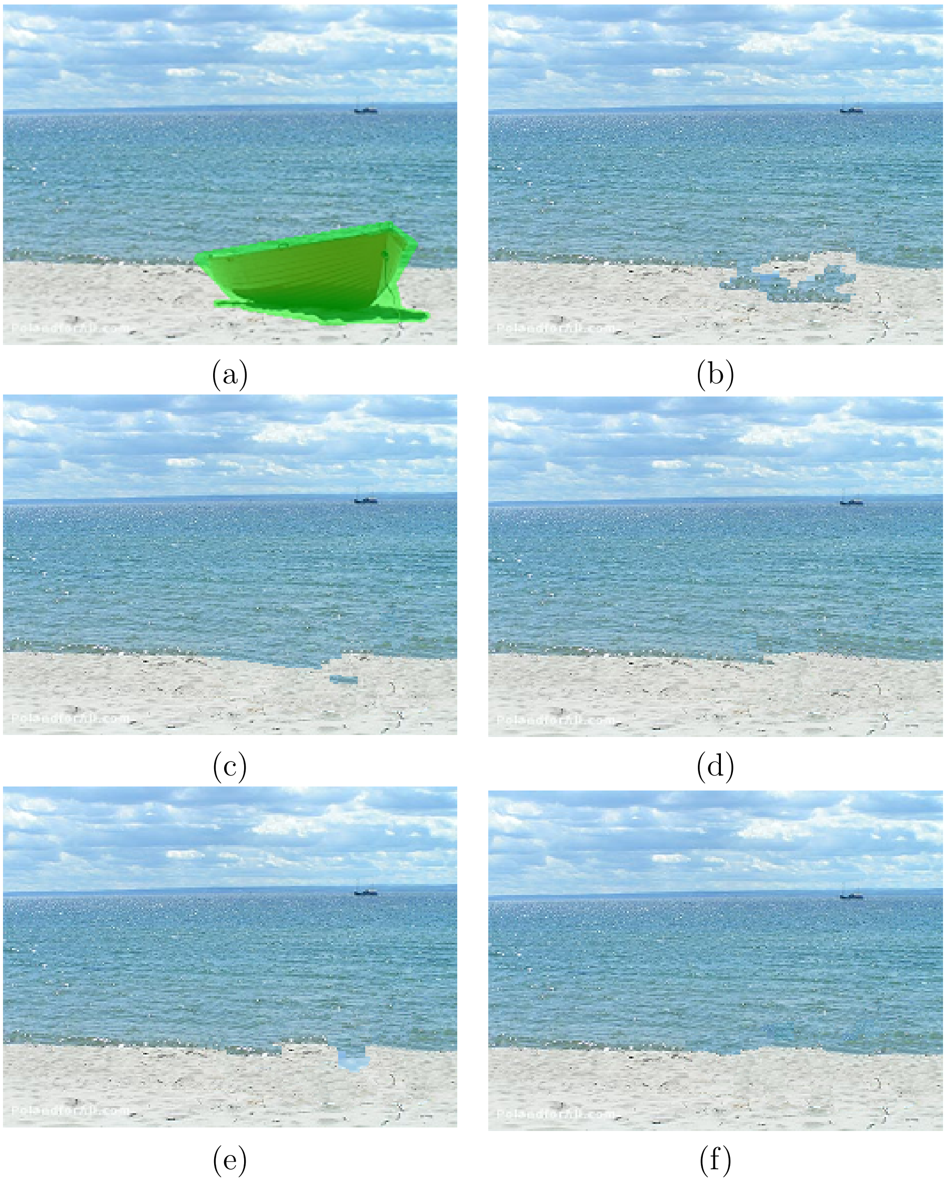
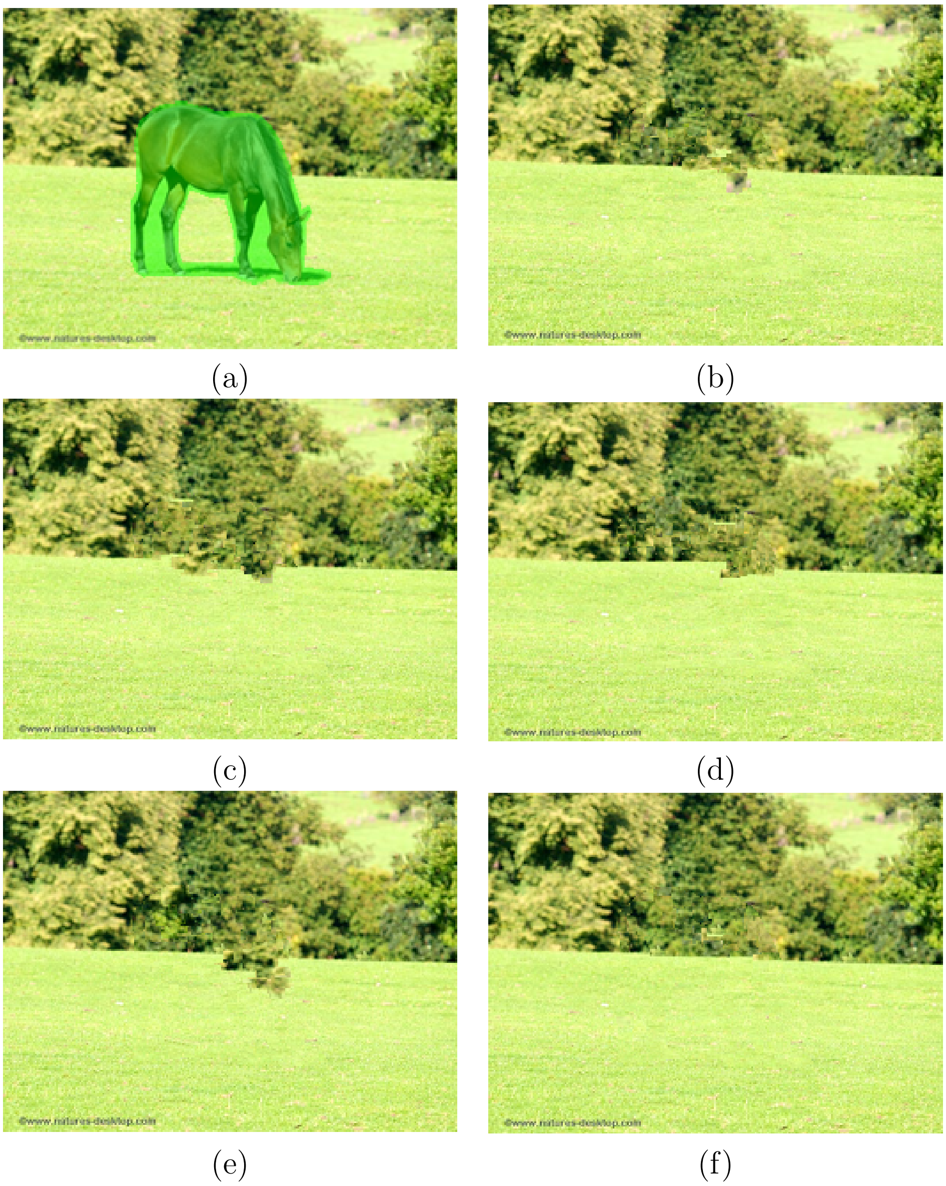
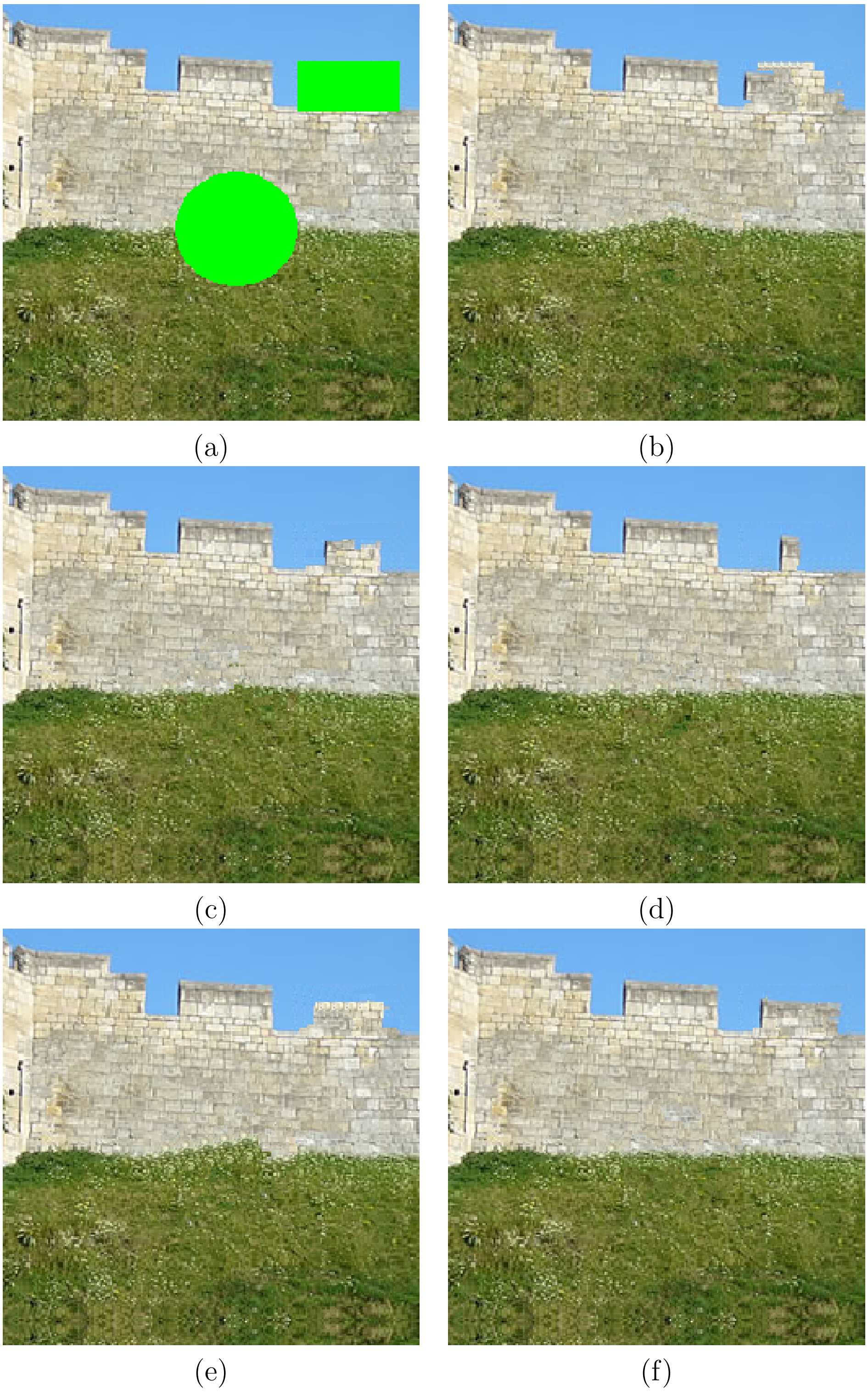


Figure 3.7. Visual comparison of 'boat' image across all five algorithms: (a) The image to be inpainted; The ouputs using approach of (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18]; and (f) our proposal.



(a) (b) (c) (d) (e) (f)

Figure 3.8. Visual comparison of 'horse' image across all five algorithms: (a) The image to be inpainted; The ouputs using approach of (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18]; and (f) our proposal.


(a)

(b)

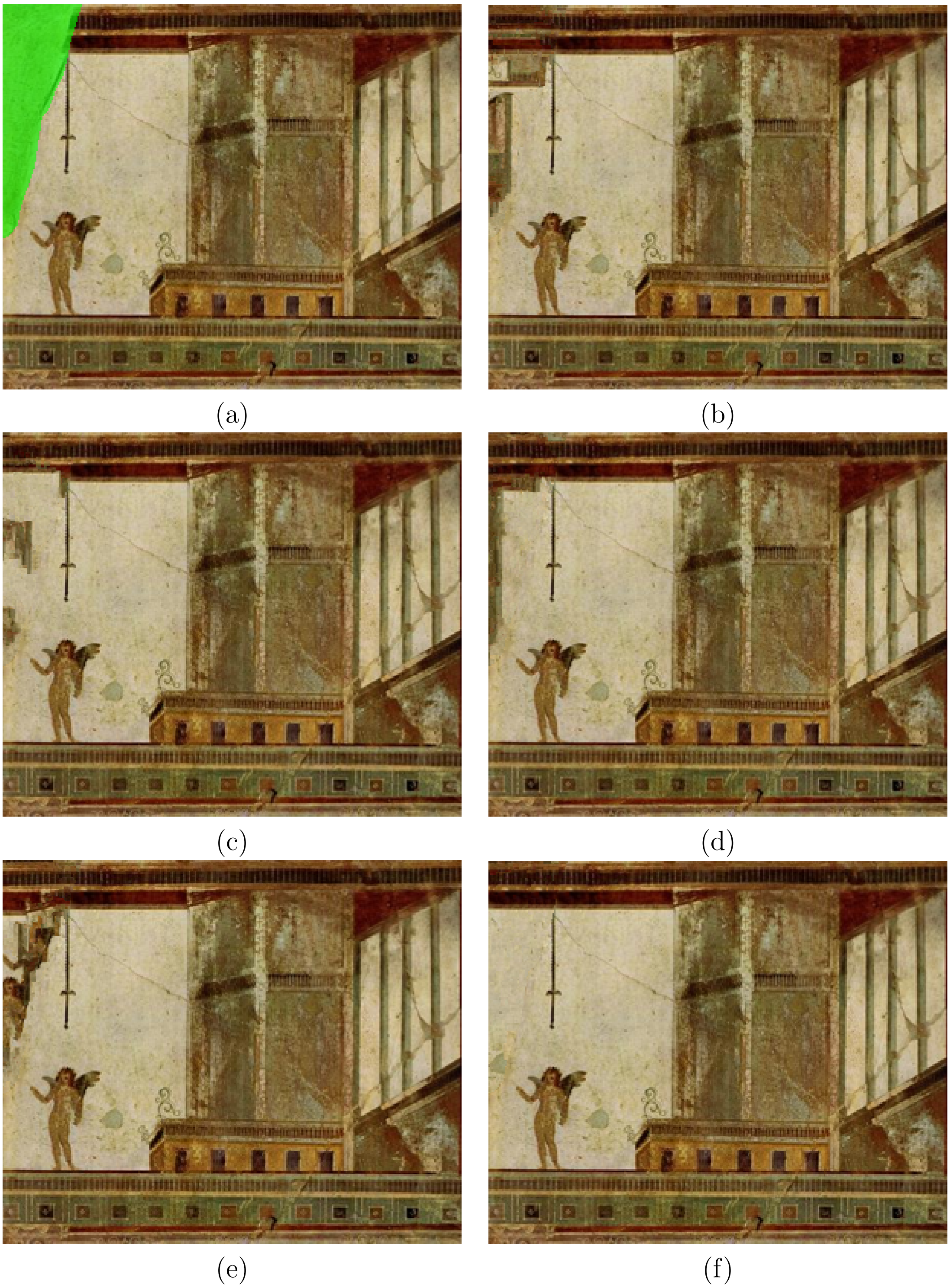
(c)

(d)

(e)

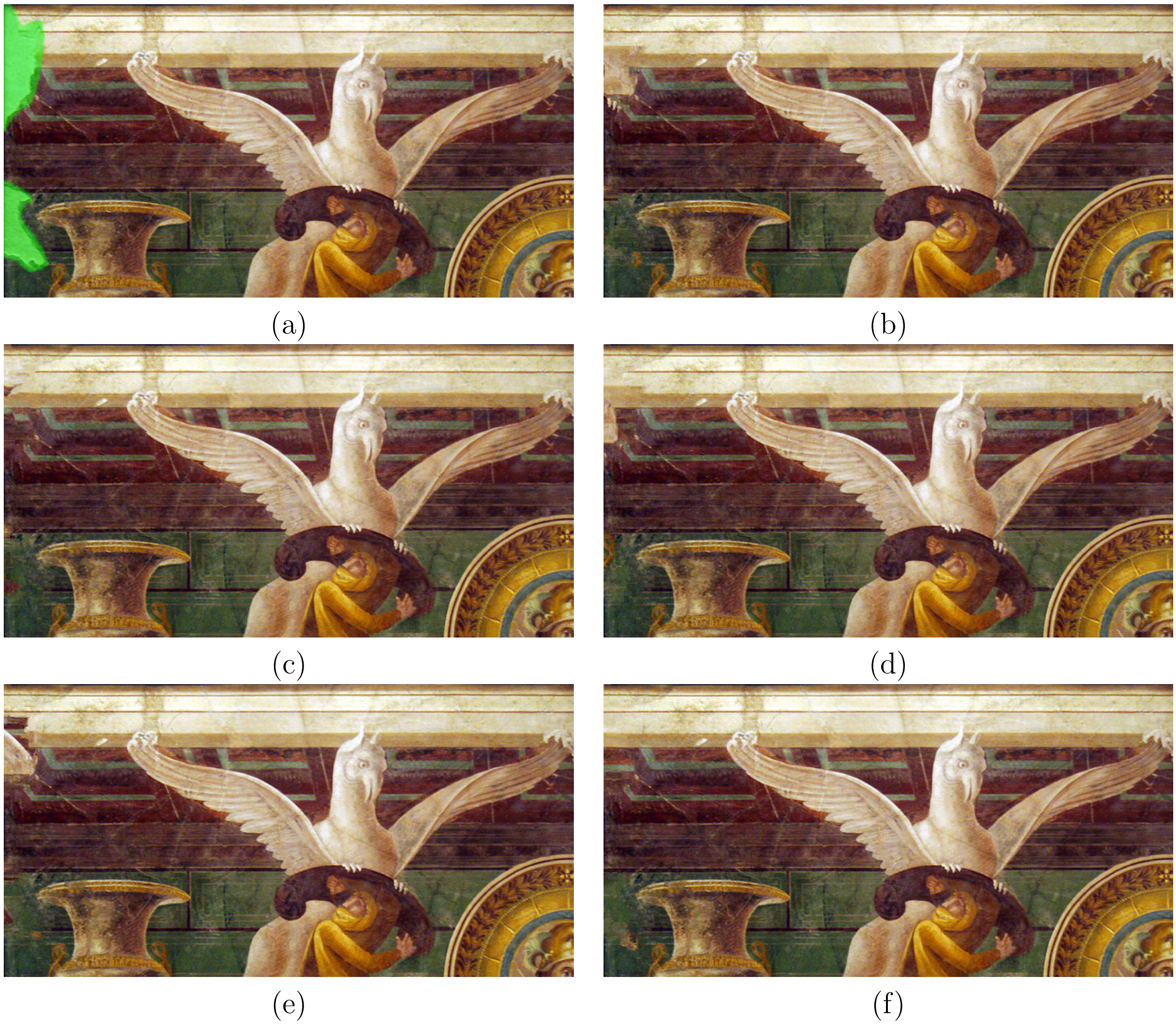
(f)

Figure 3.9. Visual comparison of 'wall' image across all five algorithms: (a) The image to be inpainted; The ouputs using approach of (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18]; and (f) our proposal.



(a) (b) (c) (d) (e) (f)

Figure 3.10. Visual comparison of 'angle' image across all five algorithms: (a) The image to be inpainted; The ouputs using approach of (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18]; and (f) our proposal.



(a) (b) (c) (d) (e) (f)

Figure 3.11. Visual comparison of ‘affresco’ image across all five algorithms: (a) The image to be inpainted; The ouputs using approach of (b) Criminisi *et al.* [17]; (c) Wu and Ruan [20]; (d) Zhang and Lin [21]; (e) Cheng *et al.* [18]; and (f) our proposal.

3.2 A hybrid approach for high-quality fast image completion

In previous section, a greedy strategy has been introduced in conjunction with a hierarchical representation for image completion. This strategy is applied not only for the lowest resolution but also for higher resolution. The greedy strategies generally have computation time better than the global optimization strategies. In addition, they take into account the human vision features because the priority is designed based on the salient structures which are sensitive to human vision system. However, they suffer from the common problems of greedy algorithms, such as, *local optimization*, *patch selection*, *patch search*, etc. The global optimization strategies often have better results but they are more computationally expensive than the greedy ones because the time complexity increases linearly with the number of labels and the number of unknown pixels. In practice, image completion is often applied with user interaction and needs quick feedback.

In this section, we would like to propose a novel improvement of previous framework for high-quality and fast image completion by combining a greedy strategy and a global optimization strategy based on a pyramidal representation of the image. There are some reasons when applying the greedy strategy for lower resolution image.

- Firstly, the human visual system is sensitive to salient structures that are stable and persistent through different scales. In other words, the most relevant structural information of an image remains visible and attractive at different levels of resolution. Therefore, this observation is exploited when performing image inpainting.
- Secondly, at low resolution the inpainting would be less sensitive to local singularities and noise effect.
- Finally, it is much less computationally demanding than when processing the original full resolution image and it copes with the problem of large regions inpainting.

The second operation consists of restoring the damaged region on the original high resolution image by exploiting spatial information contained in the lowest resolution image provided at the first step. At higher resolution, the inpainting problem is modelled as an optimal graph labeling where an offset-map represents the selected label for each unknown pixel. The offset map is initialized from an interpolation of lower resolution and refined for current resolution by optimizing an energy function using multi-label graph cut technique [52, 53]. Because each unknown pixel in the damaged region could originate from any pixel in the source region, the global optimization strategies can be computationally unfeasible. Some existing global optimization methods consider possible assignment fairly but this does not fit the human perception criteria. In term of inpainting quality, fair assignments may lead to unexpected bias for optimization. In terms of speed, a huge label set requires high computational load.

A hybrid approach combining the greedy and global optimization strategies is the best solution to enhance not only inpainting quality but also computational time. A full description of our combination is given in the Algorithm 3.2 [62].

Algorithm 3.2. The improved framework.

```
$G_0 = I$;
$\{G_1, G_2, ..., G_N\} = buildPyramid(G_0)$;
Complete $G_N$ with the scheme in [17] using window-based priority in section 3.1.2.1;
$om_N = generateShiftMap(G_N)$ ;
for detail level $i = N - 1$ downto 0 do
    $om_i = interpolate(SM_{i+1})$ ;
    $om_i = optimize(om_i, G_i)$ ;
end
$G_0 = restore(om_0, G_0)$;
Output $= G_0$ ;
```

The main improvement of this algorithm is using an offset-map, om_i, at higher levels ($i > 0$) instead of using directly the pyramidal images. At the lowest resolution, the image is completed by the same strategy as mentioned in previous section, an exemplar-based strategy. Then, a relationship map between the damaged pixels and source pixels, called offset-map, is generated as an initial guess map for next stage. In this situation, the inpainting problem is considered as an optimal graph labeling where the offset-map represents the selected label for each unknown pixel. This offset-map is then interpolated and refined for higher resolution image gradually to obtain the latest one which is the best solution for inpainting problem at original resolution. Because only a few labels relative to the initial guess are considered based on the optimized map instead of all possible labels, this implementation is much faster than the previous framework in which patch searching is performed on the whole image; furthermore, the quality of inpainted image will be significantly improved by a global optimization algorithm.

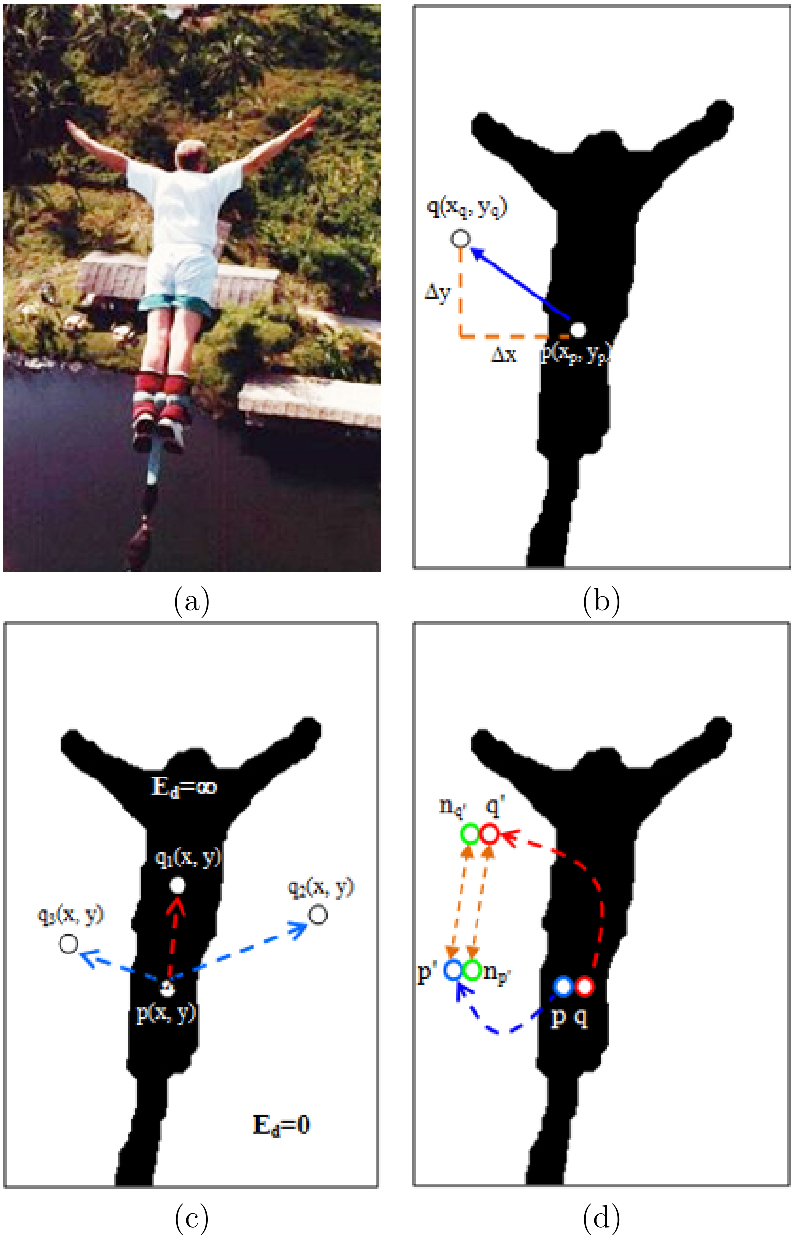


Figure 3.12. The operators in global optimization algorithm. (a) An original image; (b) An offset; (c) Data term; (d) Smoothness term.

3.2.1 Offset operator

The nature of inpainting is filling in the hole by the known pixels. In other words, the intensities of unknown pixels are copied from the known pixels. This formed naturally a relationship between the unknown pixels, $p(x_p, y_p) \in \Omega$, and known pixels, $q(x_q, y_q) \in \Phi$. And this relation is characterized by an offset-map which determines a shift from known pixel to unknown pixel for each coordinate in the image (see Figure 3.12-a). The map can be formulated by equation (3.10):

$$om(p) = \begin{cases} (\triangle x, \triangle y) & p \in \Omega \\ (0, 0) & \text{otherwise} \end{cases} \tag{3.10}$$

Then the output pixel $O(p)$ will be derived from the input pixel $I(p + om(p))$. Because the objective of inpainting is to restore the damaged regions so that it still looks natural, the offset-map must be determined to satisfy this criteria. Authors of [27] proposed a solution to evaluate the offset-map by designing energy function and optimizing it. The energy function is defined as follows:

$$EM = \alpha \sum_{p \in \Omega} E_d(om(p)) + (1 - \alpha) \sum_{(p,q) \in NB} E_s(om(p), om(q)) \tag{3.11}$$

where E_d is the data term related to external constraints and E_s is a smoothness term defined over a set of neighboring pixels, NB. The parameter α is a user defined weighting factor, which is set to $\alpha = 0.5$ in our case to balance the two terms. Thereafter, EM is used to minimize the energy related to both data and smoothness terms. Several optimization approaches have been proposed in the literature. In our method an optimization scheme based on graph cuts technique [52, 53], known for its computational advantages, is used.

3.2.2 Data term

The data term, E_d, linked to external constraints, measures how appropriate is a label or an offset. During the completion process, for each pixel in the target region, an offset is assigned to the pixel in the known regions. This offset is used in the computation of the data term to avoid including pixels from the missing regions. On the other hand, the data term dictates that no pixels in the hole are used in the output image. The detail of data term is given by equation (3.12):

$$E_d(om(p)) = \begin{cases} \infty & (x + \triangle x, y + \triangle y) \in \Omega \\ 0 & \text{otherwise} \end{cases} \tag{3.12}$$

In some cases, the specific pixels in the input image can be forced to appear or disappear in the output image by setting E_d. For example, we can use saliency map to weigh the data term. Therefore, a pixel with a high saliency value will be kept; whereas, a pixel with a low saliency value is removed. Figure 3.12-c illustrates visually how to adapt the value of the data term.

3.2.3 Smoothness term

The second component of the energy function is the smoothness term representing the discontinuity between two neighboring pixels $p(x_p, y_p)$ and $q(x_q, y_q)$. In [42], the authors proposed an effective formula, expressed in equation (3.12), for smoothness term taking into account both color differences and gradient differences between corresponding spatial neighbors in the output and input image to create a coherent stitching.

$$E_s(om(p), om(q)) = \begin{cases} 0 & om(p) = om(q) \\ \beta\delta M(om(p)) + \gamma\delta G(om(p)) & otherwise \end{cases} \tag{3.13}$$

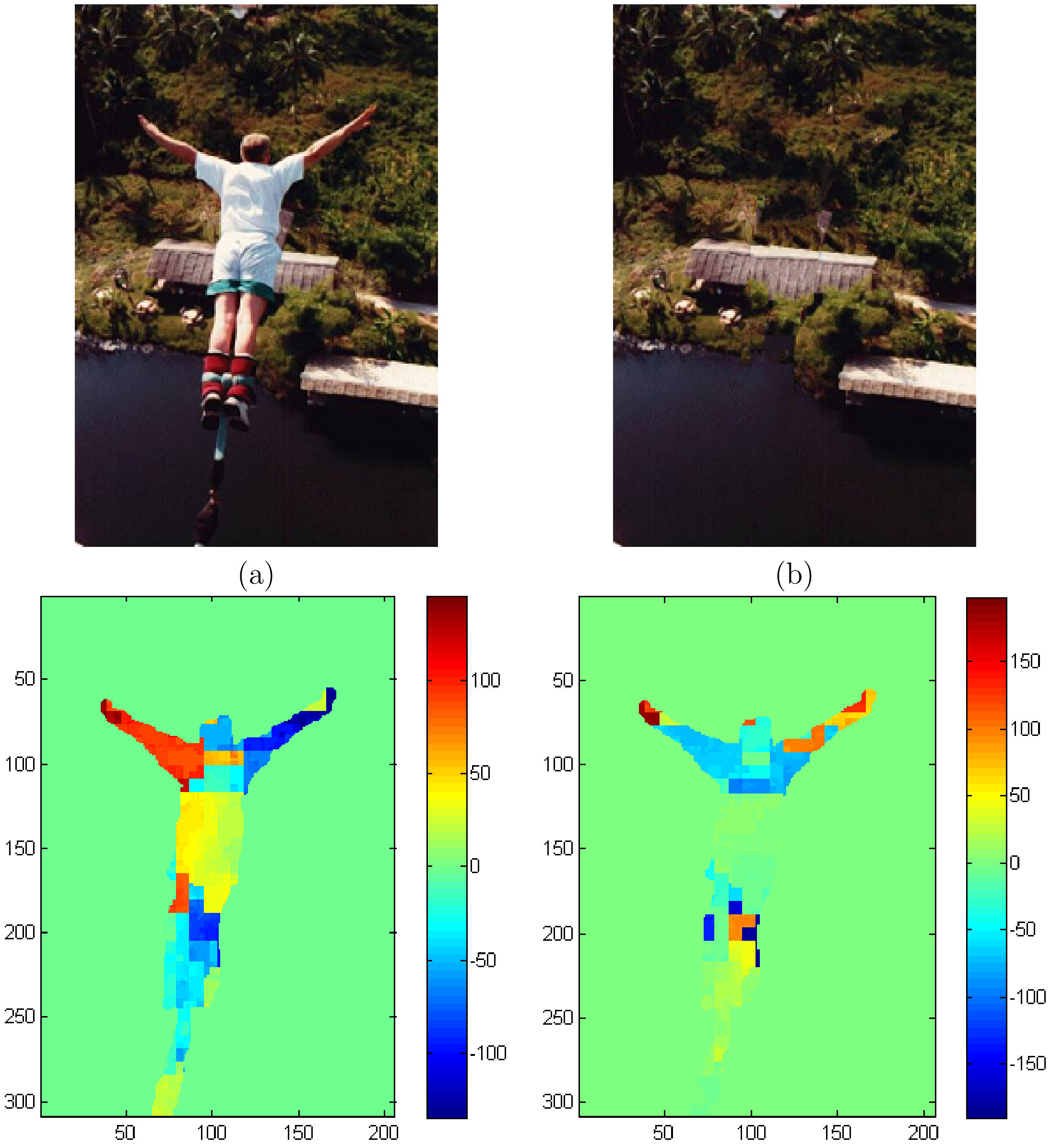
where β and γ are weighting factors balancing these two terms, and are set to $\beta = 1, \gamma = 2$ in our experiment. The terms δM and δG denote the differences of magnitude and gradient, respectively. They are defined as follows:

$$\delta M(om(p)) = ||I(n_{p'}) - I(q')|| + ||I(n_{q'}) - I(p')|| \tag{3.14}$$

$$\delta G(om(p)) = ||\nabla I(n_{p'}) - \nabla I(q')|| + ||\nabla I(n_{q'}) - \nabla I(p')|| \tag{3.15}$$

where I and ∇I are the magnitude and gradient at these locations, $p' = p + om(p)$ and $q' = q + om(q)$ are locations which are used to fill for pixels p and q, respectively, and $n_{p'}$ and $n_{q'}$ are two 4-connected neighbors of p' and q', respectively. Figure 3.12-d depicts an intuitive way for evaluating the smoothness term. The main idea is based on the fact that if a pixel is used for filling, then its neighbors should be also filled as neighbors in the inpainted regions. Moreover, the difference between filled pixels and their neighbors in the target region and known region should be as small as possible.

Figure 3.13 illustrates an example of the offset map at the original resolution image after optimization by the graph-cuts algorithm. Each offset is the 2-D coordinates including horizontal and vertical relationships. The output generated by two corresponding offset maps is shown in Figure 3.13-b.



(a) (b)

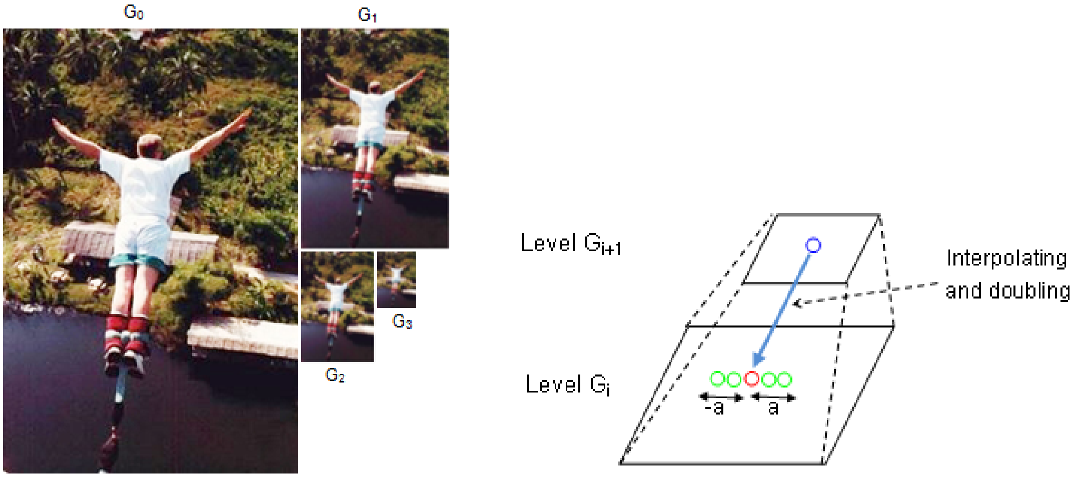
(c) (d)

Figure 3.13. Illustration of the offset values for a commonly used image. (a) Image to be inpainted; (b) A corresponding output; (c) Horizontal offset map; and (d) Vertical offset map.

