Structure Aware Image Repair
as an Image Structure Completion and Shape Recognition Problem
and its Application to Shadow Removal

by

LIU Bowen

A Thesis Submitted to

The Education University of Hong Kong

in Partial Fulfillment of the Requirement for

The Degree of Doctor of Philosophy

March 2019

Statement of Originality

I, LIU Bowen, hereby declare that I am the sole author of the thesis and the material presented in this thesis is my original work except those indicated in the acknowledgment. I further declare that I have followed the University's policies and regulations on Academic Honesty, Copyright and Plagiarism in writing the thesis and no material in this thesis has been submitted for a degree in this or other universities.

Abstract

This study aims to develop a robust image completion method with enhanced structure-preservation ability and an in-depth understanding of the contents of images. Current patch-based image completion methods transfer information from the known region into the hole. This study enhanced the transfer process with a system that can capture multiscale structures and expand the information sources with an object database and an efficient retrieval method. The technique developed in this study can be applied to other image editing tasks that required the transfer of image information. This study has three main contributions.

This study started with improving the structure information transfer in the repair process. The first contribution focuses on the reconstruction of structures in the process of repairing damaged images. We designed a Dynamic Patch System (DPS) for implicit enhancement of the structure preservation ability of the patch-based image completion framework. It enables the use of adjustable patches to capture various scale structures in images. In addition, this DPS attempted to balance the computational workload in various image pyramid levels. Our approach and previous methods are applied to damaged images with complex structures. The results show that our approach can repair images with decent and connected structures with better run-time performance.

The image completion is essentially a process of information transfer. Image editing tasks that can be abstracted as an information transfer process can be converted into an image completion task. The second contribution is the adoption of the image completion approach with the DPS to the task of shadow removal. We propose a new perspective on the task of shadow removal in which shadowed images are impaired in their illumination fields. Based on this perspective, the task of shadow removal can be converted to a task of image completion. We first decomposed the shadowed images into their illumination and reflectance, which are processed separately to avoid interference. The illumination is repaired with a patch-based "search and lighten" iteration to transfer light information from the lit region to the shadow region. The reflectance is optimized with our image completion approach using the DPS. The repaired illumination and optimized reflectance are then combined to generate shadow-free images. The results generated with this approach display better color consistency than those generated with

previous methods and are considered more visually pleasing in a user study.

Previous image completion methods assume that information transfer within the image is sufficient to repair the damaged region. But this assumption may be violated when unique structures exist, such as object contours that cannot be duplicated in many situations. The third contribution of this study is the introduction of a novel shape descriptor called Directed Chords Pattern (DCP) to repair object contours by referring to an external shape database. Our DCP shape descriptor uses the chord distribution at each sampling point to extract a shape's features and provides a similarity metric between shapes. Our shape matching method with DCP achieves an accuracy of 88.67% in benchmark testing with the MPEG-7 dataset and is competitive with previous methods while allowing a more flexible computational workload. We provide examples in which we repair horses in images. Equipped with the Weizmann horse database, our approach can generate reasonable structures for damaged horse contours according to the estimated transformation based on the DCP, which is challenging with existing completion methods.

The image-repair technique presented in this study has enhanced structure-preservation ability implicitly with DPS and explicitly with DCP. The application scene of DPS is extended to the task of shadow removal. The DPS may be adapted to arbitrary patches other than square. The DCP shape descriptor proposed in this study can be further improved and applied to other tasks in the future. For example, it can be applied to the task of gesture recognition.

Keywords: Visual Completion, Image Completion, Structure Preservation, Dynamic Patch System, Directed Chords Pattern, Shadow Removal, Color Consistency.

To my parents, for their love and support.

Acknowledgments

I would first like to express my sincere gratitude to my principal supervisor, Prof. Kong Siu Cheung for his continuous support of my PhD study and related research. I was deeply impressed by his attitude in academic pursuits. His guidance helped me throughout my PhD study, and he backed me up when I was having a difficult time. This thesis would not have been possible without his support. My sincere thanks also go to Prof. Lo Sing Kai, who gave me helpful and practical advice on study and career management and offered me opportunities to study aboard and broaden my horizons.

I would also like to thank my colleagues for their continued support. I would like to thank Ms. Hui Ching Wan Wendy, Ms. Cheung Wai Yin Nikita, and Ms. Kam Lok Sze Iris for daily support and arrangement of my work station. I would also like to thank Mr. Lai Wai Ming and Ms. Ma Yunsi Tina for their collaboration and advice during my study. I would also like to thank my other colleagues for making my experience at EdUHK exciting and fun.

I would like to thank the staff in graduate school for their support in my study. I would like to thank Ms. NG, Oi Ting Sarah for her help in various kinds of application. I would like to thank Ms. LAI, Tsz Yan Yannis for her support in my viva examination. I would like to thank Ms. CHUNG, Yin Tung Crystal for her helpful and patient illustrations. I would also like to thank Ms. LI, Yik Man Rachel and Ms. WONG Sui Ling Charlotte for their support at the beginning of my study.

Last but not least, I would like to express my deepest gratitude to my family and friends. I am truly grateful to my parents for their love and support of my PhD study and the pursuit of my goals. I would like to thank Ms. Ching Po Ying Alisa for her company and for providing me with mental support during difficult times. I would like to thank Dr. Li Yu Peng Ivan for his professional suggestions and discussions. I would also like to thank my roommate Dr. Wang Mo, and Dr. Xu Kun for making my hall life fun.

Table of Contents

List of Abbreviations

AHF Arch Height Function. 20

AHP Arch Height Pattern. 80–82, 84

ANOVA ANalysis Of VAriance. 47, 73

AP Angular Pattern. 80–82, 84

BAS Beam Angle Statistic. 22, 76

CDD Curvature Drive Diffusion. 11

CPD Coherent Point Drift. 88, 89, 91

CSS Curvature Scale Space. 81

CUDA Compute Unified Device Architecture. 32

DCP Directed Chords Pattern. ii, iii, 6, 9, 77, 79–82, 84, 89–92, 99–102, 104, 117

DCS Directed Chords Sequence. 78, 79

DPS Dynamic Patch System. ii, iii, 8, 23, 26, 33, 39, 46, 47, 101, 102, 117

DTW Dynamic Time Warping. 86, 87

DW-SSD Distance Weighted Sum of Squared Difference. 58

EM Expectation-Maximization. 90, 117

FastDTW Fast Dynamic Time Warping. 88

GMM Gaussian Mixture Model. 77, 88, 89, 99, 117

GPGPU General-Purpose computing on Graphics Processing Unit. 32

GPU Graphics Processing Unit. 32, 42, 69

HVS Human Vision System. 1, 27

IDSC Inner Distance Shape Context. 22, 76

LAP Linear Assignment Problem. 86

LBP Local Binary Pattern. 22, 58, 59

LoG Laplacian of Gaussian. 59

LSD Local Shape Descriptor. 22, 23

MAP Maximum a Posteriori. 55

MD Mean Difference. 47, 73

MRF Markov Random Field. 11, 16

PDE Partial Differential Equation. 2, 4, 10, 11, 25, 96

QAP Quadratic Assignment Problem. 87

RMSE Rooted Mean Square Error. 63, 67

SC Shape Context. 21, 22, 76

SSD Sum of Squared Difference. 29, 30, 39, 59

TV Total Variation. 11

List of Figure

List of Tables

1 Introduction

The visual system is one of the most important parts of the central nervous system. Visual perception is thought to be an advancing function in biological evolution. Reflected light is captured by optical structures (such as the eyes and retina), and organisms receive visual information by processing these optical signals into electrical signals. Organisms can use this visual information to accomplish various kinds of complex tasks in physical environments, such as sensing the surroundings and objects, estimating the distance between viewed objects and guiding body motions. The human brain processes visual information efficiently for learning and recognition, and it can reason based on visual information and make interpretations based on visual perception. Although our knowledge of the mechanism of the human brain and Human Vision System (HVS) is incomplete, research shows that the process of visual perception includes the reconstruction of objects. One of the clearest and most convincing pieces of evidence lies in the visual completion, which refers to the perceptual filling of the occluded parts of objects. Figure 1 provides several examples of this phenomenon. Figure 1(a) is the famous Kanizsa triangle. Two triangles are perceived, although only line segments and incomplete circles are explicitly shown in the visible region.

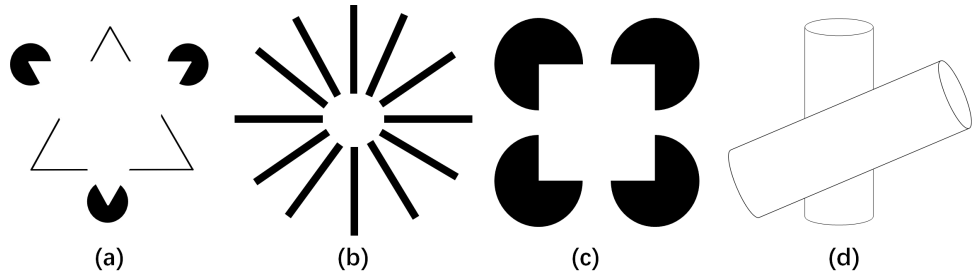

Figure 1: *Examples of the visual completion ability of the human vision system. (a) Kanizsa triangle. (b) Ehrenstein illusion of a bright disc. (c) A square can be viewed in the center. (d) Two cylinders can be recognized even though one of them is occluded.*

The visual completion ability of our vision system plays an important role in object recognition. The light reflected from objects is partially captured by our eyes because we are only able to perceive an object from a single view, which suggests that only part of an object's information is obtained. However, given only partial information of the object (for example, the

silhouette or only part of it), human beings can easily perceive objects as a complete whole.