3.2.4 Offset-map interpolation

A full offset map is first inferred from a completion at the lowest level of the pyramidal representation of the input image. Then, it is interpolated to higher resolutions using the *nearest neighbor algorithm*, and the offset-map values are *upscaled* by simply doubling each value to match the image at higher resolution.

In the higher level, only small shifts relative to the initial guess are examined. This means that only some neighbors of its parent is considered instead of all possible labels. In our implementation, the relative shift for each coordinate varies in range $[-a, a]$, so it takes $(2a + 1)^2$ labels for both direction. It is important to note that the data and smoothness terms are always computed with respect to the actual shifts and not to the labels.



(a) (b)

Figure 3.14. Interpolation of Offset-map. (a) Image gaussian pyramid; (b) Interpolation between two adjacent levels.

Figure 3.14-a illustrates an example of the Gaussian pyramidal decomposition and the associated reconstruction scheme (Figure 3.14-b).

3.2.5 Experimental results

The efficiency of our improvement is illustrated through the experimental results. This section evaluates the results of our approach and compares it with the state-of-the-art methods. A series of natural images is selected for testing. The parameters of the algorithm are kept constant throughout the experiments.

3.2.5.1 Comparative study

Firstly, we would like to evaluate the performance of our proposal based on a comparison with some state-of-the-art methods belonging to the same category of inpainting method. Four methods listed in Table 3.2 are selected for this evaluation. Four input images, commonly used for inpainting evaluation, of size 200 × 200 respectively *yokoya*, *kidstatue*, *cameraman* and *student* are chosen for the experiments, see Table 3.4.

Table 3.4. Selected testing images

Image	**Name**	**Size**	**Hole size**	**Type**
	yokoya	200 × 200	6.53%	natural
	kidstatue	200 × 200	19.86%	natural
	cameraman	200 × 200	12.78%	natural
	student	200 × 200	7.8%	natural

The obtained results, shown in Figure 3.15, demonstrate the efficiency of our method in terms of visual quality compared to the state-of-the-art.

3.2.5.2 Comparison with unoptimized version

Secondly, a comparison with unoptimized version has been performed in order to confirm the quality of our results. Our method is developed based on a previous idea presented in [61] but with the use of a multi-resolution patch matching instead of an offset-map optimization, called here unoptimized version.

Results obtained with the latter are compared to those of the proposed approach (see Figure 3.16). It is clearly shown that our method (Figure 3.16-c) is slightly better than the method proposed in [61]. However, the performance in term of computational complexity of our proposal is rather better than the unoptimized approach. This issue is discussed in more details in the next section.

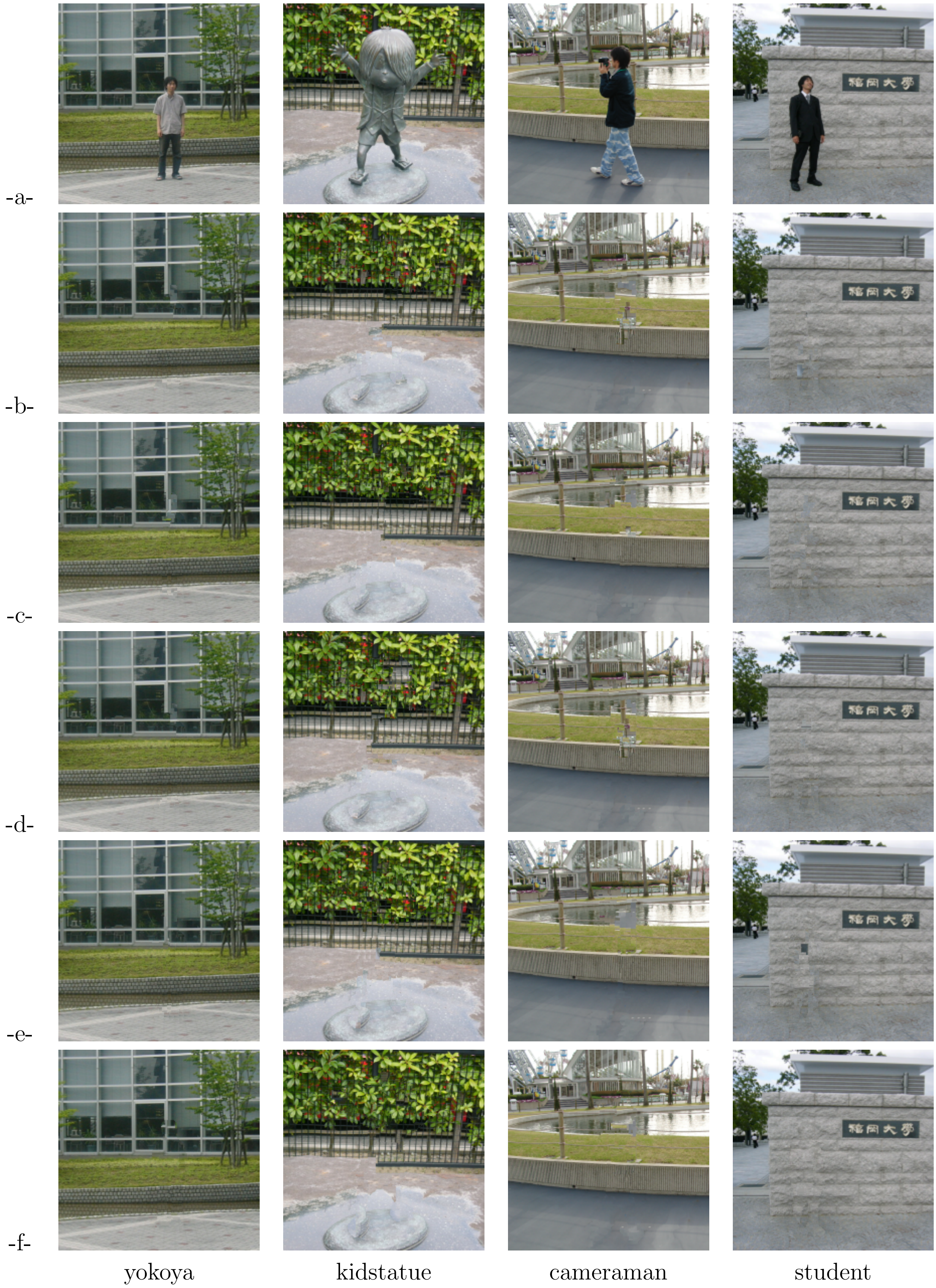


Figure 3.15. Inpainting results for four commonly used images: a) the original images; results when using method of b) Criminisi *et al.* [17]; c) Wu and Ruan [20]; d) Zhang and Lin [21]; e) Cheng *et al.* [18]; and f) Our proposal.

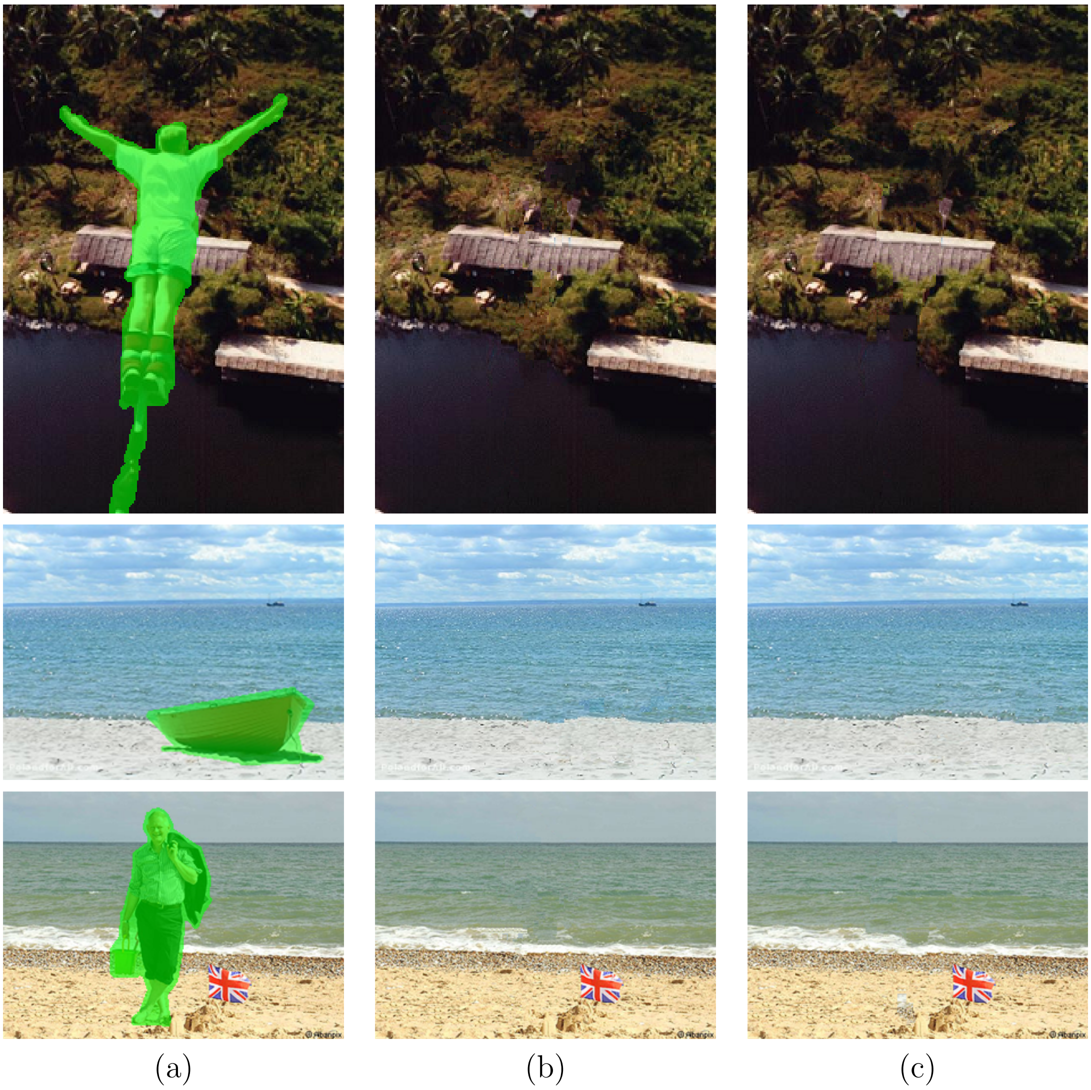


Figure 3.16. A comparison between our proposal and the unoptimized version described in [61]. (a) Image to be inpainted; Inpainting results using (b) unoptimized version and (c) our proposal.

3.2.6 Performance evaluation

The visual quality of the results obtained with the proposed approach has been discussed in the previous sections. In the following we focus on the performance evaluation in terms of computational efficiency. For the sake of completeness and clarity the performance of the method is evaluated at each stage at the global level. The benefits gained at each stage are demonstrated through the obtained results.

3.2.6.1 Local performance evaluation

In the following, the greedy strategy using window-based priority and path-match search procedure is analyzed in order to estimate the contribution of each stage in our algorithm. The global optimization strategy adopted in the proposed method is also analyzed to confirm the outstanding performance in comparison with other approaches.

Table 3.5. Time performance of each stage in percentage (%).

Image ID	**The lowest resolution**, G_3			**Higher resolutions**		
	priority calculation	patch selection	total	G_2	G_1	G_0
yokoya	0.06	0.44	0.60	1.98	6.46	66.50
cameramen	0.41	0.80	1.22	2.85	19.99	43.33
kidstatus	0.20	0.78	0.99	5.02	9.57	57.12
student	0.05	0.44	0.49	5.69	17.45	59.31
bungee	0.38	1.04	1.42	4.65	15.57	55.42
angle	0.03	0.37	0.41	3.17	8.02	66.91
silenus	0.19	0.77	0.96	3.75	7.07	73.48
boat	0.09	0.99	1.09	1.30	19.01	59.82
seaman	0.24	1.43	1.68	4.75	24.68	44.90
AVG	***0.18***	***0.78***	***0.98***	***3.68***	***14.20***	***58.53***

The performance analysis of the proposed method at the different stages, expressed in percentage, is given in Table 3.5 for a set of nine images and illustrated visually in the chart

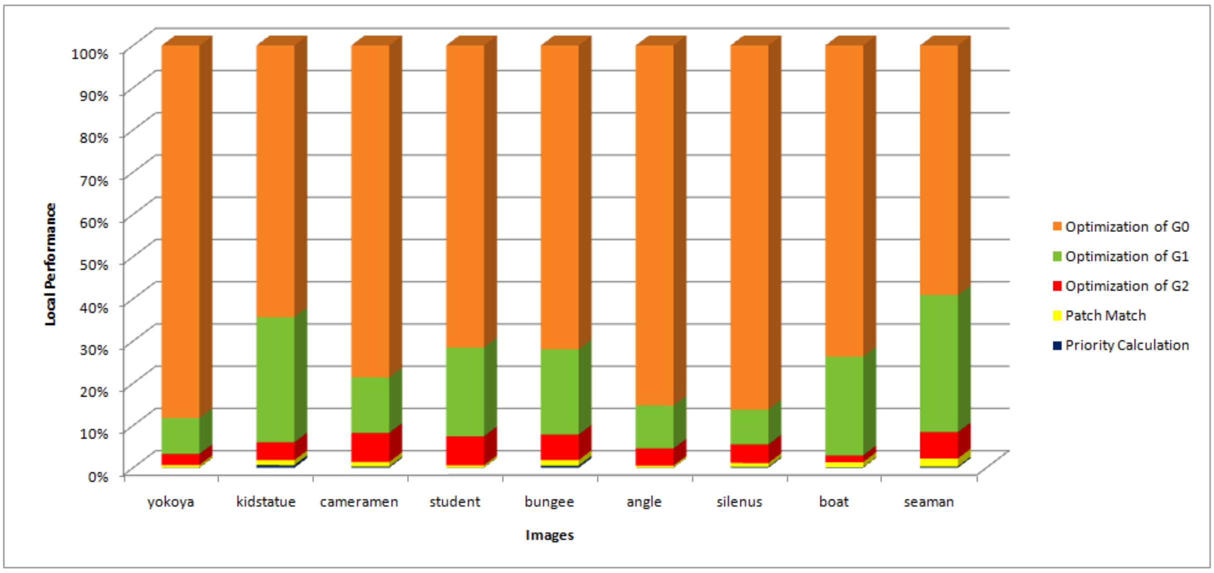


Figure 3.17. Analysis of local performance of our method.

of Figure 3.14. In this case, the level of pyramid is set to 4 (depending on the size of input image). The greedy strategy is applied at the lowest version which has the smallest size. Thus, the time complexity for this strategy is very low. There are two main sub-stages in this strategy: *window-based priority* and *patch match search*. From Figure 3.17, it is clearly shown that these steps account for a very small portion of the whole process (average time about 0.18% for window-based priority, 0.78% for patch-match searching and 1% for entire completion of the lowest resolution).

It is worth noting that the execution time is mainly due to global optimization at higher resolutions. The higher resolution levels are the most time-consuming as shown in Figure 3.17. The average time of global optimization strategy for the levels G_2; G_1; G_0 grows non-linearly and corresponds to 3.68%, 14.2% and 58.53%, respectively. The greedy strategy takes less than 1% of the whole execution time, whereas the global optimization consumes nearly 76.4%. This analysis shows that combining the greedy strategy with the global optimization algorithm not only guarantees the perceptual quality of the inpainted image but also improves the computational performance of the whole process. Furthermore, the use of hierarchical representation also significantly reduces the execution time.

In order to evaluate the visual quality of each stage, two original images in Figure 3.18 are selected as input and the corresponding outputs of each stage in our algorithm are displayed immediately below. The lowest resolution G_3 (see Figure 3.18-b) is completed by a greedy strategy based on some characteristics of the human visual system in order to preserve the perceptual continuity of the structures. This feature is kept and refined for higher resolution in G_2, G_1, and G_0 using a global optimization strategy, namely graph-cut algorithm.

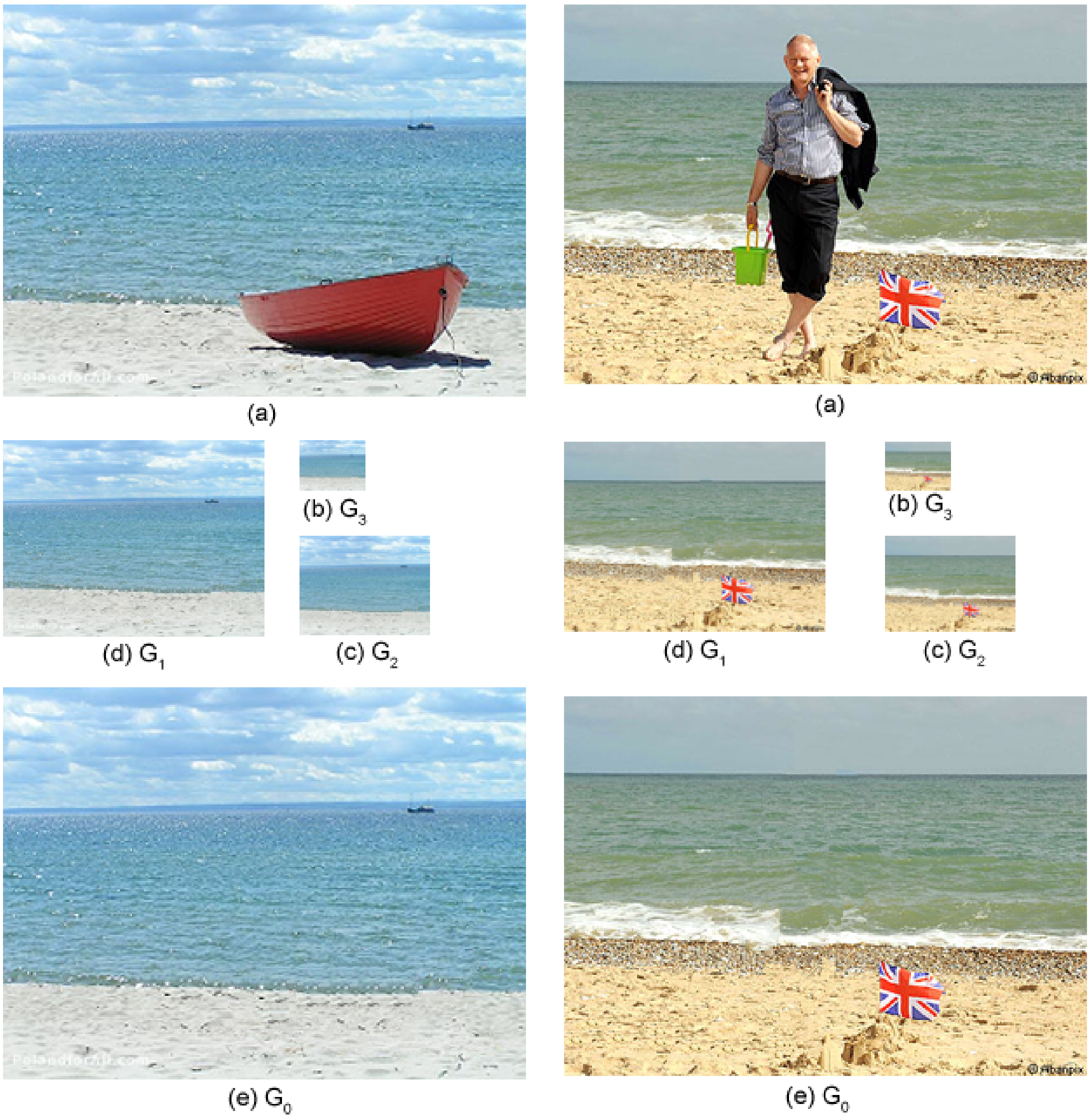


Figure 3.18. An example of outputs for each stage in our algorithm.

3.2.6.2 Global performance evaluation

To increase the reliability of our framework, we analyzed the global performance of the method and compare with other approaches. The performance is evaluated and compared with two groups of methods: methods of the second group and methods belonging to greedy and global optimization strategies. For the sake of fair comparison, all source codes of the compared methods have been written in C/C++ language and implemented on the same PC with an Intel Core i5 2.8GHz CPU and 4GB RAM.

Firstly, a number of approaches from the second group are chosen for evaluating the performance of our method as in Table 3.2 using four images of size 200 × 200 pixels, shown in Figure 3.15. The computation time obtained for the considered methods is given in Table 3.6 in seconds. Here, the bold values show the best performance respects to running time.

Table 3.6. Computational time (in seconds) in comparison with the state-of-the-art methods of the second group

Image	*yokoya*	*kidstatue*	*cameraman*	*student*
Hole size	6.53%	19.86%	12.78%	7.8%
Criminisi *et al.* [17]	**3.22**	10.09	5.75	4.74
Wu and Ruan [20]	3.64	10.29	6.05	5.98
Cheng *et al.* [18]	3.58	11.60	6.91	5.64
Zhang and Lin [21]	38.38	251.74	111.84	59.52
Our proposal	3.71	**4.53**	**3.29**	**3.84**

Secondly, the comparison is extended to other inpainting methods corresponding to algorithms proposed by *Criminisi et al.* [17] representing greedy strategy; *Dang et al.* [61] for the unoptimized approach; and *Pritch et al.* [27] for global optimization strategy.

Figure 3.19 illustrates the results obtained with the proposed approach in comparison to the others. Figure 3.19-a gives images to be inpainted where inpainting areas cover respectively 12.6%, 5.83%, 7.74%, 10.73% and 14.87% of the whole image. As it can be seen from these results, the proposed method outperforms the other considered methods. Indeed, our results look more natural and more coherent than those of other approaches. Moreover, the values in Table 3.7 depicting the computational time in seconds for the selected methods show an outstanding performance of our approach in comparison with the others.

Table 3.7. Computational time (in second) in comparison with both greedy and global optimization strategies.

Image	*bungee*	*angle*	*silenus*	*boat*	*seaman*	*pumpkin*
Size	206 × 308	300 × 252	256 × 480	300 × 225	300 × 218	473 × 332
Hole size	12.6%	5.83%	7.74%	10.73%	14.87%	5.1%
Criminisi *et al.* [17]	16.30	8.20	38.29	24.54	27.31	28.98
Dang *et al.* [27]	15.92	16.36	63.18	50.18	55.16	54.57
Pritch *et al.* [61]	35.39	13.24	57.68	21.18	15.50	37.35
Our proposal	**3.32**	**5.81**	**7.53**	**7.25**	**5.97**	**6.31**

3.3 Conclusion

In this chapter, a novel framework of image completion is introduced using multi-resolution image representation. The multi-resolution patch ensures the texture and geometric structure with the fixed size patches instead of using different patches or dynamic size patches. The chosen patches are more consistent and adaptive. A common problem to most exemplar-based methods is the effect of priority. In currently presented version, a window-based priority which classifies patches in a more suitable way and leads to the pleasant results is introduced. In addition, patch selection step is improved by using a combination of color and gradient channel instead of using only color channels as other methods.

Although the current proposal results in acceptable outputs, it still has a low performance because of patch searching. An improvement for computational time performance in our framework is also introduced which combines a greedy strategy and a global opti-

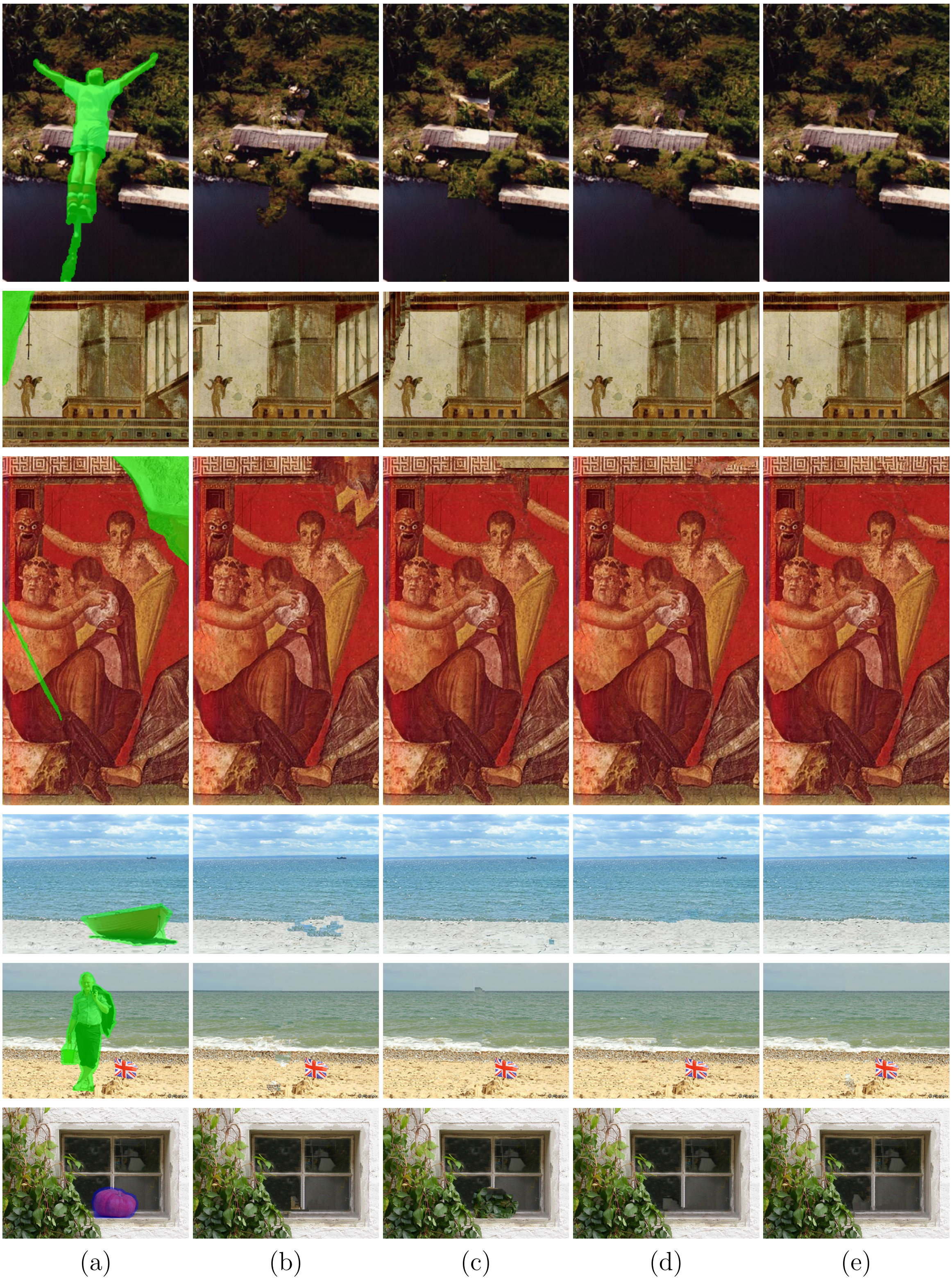


Figure 3.19. Performance evaluation. (a) Image to be inpainted; Outputs when using methods of (b) Criminisi *et al.* [17]; (c) Pritch *et al.* [61]; (d) Dang *et al.* [27]; and (e) our proposal.

mization strategy, called hybrid approach. This approach is not guaranteed to give the global optimum, but the results are very good as can be seen in the examples and the computational time is acceptable for real applications.

Comparison with some new and old methods of the second group is carried out. A series of images including natural and real archaeological images was used for testing. The experimental results show that our approach not only produces better quality output images but also implements much faster than current methods.

For future research, a perceptual patch similarity that is more stable and effective and a numerical estimation for quality of the restored color images will be investigated, and an extension of our framework to use multiple source images will be developed. In addition, a video completion is being studying.

CHAPTER 4

Inpainted Image Quality Assessment

"I have found a very great number of exceedingly beautiful theorems."

Pierre de Fermat

Contents

4.1 What is inpainting quality?

Image inpainting has been become a very active field of research in many real world applications such as digital cinema, computational photography, archaeological image restoration and archival documents restoration [8]. A large amount of research has gone into the development of new inpainting algorithms. However, much less effort has been devoted to the development of objective image quality metrics dedicated to inpainting purpose.

Alternatively, image quality assessment (IQA) plays a prominent role for many applications, including video streaming monitoring, medical imaging and lossy compression control among others. IQA in its broad sense refers to the problem of evaluating the level of perceptual quality of an image. Many interesting methods for predicting the image quality have been proposed in the literature [63–69]. In general, the subjective evaluation is still the most reliable solution, but it could not be used in real-time applications and are tedious and time consuming.

It is worth noticing that the objectives of IQA in the case of image inpainting problem are substantially different from the classical image quality evaluation problem. Indeed, in the case of inpainting problem, the intent is to evaluate the quality of the restored image. It is also worth understanding that inpainting could be considered as a special image restoration problem in its broad sense. In both problems, inpainting and classical image restoration, the existing IQA metrics could not be directly applied because of the specific needs. For example, in the case of image inpainting the recovered region is totally different from the original one.

The main objective of image inpainting aims at restoring the missing parts or replacing some parts of the image in a visually plausible way, i.e. make it difficult for Human Vision System (HVS) to detect that the image has been modified. Hence, the intent of Image Inpainting Quality Assessment (IIQA) metric is then to evaluate the visual quality of the inpainted regions in terms of spatial coherence with the other parts of the image and human attention.

Very often, the results of image inpainting are evaluated subjectively or by using some objective metrics not well adapted to the specificities of their criteria. However, the subjective evaluation experiments are time consuming, complex and unpredictable due to some uncontrolled human factors such as fatigue, visual discomfort, knowledge, etc.

This chapter will discuss methods, listed in Table 4.1, that have been extensively used for the evaluation of inpainting quality, as well as more recently proposed methods having the same goal. An overview of general image quality metrics that could be applied for inpainting quality evaluation is given in more details in the next section.

Table 4.1. The selected IQA and IIQA metrics.

Year	Authors	Name	Type	Reference
-	-	MSE and PSNR	IQA	Full
1993	Daly [64]	VDP	IQA	Full
2004	Wang *et al.* [65]	SSIM	IQA	Full
2006	Bonnier *et al.* [68]	SSIM_{IPT}	IQA	Full
2006	Sheikh *et al.* [66]	VIF	IQA	Full
2007	Chandler and Hemami [69]	VSNR	IQA	Full
2008	Wang *et al.* [70]	PWIIQ	IIQA	Full
2009	Ardis and Singhal [71]	ASVS and DN	IIQA	No
2010	Mahalingam and Cheung [72]	GD_{in} and GD_{out}	IIQA	No
2012	Oncu *et al.* [73]	BorSal and StructBorSal	IIQA	Full

4.2 Overview of image quality assessment

As mentioned earlier, objective metrics for image quality assessment provide a replacement for the resource-intensive perceptual experiments. It was proven that in the field of inpainting, the focus of research has not been put on the development of objective metrics, but on the development and improvement of inpainting methods. Thus, in the remainder of this section a number of objective metrics that have been more extensively used for general image quality evaluation will be discussed. These metrics will be further applied in some situations of inpainting in order to determine whether they can provide similar, or better results than inpainting-specific objective metrics.

4.2.1 MSE and PSNR

Perhaps the simplest and oldest objective image quality measure is the Mean Square Error (MSE) and its extension, the Peak Signal to Noise Ratio (PSNR). Unfortunately, they have been still widely used due to the fact that they are easy to implement thus convenient to use for optimization purposes [63].

Assume that $\mathbf{x} = \{x_i | i = 1, 2, ..., n\}$ is an "*original image*" which has perfect quality, and that $\mathbf{y} = \{y_i | i = 1, 2, ..., n\}$ is a "*distorted image*" whose quality is being evaluated. Where n is the number of image samples (pixels) and x_i and y_i are the intensities of the i-th samples in images $\mathbf{x}$ and $\mathbf{y}$, respectively. Note that this indexing arrangement does not account for the spatial positions of, or relationships between pixels, but rather, orders them as a one-dimensional (1-D) vector.

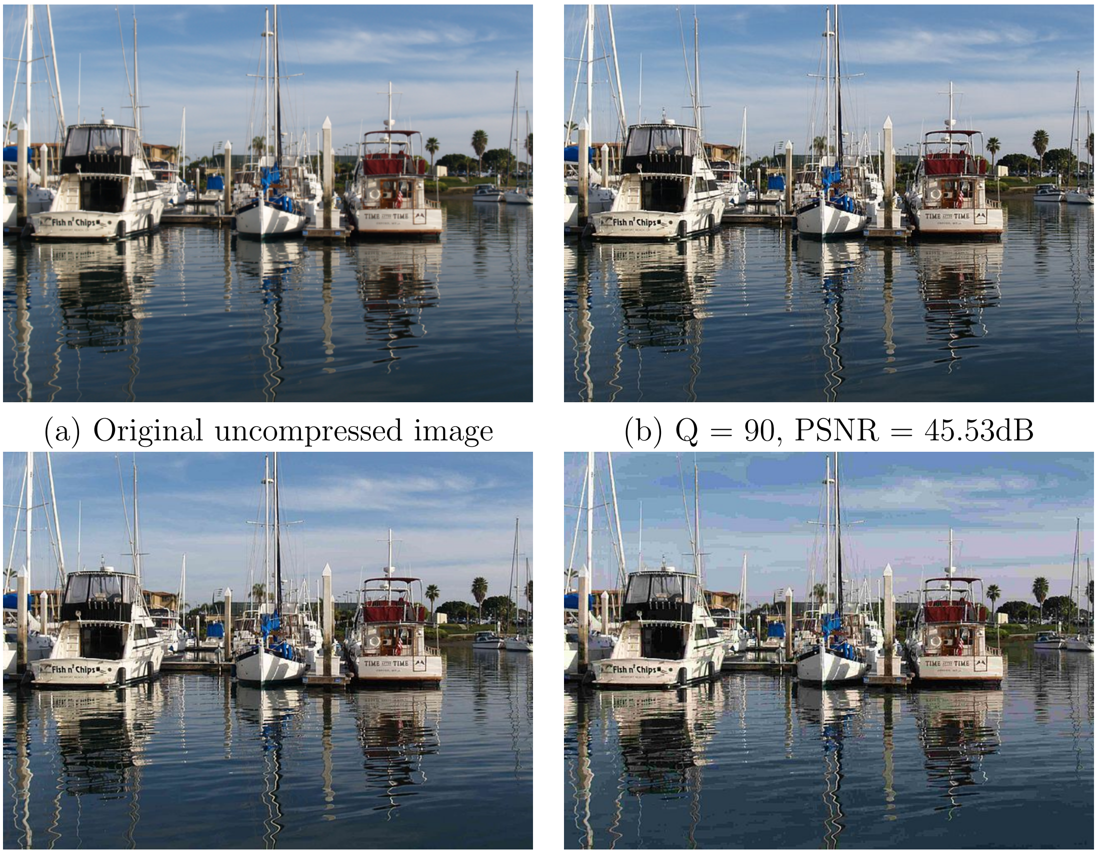
Since the MSE can be defined exactly using this 1-D representation, it is apparent that the MSE does not make use of any positional information in the image, which might be valuable in measuring image quality. Therefore, the MSE is defined as equation (4.1).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \tag{4.1}$$

The PSNR computes the peak signal to noise ratio between two images and gives a value in decibels as a result. The higher the value obtained for PSNR, the better the quality of the reproduction. First step in computing the PSNR is to obtain the MSE value for the two images. Then, the PSNR is calculated as:

$$PSNR = 10log_{10}\frac{L^2}{MSE} \tag{4.2}$$

In equation (4.2), the parameter L is the dynamic range of allowable image pixel intensities. For images that have allocation of 8bits/pixel of gray-level, $L = 255$. Alternately, the MSE and PSNR were computed for color images by converting the images to the YCbCr or HSL color space, which separates the intensity (luma) channel from the color channels. This choice was made because the human eye is more sensitive to intensity information. Taken this into consideration, the MSE and PSNR are calculated on the Y or H channel [63].



(a) Original uncompressed image

(b) Q = 90, PSNR = 45.53dB

(c) Q=30, PSNR = 36.81dB

(d) Q=10, PSNR = 31.45dB

Figure 4.1. An example of PSNR values for jpeg compressed images at various quality levels.

Figure 4.1 lists some PSNR values corresponding the compressed images at various quality levels ($Q = 90, 30, 10$). Although, it has been proven that these two metrics do not correlate well with perceived image quality, these measures are easy to compute and extensively used throughout the literature of image processing, communication, and many other signal processing fields.

4.2.2 VDP

A visible difference predictor (VDP) is an algorithm for the assessment of image fidelity which was proposed by *Scott Daly* in [64], and which aims at quantifying the visual fidelity of distorted images. The VDP model interprets early vision behaviour, from retinal contrast sensitivity to spatial masking and pooling for decision stage. A use of the VDP consists of three main stages such as components for *calibration* of the input image, a *human visual system* (HVS) model, and a method for displaying the HVS predictions of the *visible differences*.

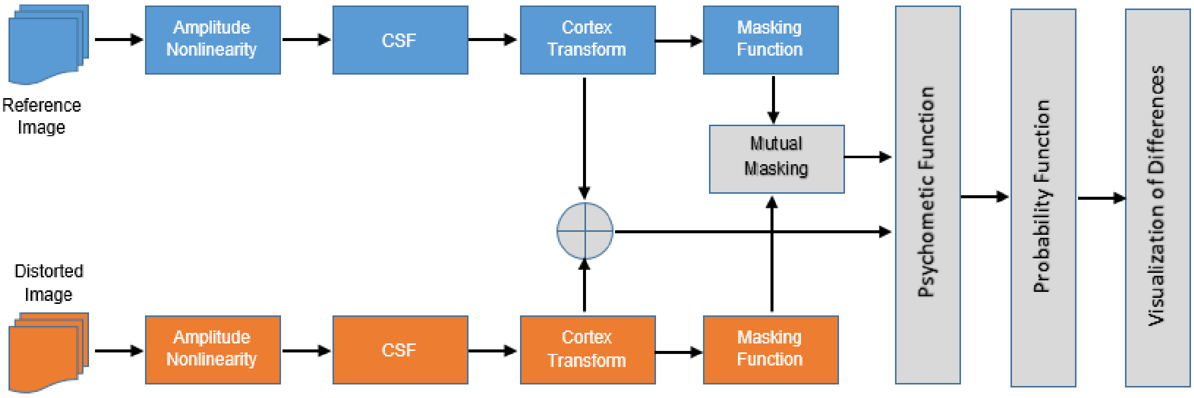


Figure 4.2. Block diagram of the Visible Differences Predictor.

The key element of the VDP is the human visual system model shown in Figure 4.2. It concentrates on the lower-order processing of the visual system such as the optics, retina, lateral geniculate nucleus, and striate cortex. The HVS model consists of a number of processes that limit visual sensitivity. Three main sensitivity variations are accounted for, namely, as a function of light level, spatial frequency, and signal content. Sensitivity, S, is defined as the inverse of the contrast required to produce a threshold response:

$$S = \frac{1}{C_T} \tag{4.3}$$

where C_T is generally referred to as simply the minimum contrast value for which the stimuli starts to be visible. The contrast is defined by:

$$C = \frac{L_{max} - L_{mean}}{L_{mean}} \tag{4.4}$$

where L_{max} and L_{mean} refer to the maximum and mean luminance of the waveform. The variations in sensitivity as a function of light level are simulated by amplitude nonlinearity. Each input luminance L_{ij} is transformed by a simplified version of the retinal response to an “amplitude nonlinearity” value R_{ij} defined as:

$$R_{ij} = \frac{L_{ij}}{L_{ij} + (c_1 L_{ij})^b} \tag{4.5}$$

where the constants $c_1 = 12.6$ and $b = 0.63$ apply when luminance is expressed in cd/m^2. For this model, the adaptation level for an image pixel is solely determined from that pixel.

The variation as a function of spatial frequency is modeled by the contrast sensitivity function (CSF), implemented as a filtering process. A Fast Fourier transform is applied to the value R_{ij}. The resulting magnitudes $f_{uv}(R_{ij})$ are filtered by a CSF which is a function of the image size in degrees and light adaptation level L_m.

The variation in sensitivity due to signal content is referred to as masking. Masking effects are modeled by the detection mechanism, which is the most complicated element of the VDP. It consists of four sub-components: image channel decomposition, spatial masking, psychometric function, and probability summation. During the image channeling stage, the input image is fanned out from one channel to 31 channels or bands. Each channel is associated with one cortex filter which consists of a *radial filter* (*dom*, difference of mesa filter) and an *orientation filter* (*fan* filter). The total number of radial filters is six resulting in five frequency bands and one base band for a typical image of standard size. Each of these bands except for the base band is further fanned out into six channels of different orientation. Thus, five frequency bands times six orientations per bands plus one base band results in 31 channels.

The VDP is a relative metric because it does not describe an absolute metric of image quality but instead address the problem of describing the visibility of differences between two images. As a result of the approach taken, the VDP can be used for all types of image distortions including blur, noise, banding, blocking, etc.; but the reference image is required.

4.2.3 SSIM and SSIM$_{\mathbf{IPT}}$

The SSIM shorting for Structural SIMilarity metric was introduced by *Wang et al.* [65] to overcome the limitation of traditional metrics such as MSE or $PSMR$ while keeping the measure relatively simple in comparison with other fully HVS-based metrics. This is a full reference metric where the measuring of image quality based on an initial uncompressed or distortion-free image as reference. The SSIM is designed to improve on traditional methods like MSE and PSNR which have proven to be inconsistent with human eye perception.

The principal idea underlying the structural similarity approach is that the Human

Visual System (HVS) is highly adapted to extract structural information from visual scenes, and therefore, a measurement of structural similarity (or distortion) should provide a good approximation to perceptual image quality [65]. The SSIM algorithm separates the task of image similarity measurement into three comparisons: luminance, contrast, and structure.

The overall SSIM metric for the whole image can be computed as the mean of local values using sliding window approach. In [63, 65], an 11 × 11 circular-symmetric Gaussian window, $\mathbf{w} = \{w_i | i = 1, 2, ..., n\}$, with standard deviation of 1.5 samples, is used as a weighting function and normalized to unit sum ($\sum_{i=1}^{n} w_i = 1$). And the local comparison is then calculated within the local window.

First, the local luminance of each signal (from image patches taken from the same locations in $\mathbf{x}$ and $\mathbf{y}$, respectively) is estimated by the mean intensity:

$$\mu_x = \sum_{i=1}^{n} w_i x_i \tag{4.6}$$

The local luminance comparison function, $l(x, y)$, is then constructed as a function of μ_x and μ_y :

$$l(x, y) = \frac{2\mu_x \mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{4.7}$$

where the constant C_1 is included to avoid instability for homogeneous or flat regions and set to $C_1 = (K_1 L)^2$. L is the dynamic range of the pixel values ($L = 255$ for 8 bits/pixel gray level) and $K_1 \ll 1$ is a small positive constant.

Second, the local standard deviation (the square root of variance) is used as an estimate of the signal contrast. An unbiased estimate in discrete form is given by:

$$\sigma_x = \left(\sum_{i=1}^{n} w_i (x_i - \mu_x)^2 \right)^{\frac{1}{2}} \tag{4.8}$$

The local contrast comparison, $c(x, y)$, is then designed as a function of σ_x and σ_y :

$$c(x, y) = \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{4.9}$$

where $C_2 = (K_2 L)^2$ and $K_2 \ll 1$ are constants to prevent the possible division by zero.

Third, the signal is normalized by its own mean and standard deviation. The structure comparison $s(x, y)$ is then conducted on the resulting normalized signals. Notice that the correlation between $(x - \mu_x)/\sigma_x$ and $(y - \mu_y)/\sigma_y$ is equivalent to the correlation coefficient between $\mathbf{x}$ and $\mathbf{y}$. Thus, we define the structure comparison function as follows:

$$s(x,y) = \frac{2\sigma_{xy} + C_3}{\sigma_x + \sigma_y + C_3} \tag{4.10}$$

In a discrete implementation, σ_{xy} can be estimated as:

$$\sigma_{xy} = \sum_{i=1}^{n} w_i(x_i - \mu_x)(y_i - \mu_y) \tag{4.11}$$

Finally, three comparisons of equations (4.7), (4.9), (4.10) are combined into the overall SSIM Index between $\mathbf{x}$ and $\mathbf{y}$:

$$S(x,y) = [l(x,y)]^{\alpha}[c(x,y)]^{\beta}[s(x,y)]^{\gamma} \tag{4.12}$$

where α, β, γ are positive parameters that adjust the relative importance of the three components. To simplify the expression and reduce the number of parameters, we set $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$. This results in a specific form of the SSIM index as in [65]:

$$S(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{4.13}$$

By applying such a sliding window approach across the image, an SSIM Index map is obtained. Finally, the SSIM Index map is combined into a single measurement that encapsulates the overall quality of the image:

$$MSSIM(X,Y) = \frac{1}{m}\sum_{j=1}^{m} SSIM(x_j, y_j) \tag{4.14}$$

where X and Y are the reference and the distorted images, respectively; x_j and y_j are the image contents at the j-th local window; and m is the number of local windows of the image. A system diagram of the proposed quality assessment system is shown in Figure 4.3. Signals $\mathbf{x}$ and $\mathbf{y}$ denote local windows for which the similarity measure is computed.

The $SSIM_{IPT}$ metric was introduced by *Bonnier et al.* in [68] and represents the color version of the Structural Similarity Index (SSIM). According to *Bonnier et al.*, the SSIM index works with luminance only. Thus, it is adapted to color images by applying the SSIM index on each channel of the image in color space IPT [74] as follows:

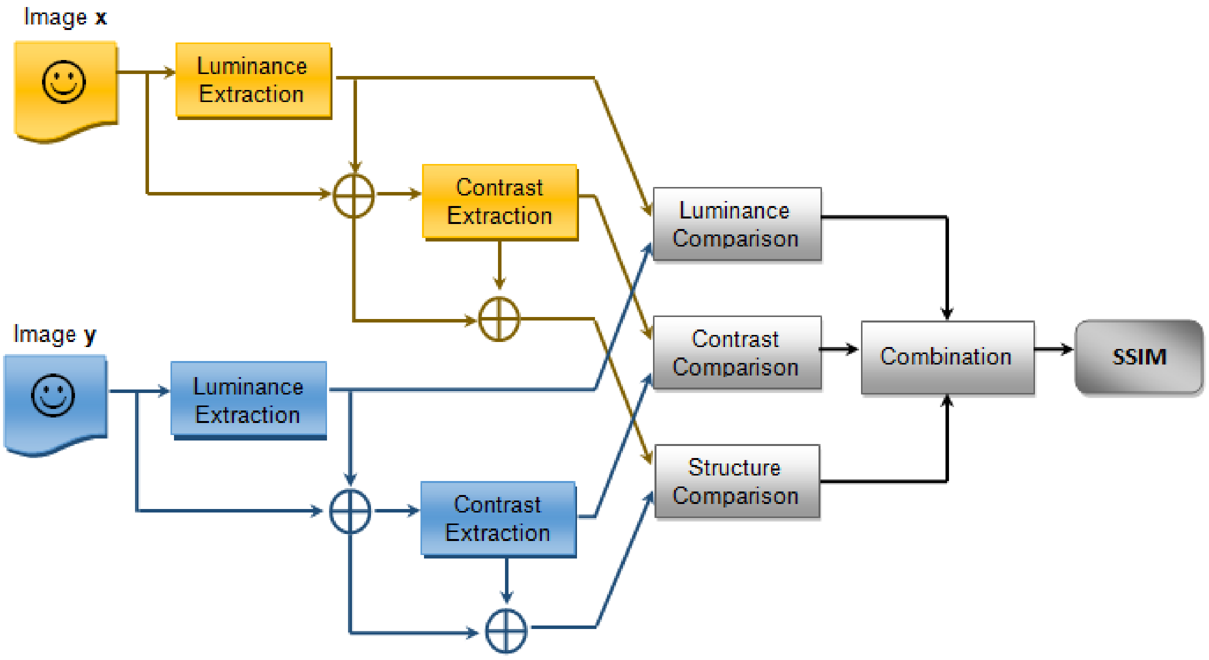


Figure 4.3. SSIM flowchart for a local window

$$SSIM(X,Y)_{IPT} = SSIM(X,Y)_I.SSIM(X,Y)_P.SSIM(X,Y)_T \tag{4.15}$$

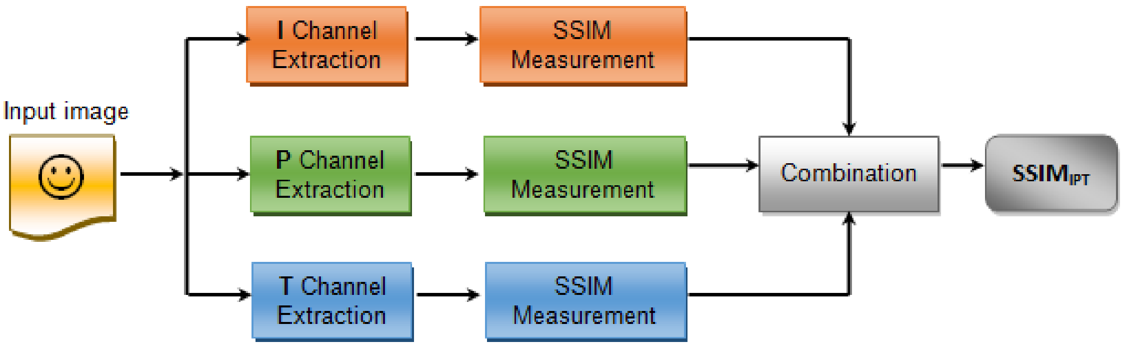


Figure 4.4. Global flowchart for $SSIM_{IPT}$ metric.

4.2.4 VIF

Sheikh et al. [66] have adopted a new paradigm for image quality assessment, the visual information fidelity (VIF) paradigm which is an information theoretic framework based on

Natural Scene Statistics (NSS) modeling in concert with an image degradation model and an HVS model.

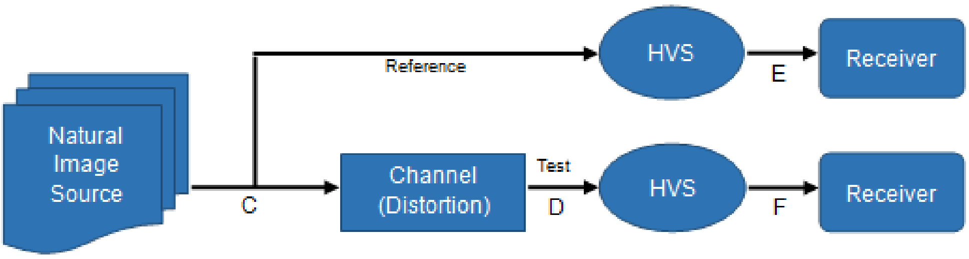


Figure 4.5. Mutual information between $\mathcal{C}$ and $\mathcal{E}$ quantifies the information that the brain could ideally exact from the reference signal, whereas the mutual information between $\mathcal{C}$ and $\mathcal{F}$ quantifies the corresponding information that could be extracted from the distorted signal.

It is known that when images are filtered using oriented band-pass filters (eg. a wavelet transform), the distribution of resulting (marginal) coefficients are highly peaked around zeros and possess heavy tails. Such statistical descriptions of natural scenes are labeled as NSS and it has been an active area of research. The VIF scheme utilizes the Gaussian scale mixture (GSM) model for wavelet NSS [75]. First, it performs a scale-space-orientation wavelet decomposition using steerable pyramid [76] and models each sub-band in the source as:

$$\mathcal{C} = \mathcal{S}.\mathcal{U} \tag{4.16}$$

where $\mathcal{S}$ is a random field of scalars and $\mathcal{U}$ is a Gaussian vector random field. The GSM model has been shown to capture key statistical features of natural images. The distortion model describes how the statistics of an image are distributed by a generic distortion operator. The distortion model chosen for VIF provides important functionality while being mathematically tractable and computationally simple. It is a signal attenuation and additive noise model in the wavelet domain:

$$\mathcal{D} = \mathcal{G}\mathcal{C} + \mathcal{V} \tag{4.17}$$

where G is a deterministic scalar gain field and $\mathcal{V}$ is a stationary additive Gaussian noise random field. This model captures important, and complementary, distortion types: blur, additive noise, and global or local contrast changes.

The VIF then assumes that the distorted and source images pass through the human visual system and the HVS uncertainty is modeled as *visual noise*: $\mathcal{N}$ and $\mathcal{N}'$ for the source and distorted image respectively; where $\mathcal{N}$ and $\mathcal{N}'$ are zero-mean uncorrelated multivariate Gaussians with the same dimensionality:

$$\mathcal{E} = \mathcal{C} + \mathcal{N} \quad \text{(reference image)} \tag{4.18}$$

$$\mathcal{F} = \mathcal{D} + \mathcal{N}' \quad \text{(test image)} \tag{4.19}$$

where $\mathcal{E}$ and $\mathcal{F}$ denote the visual signal at the output of the HVS model from the reference and the test images in one sub-band, respectively from which the brain extracts cognitive information (see Figure 4.3). With the source, distortion, and HVS models as described above, the VIF criterion is then evaluated as follows:

$$VIF = \frac{\sum_{j \in allsubband} I(C^j; F^j | s^j)}{\sum_{j \in allsubband} I(C^j; E^j | s^j)} \tag{4.20}$$

where $I(X;Y|Z)$ is the conditional mutual information between X and Y, conditioned on Z; s^j is a realization of S^j for a particular image and the index j runs through all the sub bands in the decomposed image.

Moreover, the VIF is extended to video quality assessment [77]. The authors justified the use of VIF for video by first motivating the GSM model for the spatio-temporal natural scene statistics. The model for video VIF is then essentially the same as that for VIF with the exception being the application of the model to the spatio-temporal domain as opposed to the spatial domain.

4.2.5 VSNR

The VSNR denotes a wavelet-based Visual Signal-to-Noise Ratio [69]. The VSNR aims to evaluate the effect of *supra-threshold* distortion by using parameters for the HVS model derived from experiments where the stimulus was an actual image as against sinusoidal gratings or Gabor patches. The metric takes into consideration low level as well as mid-level features of the HVS. The low level properties such as contrast sensitivity and visual masking are employed in the wavelet domain and compared to the detection threshold.

The VSNR first computes a difference image from the reference and distorted images. This difference image is then subjected to a discrete wavelet transform. Within each subband, the VSNR then computes the visibility of distortions, by comparing the contrast of the distortion to the detection threshold and then computes the root-mean-squared (RMS) contrast of the error signal (d_{pc}):

$$d_{pc} = \frac{1}{\mu_{L(I)}} \left(\frac{1}{N} \sum_{i=1}^{N} [L(E_i + \mu_I) - \mu_{L(E+\mu_I)}]^2 \right)^{\frac{1}{2}} \tag{4.21}$$

where $\mu_I = (1/N) \sum_{i=1}^{N} I_i$ and $\mu_{L(I)} = (1/N) \sum_{i=1}^{N} L(I_i)$ denote the average pixel value and average luminance of I, respectively; and $\mu_{L(E+\mu_I)} = (1/N) \sum_{i=1}^{N} L(E_i + \mu_I)$ denotes the average luminance of the mean-offset distortions $E + \mu_I$.

Finally, using a strategy inspired from what is termed as global precedence in the HVS, VSNR computes a global precedence preserving contrast (d_{gp}).

$$d_{gp} = \left(\sum_{m=1}^{M} [C^*(E_{f_m}) - C(E_{f_m})]^2 \right)^{\frac{1}{2}} \tag{4.22}$$

where $C(E_{f_m})$ denotes the actual contrast of the distortions within the band centered at f_m and $C^*(E_{f_m})$ is global-precedence preserving contrast. For more detailed specifications of these equations, the reader can be referred to [69].

According to the above, the final VSNR metric, in decibels, corresponding to two images is given by a combination of perceived contrast of the distortions d_{pc} and discontinuity of global precedence, d_{gp} as following:

$$VSNR = 10log_{10} \left(\frac{C^2(I)}{\alpha d_{pc} + (1-\alpha)\frac{d_{gp}}{\sqrt{2}}} \right) \tag{4.23}$$

where $C(I) = \sigma_{L(I)}/\mu_{L(I)}$ denotes the root mean square contrast of the original I; and parameter $\alpha \in [0, 1]$ determines the relative contribution of each distance.

4.3 Overview of inpainting quality assessment

Early inpainting algorithms have been specifically designed for filling in small missing or damaged regions in digital images and later in vintage films. Their main objective was, at that time, to reconstruct the image such that the result would appear to be as close to the original (undamaged) image as possible. Consequently, many researchers evaluated the quality of the images inpainted with their newly proposed method by employing simple objective metrics, such as MSE, PSNR, etc. or more complex metrics such as SSIM, VIF, VSNR, etc. [72].

As the domain of digital image inpainting spread, to include applications that require a larger area to be inpainted, the objective of the inpainting process also changed, from trying to achieve high fidelity with the original, not degraded, image, to making the inpainted area as visually pleasing as possible. In other words, the modifications brought to the original image through inpainting should not be discernible by human observers. Given this new objective of inpainting, it becomes obvious that evaluation methods previously used (i.e. MSE, PSNR, SSIM) could no longer provide a good assessment of the inpainting quality. Thus, new inpainting evaluation methods that would take into account the subjective nature of the human visual system were required.

At present, a few interesting works on objective IIQA have been published recently [70–73]. These metrics will be described briefly in the next sub-section and applied to inpainted images in comparison with our proposal in order to evaluate the performance of image inpainting quality.

4.3.1 PWIIQ

Authors of [70] proposed a full-reference assessment for image inpainting quality based on parameter weight. This work was inspired from three observations:

1. *Blur produced during inpainting.*
2. *Non-continuity of edge/margin structure.*
3. *Mottle appears in the monotone texture color region.*

Taking into consideration these factors, a special definition of image evaluation index inspired from the SSIM metric [65], Q, on inpainting result is given by:

$$Q = [L(x,y)]^{\alpha}[C(x,y)]^{\beta}[G(x,y)]^{\gamma} \tag{4.24}$$

where L, C, G stand for variance of image luminance, definition and gradient similarity; and α, β, γ are positive constants to adjust the important degree of each term. This metric has the same form as SSIM index but it is modified to apply for inpainting quality.

First, the luminance function, $L(x,y)$, is derived from [65] and defined as follows:

$$L(x,y)=\frac{\sum_{j=1}^{M} l(x_j,y_j)}{M} \tag{4.25}$$

where M is the number of patches in the original image; and $l(x,y)$ is the local luminance comparison which is defined by equation (4.7).

Second, the definition function, $C(x,y)$, is computed by estimating high-frequency parameter in the FFT result of the two images:

$$C(x,y)=\frac{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}|FFT_y(i,j)|-|FFT_y(0,0)|}{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}|FFT_x(i,j)|-|FFT_x(0,0)|} \tag{4.26}$$

where $FFT(i,j)$ stands for all frequency parameter of the original image, and $FFT(0,0)$ is the average gray value of image, that is the DC component, so the gap of the two is the high-frequency parameter of image. The larger the gap is, the more the high-frequency parameters are. This means that the corresponding image is clearer. Compare the high-frequency parameters of being tested image with original one, if the two are nearer, and then the being tested image is clearer and the image quality is better.

Third, the gradient similarity, $G(x,y)$, is regarded as the similarity between the main structure information in two images and calculated by equation (4.27):

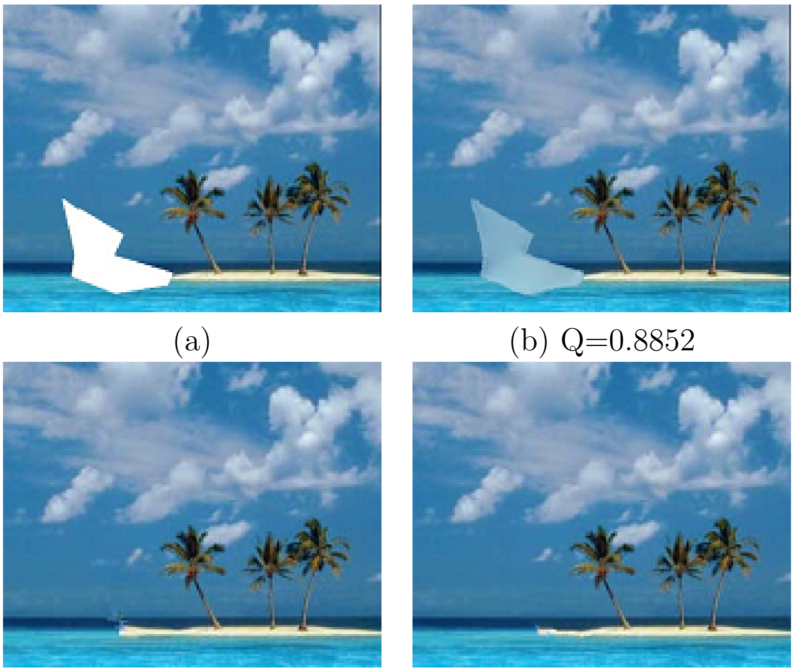
$$G(x,y)=\frac{2\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}G_x(i,j)G_y(i,j)+C_2}{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[G_x(i,j)]^2+\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[G_y(i,j)]^2+C_2} \tag{4.27}$$

where $G_x(i,j)$ and $G_y(i,j)$ are respectively the gradient range of x and y of the original image at the coordinates (i,j). C_2 is a positive number close to zero, which is brought in to avoid the case that denominator equates zero.

This metric is designed based on parameter weight through analyzing the reason for subjective distortion feeling after image inpainting. In practical applications of image inpainting algorithms, mostly there is no standard image as a reference, so how to deal with inpainting evaluation about non-reference image will be still a big challenge. Figure 4.6 shows some values of PWIIQ index in some cases of inpainting.

4.3.2 ASVS and DN

Two types of observable artifacting in an inpainted image, referred to as *in-region* and *out-region*, are defined in [71]. These types of artifacts are evaluated by the variation



(a)

(b) Q=0.8852

(c) Q=0.9657

(d)Q=0.9919

Figure 4.6. An evaluation of PWIIQ index for inpainting algorithms; (a) the original image with the white hole; The results of (b) Total variation model; (c) Criminisi *et al.* [17] and (d) Chen *et al.* [78].

of the saliency map before and after inpainting corresponding to two criteria: average squared visual salience (ASVS) and degree of noticeability (DN). The saliency maps used for computing the metric values are generated by using version 3.1 of the *iLab Neuromorphic Vision Toolkit* (iNVT), at 1:16 discretization of scale-4 (noiseless) expected visual cortex stimulation with 0.1 ms observation cutoff, 4 orientation channels, 3 center scales (2 to 4) and 2 center-surround channels (3, 4). An example of saliency map is shown in Figure 4.7. The lighter region shows more attention.

The first metric, ASVS, is a no-reference measure and it determines if inpainted pixels are relatively noticeable as compared to the remainder of the scene:

$$ASVS = \frac{\sum_{\Omega}[S'(p)]^2}{|\Omega|} \tag{4.28}$$

where $S'(p)$ is the pre-saccadic saliency map value (a non-negative upper-bounded number), as computed by a computational human attention model, that corresponds to post-inpainting pixel p in the set of inpainted pixels Ω. As in the equation (4.28), the

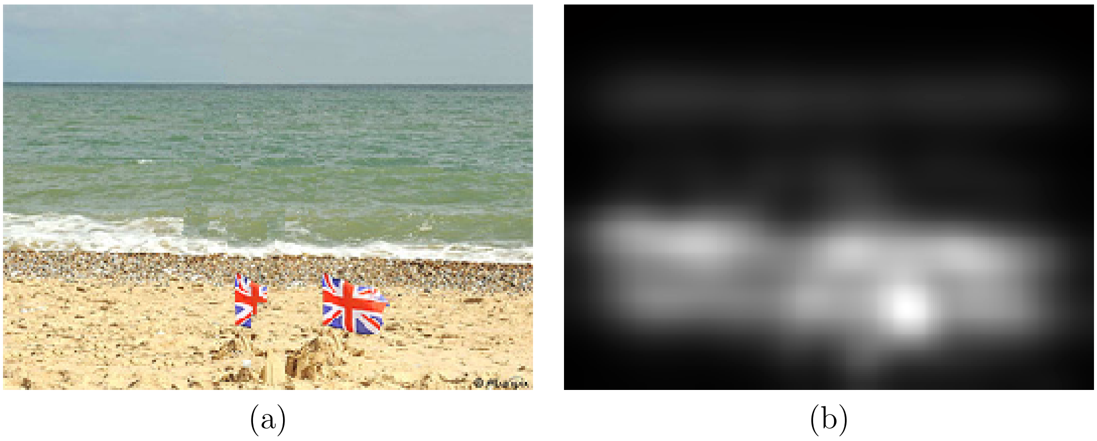


Figure 4.7. An example of saliency map is used for approach in [71]. The saliency map (b) of the original image (a) is generated by iNVT tool.

ASVS metric takes into account only the inpainted area. Thus, this metric considers for *in-region* artifacting which occurs when the pixels in the inpainted region draw more attention than they are suppose to. As noted in [71] the *in-region* artifacting is related to distinct coloration or structure, created after inpainting, that cannot be observed anywhere else in the image. Consequently, this newly introduced color or structure will result in high saliency values, and thus higher ASVS score.

Another class of artifacting specific to inpainting applications is represented by the *out-region* artifacting. As its name suggests, this class encompasses cases where the inpainting process modifies the flow of attention outside the inpainting region. The latter usually happens when an inpainting algorithm cannot successfully continue a locally repeating color or structure, from outside the gap, inside it. The discontinuation thus introduced indirectly draws the attention towards the inpainted area, by decreasing the flow of attention drawn by otherwise salient areas and increasing the attention in the neighbourhood area of the inpainting domain.

In order to account for *out-region* artifacting, *Ardis et al.* [71] proposed another metric, namely the *Degree of Noticeability* (DN), which takes into consideration both classes of artifacting, being calculated as:

$$DN = \frac{|\Omega|}{|\Omega| + |\Phi|}\text{IN-REGION}(I) + \frac{|\Phi|}{|\Omega| + |\Phi|}\text{OUT-REGION}(I) \tag{4.29}$$

where IN-REGION(I) = $ASVS$ and OUT-REGION can be computed by using the following equation:

$$\text{OUT-REGION} = \frac{1}{|\Phi|} \sum_{p \in \Phi} [S'(p) - S(p)]^2 \tag{4.30}$$

where $S'(p)$ and $S(p)$ are the saliency map values corresponding to a pixel p belonging to the inpainting domain before and after inpainting, respectively. As the authors' suggestion, higher scores for $ASVS$ and DN can be interpreted as an indicator of highly visible artifacts and thus a poor inpainting performance.

Although these metrics are correlated well to subjective scores, they needed to be confirmed by further experiments because in the current experiment, only five observers are considered.

4.3.3 GD$_{\text{in}}$ and GD$_{\text{out}}$

Other two saliency-based metrics have been proposed by *Mahalingam* [72], resembling the approach presented earlier by *Ardis et al.* Authors have defined a normalized gaze density measure in the saliency map corresponding to the inpainted and original images and introduced two metrics, GD_{in} and GD_{out}. These metrics show that if there is any change in the saliency maps corresponding to the inpainted and original image, this change is related to the perceptual quality of the inpainting.

By comparing the gaze density within and outside the hole region of images inpainted by both the techniques, the perceptual image inpainting quality can be quantitatively analyzed. The gaze density of an image *inside* and *outside* the hole is given in equation (4.31) as following:

$$GD_{in} = \sum_{p \in \Omega} S'(p) \quad and \quad GD_{out} = \sum_{p \in \Phi} S'(p) \tag{4.31}$$

where $S'(p)$ is the saliency map corresponding to the inpainted image corresponding to pixel p. In order to be able to compare the values provided by these metrics on an equal footing with reference to the unmodified image, the gaze density is normalized as the following equations:

$$\overline{GD}_{in} = \frac{GD_{in}}{GD_{in}^{unmod}} \quad and \quad \overline{GD}_{out} = \frac{GD_{out}}{GD_{out}^{unmod}} \tag{4.32}$$

where GD_{in}^{unmod} and GD_{out}^{unmod} are gaze density measures of original image before inpainting and calculated similarly to equation (4.31) by replacing $S'(p)$ with $S(p)$, where S is the saliency map of the original, unmodified image.

In the original work, the saliency maps were obtained from the eye-tracking experiments [79] based on the gaze density and the gaze pattern is recorded by the faceLAB eye tracker. Unfortunately, this way has the same disadvantages as the subjective evaluation methods. However, since inpainting evaluation methods that are fully automatic are preferred to those that imply the participation of human observers. Thus, for the sake of similarity, in our experiment, the saliency maps were calculated with the SaliencyToolbox in iNVT toolkit [73]. This toolbox is more compact, easier to understand and experiment with.

4.3.4 BorSal and StructBorSal

The metrics introduced in previous section concentrated on considering the variation of saliency regions inside and outside the gap, respectively. *A. I. Oncu et al* in [73] showed that the saliency map values corresponding to a border region around the hole should be able to accurately capture the saliency change of attention flow, without the need of examining the saliency map of the whole image. The novelty here consists of considering only one border region around the gap that contains information from both the inpainting domain and its complementary area. Consequently, one full-reference metric, denoted by Border Saliency (BorSal), has been proposed and computed as a normalized gaze density measure, similar to the GD_{in} and GD_{out} metrics in [79]:

$$BorSal = \frac{\sum_{p \in Border} S'(p)}{\sum_{p \in Border} S(p)} \tag{4.33}$$

where S' and S are the saliency maps of the inpainted and reference image, respectively and p is a pixel in the region corresponding to the Border region. In order to take into account information of both inside and outside the gap, the Border region is extended three pixels inside and outside. In other words, the size of the border region was chosen to be equal to six pixels and was obtained by applying morphological operations (i.e. erosion, dilation) to the inpainting domain.