Because visual completion is critical in the process of recognition, computer scientists have devoted considerable effort to simulate this function in computer vision systems to pursue high intelligence. Many studies in the field of computer vision are conducted on the topic of repairing damaged images. Figure 2(b) is a typical example of image completion. Given an image with a damaged area, the completion task is to repair the cavity based on the information that remains in the image. Most current systems formulate the image completion problem into a numerical inference using low-level visual cues. Some studies describe the image with Partial Differential Equation (PDE) and fill in the damaged area by allowing the propagation of pixel values into the hole. Others try to repair the hole by copying and pasting patches from the existing region to reproduce the cavity's contents. However, most of these techniques rely heavily on the consistency between patches or pixel values without a deeper understanding of the scenes and objects. In fact, when conducting completion, understanding the objects and structures in the image is of vital importance. Figure 2 presents an attempt to repair the back of a rabbit. One can hardly determine the type of the curve for repair without recognizing it as part of a rabbit. We should go further by recognizing objects and capturing structures in the repair process. Such a "repair after recognition" in fact follows a similar idea with the figural familiarity theory (Palmer, 1999), which is an explanation for the phenomenon of visual completion. The figural familiarity theory suggests that people complete partly occluded figures according to the most frequently encountered shape that is compatible with the visible stimulus information. It suggests that when human brain handling partly occlude objects, they use extra knowledge outside the visible information (such as past experience) in the recognition process.

In this thesis, we aim to address two challenges in image completion: (1) **capturing the structural information** in the image and (2) **recognizing the object contours** in the damaged image and using them to facilitate the repair process. We first present our observations of the existing image completion technique using PDE-based or patch-based methods and highlight the importance of structure in the process of completion. We then note that further improvement based on the present image completion framework will require the efficient use of mid-level vision cues especially structures. We suggest proper repair of structures within an image requires

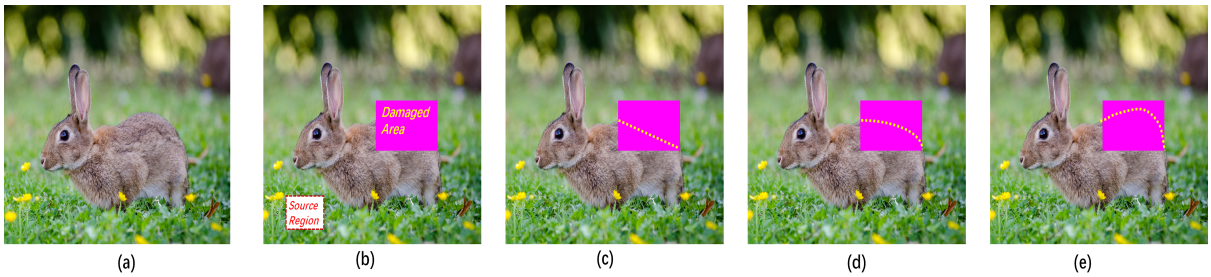

Figure 2: *How do we determine the contents that should appear in a cavity? Given a damaged image (b) (obtained by deleting an area from (a)), our task is to infer the contents within the cavity with the remaining information in the source region. (c)-(e) present various types of curves that could be used to repair the contours of the rabbit. However, current methods can hardly generate reasonable structures, and the repair is conducted without recognition of the objects.*

object information to connect the broken lines in the image. Finally, we present our image completion system with both implicit and explicit structure reconstruction functions. We also apply our image completion approach to the task of shadow removal.

1.1 Review of the existing approach

The first issue in image completion is properly capturing the structures within the image. Given a hole surrounded by the pixel values, we can begin to deduce the contents of the hole from any point on the boundary. However, both perceptual experiments and previous image repair techniques have suggested that image repair should begin with structure completion. Perceptual experiments have shown that human vision is sensitive to structures. Nill and Bouzas (1992) noted that human vision is accustomed to neglecting large uniform areas and focusing on structural regions. Kanizsa found out that the human vision system will attempt to detect T-junctions when perceiving missing areas in an image (Kanizsa, 1985; Pessoa, Thompson, & Noe, 1998). Once the T-junctions are detected, the human brain also detects the incomplete structure. Bertalmio, Sapiro, Caselles, and Ballester (2002) noted that structures have a higher priority within the repairing order in his image inpainting techniques. To properly reconstruct the structure within the image, we must first identify the structures in the image. One difficulty

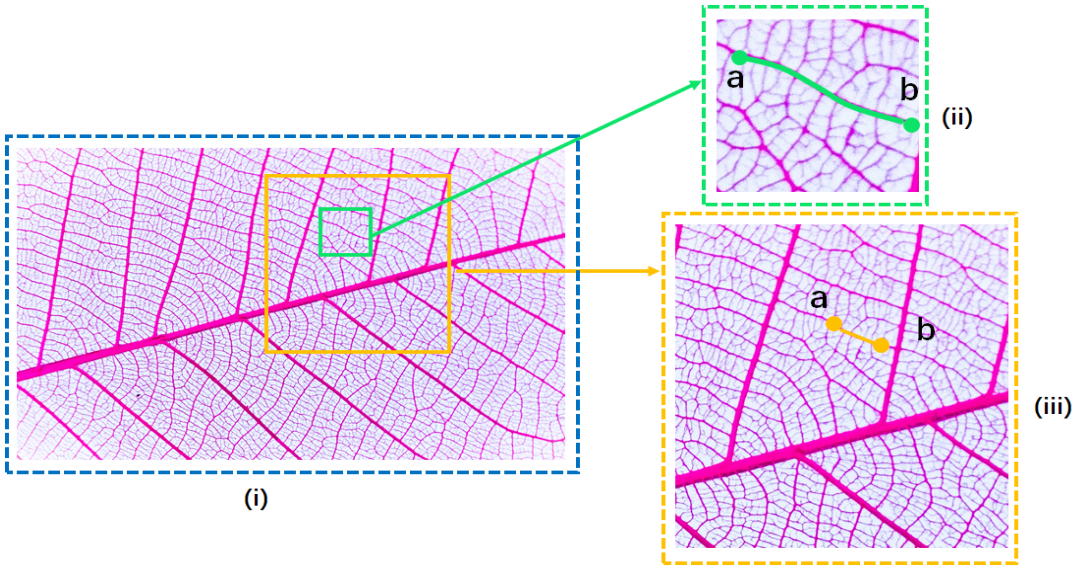

Figure 3: *Structure and scale. (i) We capture a leaf vein* ab *with different patch sizes. (ii) Line* ab *can be regarded as the main structure in a small patch. (iii) When line* ab *is observed on a larger scale, it is usually considered a component of the leaf texture.*

in identifying structures is that they vary in scale. In general, we define a structure within an image as "the lines that separate different regions". However, whether a line is a structural line or just part of a texture region largely depends on our scales of observation. This issue is illustrated in Figure 3, in which we present an identical leaf vein in patches of different scales. In Figure 3(ii), the line ab is regarded as a structure line that partitions the patch into two regions, but in Figure 3(iii), the leaf vein ab is usually considered as a component of the leaf texture when the line is observed in a larger patch. We suggest that a robust image completion system should be able to capture structures at various scales.

The second issue is how to repair the structures in an image. Although some existing image completion methods have addressed the importance of reconstructing structures within the image, reconstruction of a structure heavily relies on local geometric properties (such as gradients). Structure lines are connected via interpolation or curve-fitting, which takes advantage of the connectivity of curves. In PDE-based methods, structured lines are connected via propagation guiding by the designed equation. Previous methods repaired structures with the edge information that is near the damaged area. Nevertheless, local geometric properties are not sufficient cues to complete more complicated structures. Figure 4 provides an example of structure

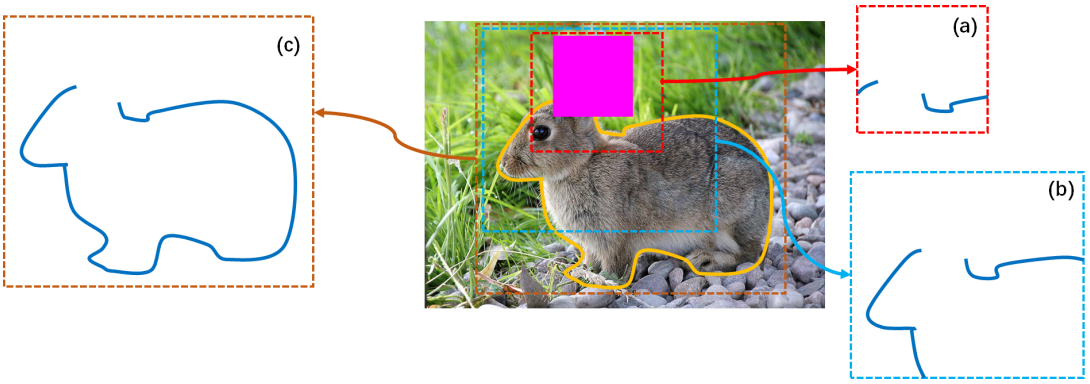

Figure 4: *Structure completion using object information. Previous methods of contour completion use local geometric features only. For example, when trying to repair the ear of the rabbit in the Figure, only part of the structures near the cavity is considered (such as the line segment in (a) and (b)). It is difficult to recover the ear of the rabbit because it is difficult to estimate the structure of the ear according to the local geometric features (such as curvature and smoothness). Part (c) includes the remaining contour of the rabbit. Given the partial object contours, it is easier to realize that the missing ear structure should be presented in the cavity.*

completion. When considering two fragments that are near the gap, one reasonable completion is to fit the curves by ensuring the smoothness of the connection. However, observation of the gap on a larger scale and consideration of more edges reveals that the line fragments combine to form the shape of a rabbit. The reconstructed structures in the gap should represent the rabbit's long ears. This reconstruction cannot be achieved through guarantee smoothness only, and additional object information is needed to infer the absent structures.

In short, our objective is to develop an image completion technique that can capture structures and take advantage of the object information that remains in the image. There are two difficulties with this goal: (1) structures are on multiple scales and are easily confused with textural patterns, and (2) generation of reasonable structures requires prior knowledge of the object.