Using the already defined $BorSal$ metric, a second inpainting quality measure is proposed, by combining the $BorSal$ metric with the $SSIM_{IPT}$ measure introduced by [68]. The later takes into account structural information in the whole image and thus can indicate the presence of artifacting due to poor structure reconstruction, while the *BorSal* metric accounts for changes in the flow of attention, post inpainting. The new metric will be denoted by *StructBorSal* and its value calculated for the entire image by adding up the value obtained according to equation (4.34) to evaluate color images:

$$StructBorSal = BorSal + SSIM_{IPT} \tag{4.34}$$

4.4 Visual coherence metric for inpainting quality

In some cases, the traditional IQA metrics can be extended and applied for inpainting. However, these metrics are suitable only in case of thin or small damaged regions. Furthermore, they are full-reference metrics. It means that it is impractical to apply for inpainting, especially, in case of large inpainting region where the original image (reference image) might not be available or the inpainted region is total different from the original one.

For the mentioned IIQA, most of these metrics are based on the HVS features directly, the saliency map. However, they do ignore an important constraint of inpainting problem related to type of artifacts that have a great influence on the quality prediction. Namely, they do not take into consideration the global visual appearance of the image that significantly affects the restoration quality.

Through the analysis of image inpainting results, we found that two main aspects need to be considered in image inpainting quality assessment. First, the completion is blindly performed without any cue from the original content of images, that is, the restored regions depend on only the rest of the image. Therefore, the new generated pixels should be consistent with the existing ones. This refers to perceptual coherence of inpainted regions that determines the new undesired visual artifacts. Second, the structures or contours often attract more the human gaze than other parts; hence more weight should be associated to these salient features.

In this thesis, a saliency map, representing an approximation of the human attention, is used as the weighting map for the evaluation. As a result, an efficient quality index for IIQA taking into account the mentioned factors is proposed. It is shown that it is possible to predict reasonably the inpainting image quality in many cases.

In the following, some notations used in previous chapters are recalled for completeness. The whole image domain, I, is composed of two disjoint regions: the inpainting region, Ω, and the known region, Φ ($\Phi = I - \Omega$). The basic unit of synthesis at pixel p is a patch, Ψ_p, centered at pixel p. The inpainted image quality index, Q, is defined through equation (4.35).

$$Q = \frac{\sum_{\Omega} C(p)^{\alpha} S(p)^{\beta}}{|\Omega|} \tag{4.35}$$

where $C(p), S(p)$ represent respectively the coherence and structure terms. The two positive parameters α and β are used to adjust the important degree of each part. In our implementation, we set $\alpha = \beta = 1$ for simplicity and reduce the number of parameters. The impact of these parameters will be further discussed in next stage. Figure 4.8 demonstrates a scheme in order to calculate the proposed index.

4.4.1 The coherence term

An inpainted region, Ω, has a global visual coherence with the rest of the image, Φ, if every new generated pixel is consistent with existing ones in terms of a structural measure. On the other hand, a local patch, Ψ_p, should be similar to the one within Φ. The coherence term for each pixel $p(x, y)$ ($p \in \Omega$) is then defined as follows:

$$C(p) = max\{SIM(\Psi_p, \Psi_q), \quad \forall \Psi_q \in \Phi\} \tag{4.36}$$

where Ψ_p and Ψ_q denote small patches around p and q, respectively. SIM is an objective function to evaluate the similarity between two patches that measures a appropriateness degree of an inpainted pixel (p) based on existing pixels (Ψ_q) and its neighbors in Ψ_p.

A good objective function needs to agree perceptually with a human observer. The MSE or $PSNR$ are used widely for patch similarity but they are insufficient to provide the desired results. The main reason for this is that they do not take into account the human visual features.

In [65], a new similarity function based on the structural information of patches was proposed. The structural similarity is defined as follows [80]:

$$SIM(\Psi_p, \Psi_q) = \frac{(2\mu_p\mu_q + C_1)(2\sigma_{pq} + C_2)}{(\mu_p^2 + \mu_q^2 + C_1)(\sigma_p^2 + \sigma_q^2 + C_2)} \tag{4.37}$$

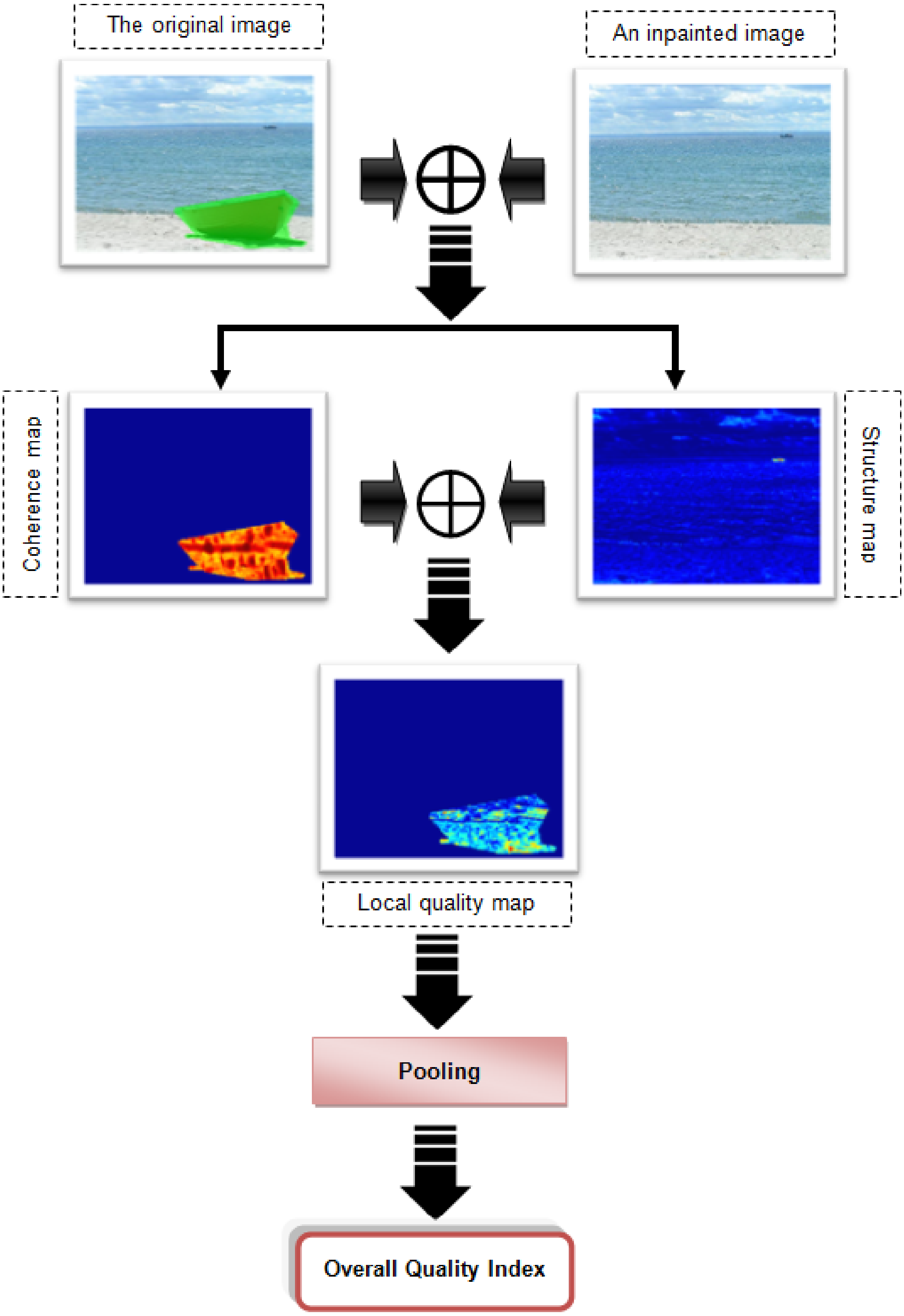


Figure 4.8. An overview scheme of local quality map.

where μ_p, σ_p and μ_q, σ_q denote the mean intensity and standard deviation set of patches Ψ_p and Ψ_q, respectively, while σ_{pq} denotes their cross correlation. C_1 and C_2 are small positive constant values to avoid instability problem when the denominator is close to zero.

This definition is very easy to understand and takes into account the HVS features but it works with luminance only, while most of image in our context includes color information. Although, human eyes are more sensitive to structure than color of an image, color information also could influence human beings judgments and it is important for HVS in quality evaluating. From this observation, we exploit the idea developed in [67] where a similarity combining structure and color information is introduced in order to compute the coherence between patches. The detail of similarity function is then defined as follows [81]:

$$SIM(\Psi_p, \Psi_q) = (1 - \theta)SS(\Psi_p, \Psi_q) + \theta HS(\Psi_p, \Psi_q) \tag{4.38}$$

where θ is a positive constant within the range $[0, 1]$ defining the relative importance between structure similarity (SS) and hue similarity (HS), corresponding to color information. The structure and hue similarity indexes are defined by equations (4.39) and (4.40), respectively:

$$SS(\Psi_p, \Psi_q) = \frac{(\sigma_{pq} + C_1)}{(\sigma_p + \sigma_q + C_1)} \tag{4.39}$$

$$HS(\Psi_p, \Psi_q) = \frac{(2\lambda_p\lambda_q + C_2)}{(\lambda_p^2 + \lambda_q^2 + C_2)} \tag{4.40}$$

where (λ_p, σ_p) and (λ_q, σ_q) denote the mean hue and standard deviation set of patches Ψ_p and Ψ_q, respectively. C_1 and C_2 are small positive constant values to avoid computation instability in very dark or homogeneous regions.

Figure 4.9 illustrates the spatial coherence notion used in the design of the IIQA. Note that the output shown in Figure 4.9-c is more spatially coherent to that of Figure 4.9-b. Indeed, some pixels in inpainted regions (the red patches) do not look coherent with the original patterns. In other terms, there are no similar patches in the source region. The blue dashed patch representing a restored pixel is more reasonable than the red ones because it is similar to a blue solid patch in the known region.

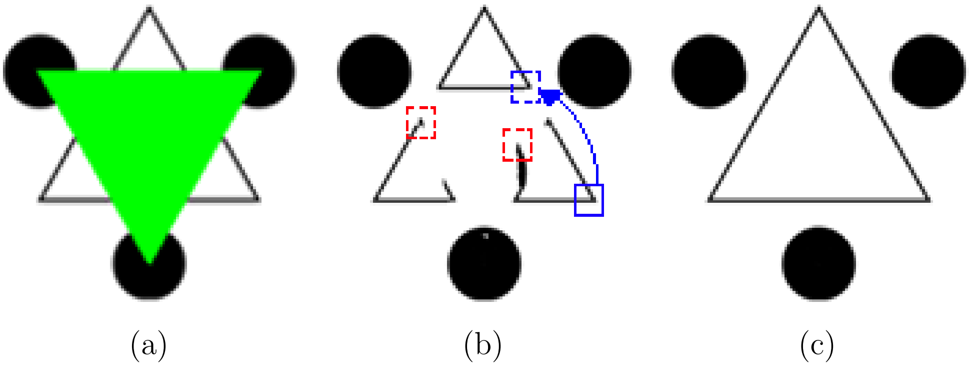

(a) (b) (c)

Figure 4.9. An example of corherence term. (a) An image to be inpainted with green mask; The output using algorithm of (b) Criminisi *et al.* [17] is less coherent than those of Dang's algorithm in (c) [61].

4.4.2 The structure term

The second term affecting image inpainting quality is the structure factor. Given an image, human observer would pay more attention to perceptually relevant regions, which usually correspond to contours and details, but less attention to the rest of the image. Thus, the contours and other relevant structures in the inpainted regions attract more human gaze than the other components. For that reason, we may identify the structure term using the information provided by a saliency map as follows:

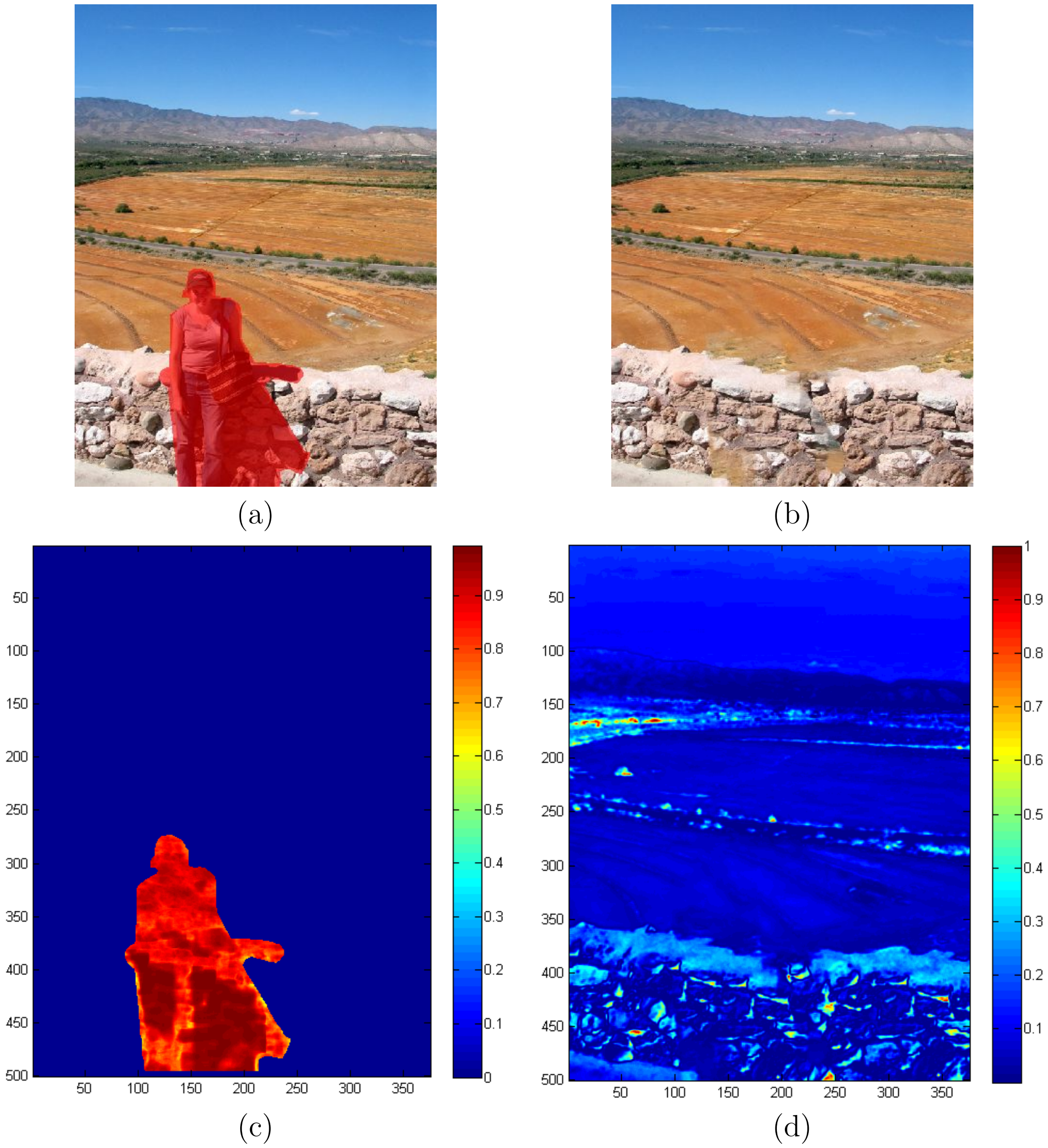
$$S(p) = \frac{SL(p)}{\max_I\{SL\}} \quad \forall p \in \Omega \tag{4.41}$$

where SL is the saliency map of the inpainted image. Several computational models have been proposed to simulate human's visual attention [82–84]. However, the high computational cost and variable parameters are still the weaknesses of these models. Authors of [85] proposed a simple and efficient method based on the idea that objects attracting the gaze of an observer should have characteristics that go beyond the average behavior of the image. A simple formulation of the aforementioned saliency map, SM, can be expressed by equation (4.42):

$$SL = ||I_\mu - I_G|| \tag{4.42}$$

where I_μ and I_G are the arithmetic mean pixel value and the Gaussian blurred version of the original image, respectively. The operation is performed in the $CIEL^*a^*b^*$ color space.

Figure 4.10 illustrates an example of coherence and structure maps by pseudo-colored mask images where the red refers to higher value and the blue refers to lower value.



(a) (b) (c) (d)

Figure 4.10. Local quality map. (a) The original image; (b) An inpainted image using Komodakis' approach [24]; The corresponding (c) coherence and (d) structure maps.

4.5 Psychophysical experiment for subjective quality assessment

Quality assessment dedicated to inpainting is rather a new field and the result of such a technique is often subject to critical evaluation from the end-user including the global homogeneity of the content and the coherence of added/suppressed regions. Therefore, the performance of a given metric, whatever its scheme is, should be assessed with regards to human judgment. However, to the best of our knowledge, there is no exhaustive inpainting database gathering state of-the-art methods and their associated subjective scores for a significant set of images. Based on this observation and for the sake of proposing an objective metric for the evaluation of digital image completion quality, a psychophysical experiment was performed as part of the work for this thesis and with the goal of subjective quality assessment of the inpainting results. Moreover, the subjective evaluation is of great significance for the current work, as the data thus obtained will be used further on for validating objective methods of quality assessment. The latter provide the means to ensure the correlation of the metrics with perceived quality. This section will discuss the experimental setup and the obtained psychophysical results.

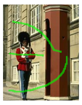
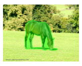
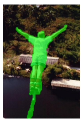
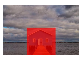
4.5.1 Test setup

The experiment was setup in an uncontrolled environment, as a web-based experiment because of some reasons as follows:

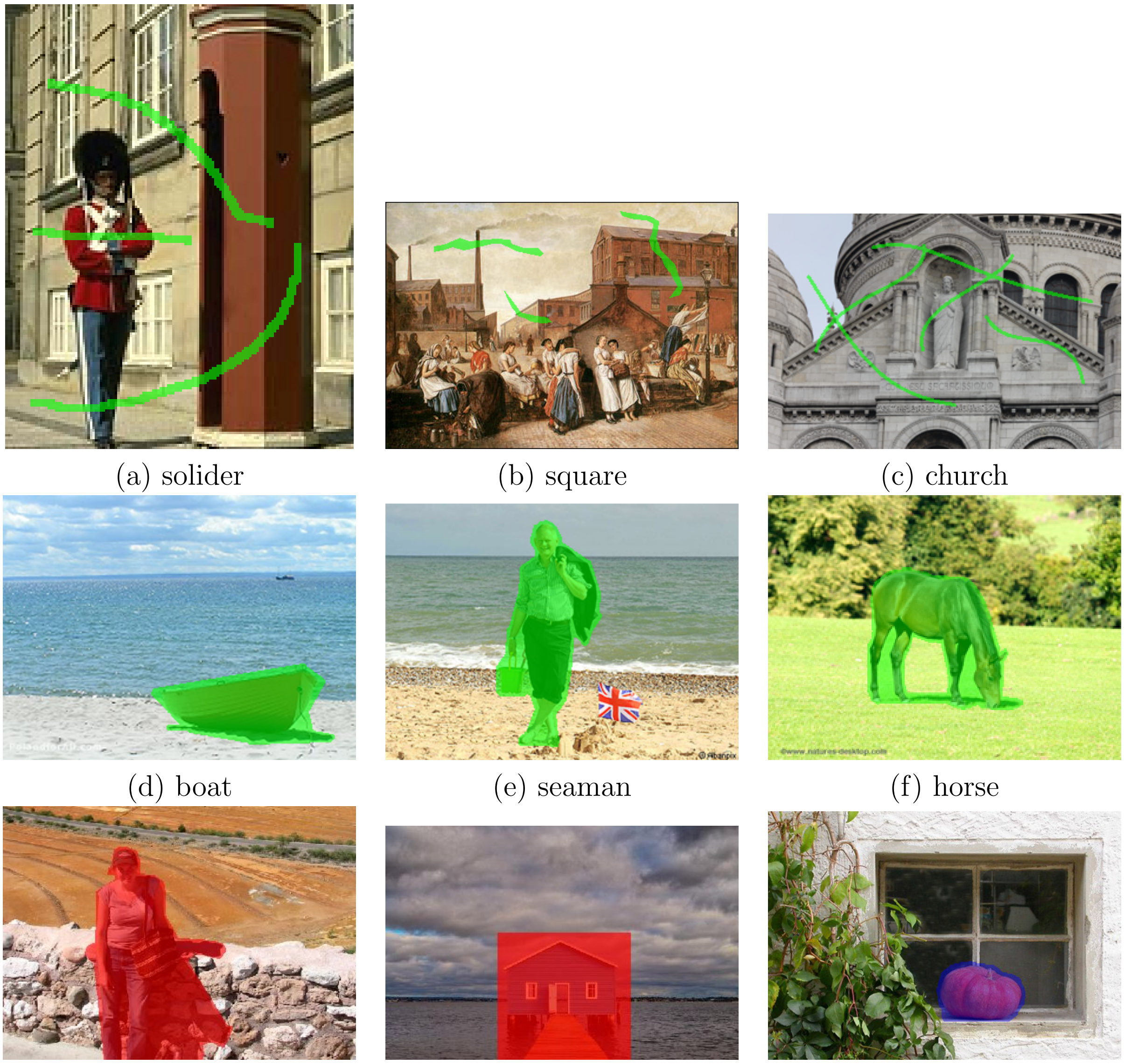
- *It is easy to acquire a large number of participants for achieving high statistical power making it thus possible to draw meaningful conclusions from the experiment.*
- *It is feasible to carry out the experiment around the clock and reach different categories of participants such as experts in image quality, professional restorers, naive observers, etc.*
- *The difference between results obtained from controlled and uncontrolled experiments is negligible. Thus, the scores recording in this experiment are reliable.*

For image database, a set of ten input images were encoded into PNG format in two cases: small and large inpainted regions. In the former case, three images and five different inpainting algorithms are used. For the latter case, seven images are divided into two subcategories: greedy strategy or global optimization strategy. Figure 4.11 shows a snapshot of the used images. The damaged regions are marked by green or red color. The detailed specification of these images is given in Table 4.2. As mentioned in Chapter 1, the hole size is not able to used as the criterion for classifying the damaged regions. In this thesis, all damaged images were classified by the definition (equation 2.1).

Table 4.2. The parameters of images considered for inpainting algorithm evaluation.

Image	Name	Resolution	Size	Hole size	H-type
	soldier	96	142 × 179	4.52%	Small
	square	72	422 × 295	1.98%	Small
	church	72	300 × 200	3.6%	Small
	boat	96	300 × 225	10.73%	Large
	seaman	96	300 × 218	14.87%	Large
	horse	96	300 × 225	14.47%	Large
	bungee	72	206 × 308	12.60%	Large
	lady	72	375 × 500	11.55%	Large
	house	72	400 × 265	21.21%	Large
	pumpkin	96	372 × 332	5.1%	Large

To try and cover all inpainting methods would be infeasible. Thus, this thesis looks at only a selection of algorithms that take different approaches in their shared goal of filling in missing parts of an image. In case of small inpainted regions, five inpainting algorithms have



(a) solider

(b) square

(c) church

(d) boat

(e) seaman

(f) horse

(g) lady

(h) house

(i) pumpkin

Figure 4.11. Sets of images used for the subjective evaluation. Images (a), (b), (c) are used in the small inpainted regions; In the case of the large inpainted region, images are divided into two sub-group such as images (d), (e), (f) for greedy strategy; and images (g), (h), (i) for gloabl optimization strategy.

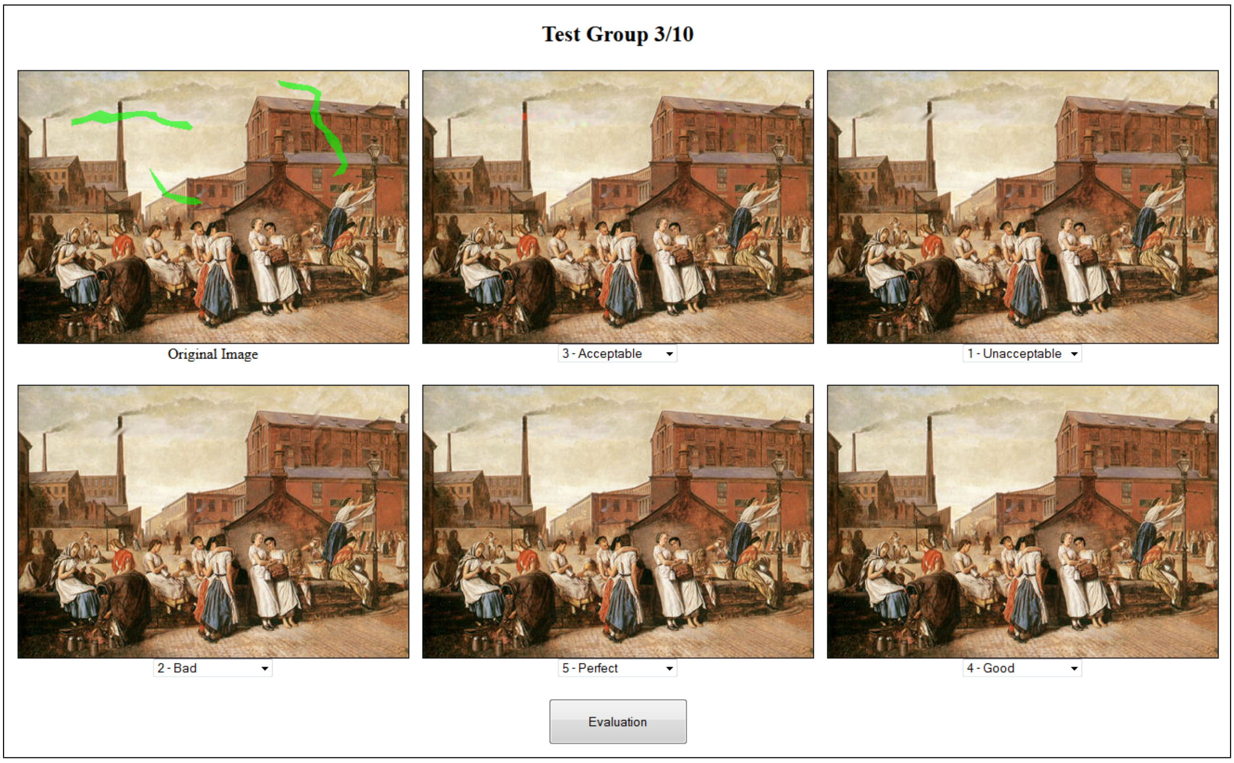
been selected for evaluation. For the large inpainted regions, eight inpaintig algorithms including both greedy and global optimization strategies are applied for restoring. All algorithms were programmed with C/C++ programming language and implemented on the same computer with an Intel Core i5, 2.80 GHz CPU and 4GB RAM. The Table 4.3 lists these algorithms, where the *Year* of publication, *Authors*' name are provide in first two columns; *Type* column indicates the category the algorithm belongs to, depending on the approach it takes; and *H-type*, as mentioned before, shorts for 'Hole type' determining for which type algorithm provides the best results.

Table 4.3. The selected inpainting algirthms.

Year	Authors	Type	H-type
2000	Bertamío *et al.* [29]	PDE	Small
2005	Tschumperlé *et al.* [12]	PDE	Small
2009	Bugeau *et al.* [86]	Hybrid	Small
2006	Wu *et al.* [20]	Exemplar	Large
2004	Criminisi *et al.* [17]	Exemplar	Large
2011	Zhang *et al.* [21]	Exemplar	Large
2012	Dang *et al.* [61]	Exemplar	Large
2013	Dang *et al.* [62]	Exemplar	Large
2007	Komodakis *et al.* [41]	Global optimization	Large
2007	Wexler *et al.* [26]	Global optimization	Large
2009	Pritch *et al.* [27]	Global optimization	Large

A panel of forty-five observers, having a normal acuity and no color blindness, participated to the subjective experiment. The observers were given clear instructions, focused on the overall quality rating of the inpainted images. Accordingly, before commencing the experiment the observers were informed and trained on the system on a set of test images, as follows:

- *One image that corresponds to the original state of a damaged painting, with visible artifacts.*
- *A series of result images corresponding to the restored versions of the same image after repairing the artifacts.*



Test Group 3/10

3 - Acceptable

1 - Unacceptable

2 - Bad

5 - Perfect

4 - Good

Figure 4.12. An example of images presented during the experiment.

Figure 4.12 shows an example of such a test group presented during the experiment. The top-left image shows the degraded image and the remaining images are the restored versions. The instructions also stated that this positioning between the original and restored images will be kept as such, throughout the entire experiment. In order to score for the test images, participants were asked to select a score in the combo box below each restored image. Scores were given on discrete scale ranging from *1 - Unacceptable* to *5 - Perfect* in the combo box and are described as in Table 4.4

Table 4.4. Mean opinion score (MOS).

Score	Quality	Impairment
1	*Unacceptable*	Too many artifacts and very annoying to viewers.
2	*Bad*	Many artifacts are easy to detect.
3	*Acceptable*	Some artifacts but not affecting image interpretation.
4	*Good*	A few artifacts but difficult to detect.
5	*Perfect*	Almost no artifact.

Observers were allowed to change their vote within a same series of images until validation. This has been done to account for the unavailability of a real original image and availability of different versions of results. The inpainted images were randomly presented and shown without including the name of inpainting methods to avoid any bias or influence.

4.5.2 Evaluation methodology

The experimental method for the subjective quality assessment was chosen to be the Mean Opinion Score (MOS) [87]. This method was initially designed for audio quality evaluation, but it has been widely used for image quality evaluation. Historically, and implied by the word Opinion in its name, MOS was a subjective measurement whose values are scored from observers as they perceived it. Participants were presented with a group of inpainted images at a time. Given inpainted images, the participants were asked to judge the overall IQ of the inpainted image using the ITU-R five point quality scale, labelled with the adjectives [87] as previous section. In order to be able to analyse the subjective data obtained, each of the five adjectives in the descriptive quality scale had an equivalent numerical value. Accordingly, *Unacceptable* corresponded to a 1 score and *Perfect* to a 5 score. Once the score values are obtained from observers, the MOS associated Confidence Interval (CI) is then computed thanks to the following equation giving the average opinion of the observers regarding the submitted question.

$$\overline{u}_{jk} = \frac{\sum_i u_{i,j,k}}{S.N} \tag{4.43}$$

where $\overline{u}_{jk}$ is the mean score of the observer i for the inpainting method j and image k; N represents the number of observer; S is the dynamic range of the scores ($S = 5$) in order to normalize the MOS value in range $[0, 1]$.

In order to evaluate as well as possible the reliability of the results, a confidence interval is associated to the MOS. It is commonly adopted that the 95% confidence interval is enough. This interval is designed as:

$$[\overline{u}_{jk} - \delta_{jk}, \overline{u}_{jk} + \delta_{jk}] \tag{4.44}$$

where

$$\delta_{jkr} = 1.95\frac{\sigma_{jk}}{\sqrt{N}} \tag{4.45}$$

The standard deviation, σ_{jk}, is defined as:

$$\sigma_{jk} = \sqrt{\sum_{i=1}^{N} \frac{(\overline{u}_{jk} - u_{ijk})^2}{N-1}} \tag{4.46}$$

The second part of this study refers to the evaluation of objective quality assessment methods. For this set of test images, the scores given by quality metrics are calculated for each image, and compared with the subjective scores given by observers to the same image. Consequently, the performance of the metrics is quantified by the correlation between these two scores. Correlation analysis shows the degree by which the objective values can be predicted, or be explained, by the subjective values. Higher correlation values indicate higher performance of the metrics.

There have been several tools on the correlation analysis in literature. In our research, two standard correlation measures as in the framework of the VQEG group [88] have been used: *Pearson's correlation coefficient* (PCC) and *Spearman rank order correlation* (SCC).

1. *Pearson's correlation coefficient* [89]: Pearsons correlation coefficient is used for data on the interval or ratio scales, and is based on the concept of covariance. The product-moment statistic is given by:

$$PCC = \frac{\sum_i^n (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_i^n (X_i - \overline{X})^2}\sqrt{\sum_i^n (Y_i - \overline{Y})^2} \tag{4.47}$$

where $PCC \in [-1, 1]$, n denote the number of samples in X and Y which are variables associated with subjective and objective results, respectively.

2. *Spearman rank order correlation* [89]: The Spearman rank correlation coefficient (or Spearman's rho), is used with ordinal data and is based on ranked scores. The SCC is the nonparametric analog to PCC is defined as the PCC between the ranked variables. Accordingly, the process for Spearman's correlation first requires ranking the X and Y scores. The analysis is then performed on the ranks of the scores, and not the scores themselves. The paired ranks are then subtracted to get the differences, $d_i = X_i - Y_i$, which are then squared to eliminate the minus sign. If there is a strong relationship between X and Y then paired values should have similar ranks. The test statistic is given by:

$$SCC = 1 - \frac{6\sum_i^n d_i^2}{n(n^2-1)} \tag{4.48}$$

Following the methodology above, the performance of the considered metrics is evaluated over the entire image database.

4.5.3 Psychophysical results

A total of forty-five observers participated in the psychophysical experiment for inpainting quality rating. Before proceeding with the interpretation of the obtained results and data analysis, in ITU-R 500-10 standard [90] a procedure for screening the observers is recommended. The results analysis is able to eliminate from the final calculation either a particular score or an observer. This rejection allows to correct influences induced by the observer's behavior, or bad choice of test images. The most obstructing effect is incoherence of the answers provided by an observer, which characterizes the non-reproductibility of a measurement. These average values are function of two variables: the presentations and the observers. Then, we check if this distribution is normal by using the β_2 test, where β_2 is the kurtosis coefficient, i.e. the ratio between the fourth-order moment and the square of the second-order moment, is given by:

$$\beta_{2jk} = \frac{\frac{1}{N}\sum_{i=1}^{N}(\overline{u}_{jk} - \overline{u}_{ijk})^4}{\left(\frac{1}{N}\sum_{i=1}^{N}(\overline{u}_{jk} - \overline{u}_{ijk})^2\right)^2} \tag{4.49}$$

If β_{2jk} is between 2 and 4, we can consider that the distribution is normal. In order, to compute P_i and Q_i values that will allow to take the final decision regarding the outliers, the observations u_{ijk} for each observer i, each degradation j, and each image k, is compared thanks to a combination of the MOS and the associated standard deviation. The different steps of the algorithm are summarized below, while the reader is referred to [90] for more details:

Algorithm 4.1. Algorithm for outliers rejection.

```
foreach observer $i$ do
    if ($2 \leq \beta_{2jk} \leq 4$) then
        /* Normal distribution                                              */
        if ($u_{ijk} \geq \overline{u}_{jk} + 2\sigma_{jk}$) then
            $P_i = P_i + 1$;
        end
        if ($u_{ijk} \leq \overline{u}_{jk} - 2\sigma_{jk}$) then
            $Q_i = Q_i + 1$;
        end
    end
    if ($u_{ijk} \geq \overline{u}_{jk} + \sqrt{20}\sigma_{jk}$) then
        $P_i = P_i + 1$;
    end
    if ($u_{ijk} \leq \overline{u}_{jk} - \sqrt{20}\sigma_{jk}$) then
        $Q_i = Q_i + 1$;
    end
    /* Finally, we can carry out the following eliminatory test            */
    if ($\frac{P_i+Q_i}{JK} > 0.05$ and $\frac{P_i-Q_i}{P_i+Q_i} < 0.3$ ) then
        Eliminate scores of observer i;
    end
end
```

where J is the total number of degradations and K is the total number of images. Consequently, from a total of 38, results of 11 participants were rejected and only 27 considered for further evaluation. A demographic distribution of the observers is shown in Figure 4.13, while the Figure 4.14 shows the charts of MOS calculated from the raw data.