1.2 Outline of the study

Here we give a brief overview of the remainder of this thesis (also shown in Figure 5). In Chapter 2, we review the literature of visual completion in the field of computer vision, most

of which is under the topic of image completion or image inpainting. We include a brief review of previous methods and analyze the difficulties encountered, and their solutions. Based on the review of previous techniques, we developed our approach based on the patch-based framework and made further improvements in structure preservation.

1.2.1 Image completion with implicit structure preservation

Our study begins with improving the structure-preservation ability based on the patch-based image completion framework (Chapter 3). We present a novel structure-preserving image completion approach equipped with dynamic patches. We formulate the image completion problem into an energy minimization framework that simultaneously accounts for coherence within the hole and global coherence. The completion of the hole is achieved through iterative optimizations combined with a multi-scale solution. To avoid abnormal structures and disordered texture, a Dynamic Patch System (DPS) is developed to achieve efficient structure restoration. Our DPS functions in both horizontal and vertical directions of the image pyramid. In the horizontal direction, we conduct a parallel search for multi-size patches at each pyramid level and design a competitive mechanism to select the most suitable patch. In the vertical direction, we use large patches at a higher pyramid level to maximize structure restoration and use small patches at a lower pyramid level to reduce computational workload.

1.2.2 Color-consistent shadow removal from an image completion prospective

Our enhanced patch-based image completion technique is applied to the task of shadow removal based on the view that the shadow image can be regarded as an image with a damaged illumination field (Chapter 4). A novel patch-based shadow removal approach was developed that maintains the color consistency between the shadow removed region and the lit region. By conducting intrinsic image decomposition, we are able to directly remove the shadow from the illumination of the image without the interference of color difference. Unlike previous methods that conduct color correction after removing shadows, our approach first optimizes the reflectance of the image and generates a guided map. The optimized reflectance not only guarantees the color consistency of the final result, it provides guidance for illumination recovery

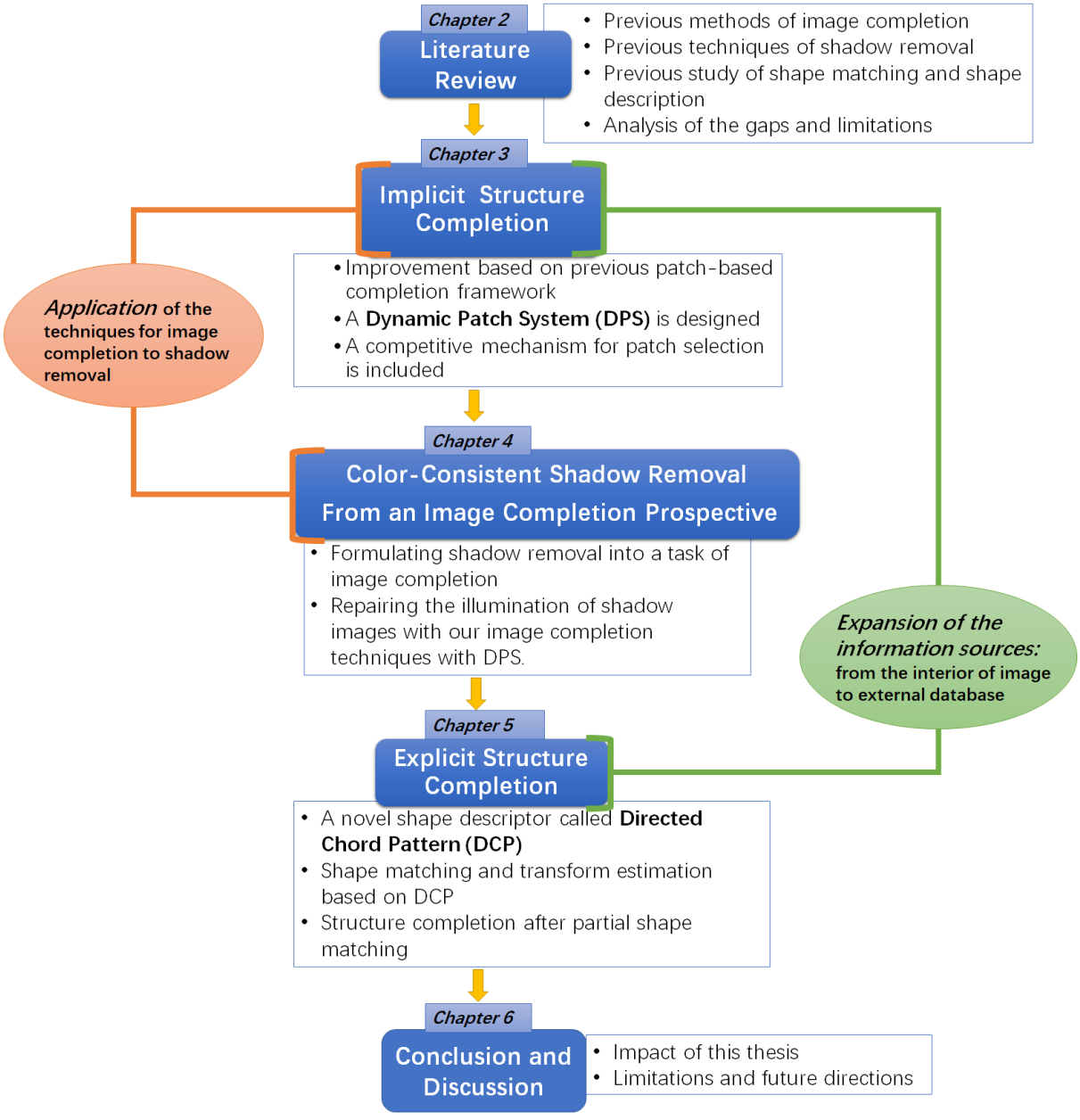

Figure 5: *An overview of the thesis*

as well. Based on the insight that a shadow image is an image with impaired illumination, we formulate the illumination recovery into an image completion framework. We recover the illumination by minimizing an energy function that simultaneously accounts for luminance, texture, and neighbor coherence. The minimization of the function is achieved through a "Search and Lighten" iterative optimization. The shadow-free images, generated by combing optimized reflectance and repaired illumination, preserve the texture well and are consistent with the lit region in color.

1.2.3 Explicit structure completion via partial shape matching

Our image completion with implicit enhancement of structure preservation can handle copyable structures properly, but repair of non-copyable structures (such as a unique object contour) requires extra information outside the image and explicit repair of the structure is needed to ensure reasonable results. Object contour is often a salient structure in the image, and we attempt to complete the contour via partial shape matching (Chapter 5). The repair of the contour is an inferential task. The task is actually an attempt to infer the original shape given only fragments extracted from the shape. This task requires two items: 1) an object contour database as an extra information sources and 2) an efficient contour retrieval method from the database. The difficulties lie in the latter. A shape descriptor named DCP was designed to match the damaged contour in the image and the shape templates. The DCP uses the spatial relationship between a sample point and the chords in its neighbor to describe the feature at that point. Based on the correspondence of DCP, we can repair the damaged contour via the estimated transformation of the shape template.

1.3 Contributions of this study

There are three main contributions in our study:

1. Dynamic Patch System (DPS). This system allows the program to use patches of various sizes in the repairing process. It is specially developed for the patch-based image completion framework.

2. Shadow removal using image completion technique. We formulate the task of shadow removal into the task of image completion. We developed a model that can unify these two tasks and successfully adapt the technique of image completion for the task of shadow removal.

3. Directed Chords Pattern (DCP). It is a novel shape descriptor proposed in this study. It uses the point-chords position to extract shape features and is designed for partial shape matching

2 Literature Review and the Research Questions of the Study

2.1 Image repair

Many frameworks and algorithms have been developed to perform the task of image repair and have achieved sound results. The techniques can be classified into two main categories: pixel-based methods and patch-based methods. In recent years, with the impact of the deep learning techniques, learning-based methods for image completion have also become popular.

2.1.1 Diffusion-based inpainting

Image inpainting applies undetectable modifications to a picture (Bertalmio et al., 2002; Brendt, 2011) by filling the selected area along the isophote line with automatic detection of source information. It operates at the pixel level and can handle small gaps and minor structures. The information in the known region propagates according to Navier-Stokes equations. The technical framework of inpainting can be improved with various diffusion mechanisms. Weickert (1996) proposed edge enhancing diffusion, which features decreasing diffusivity and increasing contrast in the direction perpendicular to the edges. Weickert (1999) proposed coherence enhancing diffusion, which enhances the diffusivity along the coherent direction. Chan and Shen (2001) took Euler's elastica into consideration to handle curve structures. Tschumperle (2006) used a trace-based PDE model to regularize images with multiple color channels. Later studies based on inpainting also attempted to improve the executive speed. Telea (2004) described a fast matching technique using the weighted average of the calculated pixels. The method's advantages include lower time consumption and easy implementation.

The image inpainting problem can also be formulated into a variational framework. The variational formulation is initially applied to a discollusion problem, in which an image is reconstructed according to a set of level lines (Masnou & Morel, 1998). These level lines are defined by bounded variational functions. Ballester, Bertalmio, Caselles, Sapiro, and Verdera (2001) modified the inpainting problem into a variational framework. The input image is regarded as a function of bounded variation. The variational approach for image repair is based on joint interpolation of both gray levels and the isophotes direction. The interpolation is

conducted by solving a series of coupled second-order PDEs. The Total Variation (TV) image model, which is initially used for denoising and the deblurring problem (Rudin, Osher, & Fatemi, 1992), has also been applied to the inpainting problem. The model's energy function is in a TV norm. Chan and Shen (2001) used a new Curvature Drive Diffusion (CDD) model to improve the diffusion mechanism. The CDD model was derived from TV inpainting and aims at realizing the connectivity principle. The conductivity coefficient is determined by the curvature of the isophotes. Levin, Zomet, and Weiss (2003) modeled an exponential family distribution over the image according to the histograms of the local features. The hole is then inpainted using a variational framework with the specific distribution. Although most inpainting techniques perform excellently with small gaps and holes, the inpainted results may generate unnatural blocks when applied to large cavities or textured regions.

2.1.2 Texture synthesis and exemplar-based approaches

Exemplar-based methods perform better in texture acquisition than the pixel-based method. They work at the patch level using texture synthesis techniques (Efros & Freeman, 2001; Kopf et al., 2007). In fact, the task of texture synthesis can be regarded as an extreme situation in which the cavity occupies most of the image. Efros and Leung (1999) proposed a valuable non-parametric method for texture synthesis, in which sample patches are acquired from a textured example and embedded into the target area. Further improvement can be achieved by modifying the sampling approaches and search mechanism. Liang, Liu, Xu, Guo, and Shum (2001) presented a real-time texture synthesis algorithm that generates a high-quality texture. The algorithm avoided mismatching features on the patch boundaries with the use of sample patches according to a nonparametric estimation of the Markov Random Field (MRF) density function. Kwatra, Schödl, Essa, Turk, and Bobick (2003) used the graph cut technique to determine the optimal patch region between the input and output textures. Unlike techniques that use dynamic programming to combine patches, this method can optimize seams in any direction. Lefebvre and Hoppe (2005) generated textures with a deterministic search process using a model derived from MRF. The synthesis process is accelerated with tree-structured vector quantization.

Exemplar-based methods for image repair operate on the image patch level. The patches are placed to fill in the unknown area under specific automatic guidance, which can either determine the fill-in ordering or provide extra information on gradient or structures. Drori, Cohenor, and Yeshurun (2003) presented a method for completion of missing parts caused by extraction of foreground or background objects from an image. Their method automatically embeds adaptive image fragments into the image according to a confidence map. Criminisi, Perez, and Toyama (2003) combined the texture synthesis technique with the inpainting method. The combined method takes advantage of inpainting to propagate one-dimensional elements in the image, such as lines and object contours. The texture synthesis method is used to generate large textural regions. Bertalmio, Vese, Sapiro, and Osher (2003) had a similar idea of using different techniques to repair structure and texture respectively. Jia and Tang (2003) used tensor voting to automatically infer missing information from an impaired image. Their method translated image color and texture information into an adaptive N-dimensions tensor, and the optimal values in the N-dimensions texture space are determined by a voting process. J. Shen, Jin, Zhou, and Wang (2007) used a gradient-based model to determine the filling order. The damaged image is repaired by solving the Poisson equation with an image gradient map.

Darabi, Shechtman, Barnes, Goldman, and Sen (2012) developed an image melding framework that combines inconsistent images using patch-based synthesis. This powerful framework enables users to complete the task with larger patches. However, the cost of using larger patches is a longer executive time and higher requirements for hardware. In fact, the size of the patch is a crucial parameter in most patch-based methods. The executive time differences can reach 90% between an application of 5×5 and 10×10 patches if the other experimental conditions remain the same. Because the use of larger patches for repair requires more computation in each iteration. PatchMatch (Barnes, Shechtman, Finkelstein, & Goldman, 2009) is a powerful tool for fast iterations and rapid calculation of the distance between patches. This patch-based method is a randomized structural algorithm for image editing. It first fills the hole with randomly selected patches and then searches for a better match in the concentric neighborhood randomly accompany with good match propagation. Barnes, Shechtman, Goldman, and Finkelstein (2010) later developed PatchMatch into a generalized correspondence algorithm

that can find k nearest neighbors and search across scales, rotations, and translations. Another valuable framework is to formulate the task of completion as a global optimization problem (Wexler, Shechtman, & Irani, 2007). The optimal result is achieved by reaching the greatest coherence of the image or video. The optimization problem is usually solved with a greedy algorithm. This framework has become a popular approach for texture synthesis and image completion (Kopf et al., 2007; Arias, Facciolo, Caselles, & Sapiro, 2011; Kwok, Sheung, & Wang, 2010).

2.1.3 Learning-based methods with deep architecture

Learning-based methods with deep architecture have achieved great success in many visual perception tasks such as image classification, object recognition, and face detection. Besides the breakthrough of in high-level vision tasks, attempts have been made to handle low-level vision tasks with deep architecture. Xie, Xu, and Chen (2012) proposed a technique which combined sparse coding with neural networks. Their network is pre-trained with denoising auto-encoder. Their technique can conduct blind inpainting tasks in which the location of the corrupted pixels is not provided. Ren, Xu, Yan, and Sun (2015) used Shepard interpolation to disclose the limitation of convolution neural network for handling image processing tasks that required translation variant interpolation. Early learning-based methods are developed for handling small gaps or holes and the inpainting task is regarded as a variant of the denoising problem. They may suffer from similar problems as the diffusion-based methods when handling large holes. In addition, the deep neural networks' ability to extract high-level visual features is not fully maximized.

In 2016, Pathak, Krahenbuhl, Donahue, Darrell, and Efros (2016) proposed an unsupervised visual feature learning algorithm driven by context-based pixel. Similar from Xie et al. (2012) which used the denoising auto-encoder, they proposed the Context Encoder that is a convolutional neural network that captures not just appearance but also the semantics of visual structures. Their network architecture is a simple encoder-decoder pipeline in which the encoder and decoder are connected through a channel-wise fully-connected layer. Although this algorithm can only handle fixed size images with low resolution, it provides a new way

to handle the task of image completion with neural networks. Later studies made further improvements on their network structure. One category of the latter studies is to improve the visual quality of the output results. Iizuka, Simo-Serra, and Ishikawa (2017) added a global discriminator network and local discriminator network after the completion network. The three networks are trained in an adversarial style. The completion network is trained to fool both discriminator networks. C. Yang et al. (2017) developed a joint optimization framework with texture and image content constraints and used a multi-scale neural patch synthesis algorithm for repairing high-resolution image. Their framework contains two networks: a content network which is similar to the encoder-decoder structure in the previous study, and a texture network which is trained to guarantee the generated contents are similar to the existing contents. Yeh et al. (2017) used the Generative Adversarial Network (GAN) to find the optimal encoder. The optimal encoder is then passed to the generative model to infer the missing contents. Their approach does not require masks for training and can infer on holes of arbitrary shape. Yu et al. (2018) introduced a contextual attention layer into the completion network to explicitly utilize the texture information with the image. A coarse-to-fine architecture is also used to improve the output quality.

In addition to improving the visual effect of the repaired image, some studies attempted to develop completion network for specific kind of contents, for example, face images. Y. Li, Liu, Yang, and Yang (2017) developed a network for face completion, which is also a combination of GAN and encoder-decoder structure similar to the approach by Iizuka et al. (2017). Q. Sun et al. (2018) developed a two-stage inpainting framework for identity obfuscation. It first obtains facial landmark in images then use the GAN to generate contents of the head for the human object. Dolhansky and Canton Ferrer (2018) designed a type of conditional GAN, called ExGAN, which uses examplar information to generate high-quality inpainting results. It has a successful application on the close-to-open eye inpainting tasks.

Although learning-based methods with deep architecture are able to generate plausible results and even generate contents that do not exists with the damaged image, they are still limited by the time-consuming training and massive data requirements. Most of the deep learning based methods are trained on powerful hardware but it still takes several days or weeks to finish the

training. This may cause difficulties in actual application. For example, if one would like to expand the ability of a completion network, he not only need to expand the training dataset but also retrain the network. In addition, specific training data are required to enable the completion network to repair specific contents. It is unlikely that a completion network that trained with face images can repair any arbitrary object or scene in the image. Current learning based methods with deep architecture rely heavily on the knowledge extracted from the training dataset. But in some situation, massive training data of specific contents may not be available.

2.1.4 Image blending

The last vital step in most of the exemplar-based methods is to arrange the retrieved patches and combine them to form visually pleasing content. If the retrieved patches are not stitched together properly, the formed content may be not consistent in terms of texture, color or contrast and may have structural fractures and unconnected edges. In addition, the transition between the hole and the known region plays an important role in the overall effect. A sound completed result usually has imperceptible edges that seamlessly connects the cavity and the known region. Many image blending techniques have been developed to achieve a seamless connection between the hole and the known region. Burt and Adelson (1983) introduced the seminal image stitching technique, which combines images via pyramidal image decomposition. Sunkavalli, Johnson, Matusik, and Pfister (2010) applied similar multi-scale techniques with a preliminary treatment called image harmonization that transfers the appearance of an image. Gracias, Mahoor, Negahdaripour, and Gleason (2009) developed a fast method to combine a set of registered images into a mosaic with the use of watershed segmentation and graph cut optimization. Chen, Cheng, Tan, Shamir, and Hu (2009) developed a system that can automatically select suitable images from the Internet and generate high-quality combined pictures with blending boundary optimization. Whitaker (2000) proposed an image blending technique via level set comparison.

2.2 Shadowed image processing

Adapting solutions or techniques from one image-processing tasks to another may unexpectedly create better solutions. However, such adaptation cannot be generated by random combination and without understanding of the similarities between tasks. As we mentioned in the review of the previous methods of image repair, the current patch-based technique originated from texture synthesis techniques. This successful adaptation can be attributed to the observation that the task of texture synthesis can be regarded an extreme situation in image completion. Inspired by this adaptation, we sought to adapt the techniques of image completion to the task of shadow removal. We now discuss the previous methods for removal of shadows and examine the similarities between these two tasks.

2.2.1 Shadow detection

The first step in removing shadows in images is to detect the shadow regions. Many methods have been proposed to locate shadow regions in images. Previous shadow detection methods can be classified as interactive shadow detection or automatic shadow detection. Interactive methods require user assistance to specify shadow samples. L. Zhang, Zhang, and Xiao (2015) used specified shadow samples and lit samples to construct a trimap to mat the shadow region. Shor and Lischinski (2008) generated three different types of shadow masks by calculating the distance between pixels and the user-indicated shadow seeds. The method described by F. Liu and Gleicher (2008) requires users to mark an approximate shadow boundary with a brush tool and can precisely locate the penumbra area based on the user-specified boundary.