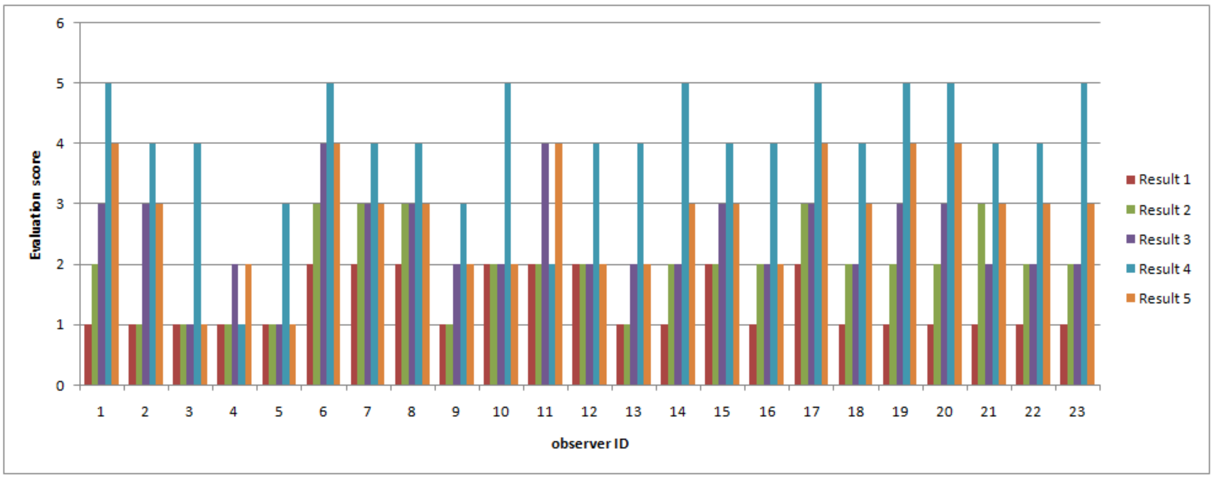
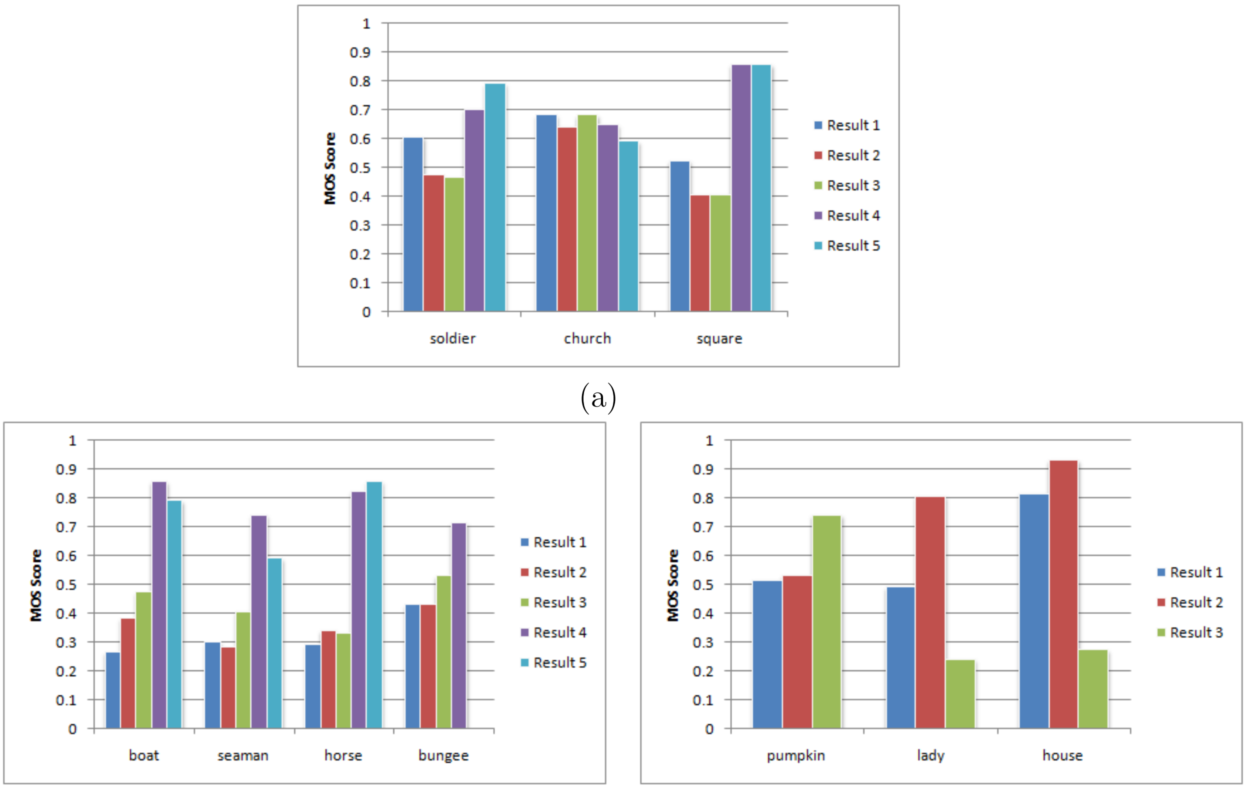


Figure 4.13. A subjective judgment based on the web-based experiment.



(a)

(b)

(c)

Figure 4.14. Mean Opinion Score for (a) small inpainted regions; and the large inpainted regions completed with (b) greedy strategy or (c) global optimization strategy.

From the Figure 4.14, it is clearly possible to conclude about the visual performance of restored results. The MOSs obtained from this study are of a high importance since it will allow to study the performance of the proposed inpainting quality metric thanks to the evaluation of the precision and accuracy of the predicted scores with regards to human judgement.

4.6 Experimental results and discussion

Following the methodology presented in previous section, the performance of a set of quality metrics will be considered in this section. The set includes seven general image quality metrics in section 4.2 and seven inpainting quality metrics in section 4.3 and our proposed metrics in section 4.4.

Based on raw perceptual data, the Mean Opinion Score (MOS) is calculated for each image in the database. For each image and each metric, an objective score will be calculated, describing the quality of the reproduction. Then, the Pearson product-moment (PCC) [89] coefficient and Spearman's rank correlation coefficient (SCC) [89] between MOS and the objective scores are calculated in order to evaluate the performance of the metrics considered.

In order to achieve a fair evaluation of the metrics, the investigation of their performance was carried out in two cases: small inpainted regions and large inpainted regions.

4.6.1 Case 1: Small inpainted regions

Early inpainting algorithms have been tried to simulate the work of the restoration in the museums. Thus, the main function of the inpainting algorithms is to restore the small or thin damaged regions and to make the input image as close to the original image as possible. Many publications have been introduced in this context. In this thesis, three images have been used (see Table 4.2) and restored by five approaches described in [6], [12], [86], [17] and [61]. Fifteen results were then obtained and used to evaluate the objective metrics.

In an attempt to compare with full-reference image quality metrics and existing inpainting quality metrics, the selected images are used as the ground-truth and the small or thin artifacts are added as the pseudo-inpainted regions for restoring. With the original goal of inpainting algorithms to bring the original degraded image to a state that is as close as possible to the state of the final restored image, the inpainting results will be compared with the reference image and evaluated using the metrics in previous sections. Figure 4.15 illustrates one of the obtained results using these algorithms and the pseudo-inpainted regions marking with green color occupy 4.52% of the total image.

Seven image quality metrics and five inpainting quality metrics have been applied and calculated in order to compare with our proposals. The corresponding PCC metrics and SCC metrics are calculated and presented in Table 4.5 and illustrated by Figure 4.16. Where Q_1 and Q_2 are two proposed metrics for gray and color level versions, respectively. The first metric Q_1 uses the coherence term with similarity measurement in equation (4.37), while the second metric Q_2 applies the similarity measurement in equation (4.38).

From these metrics, it could be found that in some cases, the PCC and SCC values of $PWIIQ$ metrics, for example, are higher than the proposed metrics. This is caused by the fact that in these case, the $PWIIQ$ metrics is a full-reference metric which could use both distorted and reference images for evaluation. Moreover, the interpolation-based inpainting methods sometimes work better than exemplar-based inpainting methods in case of small or thin degraded area. Whereas, our metrics are no-reference metrics which cannot work well without valid patches.

However, the obtained results indicate that most of the considered metrics have low PCC as well as low SCC. Thus, it can be concluded that, for the considered image database, they can not accurately predict perceived image quality. It could be noticed that $ASVS$, DN, GD_{in} and GD_{out} metrics determine the presence of inpainting artifacts, that is, they intend to produce lower values for better results. As we can see, our indexes is the most consistent with MOS values and produces the highest mean value for both PCC and SCC measures.

4.6.2 Case 2: Large inpainted regions

The main challenging case of inpainting quality assessment is the one corresponding to large inpainted regions. Because many published methods achieve impressive results in recovering the large damaged regions, there is no successful metric in order to evaluate the obtained results. Thus, it is really meaningful and valuable if it can assess quality of a large restored region. This section will investigate the performance of selected metrics in the case of large inpainted regions.

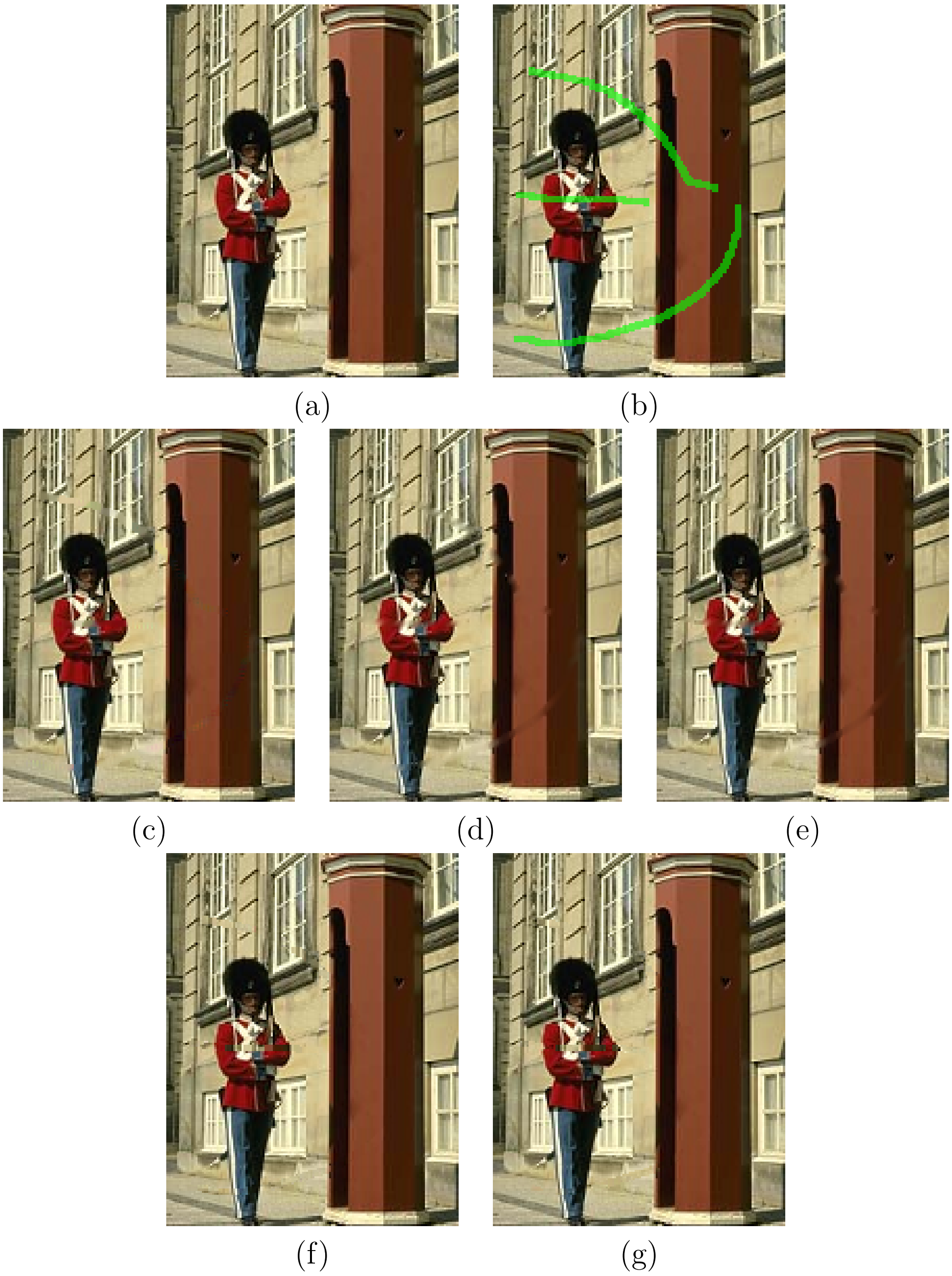
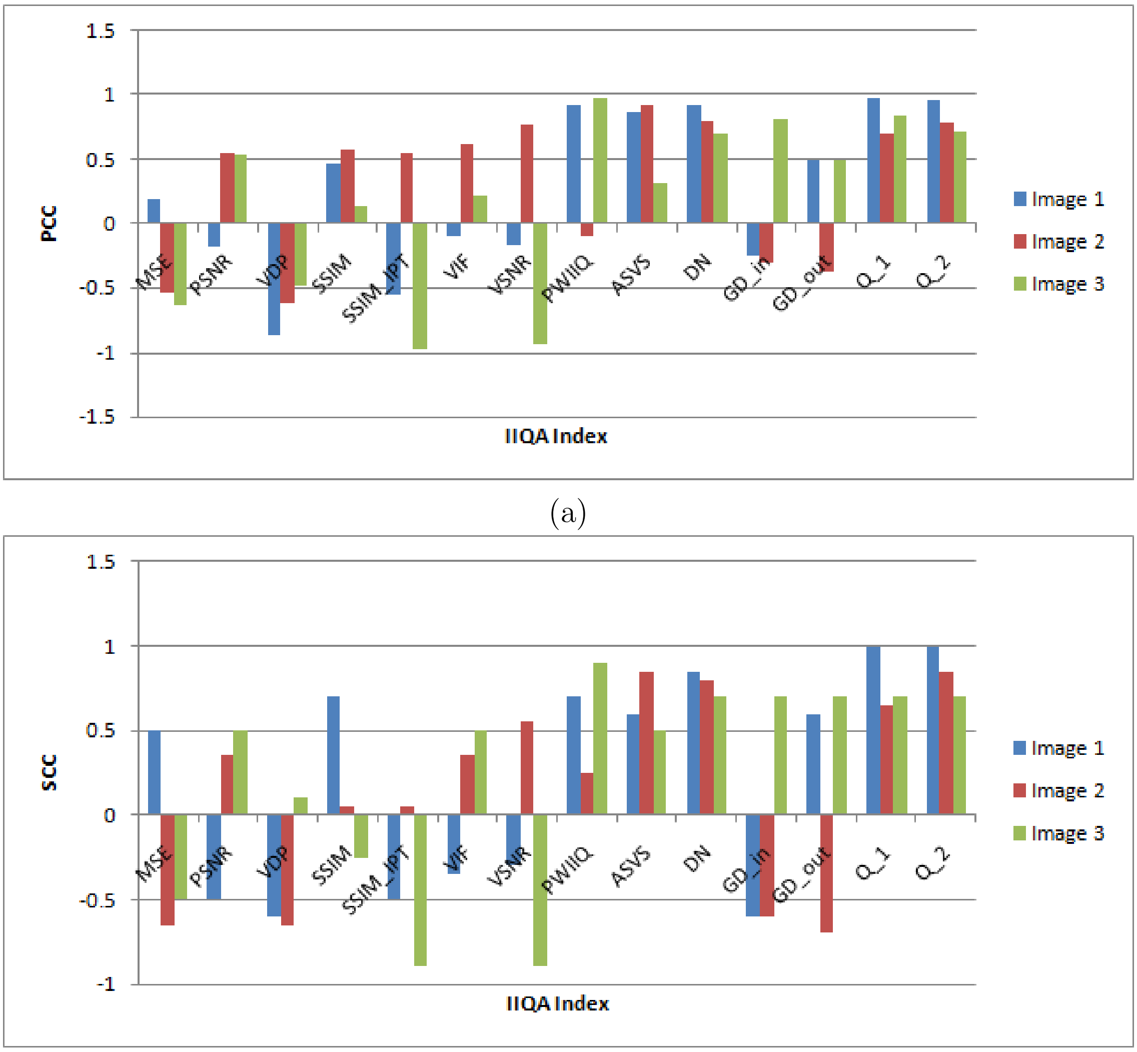


Figure 4.15. The inpainting results (image 2). (a) The original image; (b) A pseudo-inpainted image; The outputs when using the methods of (c) Bertamío *et al.* [6]; (d) Tschumperlé [12]; (e) Bugeau *et al.* [86]; (f) Criminisi *et al.* [17] and (g) Dang *et al.* [61].



(a)

(b)

Figure 4.16. Performance of the quality metrics in case of small inpainted region. (a) Pearson correlations (PCC); (b) Spearman correlations (SCC).

As mentioned in previous sections, we can divide the inpainting methods in this case into two sub-groups such as greedy strategies and global optimization strategies. In our experiments, we selected the methods in [17], [21], [20], [61] and [91] for greedy strategies and [26], [24] and [27] for global optimization strategies.

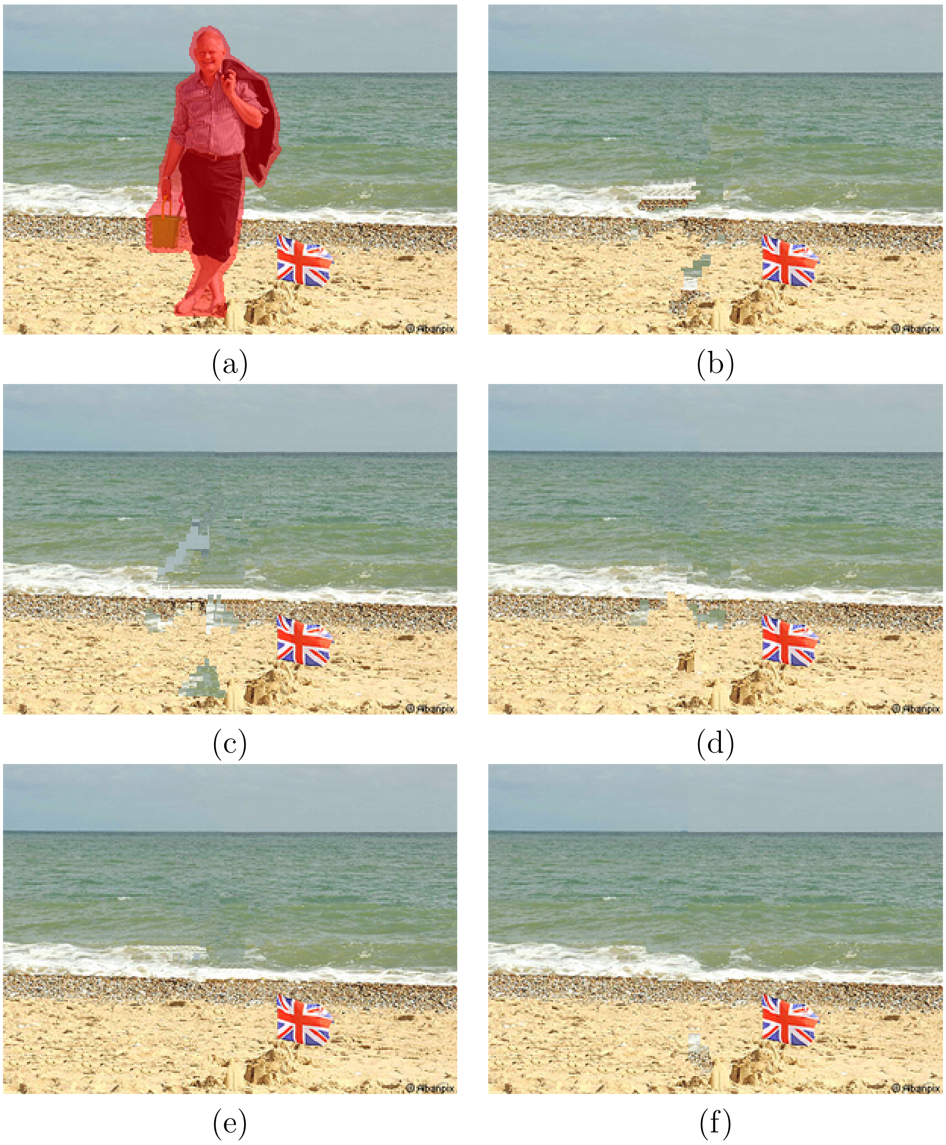
For test images, four images are used as inputs in case of greedy strategy and three images are selected as inputs in the other case (see Table 4.2). Figures 4.17 and 4.18 give an example of restoration in each case. Consequently, twenty restored images are used to evaluate in case of greedy strategy and nine other restored images are used to evaluate in

Table 4.5. The PCC and SCC indexes in case of small inpainting regions.

	PCC			**SCC**		
Image ID	1	2	3	1	2	3
MSE	0.1949	-0.5413	-0.6301	0.50	-0.65	-0.50
$PSNR$	-0.1733	0.5508	0.5385	-0.50	0.35	0.50
VDP	-0.8640	-0.6179	-0.4814	-0.60	-0.65	0.10
$MSSIM$	0.4745	0.5836	0.1360	0.70	0.05	-0.25
$MSSIM_{IPT}$	-0.5557	0.5540	-0.9838	-0.50	0.05	-0.90
VIF	-0.0998	0.6222	0.2167	-0.35	0.35	0.50
$VSNR$	-0.1678	0.7706	-0.9416	-0.30	0.55	-0.90
$PWIIQ$	0.9220	-0.0918	0.9830	0.70	0.25	0.90
$ASVS$	0.8614	0.9235	0.3103	0.60	0.85	0.50
DN	0.9157	0.8046	0.6971	0.85	0.80	0.70
GD_{in}	-0.2531	-0.3058	0.8066	-0.60	-0.60	0.70
GD_{out}	0.4999	-0.3718	0.4900	0.60	-0.70	0.70
Q_1	0.9747	0.7025	0.8355	1.00	0.65	0.70
Q_2	0.9632	0.7884	0.7178	1.00	0.85	0.70

case of global optimization strategy for IIQA evaluation.

Since the inpainting regions are large and the main objective of these methods is to restore the damaged images in a visually plausible way without refering to an original image, the full-reference metrics, such as MSE, $PSNR$, $MSSIM$, VIF, VDP, etc. could not be applied. Therefore, four indexes, $ASVS$ and DN, GD_{in} and GD_{out} are used in this



(a) (b) (c) (d) (e) (f)

Figure 4.17. The inpainting results for greedy strategies. (a). Image to be inpainted; The outputs when using the approaches of (b) Criminisi *et al.* [17]; (c) Wu and Ruan[21]; (d) Zhang and Lin [20]; (e) Dang *et al.* [61] and (f) Dang *et al.* [91].

case. In the original work, the saliency maps used to calculate for the two metrics GD_{in} and GD_{out} were recorded using the Seeing Machines faceLAB eye tracker. However, since inpainting evaluation methods that are fully automatic are preferred to those that imply the participation of human observers. Thus, for the sake of similarity, in our experiment, the saliency maps were computed with the SaliencyToolbox in iNVT toolkit [73]. Similarly, two other metrics, $ASVS$ and DN, were also computed using the saliency maps obtained

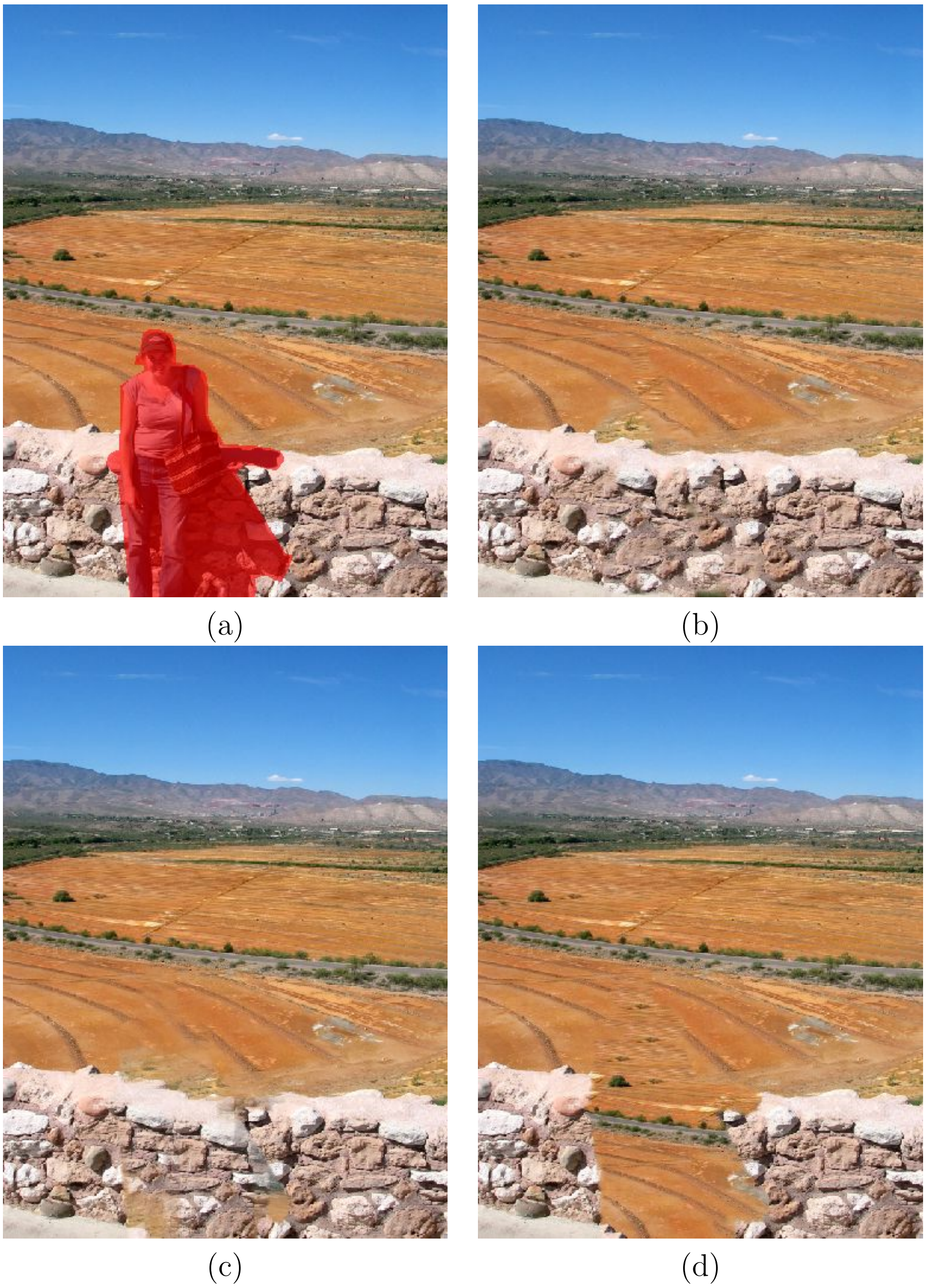


Figure 4.18. The inpainting results for global optimization strategies. (a) Image to be inpainted; The outputs when using the approaches of (b) Wexler *et al.* [26]; (c) Komodakis [24] and (d) Pritch *et al.* [27].

Table 4.6. The PCC and SCC indexes for greedy strategies.

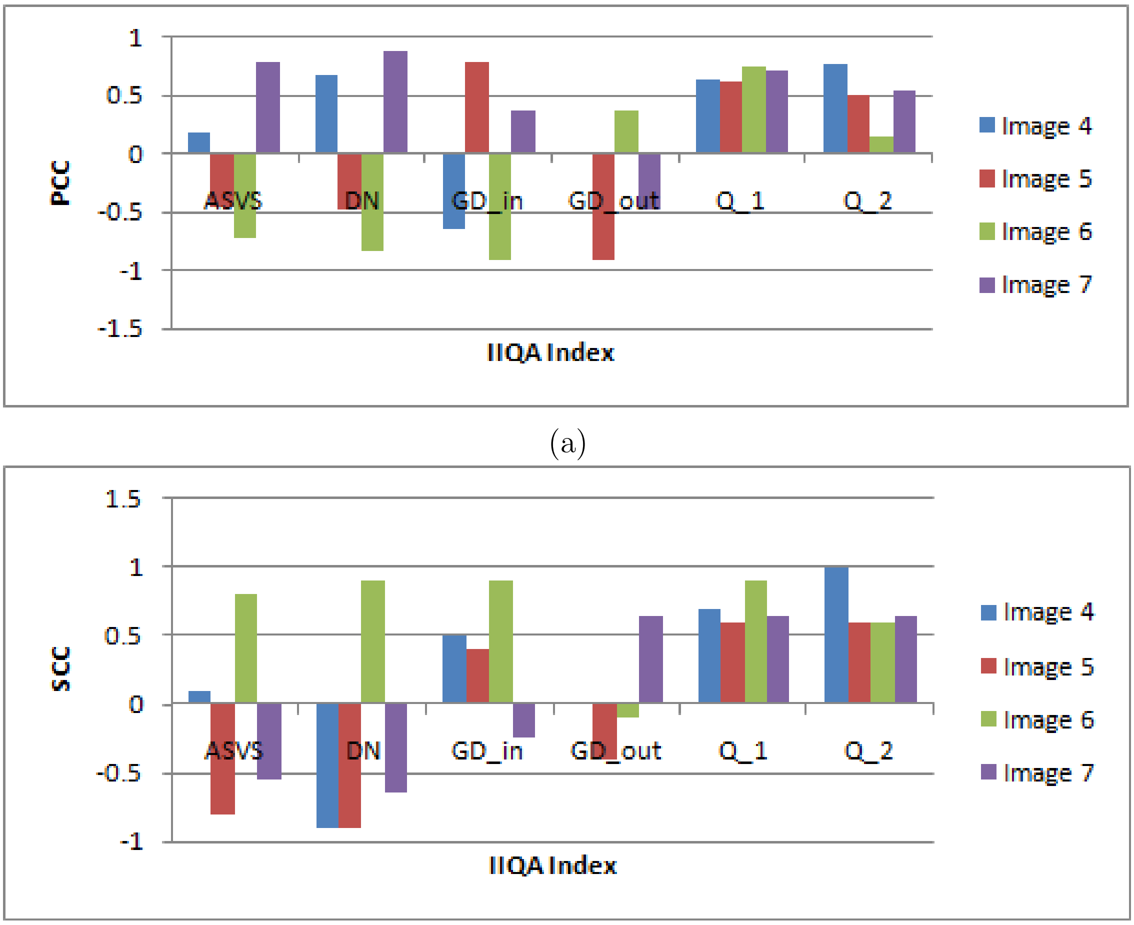
	PCC				**SCC**			
Image ID	4	5	6	7	4	5	6	7
$ASVS$	0.1946	-0.4651	-0.7154	0.7891	0.1	-0.8	0.8	-0.55
DN	0.6870	-0.4790	-0.8379	0.8793	-0.9	-0.9	0.9	-0.65
GD_{in}	-0.6571	0.7918	-0.9132	0.3815	0.5	0.4	0.9	-0.25
GD_{out}	0.0111	-0.9129	0.3777	-0.4756	0	-0.4	-0.1	0.65
Q_1	0.6339	0.6250	0.7516	0.7185	0.7	0.6	0.9	0.65
Q_2	0.7807	0.4999	0.1497	0.5429	1	0.6	0.6	0.65

Table 4.7. The PCC and SCC indexes for global optimization strategies.

	PCC			**SCC**		
Image ID	8	9	10	8	9	10
$ASVS$	0.7135	0.7135	0.4242	-0.5	-1	-0.5
DN	0.9073	0.9921	-0.9885	-1	-1	1
GD_{in}	-0.7203	0.9954	0.9989	-1	-1	-1
GD_{out}	0.2970	-0.8869	-0.8638	-1	0.5	0.5
Q_1	0.5140	0.8446	0.9894	0.5	1	1
Q_2	0.4941	0.8515	0.9989	1	1	1

from this toolkit. The output results are also subjectively evaluated and MOS are available for studying the performance of the considered metrics.

The PCC and SCC values given in Tables 4.6 and 4.7 clearly confirm the efficiency



(a)

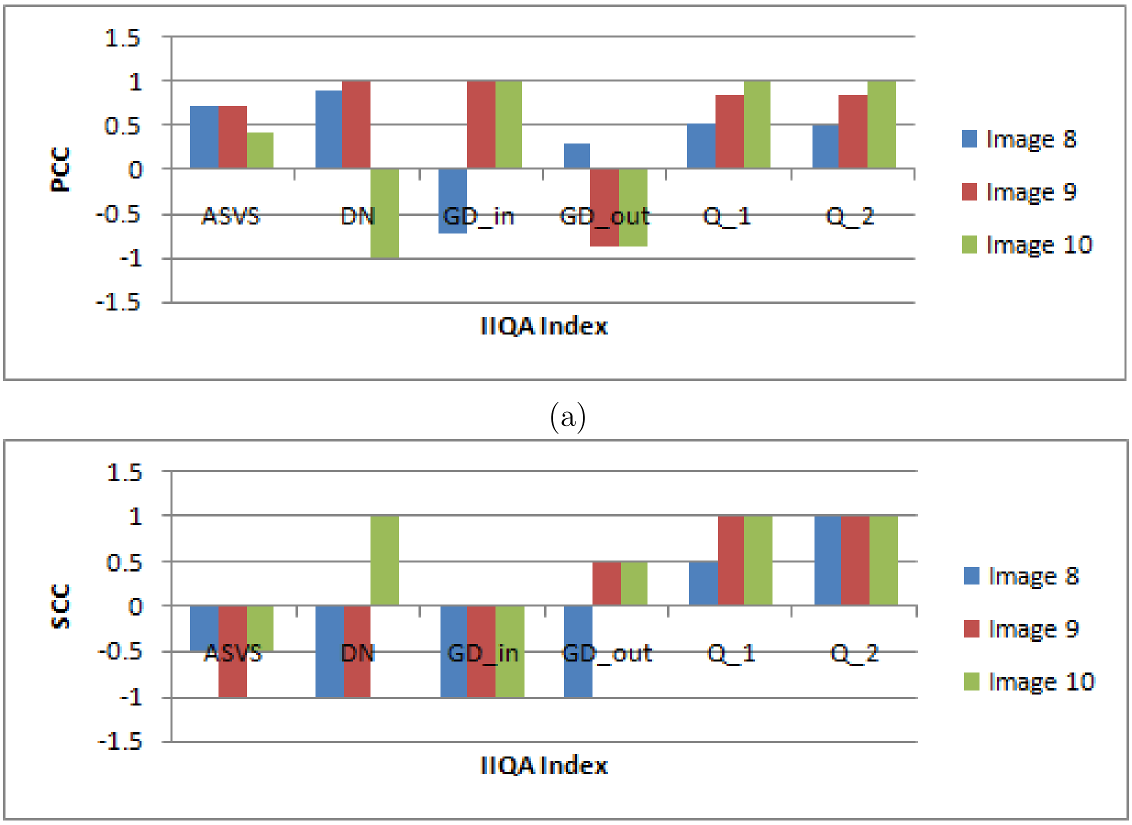
(b)

Figure 4.19. Performance of the quality metrics in case of large inpainted region for greedy strategy. (a) Pearson correlations (PCC); (b) Spearman correlations (SCC).

of the proposed index in quantifying the quality of inpainting results. To illustrate more intuitively, they are represented by charts in Figures 4.19 and 4.20. Indeed, the mean values of the proposed indexes are highest. This indicates that our index appears promising for inpainting quality assessment.