Unlike interactive methods, automatic shadow detection aims to locate the shadow with minimal user operation; however, automatic shadow detection is an extremely difficult task. Given local information only, a dark pixel can be caused either by shadow or by the reflectance at the corresponding scene point. Thus, the design of an automatic shadow detection method requires a deeper understanding of the scenes in the images. Finlayson, Hordley, Lu, and Drew (2006) located the shadow region by comparing the edges in the original RGB image and those in an illuminant-invariant image. This method can precisely locate the shadow in

high-resolution images generated by calibrated sensors (Lalonde, Efros, & Narasimhan, 2010). The core of this method lies in the illuminant-invariant image, which may not be available for web-quality consumer photographs. Some researchers have attempted to detect shadows with empirical and data-driven methods that take advantage of the machine learning technique. Zhu, Samuel, Masood, and Tappen (2010) classified regions according to both shadow variant features (such as intensity difference and smoothness) and shadow invariant features (such as texture and gradient), and developed a binary conditional random field to label the shadow region. Panagopoulos, Wang, Samaras, and Paragios (2011) proposed a higher-order MRF illumination model that takes advantage of both the low-level shadow evidence and the high-level prior knowledge to estimate the cast shadow and illumination environment. R. Guo, Dai, and Hoiem (2013) proposed a shadow detection method based on pair wise region relationship classification. Qu, Tian, He, Tang, and Lau (2017) tackled the shadow detection problem with a deep neural network called DeshadowNet, which is a multi-context architecture.

2.2.2 Shadow removal

The shadow removal described here refers to manipulation to recover the illumination of the dark pixels. Previous methods conducted the shadow removal operations in various ways based on their formulation of the shadow removal problems. Finlayson and his colleagues (Finlayson et al., 2006; Finlayson, Drew, & Lu, 2004) treated the shadow removal task as a reintegration problem in the gradient domain. Their method generates shadow-free images by nullifying the gradients on the shadow boundaries. This method relies heavily on accurate shadow boundary detection, so some researchers have attempted to make improvements in locating the shadow boundary. Mohan, Tumblin, and Choudhury (2007) used a gradient-domain shadow-edge model to describe the shadow region. The proposed model can simulate various lighting conditions. F. Liu and Gleicher (2008) constructed a shadow-free field and a texture-consistent gradient field separately to ensure texture consistency in the shadow-removal task.

Another category of shadow removal is the relighting method. These methods are mainly based on the observation that the pixels in shadow regions and those in corresponding lit regions mainly have differences in intensity, which are measured by a scale factor. The core

idea of these methods is to estimate the scale factor and use it to enhance the lightness of the shadow pixels. Fredembach and Finlayson (2006) simplified the reintegration operation to an optimization method that attempts to find the additive constant in each color channel in each shadow region. The constant is estimated according to the pixel values on both sides of the shadow boundary. Arbel and Hel-Or (2007) noted that a fixed scale factor may not be suitable to handle the shadows in penumbra areas and curved surfaces. They applied directional smoothing to correct the scale factors to eliminate abrupt variation.

Some researchers formulated the shadow removal problem into a color transfer task in which the statistical color features of one image are transferred to another. When the color transfer technique is applied to shadow removal tasks, the goal is to transfer illumination in the lit region to the shadow region. Based on this idea, a series of color transfer methods for shadow removal have been proposed. Shor and Lischinski (2008) proposed a linear mapping model to describe the affine relationship between the shadow and non-shadow regions. Xiao, She, Xiao, and Ma (2013) proposed an adaptive illumination transfer technique that varies the transformation according to the material reflectance. Both techniques use global illumination transfer operators. Xiao, Xiao, Zhang, and Chen (2013) improved their methods by using multiple illumination transfer operators between matched subregions. L. Zhang et al. (2015) used optimized illumination-recovering operators to lighten the patches in the shadow area.

2.2.3 Intrinsic image decomposition

Intrinsic image decomposition usually refers to the separation of illumination and reflectance components from an input photograph. Barrow and Tenenbaum (1978) first introduced the term “intrinsic image” which refers to a view-dependent mid level description of a scene. Any observed image is a product of two images: an illumination image and a reflectance image. This concept proposed by Barrow and Tenenbaum is extremely useful for visual inferences even though that it does not make explicit all the physical causes of image features. However, obtaining intrinsic images are difficult because the problem is highly ill-posed. The first solution to this problem was the Retinex algorithm developed by Land and McCann (1971). Tappen, Freeman, and Adelson (2003) designed an algorithm in which color information and

gray scale pattern classifiers are used to recover intrinsic images. Some researchers have attempted to handle these ill-posed tasks with additional constraints. Weiss (2001) simplified the ill-posed problem to the task of recovering intrinsic images according to a series of images taken in different lighting conditions. Although the simplified problem is still ill-posed, a maximum-likelihood solution was provided. L. Shen, Tan, and Lin (2008) took advantage of the texture information to obtain constraints on reflectance. Their method is based on the observation that different points with the same intensity-normalized texture configuration usually share the same reflectance value. Interactive methods used to obtain constraints on reflectance are also practical. Bousseau, Paris, and Durand (2009) developed a method that allows users to guide an optimization that has a closed-form solution using linear least-squares. J. Shen, Yang, Li, and Jia (2013) took user scribbles to specify local constraint cues that can further enhance the recovery results. Yue, Yang, Sun, Wu, and Hou (2017) developed a contrast-enhancement method based on constrained intrinsic image decomposition models.

2.3 Shape description and matching techniques

The object contour provides important object information. Human beings can easily recognize objects by their contours. When a damaged image contains damaged contours, repair is often difficult. Because the object contours may contain some unique features that cannot be duplicated or easily inferred from the remaining segment. Because the remaining information within the images is not sufficient for repair, we must turn to extra information outside the damaged images. An intuitive solution is to use a shape database to aid the repair. However, the best use of the damaged object contours to retrieve a proper shape template from the database and complete the contour remains unclear. Here, we come to the task of shape description and shape matching.

2.3.1 Curvature and integral invariants

For feature-based shape matching methods, shape descriptors are the foundation of the similarity measures between two shapes. The main function of a shape descriptor is to extract features from a shape. Previous researchers have devoted considerable effort to develop smart shape

descriptors. One intuitive method for processing shape is to use the boundaries of silhouette images. Regardless of the contents within the shape (such as holes or internal markings), the associated boundaries are essentially two-dimensional closed curves, and so traditional geometric parameters of curves can be applied for description. Techniques based on one-dimensional function derived from boundary coordinates are often called *Shape Signature*. *Tangent Angle* is a kind of shape signature (D. Zhang & Lu, 2001) that defined by the angle forms by the tangent line at each sample point and the horizontal axis. Another shape signature *Complex Coordinates* function uses the complex number generated from the coordinates of boundary points to describe the shape. One of the most reliable and important shape signatures is the *Contour Curvature*, which has salient perceptual characteristics and benign invariant features (Mokhtarian & Mackworth, 1992; Jalba, Wilkinson, & Roerdink, 2006). The curvature is invariant against translation, rotation, and scale.

Although curvature has significant discriminating power, the result of direct comparison using curvature may fail expectations. For one thing, digital images have a discrete nature and cause quantization in the calculation of curvature. For another, the computation of curvature requires second-order derivatives, which are not always available in the digitized boundary functions. To overcome these two difficulties, some researchers have used auxilliary techniques in the curvature computation. H.-C. Liu and Srinath (1990) applied chain code representation of the boundary before computing the contour curvature and used distance transformation before shape matching. Some researchers have attempted to find similar geometric features that have a strong correlation to curvature. Han and Poston (2001) developed a method of calculating chord-to-point distance accumulation, which can be regarded as discrete curvature. Lin, Dou, and Wang (1992) defined an Arch Height Function (AHF) for contour feature extraction. Their function has local maxima at the corner points, which are often the points with high curvature. Other researchers have sought an equivalent definition of the curvature in integral geometry, and new curvature-based descriptors have been developed. H. Liu, Latecki, and Liu (2008) proposed a unified definition of curvature for regular, polygonal, and digital planar curves. Their curvature definition is based on the statistics of the extreme points of the height functions and yields a stable curvature estimation of digitized curves. Based on this definition, Wang,

Bai, You, Liu, and Latecki (2012) developed a shape matching method by using tangent lines to calculate height functions. Manay, Cremers, Hong, Jr Yezzi, and Soatto (2006) defined curvature from a view of integral geometry and proposed a normalized area integral invariant. C. Xu, Liu, and Tang (2009) implemented this definition and proposed a matching technique based on a descriptor called *Contour Flexibility*.

2.3.2 2D shape descriptors

The features directly extracted from the boundary points are limited because they rely mainly on geometric features at the local point. In addition, some shape signatures discard part of the global geometric information and cannot properly reflect the relationship between points. Thus, researchers have attempted to design rich descriptions of shapes in higher dimensions. Except for geometric features at local points, spatial interrelation features can also be extracted from regions or chords in the shape. Broadly speaking, descriptors based on spatial interrelation features can be classified into region-based and chord-based descriptors. Region-based shape descriptors usually treat a shape as the bounded interior of simple closed (Jordan) curves and describe the shapes by dividing subregions within the shapes or constructing geometric measures inside the interiors. Chakrabarti, Ortega-Binderberger, Porkaew, and Mehrotra (2000) used a technique called *adaptive grid resolution* to represent 2D shapes using a series of grid cells whose resolution varies according to the portion of the content. With this technique, a shape can be easily expressed with quad-tree decomposition (Shusterman & Feder, 1994). Bounding box (Bauckhage & Tsotsos, 2005) is another similar shape descriptor that divides a shape of arbitrary topologies into a series of 2D lattices. Along with region-based descriptors, moments are frequently introduced for shape analysis. Shape descriptors based on moments take advantage of both the region and the contours of a shape. Different types of moments have been explored, and they mainly vary in the *moment weighting kernel*. *Invariant moment* (M. Hu, 1962) is one of the simplest moments using multiplication of the coordinates as the basis set. The *Zernike moment* (Celebi & Aslandogan, 2005) originated from the orthogonal Zernike polynomials, and the moments are expressed in the complex plane. *Radical Chelbyshev moment* (Mukundan, 2004) applies scaled orthogonal Chebyshev polynomials as the

basis set and is rotational invariance. Region-based shape descriptors, in general, have good invariance but may show poor performance when occlusions exist.

Unlike the region-based shape descriptors, which directly construct a similarity measure based on the region, chord-based shape descriptors develop similarity metrics with chords in the shape. Chord-based shape descriptors construct chords by connecting sampled points on the shape contours and recognize the shape using the statistics of the chords feature (length or radical position). Although many chord-based shape descriptors do not explicitly construct chords within the shape, the pairwise distances between boundary points are essentially the chord length. Smith and Jain (1982) accounted for all chords within the shape and generated a chord distribution for shape matching. The chord distribution is composed of two histograms: a chord lengths histogram and a chord angles histogram. These two histograms are essentially statistics on the spatial position of the chords; however, it is obvious that taking all chords into consideration for shape description may not be necessary and efficient. A better way is to classify the chords according to specific criteria and to discard useless chords. M. Yang, Kidiyo, and Ronsin (2008) discarded chords that were too short and classified the remaining chords according to their orientation. A more popular way is to construct the chords according to certain rules. Shape Context (SC) (Belongie, Malik, & Puzicha, 2002) is a powerful tool for the shape object recognition task. It encodes each sample point on the contours using the chords that begin at that point and present the local chord distribution using a log-polar histogram. Its improved version, the Inner Distance Shape Context (IDSC) (Ling & Jacobs, 2007) enhances the performance of SC by limiting the chords within the interior of the shape. The Beam Angle Statistic (BAS) (Arica & Vural, 2003) uses the angles formed by chords that originate from a boundary point and connects the rest of the points on the boundary. The BAS is scale invariant and has been successfully applied in partial shape matching (Donoser, Riemenschneider, & Bischof, 2009; Michel, Oikonomidis, & Argyros, 2011). R.-X. Hu, Jia, Ling, Zhao, and Gui (2014) developed a similar shape descriptor based on angles. They appended neighboring angular patterns at each point using a scheme similar to Local Binary Pattern (LBP).

2.3.3 Shape matching

In this section, we review the matching techniques developed based on Local Shape Descriptor (LSD). For most shape matching methods based on LSD, the matching procedure is as follows. The first step is to apply the LSD to each sample point and extract feature vectors. The second step is to estimate the correspondence between points according to the feature vectors, and the final step is to use the correspondence between point sets to find the transformation. The reliability of the correspondence depends on the design of the shape descriptors. Whether a correspondence can be found efficiently relies on the matching techniques.

Usually, two shapes are matched by minimizing the sum of the matching cost, which is usually defined by distance of the feature vectors extracted by LSD at each sample point. The task is actually to find an assignment between two point sets; thus, it can be solved with the Hungarian algorithm (Munkres, 1957) or dynamic programming (Bellman, 1966). However, these matching methods do not ensure the preservation of global geometric information. Previous researchers have attempted to enhance the matching techniques by appending extra constraints. Scott and Nowak (2006) used the order of points on the contour as a constraint in the correspondence assignment. Schmidt, Farin, and Cremers (2007) formulated the shape matching task into the shortest circle problem on a torus and proposed a rapid algorithm that can finish the matching task in sub-cubic run time. Yang and his colleagues used graph transduction to improve the performance of shape retrieval (X. Yang, Bai, Latecki, & Tu, 2008; X. Yang, Koknar-Tezel, & Latecki, 2009), and Egozi, Keller, and Guterman (2010) developed a matching scheme that accounts for meta similarities and can be compatible with various LSDs.

2.4 Relations to our work and research questions

Our research interest begins from the patch-based image completion problem. After reviewing previous methods, we realized the importance of structure preservation in generating satisfying results. The patch-based image completion approach achieves structure preservation via spatial coherence between patches. We noticed that previous methods prefer large patches in the completion and that the size of the patches is usually fixed during repair. Here we propose the

first research question:

- **Research Question 1:** *Are larger patches better for image completion?*

We address this question in Chapter 3 and find that large patches is not always the best choice. We make improvements based on the previous patch-based image completion framework by including a Dynamic Patch System (DPS). We then move from our improved approach in two directions: 1) to adapt the image completion technique to other tasks; and 2) to include prior knowledge in the repair.

The patch-based image completion methods originated from the texture synthesis techniques. Successful adaptation of texture synthesis techniques to image completion tasks can be attributed to the essential similarities between the tasks. The task of texture synthesis can be regarded as an extreme situation in which the damaged area occupies most of the image. Inspired by this adaptation, we suggest that potential exists for applying our improved image completion technique to other image processing tasks. We then target the task of shadow removal. Because the shadow area in an image is an area with insufficient lighting information, we suggest that the absence of lighting information can be considered as a special kind of damage. Realizing the similarity between the image completion task and shadow removal task, we proposed the second research question:

- **Research Question 2:** *Is it possible to formulate the task of shadow removal into the task of image completion?*

Our answer is positive, and we propose a novel shadow removal model from the view of image completion in Chapter 4.

The core of the patch-based image completion method is to use the coherence between patches to generate the contents within the damaged area. When the source of the patches is limited within the known region of the images, the generated contents can be regarded as a copy of the information from the known region. Previous patch-based methods can properly repair a copyable structure, but when handling non-copyable or unique structures (such as object contours), the repaired result is often disappointing. This failure can be attributed to two

aspects. On one hand, the information source is limited to within the damaged image. In many situations, the known region in a damaged image does not provide a ready-made structure for repair. On the other hand, repairing images relying only on the coherence between patches is essentially making inference using low-level vision cues without realization of the objects it is repairing. We attempt to address these two limitations in the case of repairing object contours.

The limitation of information sources can be solved easily. To expand the information sources for repair, we need only prepare an extra database of shape templates as prior knowledge. However, the main difficulty lies in matching the shape templates and the damaged contours in the image. Matching between shapes is essentially making inference using mid-level vision cues. Here, we come to the problem of partial shape matching and proposed the following research question:

- **Research Question 3:** *How can a damaged contour be matched with massive shape templates?*

We propose our solution in Chapter 5, in which we develop a novel shape descriptor and the corresponding shape matching schemes.

3 Image Completion with Implicit Structure Preservation

3.1 Introduction

Image completion is a challenging image editing task that plays an important role in repairing old photos, removing unwanted objects and disocclusion in image-based rendering. The goal of image completion is to fill the hole in a damaged image with contents that keep the overall visual effect realistic and harmonic. Two difficulties in the completion task, structure reconstruction and texture representation, are both essential to generate visually pleasing results.

The basic idea of image completion is to propagate the information in the known region into the hole based on the assumption that the known region contains the necessary information. The propagation can be described as a diffusion using PDE. Bertalmio et al. (2002) proposed an image inpainting technique to repair small cracks at the pixel level by propagating image Laplacians in the isophote direction. Most diffusion-based methods work at the pixel level and show excellent performance in reserving the local structure in small gaps. However, in situations that include repairing large cavities, artifacts or strange structure may occur.

Patch-based methods outperform diffusion-based techniques that operate at the pixel level in situations with large holes. Patch-based completion methods originate from the seminal work by Efros and Leung (1999), which was initially designed for texture synthesis. The adaptation of the texture synthesis technique is based on the fact that the task of texture synthesis can be regarded as an extreme situation in image completion in which the hole takes up most of the image. Based on this fact, studies such as those by Criminisi et al. (2003) and Efros and Freeman (2001) amplified the technique for texture synthesis and successfully reproduced nice textures in the hole to complete the image.

An essential criterion for evaluation of the completion result is whether structural lines are reconstructed smoothly within the hole, because the human visual system is sensitive to the structural region (Nill & Bouzas, 1992) and any artifact on a structure makes the completed image visually unpleasant. In most patch-based methods, structure restoration is conducted implicitly by pursuing global coherence. These methods complete the hole with patches from the known region. By maximizing the coherence between patches, they hope that the struc-

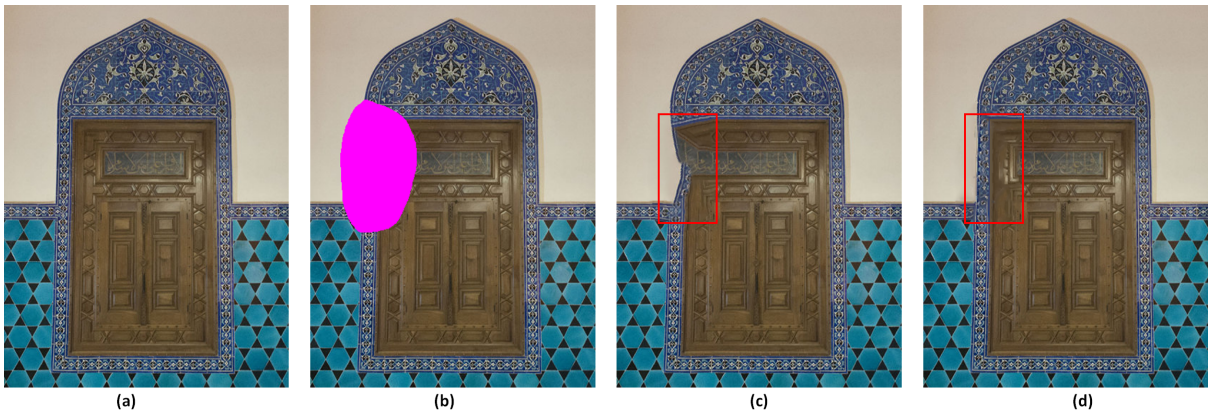

Figure 6: *(a) Original image* (399×533)*. (b) Hole is masked (magenta). (c) Result generated with state-of-the-art ImageMelding (Darabi et al., 2012). (d) Our approach.*

tures are also properly restored. However, distorted structures may still appear when complex structures are required to repair the hole. As shown in Figure 6(c), even the completed image generated by a previous advanced technique (Darabi et al., 2012) may present structural distortions in the hole. Figure 6(d) presents the image after repair with the approach we describe in this chapter. As shown in the figure, the hole is filled with nice textures, and a realistic structure is reconstructed.