4.7 Conclusion

In this section, a novel approach for inpainted image quality evaluation has been proposed. It is shown that the traditional image quality index could not be used for evaluating the inpainting results. By taking into account the specificities and objectives of image com-



(a)

(b)

Figure 4.20. Performance of the quality metrics in case of large inpainted region for global optimization strategy. (a) Pearson correlations (PCC); (b) Spearman correlations (SCC).

pletion problem and some characteristics of the humane visual system, such as perceptual saliency, two efficient metrics could be derived, the gray and color level versions.

An overview of existing image quality metrics and inpainting quality metrics has been discussed and compared with our proposals in some cases.

Eleven different digital inpainting algorithms were qualitatively evaluated by conducting a psychophysical experiment and a set of ten natural images were chosen in order to generate an inpainting database for evaluation. To avoid ranking bias, the inpainting algorithms were divided into two cases: the small and large damaged regions. The tested images were also classified into corresponding groups.

The proposed image inpainting quality indexes not only correlate with subjective evaluation but also could be applied to most of image inpainting approaches because they are no-reference inpainting quality metrics. The performed experimental results and comparison with some current metrics confirm the efficiency of the proposed indexes respect to both the Pearson's correlation, PCC, and the Spearman rank order correlation, SCC, coefficients.

Future work will include an expanded psychophysical experiment that will consider more recently natural images and current inpainting algorithms in order to reconfirm the efficiency of our proposals. An extensive research should be conducted with the aim of developing an IIQA metric for video or sequence images.

CHAPTER 5

Inpainting Forgery Detection

"It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change."

Charles Darwin

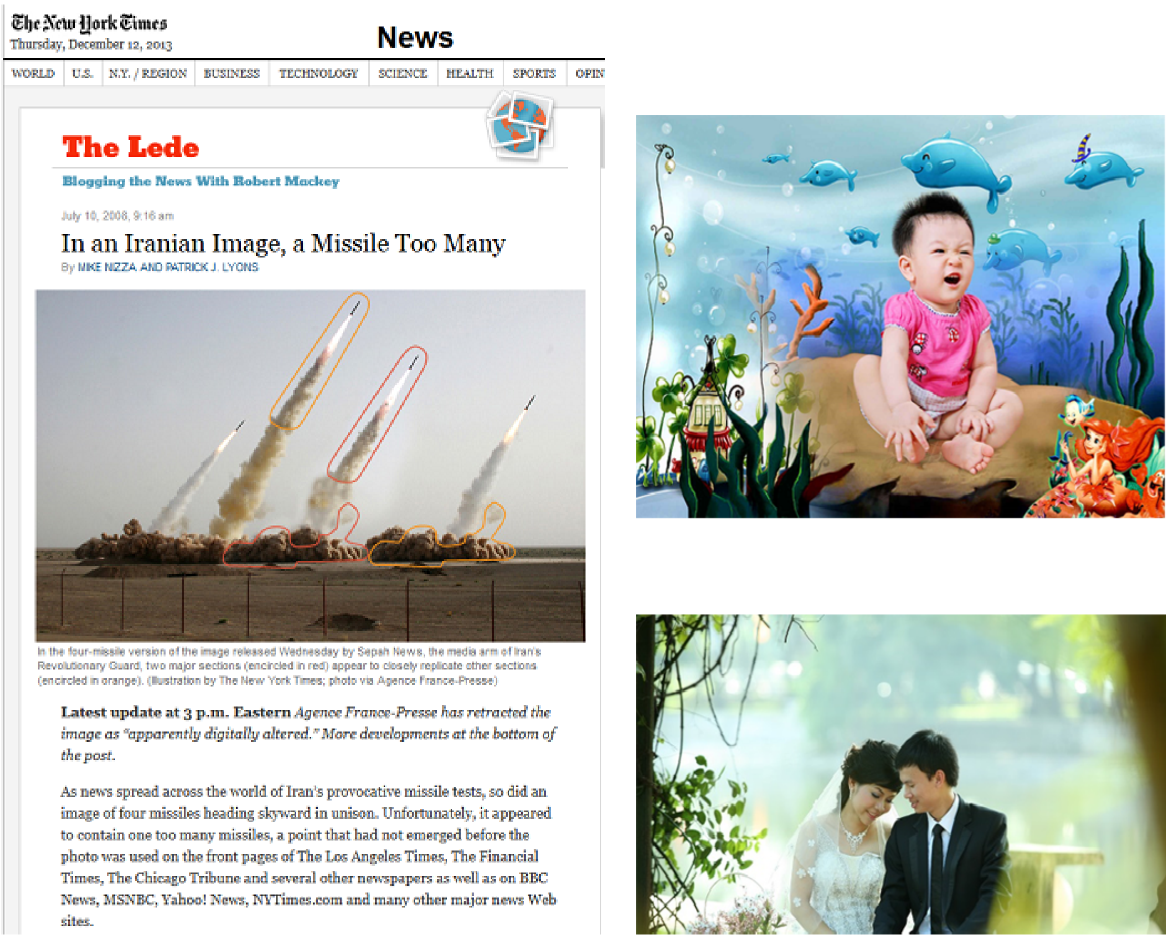
Contents

5.1 Why could inpainting be considered as forgery tool?

Today's technology allows users to manipulate digital media in ways that were simply impossible a decade ago. Tomorrow's technology will almost certainly allows us to manipulate digital media in ways that today seem unimaginable. These media are prone to several sources of accidental or intentional degradations. And as this technology continues to evolve, it will become increasingly important for the science of digital forensics to try to keep pace. As a result, there is a rapid increase of the digitally manipulated forgeries in mainstream media, on the Internet, especially on social network. Therefore, developing techniques to verify the integrity and the authenticity of the digital images became very important, especially considering the images presented as evidence in a court of law, as news items, as a part of a medical record or as a financial document, etc.

An image can become a forgery based on the context in which it is used. For instance, an image altered for fun or someone who has taken a bad photo, but has been altered to improve its appearance cannot be considered a forgery even though it has been altered from its original capture. In contrast, it is a forgery image if an altered image is used to create non-existing situations threatens to diminish the credibility and value of images or videos presented as evidence in court independently of the fact.

An example of a digital forgery is shown in Figure 5.1. As in the figure, the sub-figure 5.1-a is considered as forgery because they are illustrated in a newspaper in order to distort the truth [92]. Some objects was rescaled and duplicated to paste on the original images and create an illusion of an out-of-focus background. Two other images in Figure 5.1-b have been edited for fun. Surely, they are not tampering.



The Lede

Blogging the News With Robert Mackey

July 10, 2008, 9:16 am

In an Iranian Image, a Missile Too Many

By MIKE NIZZA AND PATRICK J. LYONS

In the four-missile version of the image released Wednesday by Sepah News, the media arm of Iran's Revolutionary Guard, two major sections (encircled in red) appear to closely replicate other sections (encircled in orange). (Illustration by The New York Times; photo via Agence France-Presse)

Latest update at 3 p.m. Eastern *Agence France-Presse has retracted the image as "apparently digitally altered." More developments at the bottom of the post.*

As news spread across the world of Iran's provocative missile tests, so did an image of four missiles heading skyward in unison. Unfortunately, it appeared to contain one too many missiles, a point that had not emerged before the photo was used on the front pages of The Los Angeles Times, The Financial Times, The Chicago Tribune and several other newspapers as well as on BBC News, MSNBC, Yahoo! News, NYTimes.com and many other major news Web sites.

(a)

(b)

Figure 5.1. An example of very realistic-looking forgeries. The image in (a) is forgery because of illustration in a newspaper [92]; but two edited photos of a wedding album in (b) are not forgery.

Naturally, digital image forensics technology is a new research and it can be divided into active evidence and passive-blind evidence based on whether the additional information is embedded into digital images in advance or not. Alternatively, active evidence is mainly about digital watermark which has been proposed as a mean for fragile authentication, content authentication, detection of tampering, etc. While this approach can provide useful information about the image integrity and its processing history, the watermark has some drawbacks:

- *The watermark must be inserted at the time of recoding before the tampering occurs.*
- *The watermark can be fragile and it is easy to be destroyed by an image processing operator such as compressing, rescaling, etc.*
- *The watermark cannot detect accurately areas where the image has been manipulated.*
- *The watermark cannot distinguish whether a manipulation is innocent, such as JPEG compression, sharpening from those which are malicious such as adding or removing parts of an image.*

On the contrary, passive techniques for image forensics should operate in the absence of any watermark or signature. These techniques work on only the assumption that although digital forgeries may leave no visual clues that indicate tampering, they may alter the underlying statistics of an image. The set of image forensic tools can be roughly divided into five groups [93]:

1. *Pixel-based methods* that detect statistical anomalies introduced at the pixel level.
2. *Format-based methods* that leverage the statistical correlations introduced by a specific lossy compression.
3. *Camera-based methods* that exploit artifacts introduced by the camera lens, sensor, or on-chip processing.
4. *Physically-based methods* that explicitly model and detect anomalies in the three dimensional interaction between physical objects, light and the camera.
5. *Geometric-based methods* that make measurements of objects in the world and their positions relative to the camera.

Early tampering using powerful image processing tools such as Photoshop or Freehand, is not too big challenge for humans. However, the development of sophisticated algorithms in image processing field, such as image inpainting, which do not leave any obvious clues after manipulating image, make it relatively difficult to detect with human vision. Thus, image tampering detection is one of the primary objectives in image forensics.

In this chapter, we are considering a specific type of image forgery using inpainting algorithm to modify the context of image, namely inpainting detection. Since the main goal of inpainting is to remove or restore a part of the image based on the remaining parts of the same image such that the restored regions are undetectable by viewers, developing techniques to verify the integrity and the authenticity of the digital images became very important. Especially when the inpainting algorithms is more sophisticated and tampered images do not reveal any visual artifacts or anomalies, the detection algorithm can still work well based on the difference of underlying statistics between these images and the original one. To restore the large damaged regions, the main operation is copying and pasting pixels' intensity values from the known sources into the unknown regions. In principal, the inpainting detection is synonymous with copy-move forgery detection. However, there is a little difference between these two problems:

- The copy-move forgery techniques are used to detect the clone area using image processing tools. While the damaged regions in inpainting problem are restored from many distinguishing patches in the known source. These patches may not belong to the same area. Thus, the duplicated regions may not exist (for example, inpainting in previous section).

- The forged region decision has been made based on the amount of shift, the distance between the original and copied regions. However, the duplicated regions may not exist in inpainted image. Thus, it is impossible to confirm an inpainted image based on a shift counter as in copy-move detection.

- In some applications, the objective of inpainting algorithm is to remove or replace an object which may include the feature points with a homogeneous background which may not include any feature point. Thus, using image feature matching to detect the duplication region is not able to apply for inpainting detection in this sense.

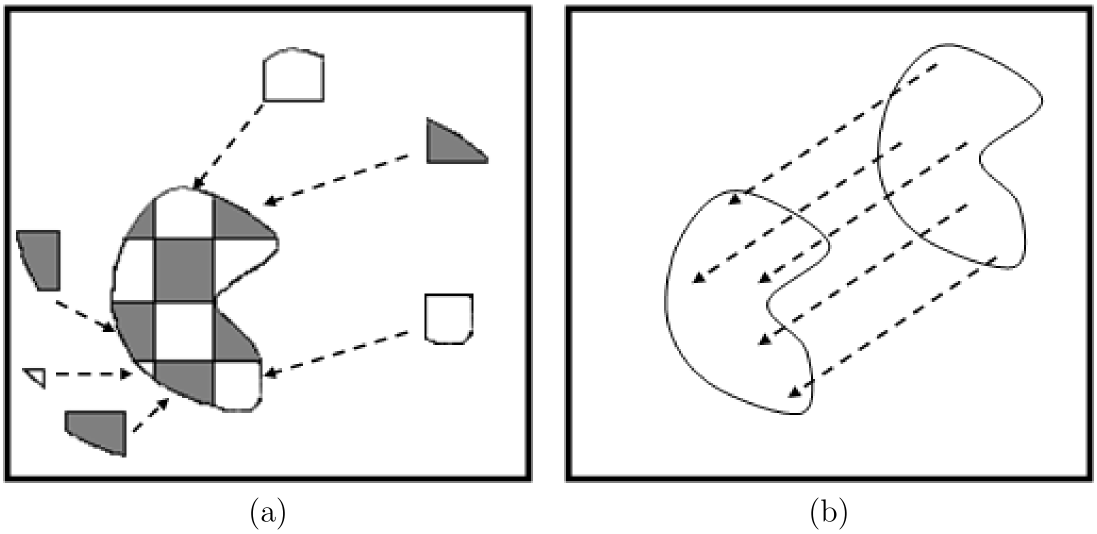


Figure 5.2. The difference between inpainting and copy-move problems. (a) inpainted region is copied from many different sources; while (b) a forged region is copied from another isoform.

Figure 3.13 illustrates an example of inpainting result in which the inpainted region is filled in by many distinguish small regions. Figure 5.2 indicates the differences between these two problems. In this context, the existing copy-move detection techniques often fail to detect the inpainting regions.

5.2 Overview of forgery

To cover and summarize all forgery techniques would be hard and complex. Thus, this section only focuses on related forgery techniques that can be referred to compare and evaluate the performance of our proposal for inpainting detection purpose. Because we are interested in only large restored regions using inpainting algorithm with processing operations similar to copy-move methods, a brief evaluation of popular copy-move forgery detections will be considered in this section and listed in Table 5.1. As described before, the columns *Year* and *Authors* present the year of the publication and the name of the authors, respectively. The *Type* indicates the category of the algorithm. The two remaining columns describe for the used feature. Where *F-type*, short for 'Feature type', shows the kind of feature extracted from image or block, and the corresponding number of dimensions is given in column *F-length*, stand for 'Feature length'. The symbol '-' is used when the number of dimension is undefined or not mentioned in the publication.

Table 5.1. The selected methods for copy-move forgery detection.

Year	**Authors**	**Type**	**F-type**	**F-length**
2003	Fridrich *et al.* [94]	Frequency	DCT	256
2009	Bayram *et al.* [95]	Frequency	FMT	45
2010	Bashar *et al.* [96]	Frequency	DWT	256
2004	Popescu *et al.* [97]	Dimensionality reduction	PCA	32
2008	Kang *et al.* [98]	Dimensionality reduction	SVD	-
2010	Bashar *et al.* [96]	Dimensionality reduction	KPCA	192
2007	Mahdian *et al.* [99]	Moments	BLUR	24
2009	Wang *et al.* [100]	Moments	HU	4
2010	Ryu *et al.* [101]	Moments	ZERNIKE	12
2006	Luo *et al.* [102]	Intensity	LUO	7
2011	Bravo-Solorio *et al.* [103]	Intensity	BRAVO	4
2010	Lin *et al.* [104]	Intensity	LIN	9
2009	Wang *et al.* [105]	Intensity	CIRCLE	4
2008	Huang *et al.* [106]	Keypoint	SIFT	128
2010	Pan *et al.* [107]	Keypoint	SIFT	128
2011	Amerini *et al.* [108]	Keypoint	SIFT	128
2010	Bo *et al.* [109]	Keypoint	SURF	64
2011	Shivakumar *et al.* [110]	Keypoint	SURF	64

Although, a large number of copy-move detection algorithms have been published, most techniques follow a common scheme as shown in Figure 5.3. According to this scheme, the first step is *pre-processing* in which some operations are performed to normalize the input image. For instance, most methods require a grayscale image as input rather than color image. Thus, the color channels must be first merged to convert into gray level. Due to differences in the computational cost as well as the detected details, two processing alternatives such as *block-based* methods and *keypoint-based* methods [111] are used to extract the feature vectors. The *matching* step is to define the similarity between two feature descriptors which is interpreted as a cue for duplicated regions. *Filtering* schemes have been proposed to reduce the probability of false matches. Different distance criteria were also formulated in order to filter out weak matches. For example, most authors proposed the Euclidean distance between matched feature vectors. *Bravo-Solorio and Nani* [103] proposed the correlation coefficient as a similarity criterion. Lastly, the *post-processing* is only used to preserve matches that exhibit a common behavior. Consider a set of matches belonging to a copied region, they are expected to be spatially close to each other in both the source and the target blocks. Moreover, matches that originate from the same copy-move action should exhibit similar amounts of translation, scaling and rotation.

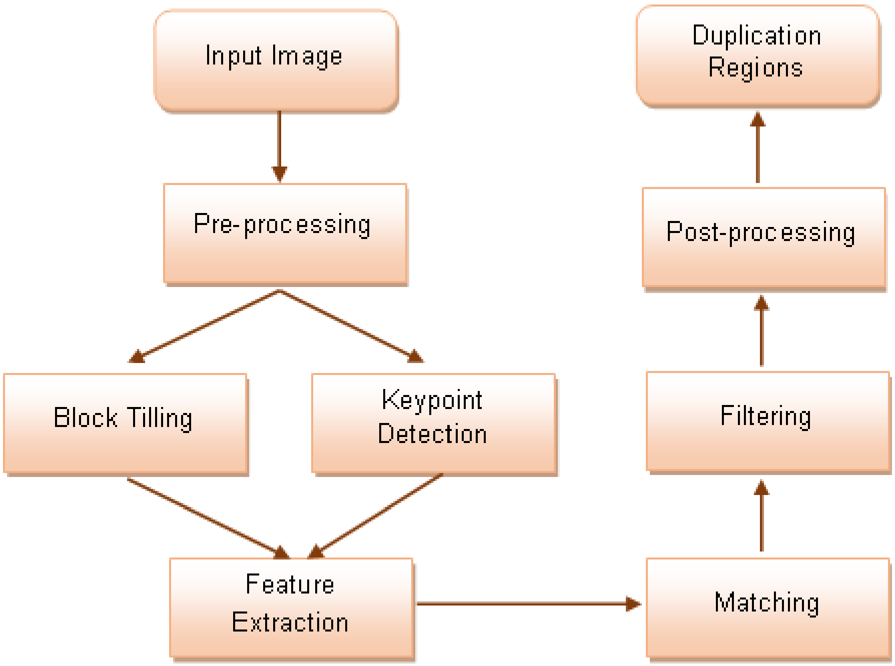


Figure 5.3. Common processing pipeline for the copy-move detection.

5.2.1 Block-based approaches

The key characteristic of the methods of this group is dividing the input image into overlapping blocks. The feature vector of each block is then extracted and used as input for block matching. The feature length depends on each extraction algorithm. Figure 5.4 shows a common illustration of the methods in this group.

There are many criteria to classify the methods of this group into sub-groups. For block matching, there are two sub-groups: *exact match* and *fuzzy match*. The difference between them is that the vector of block feature is identical or similar. Exact match only finds out exactly the same as the copy of image block and its robustness is limited. The fuzzy match is used more widely because of more practical value. For block feature extraction, most methods can be classified into four sub-groups:

- *moment-based methods.*
- *dimensionality reduction-based methods.*
- *intensity-based methods.*
- *frequency domain-based methods.*

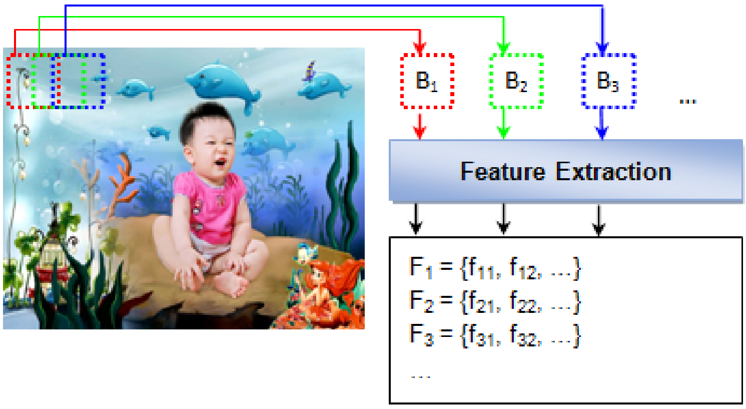

B_1
B_2
B_3
...
Feature Extraction
$F_1 = \{f_{11}, f_{12}, ...\}$
$F_2 = \{f_{21}, f_{22}, ...\}$
$F_3 = \{f_{31}, f_{32}, ...\}$
...

Figure 5.4. A common illustration of block-based approaches.

A notable constraint in this group is that the block size must be considerably smaller than the duplicated region to be detected. In this section, thirteen block-based methods covering all four groups are investigated.

5.2.1.1 Frequency-based methods

Methods in this group concentrate on transforming the pixel values within a block into spatial frequency space in order to extract and preserve high or low frequency features. In this group, there are three typical approaches based on *Discrete Cosine Transform* (DCT) [94], *Discrete Wavelet Transform* (DWT) [96] and *Fourier-Mellin Transform* (FMT) [95].

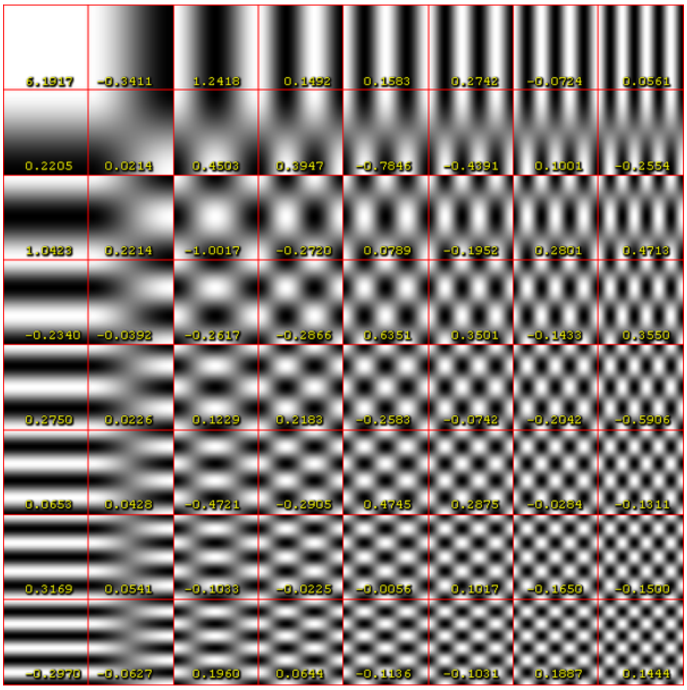


Figure 5.5. Basis functions of the Discrete Cosine Transformation with corresponding coefficients.

As one of the earliest methods, *Fridrich et al.* [94] have proposed a simple and efficient algorithm to detect the duplicated region using block matching strategy. To avoid the influence of noise and compression, 256 DCT coefficients are used as features instead of the original pixel values. In addition, an extension of DCT for block size 16 × 16 is developed to prevent too many false matches instead of simply using the standard quantization matrix of JPEG (8 × 8 as in Figure 5.5). A detailed extension of the quantization matrix is given in [94]. All feature vectors are then lexicographically sorted to reduce computational complexity. Finally, the forgery decision is made based on a 2D shift counter which accumulates the frequencies of relative distance of matching blocks.

The second frequency-based approach recalled in this section has been proposed by

Bashar et al. [96]. Since natural images might have discontinuity effects due to existence of various man-made structures, the traditional Fourier Transform (FT) or Short Time Fourier Transform (STFT) is not well suited. In this scenario, the authors indicated that a Wavelet Transform is more suitable to the effective local analysis of the image. A DWT using Haar-Wavelets is performed to convert the blocks in a natural image into DWT coefficients. These coefficients are then arranged into a vector according to decreasing local variances. Moreover, authors also introduced another transform to reduce the dimensionality of feature using Kernel Principal Component Analysis (KPCA), a variant of traditional PCA. Accordingly, the feature vector is reduced up to 192 dimensions. Although KPCA is computationally more expensive than the linear PCA, it extracts more useful features. Some results of this approach can be found in [101].

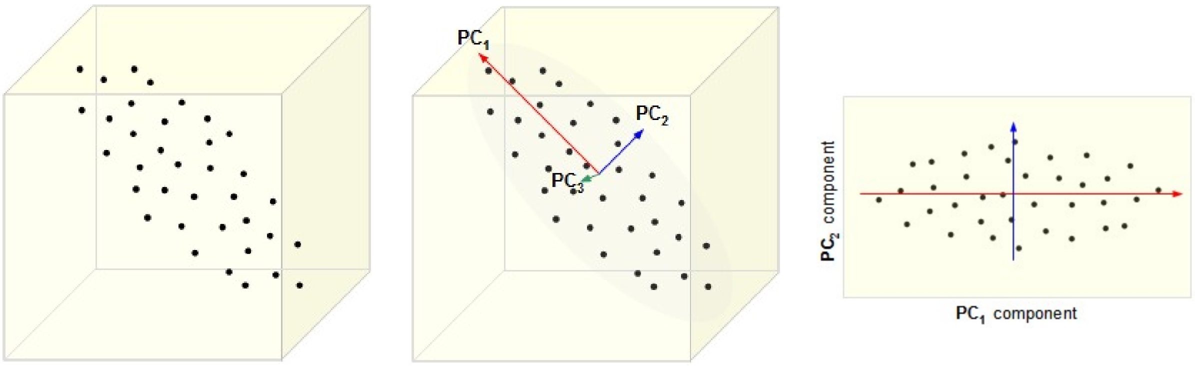
The last frequency-based approach has been introduced by *Bayram et al.* [95]. The authors recommended the use of the FMT for generating feature vectors. Essentially, these features would not be only robust to lossy JPEG compression, blurring, or noise addition, but also known to be scaling and translation invariant but sensitive to rotation. Furthermore, the counting boom filters are used, instead of lexicographic sorting, to reduce the detection time. Some outputs of this approach are available in [95].

5.2.1.2 Dimensionality reduction-based methods

The main idea of the methods in this group is to reduce the number of dimensionality of feature vector which can significantly affect the performance of similar block search. Here, we recall three typical techniques for converting and reducing the number of correlated dimensionality, such as *Principal Component Analysis* (PCA), *Singular Value Decomposition* (SVD) and *Kernel Principal Component Analysis* (KPCA).

In [97], *Popescu et al.* proposed to perform PCA to derive an alternative representation of the blocks instead of quantized DCT coefficients. For this purpose, a new linear basis was obtained by finding the eigenvectors of the covariance matrix and the projection of each block onto these basis vectors with higher eigenvalues was used as new representation to reduce dimensionality. In their paper, the method was shown to be robust to compression up to JPEG quality level 50 and to additive noise with SNR 36dB and 29dB, but not to re-sampling (scaling, rotation).

Kang et al. [98] computed the singular values of a reduced rank approximation (SVD). In the proposed passive techniques, SVD served to produce algebraic and geometric in-



(a) (b) (c)

Figure 5.6. An illustration of PCA dimensionality reduction. (a) A data set given as 3-dimensional points; (b) The three orthogonal Principal Components (PCs) for the data, ordered by variance; (c) The projection of the data set into the first two PCs, discarding the third one.

variant and feature vectors. Experimental results demonstrate the validity of the proposed approach to tampered images undergone some attacks like Gaussian blur filtering, Gaussian white noise contamination and lossy JPEG compression.

The third technique for reduction of dimensionality has been proposed by *Bashar et al.* [101] as mentioned in previous subsection. In their work, an extension of traditional PCA, called Kernel Principal Component Analysis (KPCA) which achieves non-linear dimensionality reduction through the use of kernels, is applied to reduce the dimensionality of feature. A performance evaluation of this approach is described in more details in [101].

5.2.1.3 Moment-based methods

In general, moments describe numeric quantities at some distance from a reference point or axis. In image processing, computer vision and related fields, an image moment is a certain particular weighted average (moment) of the image pixels' intensities, or a function of such moments, usually chosen to have some attractive property or interpretation. In this context, three kinds of image moments, such as *blur moment* [99], *Hu moment* [100] and *Zernike moment* [101], have been proposed.

The first moment-based method was proposed by *Mahdian and Saic* [99] where blur moment invariants are developed as block feature. Here, each block is represented by blur invariants which are function of central moments. 24 blur invariant components up to the

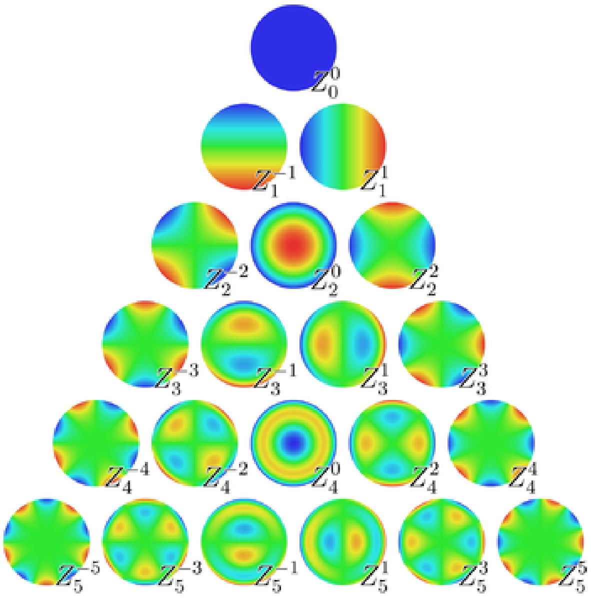


Figure 5.7. A set of complex Zernike polynomials.

seventh order are extracted to create a $24D$-feature vector. The detailed implementation of this invariant extraction is found in [99]. For color images, moment invariants of each block in each channel are computed separately, resulting in a $72D$-feature vector. To reduce the dimension, a PCA is applied and the number of dimensionality is cut down to $9D$. In addition, the KD-tree representation is constructed for searching nearest neighbors to improve the efficiency of finding neighboring blocks.

Following the same idea, *Wang et al.* [100] used first four Hu moments as features instead of blur invariants. The Hu moments are invariant with respect to scale, position, and orientation. The efficiency of this approach lies in the use of a Gaussian pyramid representation to reduce the dimension of input image as well as reduce the total number of blocks to narrow block-matching searching space. The Hu moment is applied to the fixed sized overlapping blocks of low-frequency image. Finally, similar eigenvectors are matched using a certain threshold value and some mathematical morphology operations are performed to locate the tampered parts.

The last method of this group has been recently proposed by *Ryu et al.* [101] in which Zernike moments are used as features to detect the duplicated regions. Of various types of moments defined in the literature, Zernike moments have been proven to be superior to the others in terms of insensitivity to image noise, information content, and ability to provide faithful image representation. In particular, the magnitude of Zernike moments is algebraically invariant against rotation. So this method can detect a forged region even though it is rotated. Each block is represented by a feature vector of 12 Zernike moments. For more details of the algorithm, the reader could refer to [101].

5.2.1.4 Intensity-based methods

The last group of detection techniques presented in this thesis is intensity-based approach where the feature vectors are computed directly from the pixels' intensities instead of transforming into other orthogonal projections. Four detection techniques are considered.

The first detection technique was proposed by *Luo et al.* [102]. For each block, seven characteristic features are calculated including three average components of the red, green and blue channels and four other directional information components of blocks. To calculate the last four features, the block is divided into 2 equal parts in 4 directions as Figure 5.8 and these components are computed on Y channel. The detailed implementation can be found in [102]. These characteristics features do not change significantly after some common processing operations such as Gaussian noise, Gaussian bluring, JPEG compression.

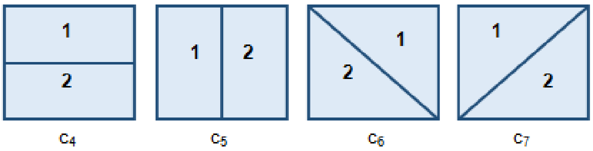


Figure 5.8. Four directions for calculation of the last four features in Luo's approach [102].

The second technique in this group was introduced by *Bravo-Solorio et al.* [103]. The authors considered the entropy of a block as a discriminating feature. Four color-dependent features are calculated considering only the pixels within a disc of diameter q, which just fits inside block. The first three features are independently computed as the average of the red, green and blue components of the pixels in the disc as in *Luo*'s approach [102], but the fourth feature is computed as the entropy of the luminance channel. The algorithm uses colour-dependent feature vectors to reduce the number of comparisons in the search stage. While 1D descriptors, invariant to reflection and rotation, are used to perform an efficient search in terms of memory usage. To reduce the number of false matches, the correlation coefficient between two features is used instead of Euclidean as other methods.

The third approach was recommended by *Lin et al.* [104]. The 9-dimensional feature vectors are extracted based on the average gray-scale intensities of a block and its sub-blocks. Firstly, each block is divided into four equal-sized sub-blocks (Figure **??**). Then, the first dimension is calculated from the average intensity of whole block. The next four dimensions denote the ratios of the average intensities of four sub-blocks to the first dimension. The last four dimensions stand for the differences of the average intensities of the four sub-blocks from the first dimension. Finally, all components of feature vector are normalized to integers ranging in $[0, 255]$. Although these 9 entities contain duplicated information, they together possess higher capability of resistance against some modifications, such as JPEG compression and Gaussian noise. There is a slight difference with the *Popescu*'s approach in sorting feature vectors. Because elements of feature vectors are integers, it is possible to use the efficient radix sort algorithm to perform lexicographical sorting over those vectors.

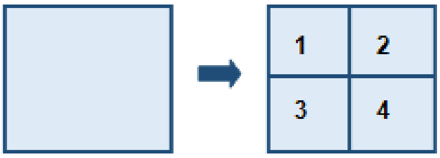


Figure 5.9. A block is divided into four equal-size sub-blosks in Lin feature [104].

Finally, *Wang et al.* [105] used the mean intensities of circles with different radii around the block center as circle features. Instead of breaking the input block into four sub-blocks as *Lin*'s method [104], authors broke the input block into four concentric circles (Figure 5.10). The mean of image pixel values in each circle region is adopted as feature. To narrow the search space and the total number of blocks, the input image is reduced in dimension by Gaussian pyramid decomposition.

5.2.2 Keypoint-based approaches

Unlike block-based algorithms in which the image is divided into overlapping blocks, keypoint-based methods rely on the identification and selection of high-entropy image regions. Nowadays, local visual features (e.g. SIFT, SURF) have been widely used for image

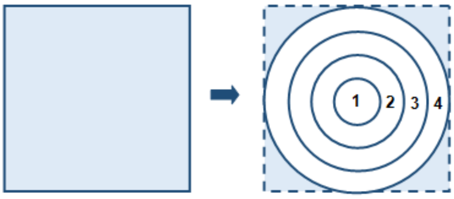


Figure 5.10. A block is divided into four concentric circles in Wang feature [105].

retrieval and object recognition, their robustness to several geometrical transformations (such as rotation, scaling, bluring) occlusions and clutter. More recently, these kinds of features have been also attempted to apply in the digital forensics domain. Figure 5.11 show a common chart to illustrate for the methods in this case.