Previous patch-based methods tended to use large patches in general. The purpose is to capture as much information as possible. However, large patches may not be suitable for repairing structures because structural lines usually appear on various scales. As shown in Figure 9(a), large patches may absorb more irrelevant information and distort structure within the hole. Also, the application of large patches creates an extra burden on hardware and leads to greater time consumption.

In this chapter, we introduce a new approach that accounts for the coherence between global and local to generate a realistic repaired image that preserves nice structure and texture. We formulate image completion into an energy minimization task that considers global and local coherence simultaneously. We develop a Dynamic Patch System (DPS) for efficient structure preservation. Our DPS functions from two perspectives in the repair process. First, it enables a parallel search for patches of multiple sizes. Patches of various sizes are selected to fill the hole with a competitive mechanism. Second, the size of the patches varies in the multi-scale

solution. Large patches are used at a lower resolution scale to maximize the information gain. Small patches are applied at a higher resolution scale to reduce the computational workload. In sum, the approach we present in this chapter makes the following contributions:

-**Efficient Structure Restoration** Mixed use of patches of various sizes can efficiently capture structure in various scales, and can avoid absorption of irrelevant information and distortion in structures.

-**Balanced Computational Workload** A multiscale solution with dynamic patches makes the computational workload adjustable in the repair process. It accelerates the repair process and reduces the computation needed for a lower image pyramid level without sacrificing the quality of the repaired image.

-**Competitive Mechanism** A parallel search for patches of various sizes is conducted with acceleration using GPU. A competitive mechanism is introduced to select suitable patches for completion.

3.2 Approach overview

Images are more than a collection of pixels. They all follow certain patterns that contain textural and structural information. Our Human Vision System (HVS) is sensitive to these patterns and thus image completion cannot be conducted with a simple copy-paste of the existing pixels. We formulate the task into an energy minimization framework to generate visually pleasing patterns in the hole. We aim to fill the hole with patches that are coherent with other patches within the hole and with the global picture. The challenge lies in discovering existing patterns in the known region and capturing various scale structures at the same time and avoiding interference from irrelevant information. Our approach optimizes an objective function that considers coherence within the hole and global coherence. That is, our objective function includes two terms: an external term and an internal term. The objective function is minimized via an iterative optimization, which contains two phases: a search phase and a vote phase. In the search phase, we conduct a parallel search to retrieve patches of various sizes. A competitive mechanism is designed to select the most suitable patch. In the vote phase, all patches are blended to form the contents within the hole. A minimum cost boundary between overlapped patches is

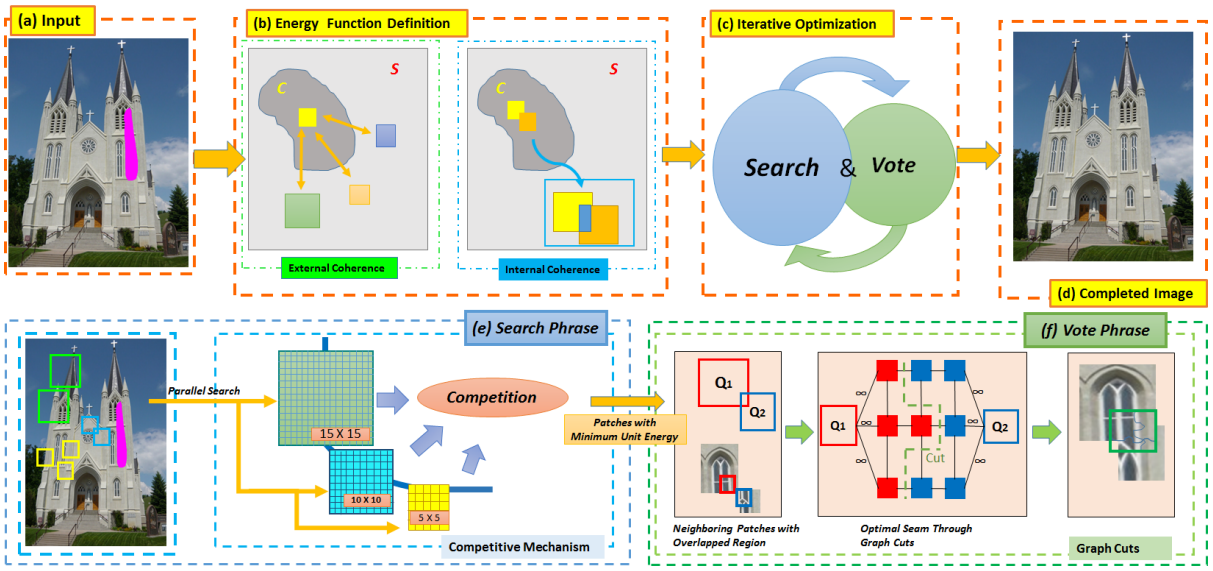

Figure 7: *Overview of our approach. (a) An image with a hole marked in magenta. (b) Definition of our energy function which considers external and internal coherence simultaneously. (c) Optimum of our energy function is achieved by "search & vote" iterative optimization. (e) Search phase. Parallel search is conducted in this phase to retrieve multiple size patches. The retrieved patch candidates compete to enter the vote phase. (f) Vote phase. Patches are combined through calculating the optimal seam with graph cuts. (d) Result with our approach.*

calculated to ensure that the patches are seamlessly connected and avoid artifacts in structure and texture. The overall repair process is accelerated with a multiple-scale solution. Unlike the previous methods that fixed the patch sizes, we use large patches at a lower resolution scale to maximize the information gain and provide a sound foundation for the next scale.

3.3 Energy minimization

Given a color image I with a hole C, we begin with the assumption that the source region $S = I - C$ contains sufficient information to generate the contents within the hole. We repair the image by minimizing the following objective function:

$$E(C|S) = \sum_{q \in C} \min_{p \in S} [w_1 E_1(Q(q), P(p)) + w_2 E_2(Q(q))], \tag{1}$$

where E_1 is the energy term for external coherence, and E_2 is the term for internal coherence. Detailed descriptions of these two terms are given in the following paragraphs. w_1 and w_2 are

the weighting factors that determine a preference between internal and external coherence. In our experiments, $w_1 + w_2 = 1$. $Q = \mathcal{N}(q)$ is an $h \times h$ patch in the hole with the target pixel q at its center. The target pixel q is set as the origin $Q_{0,0}$, and other pixels in the patch are denoted as $Q_{i,j}$. $P = f(\mathcal{N}(p))$ is an $h \times h$ patch in the known region after transformation f. p is its anchor pixel in the center. The image is assumed to be in $L * a * b$ space, and the pixels each have five channels that contain three color channels (L, a, b) and two luminance gradient channels $(\nabla_x L, \nabla_y L)$.

Mixed Similarity Measure- Metric selection plays an important role in the energy function definition. Previous patch-based methods use the Sum of Squared Difference (SSD) to measure the similarities between the two patches. However, SSD tends to give preference to uniform regions (Bugeau, Bertalmio, Caselles, & Sapiro, 2010) and may cause disordered texture and distorted structure in the repaired image. Thus, we combine the simple SSD with the Bhattacharyya weighted distance function in the study by Bugeau et al. (2010). We define the distance between two patches $A = \mathcal{N}(a)$ and $B = \mathcal{N}(b)$ as follow:

$$D_m(A, B) = D_{SSD}(A, B) \cdot D_{BC}(A, B), \tag{2}$$

where

$$D_{SSD}(A, B) = \sum_{i,j=1}^{h} ||A_{i,j} - B_{i,j}||^2, \tag{3}$$

$$D_{BC}(A, B) = \begin{cases} 1 & \rho_A = \rho_B, \\ \sqrt{1 - \sum_{i=1}^{h} \sqrt{\rho_A(i) \cdot \rho_B(i)}} & \text{else.} \end{cases} \tag{4}$$

ρ_A and ρ_B are the histograms of the patches. The distance function is defined as the multiplication of the simple SSD and Bhattacharyya metrics. Unlike the distance function defined by Bugeau et al. (2010), the Bhattacharyya metric is embedded into a piecewise function in our definition. Our definition can avoid the null measure in situations in which two patches share the exact same distribution. In these situations, our distance function will automatically degrade into a simple SSD.

E_1 constrains the similarity between the patches within the hole and those in the source region. As shown in Figure 7(b), our approach searches suitable patches in the source region and calculates the distance between the target patch and the source patches. The optimum patch minimizes the following energy term:

$$E_1(Q, P) = D_m(Q, P). \tag{5}$$

Note that the distance between two patches is calculated in both the luminance channels and the color channels.

E_2 encodes the coherence between the patches and its neighboring patches within the hole. Given a patch in the hole Q, its adjoining patches Q' share common pixels with Q. As shown in Figure 7(b), the overlapped region between the patches are considered when calculating E_2. The energy term that measures the patch differences in the overlapped regions is:

$$E_2(Q) = \sum_{Q' \in C} D_m(Q, Q'), \qquad Q \cap Q' \neq \emptyset. \tag{6}$$

The weighted factors w_1 and w_2 balance the influence of internal and external coherence in the objective function. These factors are adjustable for users to tune the final repaired results. In general, it is recommended that w_1 is set larger than w_2 to avoid trivial solution. Noted that patches may repeat too many times in the hole when w_2 is set much larger than w_1.