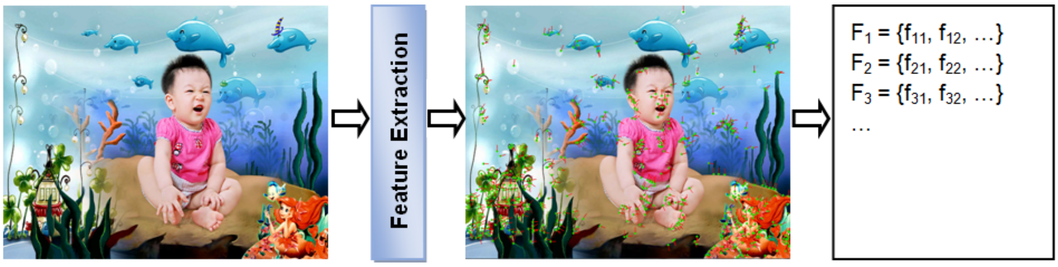

Feature Extraction
$F_1 = \{f_{11}, f_{12}, \ldots\}$
$F_2 = \{f_{21}, f_{22}, \ldots\}$
$F_3 = \{f_{31}, f_{32}, \ldots\}$
...

Figure 5.11. A common illustration of keypoint-based approaches.

In this context, each keypoint is characterized by a feature vector that consists of a set of image statistics collected at the local neighborhood of the corresponding keypoint. Consequently, fewer feature vectors are estimated, resulting in reduced computational complexity of feature matching and post-processing. Two different kinds of keypoints are examined such as *Scale Invariant Feature Transform* (SIFT) [106–108] and *Speeded Up Robust Features* (SUFT) [109, 110]. A drawback of keypoint methods is that copied regions exhibit little structure or contain the homogeneous regions, it may happen that the region is completely missed.

5.2.2.1 SIFT-based methods

SIFT feature shorting for *Scale Invariant Feature Transform* has been proposed by *D. G. Lowe* [112]. This feature is invariant to image scale, rotation and originally developed for object recognition. In fact, SIFT features are also a good solution for duplicated regions detection because of their robust performance and relatively low computational costs. There are four main steps of computation to generate the SIFT features: *i) Scale-space extrema detection*; *ii) Keypoint localization*; *iii) Orientation assignment*; *iv) Keypoint descriptor generation*. The detailed specification and source code can be found in [112].

At each keypoint, a 128 dimensional feature vector is generated from the histograms of local gradients in its neighborhood. To ensure the invariance of the obtained feature vector to rotation and scaling, the size of the neighborhood is determined by the dominant scale of the keypoint, and all gradients within are aligned with the keypoint's dominant orientation. Furthermore, the obtained histograms are normalized to unit length, which renders the feature vector invariant to local illumination changes.

In order to identify possible cloned areas, the detected SIFT keypoints are then tentatively matched based on their feature vector using the best-bin-first algorithm [106, 107] or agglomerative hierarchical clustering [108]. To handle rotation and scaling, *Pan and Lyu* [107] proposed to use RANSAC; while *Amerini et al.* [108] applied a scheme to cluster the locations of detected features before using RANSAC to estimate the geometric transformation between the original area and its copy-moved verions.

5.2.2.2 SURF-based methods

The second kind of keypoint is used for detection of duplicated regions is SUFT feature. SUFT feature known as *Speeded Up Robust Feature* has been recently published by *Bay et al.* [113]. Keypoints are found by using a so called Fast-Hessian Detector that is based on an approximation of the Hessian matrix for a given image point. The responses to Haar wavelets are used for orientation assignment, before the keypoint descriptor is formed from the wavelet responses in a certain surrounding of the keypoint. There are three main stages of computation used to generate the SIFT features: *i) Integral image*; *ii) Hessian matrix-based interest point*; *iii) Interest point descriptors*. The detailed specification can be referred to [113].

In contrast to SIFT feature, which approximates Laplacian of Gaussian (LoG) with Difference of Gaussians (DoG), SURF feature approximates second order Gaussian derivatives with box filters. Moreover, only 64 dimensions are used, reducing the time for feature computation and matching, and increasing simultaneously the robustness. The extracted SURF keypoints are then matched using divide and conquer algorithm [109] or KD-tree structure [110].

Besides achieving promising performance in detecting sophisticated forgeries with duplicated regions, the keypoint-based methods may have a limitation for some cases. One example is shown as the Figure 5.12 where an obvious duplicated square cannot be detected by this kind of method. The reason is the SIFT or SURF algorithms cannot find reliable keypoints in regions with little visual structures. Similarly, as smaller regions have fewer keypoints, they are also hard to detect.

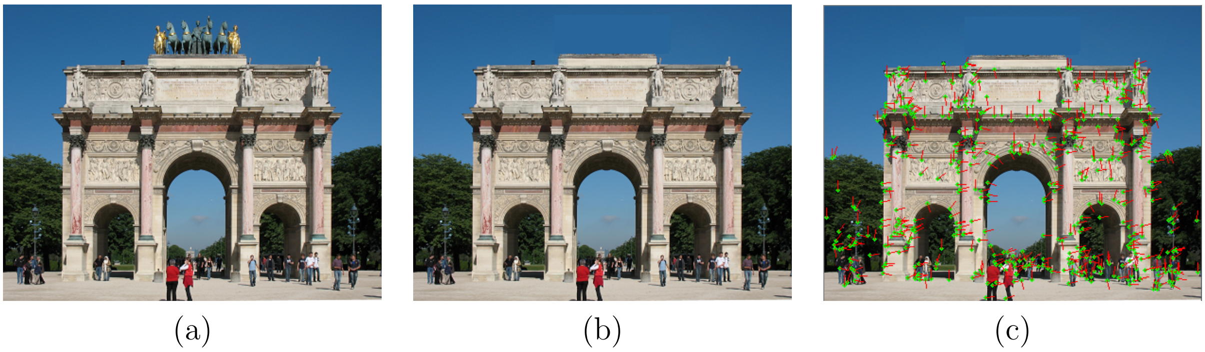
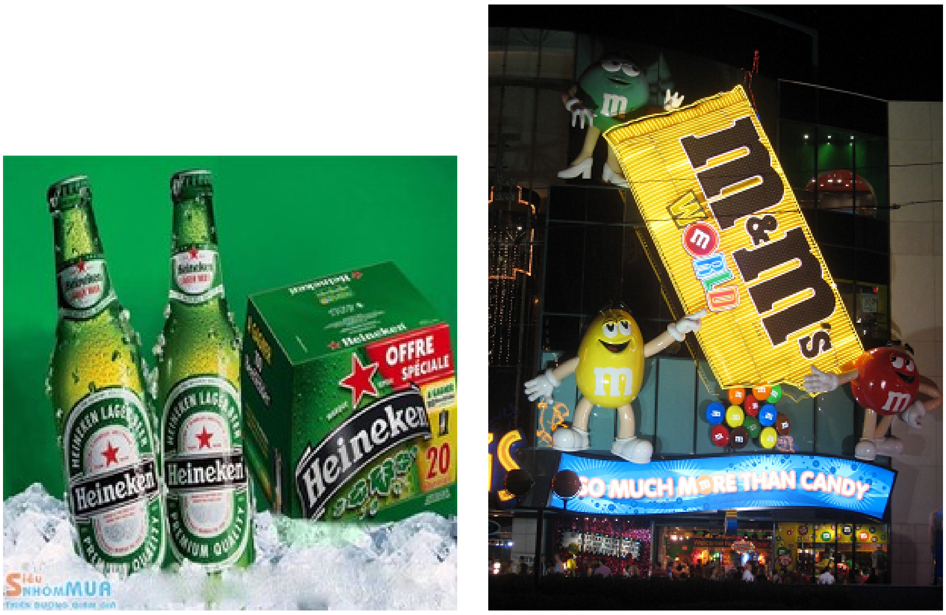


Figure 5.12. An illustration of failed detection results using keypoint-based approach. (a) the original image; (b) a forgery image using duplicated region; (c) the keypoint-detected image.

Another case of false detection is illustrated in the Figure 5.13 where the input images have intrinsically identical areas that cannot be differentiated from intentionally inserted duplicated regions. Thus, as the future work, these methods could be improved by incorporating other features or histograms of oriented gradients, or combining with other detection schemes based on intrinsic signal statistics/patterns to provide strong cues when image keypoints and features are insufficient.

5.3 A proposal for inpainting detection

In this work, we focus on detecting a specific type of digital image forgery, namely image inpainting. The intent of this operation is to modify the content of an image in the most visually plausible way. This yields to what we call *inpainting forgery*. This technique is more sophisticated and complex than copy-move forgery because the source of copied information could be non-continuous. This means that an object may be filled by a set of multiple small parts located at different places in the same image and not a single



(a) (b)

Figure 5.13. An illustration of false detection results using keypoint-based method. The images are known untampered but are detected to have duplicated regions

continuous region. Therefore, inpainting forgery is more difficult to detect than copy-move forgery as it can be seen on Figure 5.14 where state-of-the-art copy-move detection algorithms would fail.

To verify the capacity of the copy-move detection techniques for inpainting forgery, a series experiments have been carried out on a set of inpainted images. Most of copy-move detection techniques introduced in Section 5.2 have been applied. In this work, we developed the framework in [111] for this purpose. Figure 5.15 and 5.16 show an example of the results for our experiment. No detection has been found in the Figure 5.15. Figure 5.16-b detected a small copy-move part but it is uncompleted. This is certain because the forged region has been filled from many small and distinct parts. Surely, it cannot detect the forged region completely.

According to our observation, all outputs either cannot detect the forgery region or make a false detection. This confirms firmly that the copy-move detection methods is not suitable for detecting inpainting forgery. A different framework should be considered and developed for this special case.

In fact, to the best of our knowledge there is almost no study about image inpainting

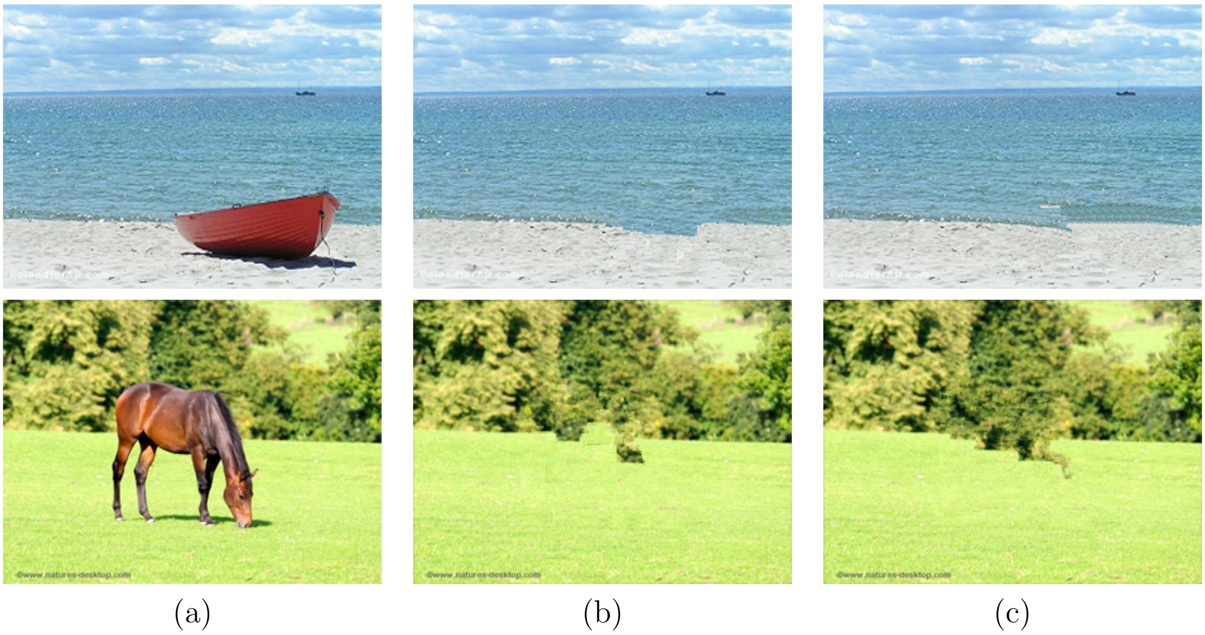
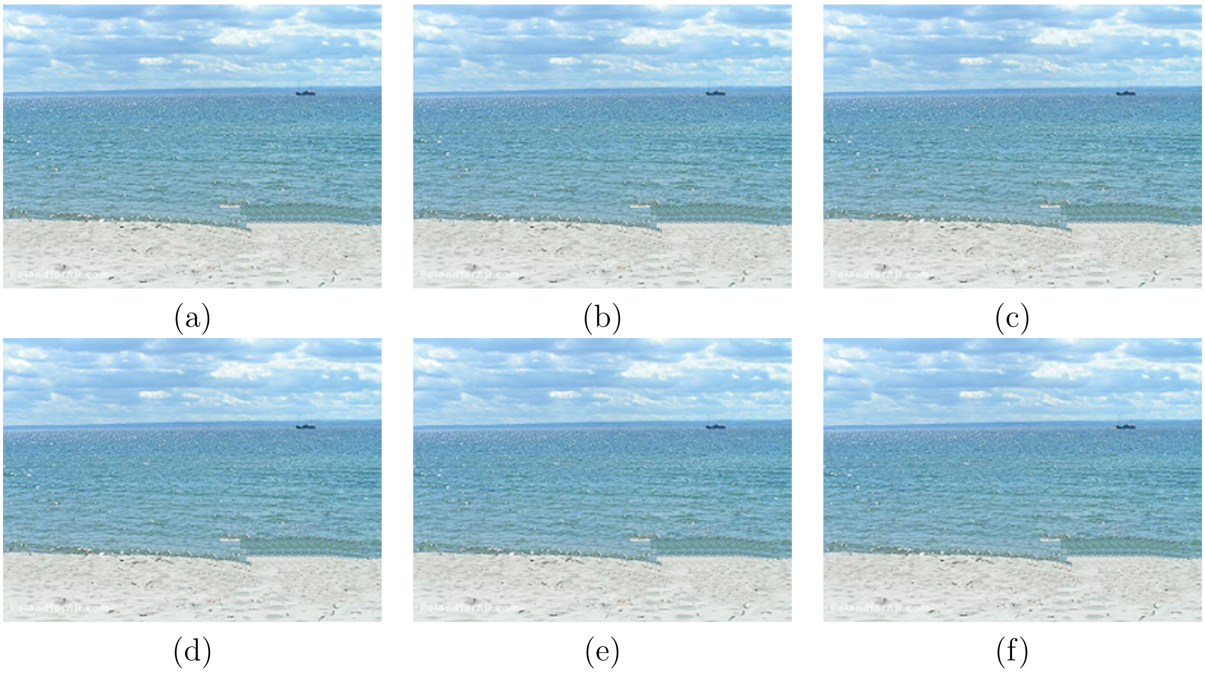


Figure 5.14. An example of an inpainting forgery. (a) The original image; an inpainted image using (b) Wu's approach [20] and (c) Zhang's approach [21].

forgery. Thus, the purpose of our work is to introduce an efficient and reliable inpainting detection method. The proposed method is designed based on the common principle used by exemplar-based inpainting algorithms. It aims at detecting whether the input image has been edited following the aforementioned principle or not. The performance of the proposed method is evaluated on a set of various natural images. We also report its robustness with regards to different inpainting techniques.

As mentioned in Chapter 1, several approaches of image inpainting have been proposed in the literature. Most of them could be categorized into two types based on the purpose [8]. The first category is geometry-oriented approaches in which the missing regions are filled by diffusing the information from the known region into the missing one. These methods are suitable for filling narrow or small area like scratches but are less efficient for large area. The second category is the texture-oriented approaches. Inspired by texture synthesis methods, these methods produce an impressive output in recovering large damaged regions. they could be further subdivided into two subgroups: exemplar-based completion methods [17, 18, 20, 21, 57] and pixel-based completion methods [26, 27, 40].

In this study, we subscribe to the exemplar-based completion methods since it is more suited for large regions restoration or hiding a whole object. In inpainting methods, the


(a) (b) (c)

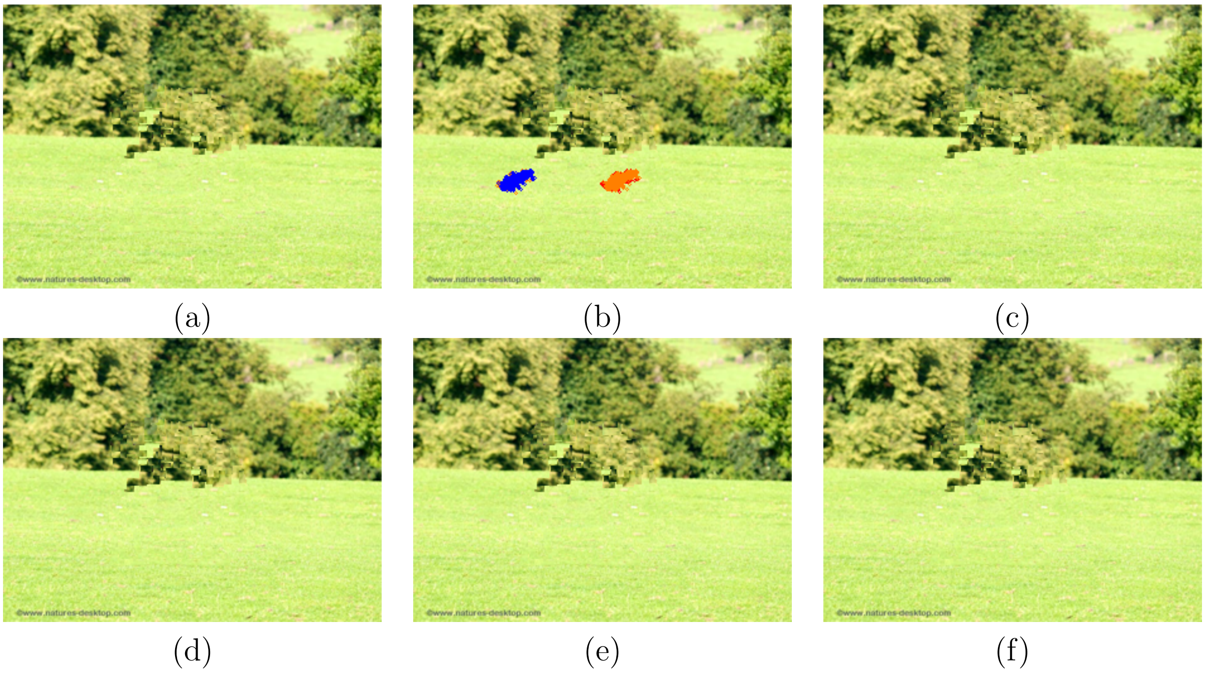
(d) (e) (f)

Figure 5.15. The detection using the copy-move forgery detection for inpainting forgery. The results when using the methods of (a) Fridrich *et al.* [94]; (b) Ryu *et al.* [101]; (c) Mahdian *et al.* [99]; (d) Wang *et al.* [100]; (e) Amerini *et al.* [108] and (f) Shivakumar *et al.* [110].

missing information is restored based on the most similar patches under a pre-defined priority. In this sense, the inpainting is similar to a copy-paste operation regarding the notion of patches. Nevertheless, as described previously, in copy-paste approaches a continuous region is duplicated and pasted into the missing one instead of having small patches coming from different parts as in the case of inpainting methods. This is why copy-move detection cannot be applied for inpainted images.

Following this observation and by analyzing principles of image inpainting algorithms in the second group, we introduce a novel approach to detect inpainted regions. The proposed algorithm can be summarized through three main steps as described below. Let consider a window centered at pixel p denoted as a patch Ψ_p. The patch size, three thresholds θ_1, θ_2 and θ_3 are global parameters for the proposed algorithm.

1. Patch matching : Since most of the patch-based inpainting algorithms rely on patch similarity analysis, the first step in our detection scheme is searching all pairs of similar patches (Ψ_p, Ψ_i) in the input image to detect the suspicious regions. A list of pair


(a) (b) (c)

(d) (e) (f)

Figure 5.16. The detection using the copy-move forgery detection for inpainting forgery. The results when using the methods of (a) Fridrich *et al.* [94]; (b) Ryu *et al.* [101]; (c) Luo *et al.* [102]; (d) Bravo *et al.* [103]; (e) Amerini *et al.* [108] and (f) Shivakumar *et al.* [110].

of patches satisfying the three following criteria, formalized by equation (5.1), is built and considered as potential candidates.

$$\begin{aligned} \exists_{i \in I} \Psi_p \simeq \Psi_i \Leftrightarrow \quad & (Sim(\Psi_p, \Psi_i) < \theta_1) \\ & \wedge (Dist(\Psi_p, \Psi_i) < \theta_2) \\ & \wedge (Card(\Psi_p \cap \Psi_i) > \theta_3) \end{aligned} \tag{5.1}$$

- The similarity between two patches, $Sim(\Psi_p, \Psi_i)$, should be less than a threshold, θ_1. This threshold is used in order to reduce the probability of false matches while preserving suitable matches.
- The distance between two similar patches, $Dist(\Psi_p, \Psi_i)$, should be greater than a threshold, θ_2. This criterion is applied to preclude nearby patches or identical patches.
- The number of the same pixels in two patches, $Card(\Psi_p \cap \Psi_i)$, should be greater than a threshold, θ_3. This constraint is introduced to ensure that at least θ_3 pixels are copied between two patches.

A selection of values for these parameters is given in the next chapter depending on the patch size. A further discussion of parameter influence is being studied and will be published in the future.

Besides, another big challenge in this step is related to minimizing the computational cost. It could take hours or even days to search similar patches satisfying equation (5.1) by brute force procedure. This is infeasible and impractical. In our algorithm, we used a very effective algorithm, the *kd*-tree [114], to get approximate nearest neighbors. Typically, the Euclidean distance is used as a similarity measure as in the inpainting methods [17, 20, 21]. In prior work [111], it has been shown that the use of *kd*-tree matching leads, in general, to better results than lexicographic sorting as in [94, 97].

2. Mask generation : A binary mask, in which pixels belonging to candidate patches are labelled as "1" and the others as "0", is generated for detecting inpainted regions. This mask is a collection of connected regions composed of pixels labelled "1" on a background of pixels labelled "0". In our experiment, we assume that only one region has been restored in the input image. In such a way, we evaluated only the largest connected area. Logically, this region is characterized by its centroid. The centroid of the largest connected region, as a result, is located based on position of all pixels belonging to this region.

3. Patch filtering : A filtering scheme is applied to reduce the false detected patches. Indeed, for each couple of matches, one of the two patches is an original and the other is the inpainted version that should be marked as forged patch. The filtering is implemented for all matches based on their distance to the calculated centroid. The patch closer to the centroid is kept and the remaining patch is discarded. The morphological operation, such as *opening*, could be applied to connect nearby regions. Finally, the largest connected region is considered as the latest output.

The detail scheme of our method is shown in Figure 5.17. As a result, an example for each step is illustrated in Figure 5.18. More experimental results will be presented in the next section.

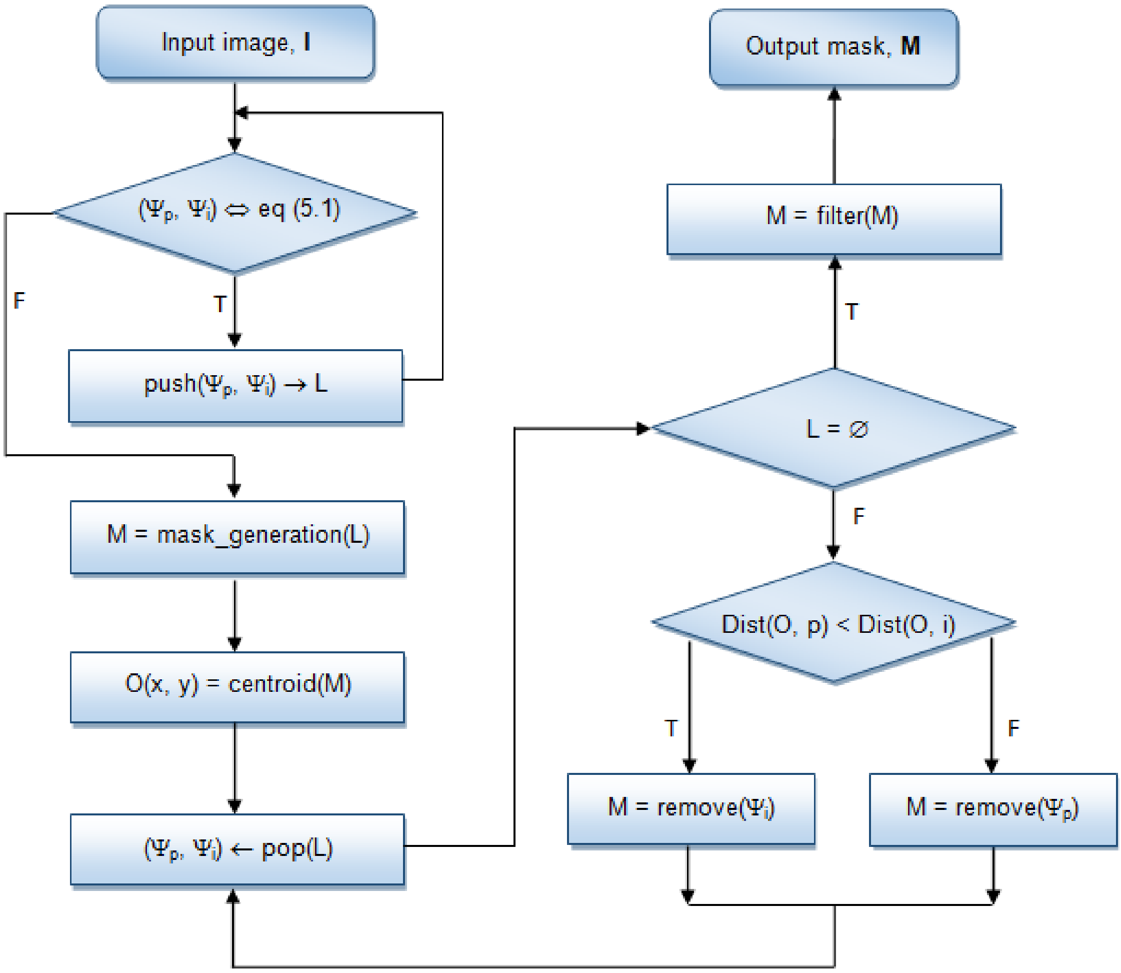


Figure 5.17. The detail scheme of our method.

5.4 Experimental results

In order to evaluate the performance of the proposed inpainting detection algorithm, a series of experiments are carried out and discussed in this section. The performance evaluation is considered comprehensively in two directions:

- *many different inpainting algorithms for one image.*
- *one inpainting algorithm for many different images.*

In addition, the ground truth images are used in comparison with the detection outputs. In consequent, the obtained results confirm our proposal for inpainting detection with visual inspection and objective measurements. The detailed discussion is given as below.

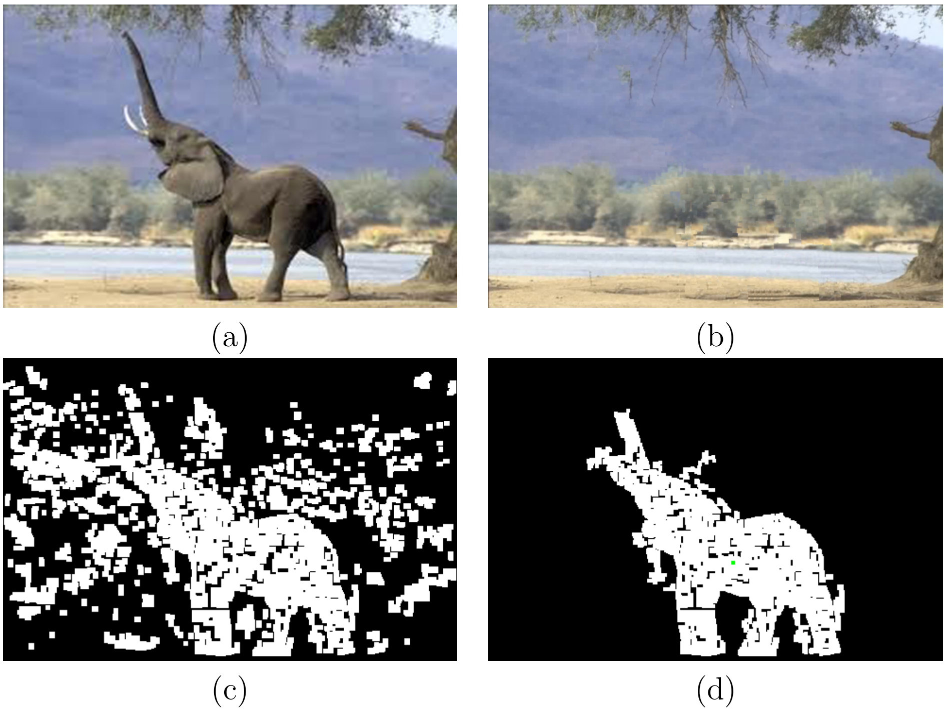


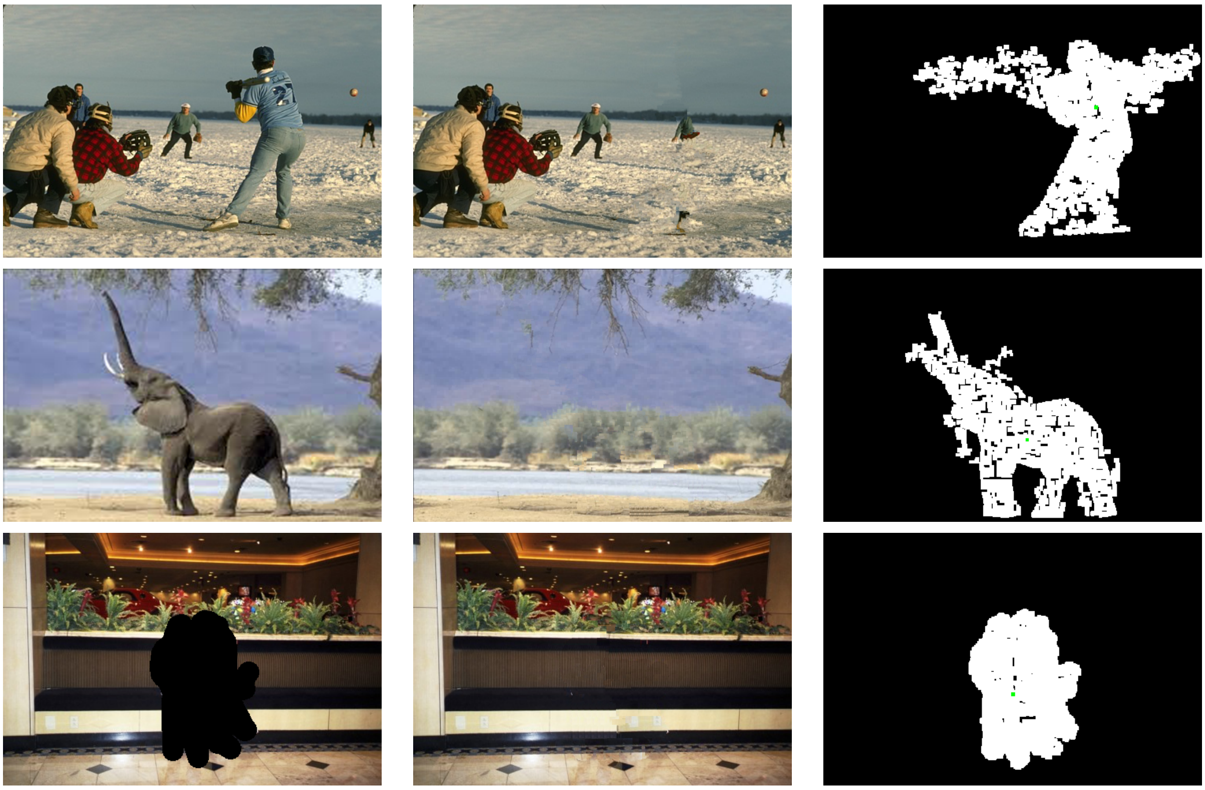
Figure 5.18. An example for each step in our algorithm. (a) original image using Criminisi's method [17]; (b) inpainted image; (c) and (d) mask images before and after filtering patches.

5.4.1 Test for inpainting forgery

We implemented the proposed detection algorithm using C language and tested on a set of natural images with various contents. The output of the algorithm is a binary mask where white pixels correspond to detected inpainted regions and black pixels are non-inpainted areas. Green patch is added to identify the centroid of the inpainted region. In order to evaluate the visual performance of our detection, an experimental scheme is applied and described as follows.

First, an original image is used as ground truth with a binary mask image identifying an area need to be inpainted. An inpainting method is applied to the latter to generate an inpainted image. Finally, the inpainted image is used as input for our detection scheme to generate the detection mask. Figure 5.19 shows some results for our detection with the inpainting method in [18].

The parameters of our algorithm have been tuned as follows: patch size $l = 5$ and thresholds $\theta_1 = 0.1$; $\theta_2 = l/2$ and $\theta_3 = l^2/3$. It can be easily noticed that the proposed approach achieves good results by localizing quite precisely the inpainted object. The results yield to a direct conclusion about the presence of forgery or not.



(a) (b) (c)

Figure 5.19. Inpainting detection results of our proposal for a set of inpainted images. (a) original images; (b) inpainted images using Criminisi's method [17]; (c) detected masks.

In order to evaluate the robustness of our method, we applied three different inpainting methods [17, 20, 21] to modify the original images. This leads to a set of inpainted images for the same input image. Figure 5.20 depicts an example for one original image (a), inpainted images using respectively methods described in [17], [20] and [21] (b) and the obtained detection masks (c). Again, the proposed approach allows detecting the presence of inpainting forgery even with the use of different inpainting techniques.

5.4.2 Performance evaluation

In order to quantify the efficiency of our inpainting forgery detection, we used objective measures namely *Precision*, *Recall* and F_1 often used for information retrieval performance

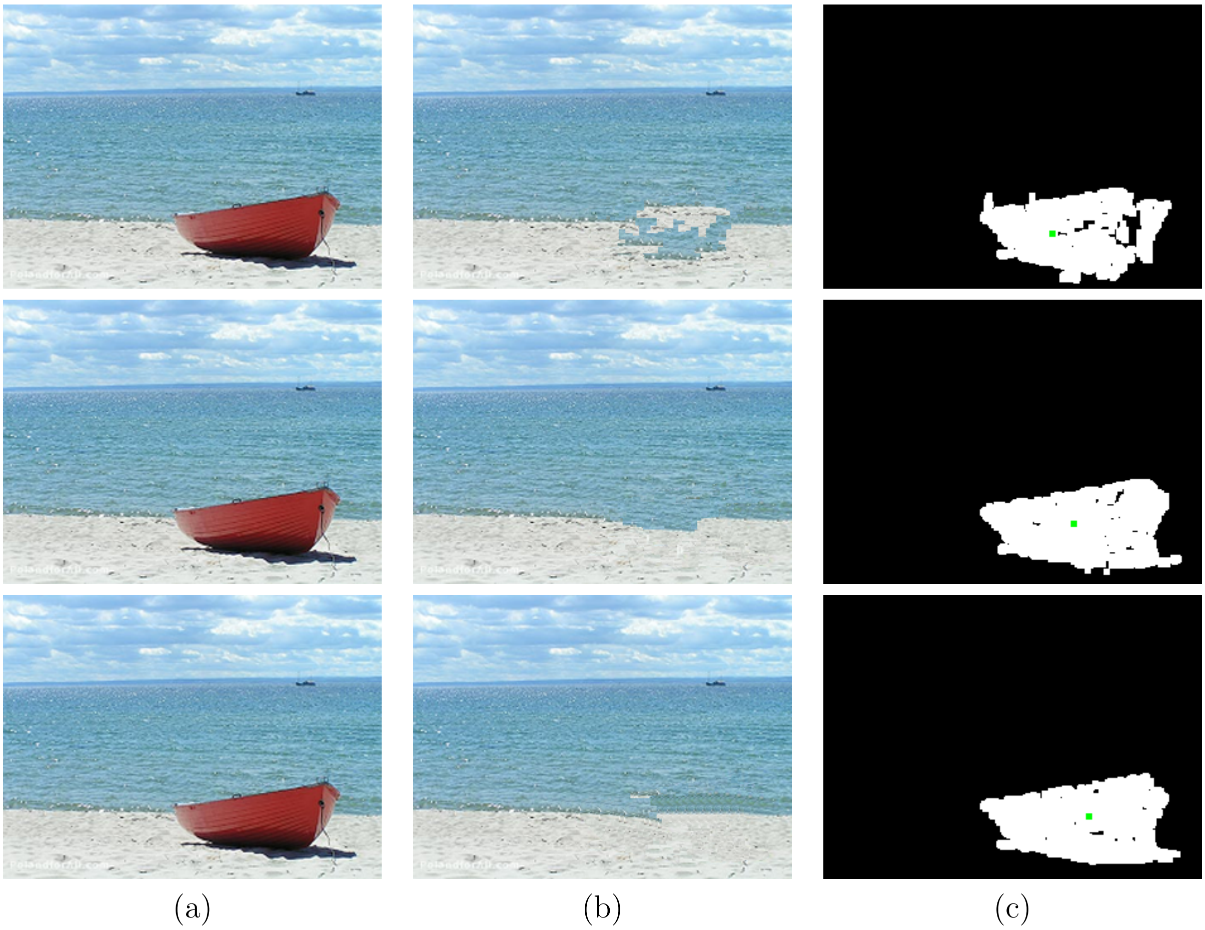


Figure 5.20. Inpainting detection results of our proposal for a set of inpainted images. (a) original images; (b) inpainted images using Criminisi's method[17], Wu's method [20] and Zhang's method [21] (top to bottom) ; (c) detected masks.

evaluation [115].

Precision and *Recall* correspond to exactness and completeness of the results. In our perspective, the *Precision* is applied to estimate the probability that a detected region is correct. This probability is defined as follows:

$$P = \frac{|I_I \cap I_D|}{|I_D|} \times 100\% \tag{5.2}$$

where I_I and I_D denote the inpainted region and detected region, respectively. The operator $|\Omega|$ counts the number of pixels in the region, Ω. Alternatively, *Recall* is used to measure the probability that a corrected region is detected. It is defined as follows:

$$R = \frac{|I_I \cap I_D|}{|I_I|} \times 100\% \tag{5.3}$$

However, there is a trade-off between *Precision* and *Recall*. Greater *Precision* might decrease *Recall* and vice versa. To consider both *Precision* and *Recall* together, the F_1 measure, the harmonic-mean of *Precision* and *Recall* is proposed and calculated as given in equation (5.4).

$$F_1 = \frac{2PR}{P+R} \tag{5.4}$$

We tested the proposed method to fifteen images. The corresponding *Precision*, *Recall* and F_1 measurements are presented in Table 5.2 and visually shown in Figure 5.20. Where, the first column shows identification of input images with the corresponding sizes in the next column, labelled with *Size*. Column IA defines the size of original inpainted area respecting to percentage of the whole input image. The columns of P, R and F_1 indicate the values of corresponding metrics. The average rates of *Precision*, *Recall* and F_1 are 78.68%, 84.49% and 80.73%, respectively. These values are nearly the same and relatively high. This shows that the detected region is not only correct but also quite complete. On the other hand, the obtained output demonstrates the important ability of the proposed method to detect inpainting forgery in the digital images. It thus confirms that our proposal is efficient for inpainting forgery detection.

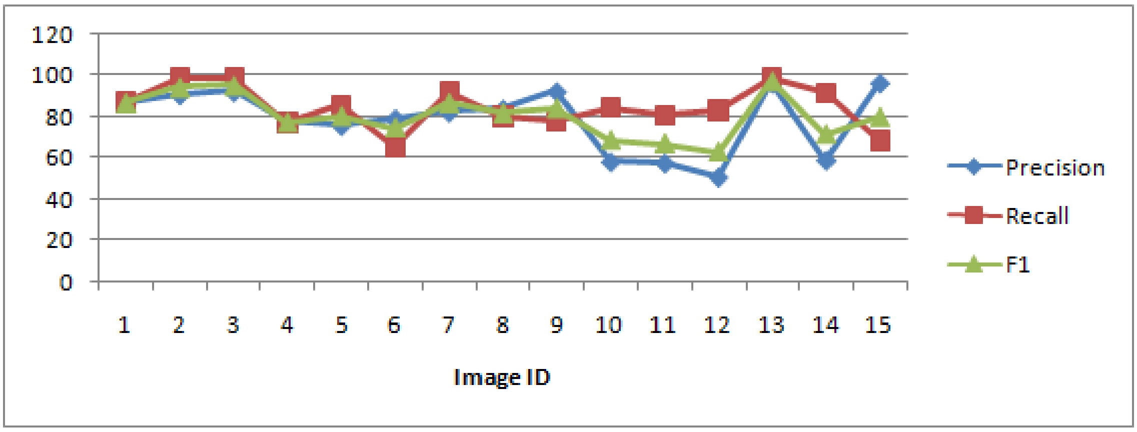


Figure 5.21. A chart of performance respect to detection rates.

Our implementation used C language based on Visual Studio platform. On a machine with an Intel Core i5 Duo 2.8GHz and 4GB memory, the average time to detect is about

Table 5.2. Detection rates for inpainting forgery.

Image ID	**Size (px)**	**IA (%)**	P (%)	R (%)	F_1 (%)
1	300 × 255	10.73	87.0401	86.85	86.94
2	300 × 255	10.73	90.80	98.47	94.48
3	300 × 255	10.73	92.28	98.77	95.41
4	206 × 308	12.60	78.02	76.93	77.47
5	206 × 308	12.60	75.90	85.58	80.45
6	206 × 308	12.60	79.10	65.09	74.87
7	300 × 255	14.47	82.51	91.93	86.97
8	300 × 255	14.47	83.80	80.08	81.90
9	300 × 255	14.47	92.34	77.80	84.45
10	300 × 218	14.87	58.11	84.14	68.74
11	300 × 218	14.87	57.50	80.49	67.08
12	300 × 218	14.87	50.71	82.79	62.89
13	480 × 320	12.97	96.59	98.52	97.55
14	481 × 321	12.56	58.91	91.43	71.66
15	480 × 320	17.19	96.57	68.43	80.10
Average			78.68	84.49	80.73

40 milliseconds. The computational time is not the same for images with the same size. It is dependent on each image's characteristics and especially on the thresholds. Please note that the computational time of an image containing large uniform or very similar areas is higher. This is caused by the fact that the most time consumption is associated with the analysis of patch similarities and neighborhoods. Logically, having a high number of similar blocks causes expensive computation. The hierarchical structure, *kd*-tree, allows us to make efficient range queries in multidimensional data. Consequently, the proposed algorithm has a low computational cost in most cases. The chart 5.22 shows the time consumption in milisecond (ms) for the set of test images.

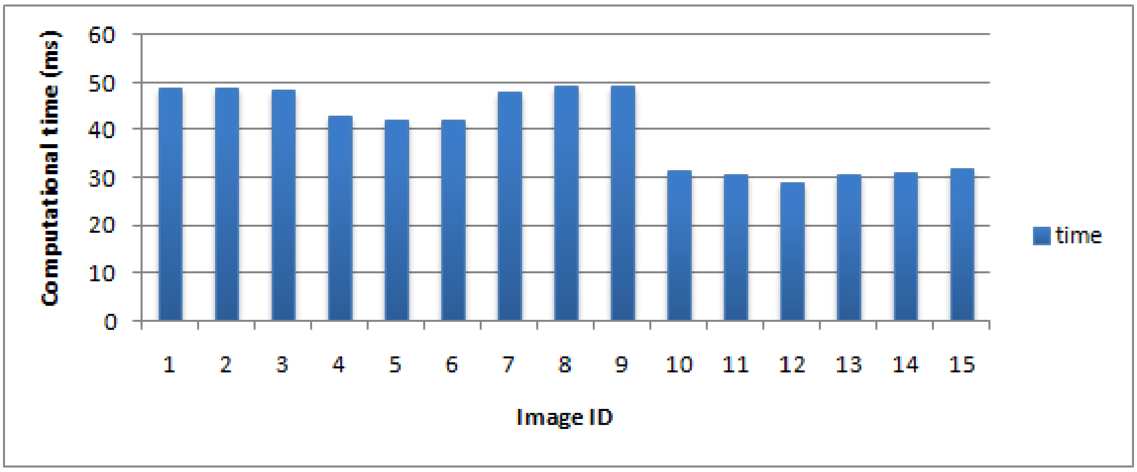


Figure 5.22. A chart of performance respect to computational time.

5.5 Summary and conclusion

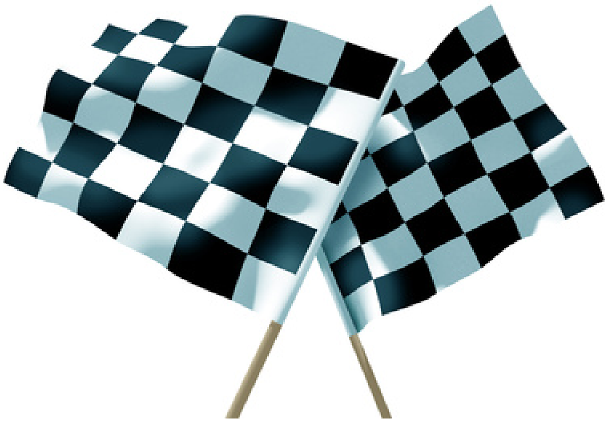
In this chapter, a notation of digital forgery was introduced. Within this field, copy-move forgery detection is probably the most actively investigated subtopic and has a particularly close connection with our problem, digital inpainting forgery. The overview of common copy-move forgery detection has been summarized and the main difference between these two problems, copy-move and inpainting forgery detections, has been indicated. Generally, the inpainting forgery detection is more difficult and sophisticated.

In this study we addressed the problem of inpainting forgery detection by proposing a novel approach relying on the behavior of most known inpainting algorithms. Therefore, based on the analysis of many exemplar-based inpainting methods, we have predicted the inpainted regions by similar patches and located it by the centroid connected component. Experimental results supported that the proposed method was appropriate to identify and localize the inpainted region with high accuracy even though the image could be modified by many different inpainting methods.

Although having achieved promising performance in detecting inpainted region, our method still contains limitation in some images. One example is shown as the top image in Figure 5.16 where the unexpected matches have been detected in the homogeneous or flat regions. This is because there are always similar patches in these kinds of region with any thresholds. Thus, it could be considered as an important future work to improve the detection performance for such case. In addition, our method is suitable to detect for exemplar-based inpainting methods but not for pixel-based inpainiting methods. A further study should be conducted to deal with such cases.

CHAPTER 6

Conclusion and Perspectives



Contents

6.1 Conclusion

The goal of the current research was to develop a framework for *digital image inpainting*. Two related trends are exploited such as *inpainting quality assessment* and *inpainting forgery detection*. The main contributions of the thesis are related to these two trends.

A specific concept of inpainting problem and the related definitions was formulated in Chapter 2. Besides, a detailed classification of existing digital image inpainting algorithms was provided. Accordingly, most of current inpainting methods can be classified into three main groups. However, because of extensive researches, it is impossible to have a common criteria for classification. Thus, there are some inpainting methods that are not included in the above groups such as multi-source image completion.

A novel framework of the second group has been proposed in Chapter 3 based on a hierarchical representation. Additionally, a window-based priority which classifies patches in a more suitable way has been also introduced and combined with the multi-resolution patch in order to ensure for more pleasant results. A comparison with all three groups of inpainting methods has been carried out with a set of natural and real archaeological images. The experimental results show that the proposed approach outperforms the methods of the state-of-the-art in regard to visual quality.

However, the proposed method suffers from some limitation mainly on computational performance side. Although, this framework produces the acceptable results in most case, it is time consuming. This is mainly due to the patch searching process. Thus, an improvement has been proposed. An offset map defining the relationship between the known pixels and unknown pixels is used instead of the multi-resolution patch for completing higher level resolution. A global optimization strategy based on graph-cut algorithm has been introduced in order to optimize the offset map by considering only some neighbors instead of an exhaustive search. The computational time of implementation has been significantly reduced. Many comparisons and analysis have been performed and discussed. They demonstrated impressive results in term of both visual quality and computational time.

Chapter 4 is devoted to the evaluation of inpainted images. Very often, the results of image inpainting are evaluated subjectively or by using some objective metrics not well adapted to the specificities of their criteria. However, the subjective evaluation experiments are time consuming, complex and unpredictable due to some uncontrolled human factors such as fatigue, visual discomfort, knowledge, etc. In this chapter, a series of traditional image quality assessment and a few existing inpainting quality assessment metrics have been investigated. Since the existing IQA metrics could not be directly applied because the specificities and goals of both image quality, in its broad sense, and image inpainting are different. Therefore, an inpainted image quality assessment is very meaningful in this case. This is the main content of this chapter. A novel approach for objective Image Inpainting Quality Assessment has been designed. The proposed metric is composed of two terms: coherence term which evaluates the consistency between the restored regions and source regions; and structure term which attracts the viewer attention. Two versions for gray-level and color image have been developed. The psychophysical experiments have been run to confirm the performance of our metrics. The obtained results showed that none of the considered metrics correlated well with perceived overall inpainting quality. However, the scores obtained from our proposal indicate a better performance when compared to some of the other existent inpainting-specific quality metrics.

The last contribution was presented in Chapter 5. In this chapter, we first defined our problem and introduced some differences with the existing forgeries. An overview of current forgery detection has been reported and many results when using existing forgery detection to detect inpainted images were shown. The demonstration pointed out an inefficiency of this approach. Many experimental results supported that the proposed methods is appropriate to identify and localize the inpainted region with high accuracy even though the image could be modified with many different inpainting algorithms.

In conclusion, the current research has *analysed*, *developed* and *evaluated* objective methods for inpainting problem. This dissertation built a complete picture of inpainting problem along with the corresponding contributions to fully address each part. A series of experimental results and comparison with existing approaches coming from literature are performed. They all confirmed the outstanding results for our proposals.

6.2 Future works

In the course of the research carried out for this thesis a number of possible directions for further research have been identified. Relating to different parts of this thesis, they can be divided into three separate issues.

For *inpainting algorithm*, despite the extensive research carried out in the field of digital image inpainting, there is still room for improvements. In the current algorithms, the patch selection is an essential step to complete the image. A perceptual patch similarity that is more stable and effective should be carefully considered. An extension of our framework could be developed for multiple source images or videos. Experimenting with high resolution images should also be considered for further research. The main issue when dealing with high resolution images as input for inpainting algorithms is to a significant increase of running time (up to several hours). However, most of the existent algorithms were not designed for high resolution images. Thus, developing new algorithms, able to inpaint high resolution images would potentially increase the quality of the inpainting. But, guaranteeing high inpainting quality would be a challenging issue.

For *inpainting quality assessment*, regarding the evaluation of inpainting quality, further work should be focused on finding a metric that does not depend on a reference image and that correlates better with perceived quality. Conducting a psychophysical experiment that considers more natural images and current inpainting algorithms, as well as quality metrics, could also significantly contribute to a better understanding of the deficiencies of current inpainting quality metrics. Identifying these deficiencies would make space for further improvements.

For *inpainting forgery detection*, although having achieved promising performance in detecting inpainted region, the detection technique could be improved further. Some parameters could be considered as threshold to reduce the unexpected matching patches in the special regions such as homogeneous or flat. Moreover, the proposed detection can apply only for inpainted images which had been edited by exemplar-based inpainting algorithms but not for pixel-based inpainting methods or interpolation-based methods. Therefore, some further studies could be developed with these directions. Finally, a forgery detection for inpainting video is also a potential and considerable direction of research.

List of Tables

List of Figures

Bibliography

[1] M. R. Banham and A. K. Katsaggelos, "Digital image restoration," *IEEE Signal Processing Magazine*, vol. 14, pp. 24 – 41, 1997.

[2] Z. K. Liu and J. Y. Xiao, "Restoration of blurred tv picture caused by uniform linear motion.," *Computer Vision, Graphics, and Image Processing*, vol. 44, no. 1, pp. 30–34, 1988.

[3] C. H. Slump, "Real-time image restoration in diagnostic x-ray imaging, the effects on quantum noise," *Proc, 11th IAPR Int. Conf. on Pattern Recognition*, vol. II, pp. 693 – 696, 1992.

[4] G. Aubert and P. Kornprobst, *Mathematical problems in image processing: partial differential equations and the calculus of variations (second edition)*, vol. 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2006.

[5] T. F. Chan and J. Shen, "Non-texture inpainting by curvature-driven diffusions (cdd)," *Journal of Visual Communication and Image Representation*, vol. 12, pp. 436–449, 2001.

[6] M. Bertalmío, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, (New York, NY, USA), pp. 417–424, ACM Press/Addison-Wesley Publishing Co., 2000.

[7] C. Guillemot and O. Le Meur, "Image inpainting: Overview and recent advances," *IEEE Signal Processing Magazine*, pp. 127–144, Jan. 2014.

[8] P. Arias, G. Facciolo, V. Caselles, and G. Sapiro, "A variational framework for exemplar-based image inpainting," *Int. J. Comput. Vision*, vol. 93, pp. 319–347, July 2011.

[9] J. M. Ogden, E. H. Adelson, J. R. Bergen, and P. Burt, "Pyramid-based computer graphics," 1985.

[10] M. Bertalmío, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *CVPR (1)*, pp. 355–362, IEEE Computer Society, 2001.

[11] S. Esedoglu and J. Shen, "Digital inpainting based on the mumford-shah-euler image model," *European J. Appl. Math*, vol. 13, pp. 353–370, 2002.

[12] D. Tschumperlé, "Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 65–82, 2006.

[13] Y.-R. Li, L. Shen, and B. W. Suter, "Adaptive inpainting algorithm based on dct induced wavelet regularization," *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 752–763, 2013.

[14] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling," in *In International Conference on Computer Vision*, vol. 2, pp. 1033–1038, 1999.

[15] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, (New York, NY, USA), pp. 341–346, ACM, 2001.

[16] M. Ashikhmin, "Synthesizing natural textures," in *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, I3D '01, (New York, NY, USA), pp. 217–226, ACM, 2001.

[17] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *Trans. Img. Proc.*, vol. 13, pp. 1200–1212, Sept. 2004.

[18] W. Cheng, C. Hsieh, S. Lin, C. Wang, and J. Wu, “Robust algorithm for exemplar-based image inpainting,” *Proceeding of International Conference on Computer Graphics, Imaging and Visualization. IEEE*, pp. 64 – 69, 2005.

[19] Anupam, P. Goyal, and S. Diwakar, “Fast and enhanced algorithm for exemplar based image inpainting,” *Image and Video Technology, Pacific-Rim Symposium on*, vol. 0, pp. 325–330, 2010.

[20] J. Wu and Q. Ruan, “Object removal by cross isophotes exemplar-based inpainting,” in *Proceedings of the 18th International Conference on Pattern Recognition - Volume 03*, ICPR ’06, (Washington, DC, USA), pp. 810–813, IEEE Computer Society, 2006.

[21] Q. Zhang and J. Lin, “Exemplar-based image inpainting using color distribution analysis,” *J. Inf. Sci. Eng.*, vol. 28, no. 4, pp. 641–654, 2012.

[22] J. Mairal, J. Mairal, M. Elad, M. Elad, G. Sapiro, and G. Sapiro, “Sparse representation for color image restoration,” in *the IEEE Trans. on Image Processing*, vol. 17, pp. 53–69, ITIP, 2007.

[23] H. Mobahi, S. R. Rao, and Y. Ma, “Data-driven image completion by image patch subspaces,” *Proceedings of the Picture Coding Symposium*, pp. 1 – 4, 2009.

[24] N. Komodakis, “Image completion using global optimization,” in *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR ’06, (Washington, DC, USA), pp. 442–452, IEEE Computer Society, 2006.

[25] O. Le Meur, M. Ebdelli, and C. Guillemot, “Hierarchical super-resolution-based inpainting,” *IEEE Transactions on Image Processing*, vol. 22, pp. 3779–3790, May 2013.

[26] Y. Wexler, E. Shechtman, and M. Irani, “Space-time completion of video,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 463–476, Mar. 2007.

[27] Y. Pritch, E. Kav-Venaki, and S. Peleg, “Shift-map image editing,” in *ICCV’09*, (Kyoto), pp. 151–158, Sept 2009.

[28] Y. Liu and V. Caselles, "Exemplar-based image inpainting using multiscale graph cuts," *IEEE Transactions on Image Processing*, vol. 22, no. 5, pp. 1699–1711, 2013.

[29] M. Bertalmío, L. A. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting.," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 882–889, 2003.

[30] K. Sangeetha, P. Sengottuvelan, and E. Balamurugan, "Combined structure and texture image inpainting algorithm for natural scene image completion," *Journal of Information Engineering and Applications*, vol. 1, pp. 7–12, 2011.

[31] V. Sairam, R. R. Sarma, S. Balasubramanian, and A. Hareesh, "A unified framework for geometry and exemplar based image inpainting," *IEEE Second International Conference on Image Information Processing (ICIIP)*, pp. 511 – 515, 2013.

[32] J. Hays and A. A. Efros, "Scene completion using millions of photographs," *ACM Transactions on Graphics (SIGGRAPH 2007)*, vol. 26, no. 3, 2007.

[33] O. Whyte, J. Sivic, and A. Zisserman, "Get out of my picture! internet-based inpainting," *Proceedings of the 20th British Machine Vision Conference*, 2009.

[34] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, pp. 313–318, July 2003.

[35] T. F. Chan and J. Shen, "Mathematical models for local nontexture inpaintings," *SIAM Journal on Applied Mathematics*, vol. 62, pp. 1019–1043, 2002.

[36] E. De Giorgi, "Some remarks on Γ-convergence and least square methods," in *Composite Media and Homogenization Theory* (G. D. Maso and G. Dell'Antonio, eds.), pp. 135–142, Birkhaüser, Boston, 1991.

[37] M. Röger and R. Schätzle, "On a modified conjecture of de giorgi.," *Math. Z.*, vol. 254, no. 4, pp. 675–714, 2006.

[38] N. U. Ahmed and K. R. Rao, *Orthogonal transforms for digital signal processing.* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1975.

[39] J. Jia and C.-K. Tang, "Image repairing: robust image synthesis by adaptive nd tensor voting," in *CVPR (1)*, pp. 643–650, IEEE Computer Society, 2003.

[40] J. Sun, L. Yuan, J. Jia, and H.-Y. Shum, "Image completion with structure propagation," *ACM Trans. Graph.*, vol. 24, pp. 861–868, July 2005.

[41] G. T. N. Komodakis and G. Tziritas, "Image completion using efficient belief propagation via priority scheduling and dynamic pruning," *IEEE Trans. on Image Processing*, vol. 16, no. 11, pp. 2649–2661, 2007.

[42] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen, "Interactive digital photomontage," *ACM Trans. Graph.*, vol. 23, pp. 294–302, Aug. 2004.

[43] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *in Proc. 8th Intl Conf. Computer Vision*, pp. 416–423, 2001.

[44] J. J. McAuley, T. S. Caetano, A. J. Smola, and M. O. Franz, "Learning high-order mrf priors of color images.," in *ICML* (W. W. Cohen and A. Moore, eds.), vol. 148 of *ACM International Conference Proceeding Series*, pp. 617–624, ACM, 2006.

[45] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra, "Texture optimization for example-based synthesis," *ACM Trans. Graph.*, vol. 24, pp. 795–802, July 2005.

[46] O. Le Meur and C. Guillemot, "Super-resolution-based inpainting," in *Proceedings of the 12th European Conference on Computer Vision - Volume Part VI*, ECCV'12, (Berlin, Heidelberg), pp. 554–567, Springer-Verlag, 2012.

[47] M. Granados, J. Tompkin, K. Kim, O. Grau, J. Kautz, and C. Theobalt, "How not to be seen - object removal from videos of crowded scenes," *Comp. Graph. Forum*, vol. 31, pp. 219–228, May 2012.

[48] D. Comaniciu, P. Meer, and S. Member, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 603–619, 2002.

[49] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference.* San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[50] S. Di Zenzo, "A note on the gradient of a multi-image," *Comput. Vision Graph. Image Process.*, vol. 33, pp. 116–125, Jan. 1986.

[51] Z. Xu and J. Sun, "Image inpainting by patch propagation using patch sparsity," *Trans. Img. Proc.*, vol. 19, pp. 1153–1165, May 2010.

[52] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, pp. 1124–1137, Sept. 2004.

[53] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, pp. 1222–1239, Nov. 2001.

[54] L. A. Vese and S. Osher, "Modeling textures with total variation minimization and oscillating patterns in image processing," *J. Sci. Comput.*, vol. 19, no. 1-3, pp. 553–572, 2003.

[55] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, (Washington, DC, USA), pp. 273–280, IEEE Computer Society, 2003.

[56] A. Oliva and A. Torralba, "Building the gist of a scene: the role of global image features in recognition," *Visual Perception, Progress in Brain Research*, vol. 155, 2006.

[57] A. Wong and J. Orchard, "A nonlocal-means approach to exemplar-based inpainting.," in *ICIP*, pp. 2600–2603, IEEE, 2008.

[58] P. J. Burt, Edward, and E. H. Adelson, "The laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, pp. 532–540, 1983.

[59] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proc. of Fourth Alvey Vision Conference*, pp. 147–151, 1988.

[60] P. Beaudet, "Rotationally invariant image operators," *International Joint Conference on Pattern Recognition*, vol. 579, pp. 579 – 583, 1978.

[61] T. T. Dang, C. Larabi, and A. Beghdadi, “Multi-resolution patch and window-based priority for digital image inpainting problem,” *the 3rd International Conference on Image Processing Theory, Tools and Applications*, pp. 280 – 284, 2012.

[62] T. T. Dang, A. Beghdadi, and M. C. Larabi, “Archaeological image inpainting,” *the 4th European Workshop on Visual Information Processing*, 2013.

[63] Z. Wang and A. C. Bovik, *Modern image quality assessment.* Morgan & Claypool, 2006.

[64] S. Daly, “Digital images and human vision,” in *Digital Images and Human Vision* (A. B. Watson, ed.), ch. The Visible Differences Predictor: An Algorithm for the Assessment of Image Fidelity, pp. 179–206, Cambridge, MA, USA: MIT Press, 1993.

[65] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 13, no. 4, pp. 600–612, 2004.

[66] H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” in *IEEE Trans. Image Processing*, vol. 15 (2), pp. 430–444, 2004.

[67] Y. Shi, Y. Ding, R. Zhang, and J. Li, “Structure and hue similarity for color image quality assessment,” *Electronic Computer Technology, International Conference on*, vol. 0, pp. 329–333, 2009.

[68] N. Bonnier, F. Schmitt, H. Brettel, and S. Berche, “Evaluation of spatial gamut mapping algorithms,” *Proceeding of the 14th Color Imaging Conference*, pp. 56–61, 2006.

[69] D. M. Chandler and S. S. Hemami, “Vsnr: A wavelet-based visual signal-to-noise ratio for natural images,” *Trans. Img. Proc.*, vol. 16, pp. 2284–2298, Sept. 2007.

[70] S. Wang, H. Li, X. Zhu, and P. Li, “An evaluation index based on parameter weight for image inpainting quality,” in *ICYCS*, pp. 786–790, IEEE Computer Society, 2008.

[71] P. Ardis and A. Singhal, “Visual salience metric for image inpainting,” *Proceedings of the SPIE*, vol. 7257, 2009.

[72] V. V. Mahalingam, *Digital inpainting algorithms and evaluation.* PhD thesis, University of Kentucky, 2010.

[73] A. I. Oncu, F. Deger, and J. Y. Hardeberg, "Evaluation of digital inpainting quality in the context of artwork restoration," in *ECCV Workshops (1)* (A. Fusiello, V. Murino, and R. Cucchiara, eds.), vol. 7583 of *Lecture Notes in Computer Science*, pp. 561–570, Springer, 2012.

[74] F. Ebner and M. D. Fairchild, "Development and testing of a color space (ipt) with improved hue uniformity," *6th IST/SID Color Imaging Conference*, pp. 8–13, 1998.

[75] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on Image Processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[76] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multiscale transforms," *IEEE Trans. Inf. Theor.*, vol. 38, pp. 587–607, Sept. 2006.

[77] H. R. Sheikh and A. C. Bovik, "A visual information fidelity approach to video quality assessment," *IEEE Transaction on Image Processing*, vol. 15, pp. 430–444, 2006.

[78] Q. Chen, Y. Zhang, and Y. Liu, "Image inpainting with improved exemplar-based approach," in *Proceedings of the 2007 International Conference on Multimedia Content Analysis and Mining*, MCAM'07, (Berlin, Heidelberg), pp. 242–251, Springer-Verlag, 2007.

[79] M. V. Venkatesh and S.-C. S. Cheung, "Eye tracking based perceptual image inpainting quality analysis.," in *ICIP*, pp. 1109–1112, IEEE, 2010.

[80] T. T. Dang, A. Beghdadi, and M.-C. Larabi, "Inpainted image quality assessment.," in *EUVIP*, pp. 76–81, IEEE, 2013.

[81] T. T. Dang, A. Beghdadi, and M.-C. Larabi, "Perceptual quality assessment for color image inpainting," *IEEE International Conference on Image Processing*, pp. 398–402, 2013.

[82] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, pp. 1254–1259, Nov. 1998.

[83] X. Hou and L. Zhang, "Saliency detection: A spectral residual approach," in *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR07). IEEE Computer Society*, pp. 1–8, 2007.

[84] C. Guo, Q. Ma, and L. Zhang, "Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform," *2013 IEEE Conference on Computer Vision and Pattern Recognition*, vol. 0, pp. 1–8, 2008.

[85] R. Achanta, S. Hemami, F. Estrada, and S. Ssstrunk, "Frequency-tuned salient region detection," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 1597 – 1604, 2009. For code and supplementary material, click on the url below.

[86] A. Bugeau and M. Bertalmío, "Combining texture synthesis and diffusion for image inpainting," in *VISAPP (1)* (A. Ranchordas and H. Arajo, eds.), pp. 26–33, INSTICC Press, 2009.

[87] I.-T. R. P.800, "Methods for objective and subjective assessment of quality," tech. rep., International Telecommunication Union., 1996.

[88] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment," tech. rep., http://www.vqeg.org, 2000.

[89] S. Huck, *Reading statistics and research.* Longman, 2000.

[90] I.-R. R. BT.500-10, "Methodology for the subjective assessment of the quality of television pictures," tech. rep., ITU, Geneva, Switzerland, 2000.

[91] T. T. Dang, A. Beghdadi, and M.-C. Larabi, "A hierarchical approach for high-quality and fast image completion," in *KSE (1)* (V. N. Huynh, T. Denoeux, D. H. Tran, A. C. Le, and S. B. Pham, eds.), vol. 244 of *Advances in Intelligent Systems and Computing*, pp. 11–21, Springer, 2013.

[92] T. N. Y. Times, "In an iranian image, a missile too many," July 2008.

[93] H. Farid, "Image forgery detection," *IEEE Signal Processing Magazine*, vol. 26, pp. 16–25, 2009.

[94] J. Fridrich, D. Soukal, and J. Lukás, "Detection of copy move forgery in digital images," in *Digital Forensic Research Workshop*, Aug. 2003.

[95] S. Bayram, H. T. Sencar, and N. D. Memon, "An efficient and robust method for detecting copy-move forgery.," in *ICASSP*, pp. 1053–1056, IEEE, 2009.

[96] M. Bashar, K. Noda, N. Ohnishi, and K. Mori, "Exploring duplicated regions in natural images," *IEEE Transactions on Image Processing*, vol. PP, p. 1, March 2010.

[97] A. C. Popescu and F. Hany, "Exposing digital forgeries by detecting duplicated image regions," tech. rep., Department of Computer Science, 2004.

[98] X. Kang and S. Wei, "Identifying tampered regions using singular value decomposition in digital image forensics," in *CSSE (3)*, vol. 3, pp. 926 – 930, IEEE Computer Society, 2008.

[99] B. Mahdian and S. Saic, "Detection of copy-move forgery using a method based on blur moment invariants," *Forensic Science International*, vol. 17, no. 2, pp. 180 – 189, 2007.

[100] J. Wang, G. Liu, Z. Zhang, Y. Dai, and Z. Wang, "Fast and robust forensics for image region-duplication forgery," *Acta Automatica Sinica*, vol. 35, no. 12, pp. 1488–1495, 2009.

[101] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using zernike moments," in *Proceedings of the 12th International Conference on Information Hiding*, IH'10, (Berlin, Heidelberg), pp. 51–65, Springer-Verlag, 2010.

[102] W. Luo, J. Huang, and G. Qiu, "Robust detection of region-duplication forgery in digital image," in *ICPR (4)*, pp. 746 – 749, 2006.

[103] S. Bravo-Solorio and A. K. Nandi, "Exposing duplicated regions affected by reflection, rotation and scaling.," in *ICASSP*, pp. 1880 – 1883, IEEE, 2011.

[104] H.-J. Lin, C.-W. Wang, and Y.-T. Kao, "Fast copy-move forgery detection," *WSEAS Trans. Sig. Proc.*, vol. 5, pp. 188 – 197, May 2009.

[105] J. Wang, G. Liu, H. Li, Y. Dai, and Z. Wang, "Detection of image region duplication forgery using model with circle block," in *Proceedings of the 2009 International Conference on Multimedia Information Networking and Security - Volume 01*, MINES '09, (Washington, DC, USA), pp. 25 – 29, IEEE Computer Society, 2009.

[106] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using sift algorithm.," in *PACIIA (2)*, pp. 272–276, IEEE Computer Society, 2008.

[107] X. Pan and S. Lyu, "Region duplication detection using image feature matching," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 857 – 867, 2010.

[108] I. Amerini, L. Ballan, R. Caldelli, A. D. Bimbo, and G. Serra, "A sift-based forensic method for copy-move attack detection and transformation recovery," *IEEE Trans. on Information Forensics and Sec.*, vol. 6, pp. 1099 – 1110, Sep. 2011.

[109] X. Bo, W. Junwen, L. Guangjie, and D. Yuewei, "Image copy-move forgery detection based on surf," in *Proceedings of the 2010 International Conference on Multimedia Information Networking and Security*, MINES '10, (Washington, DC, USA), pp. 889–892, IEEE Computer Society, 2010.

[110] B. L. Shivakumar and L. D. S. S. Baboo, "Detection of region duplication forgery in digital images using surf," *IJCSI International Journal of Computer Science Issues*, vol. 8, pp. 199 – 205, July 2011.

[111] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1841–1854, 2012.

[112] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91 – 110, Nov. 2004.

[113] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Underst.*, vol. 110, pp. 346–359, June 2008.

[114] J. S. Beis and D. G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," in *Proceedings of the 1997 Conference on Com-*

puter Vision and Pattern Recognition (CVPR '97), CVPR '97, (Washington, DC, USA), pp. 1000–1006, IEEE Computer Society, 1997.

[115] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval.* New York, NY, USA: Cambridge University Press, 2008.