3.4 Iterative optimization with dynamic patches

Directly achieving the global optimum of the objective function (Equation 1) is intractable because of the massive solution space and time-consuming energy term evaluation. The objective function is non-convex with many local minima. A practical choice is to obtain approximate solutions with iterative optimization scheme. The basic idea of this scheme is to initialize every iteration with the result from the previous iteration and constrain the objective function from increasing in each iteration. Wexler et al. (2007) developed a "search & vote" iterative scheme for image completion. Our approach also uses a "search & vote" scheme with dynamic patches. In the search phase, a parallel search with a competitive mechanism is conducted to find suitable patches. In the vote phase, the graph cuts technique (Kwatra et al., 2003) is used to

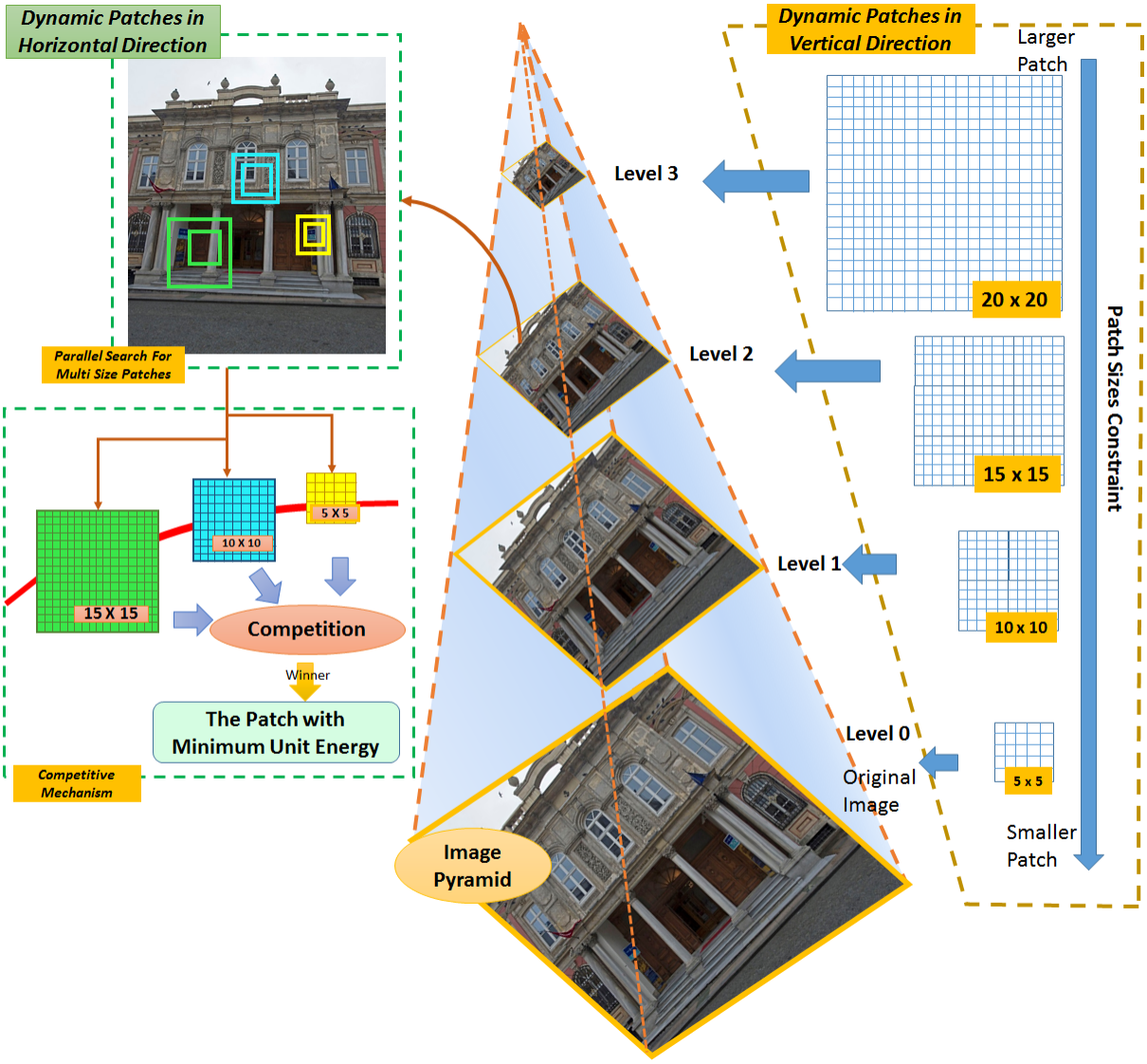

Figure 8: *Overview of the dynamic patch system. Our dynamic patches system takes effect in two directions of the image pyramid. In the horizontal direction, we conduct a parallel search for various size patches at each pyramid level. In the vertical direction, we constrain the algorithm to apply large patches at the higher pyramid level and small patches at the lower pyramid level.*

pursue a seamless connection between patches. The pseudo code of our iterative optimization is shown in Algorithm 1.

3.4.1 Parallel search with competitive mechanism

Parallel Search- PatchMatch is a randomized correspondence algorithm for effective patch searching and image editing. It begins with a random filling of the hole in the image and then rapidly seeks approximate nearest-neighbor matches between source patches. In this process, good matches are propagated, and a random search is executed to discover potential matches. To search for patches of multiple sizes, we make several modifications based on the PatchMatch algorithm. We enhance the algorithm with a parallel search for various size patches within the known region. The number of the patch sizes v is user-defined. It is obvious that the more different sizes of patches searched, the higher the computational cost; thus requiring more computing power and more memory to store patches. In our approach, Graphics Processing Unit (GPU) is used for computation. GPU is a multi-thread, highly parallel processor developed to process massive amount of data. The stream of GPU processing conducts the multi-size patch search in the form of parallel acceleration. A parallel programming model with shared memory is used to search for various size patches while taking advantage of the General-Purpose computing on Graphics Processing Unit (GPGPU) ability emerged in Compute Unified Device Architecture (CUDA). Various size patches found in this phase are stored as candidates for the competition in the next stage

Competitive Mechanism- Candidate patches must compete against each other to be the most suitable patch for a certain location. The competition proceeds by calculating the unit energy term U_{h_i} of every $h_i \times h_i$ patch:

$$U_{h_i} = \frac{D_m(P_{h_i \times h_i}, Q_{h_i \times h_i})}{h_i^2}. \tag{7}$$

The patch with the lowest U_{h_i} is selected to enter the voting phases. Supposed that v candidates attend the competition, the winner satisfies:

$$P_{win} = \arg\min\{U_{h_i}\},\ i = 1, 2 \ldots v. \tag{8}$$

In our experiments, v is set to three, which means that patches of three different sizes are searched at the same time at every pyramid level, and the one with the lowest unit energy is the most suitable. After the winning patches are selected at each location of the hole, they are blended to generate the contents of the hole.

The parallel search and competitive mechanism is part of our DPS and enables the use of patches of multiple sizes to repair damaged images. The use of patches of multiple sizes enables us to capture structures and textures on various scales. Previous methods prefer large patches with a fixed size, which easily absorb irrelevant background information in structure restoration. As shown in Figure 9, the branch of a tree is captured with a series of 10×10 patches, and much irrelevant information (such as the background and the part of the bird) is included. The problem with containing irrelevant information is significantly alleviated by the mixed use of three different sizes of patches. Besides, the use of patches of multiple sizes can lower the computational cost, because smaller patches contain fewer pixels and reduce the calculation required for energy evaluation.

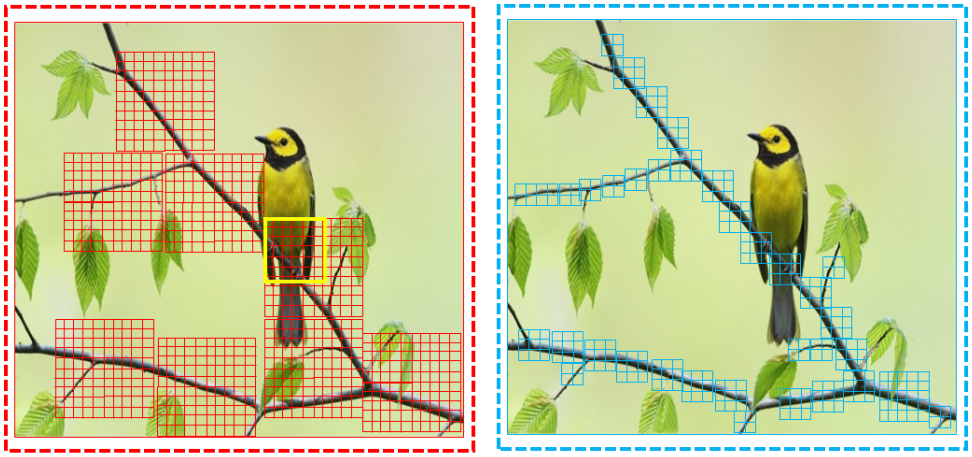

Figure 9: *Multiple size patches for structure preservation. Large patches are not always the best choice for structure preservation. (a) We attempt to capture the structure of the branches with* 10×10 *patches. Part of the bird's body and the leaves are absorbed into the patches. (b) We attempt to capture the same structure with a series of patches in different sizes (*2×2*,* 3×3*, and* 5×5*). The intake of irrelevant information is significantly reduced.*

3.4.2 Vote phase with graph cuts

The contents of the hole are generated by combining all the winning patches retrieved from the search phase. In the method of Wexler et al. (2007), the color of the pixel is decided by the median of all votes in a weighted voting scheme. This scheme works well when using patches of a fixed size. Nevertheless, it could not solve the problem of discontinuity when patches of various sizes are combined. If the overlapping areas of patches of various sizes are not handled properly, textural disorder or structural fracture may occur. Thus, we apply the technique of graph cuts (Kwatra et al., 2003) to connect partly overlapped patches seamlessly in the voting phases. Suppose that two patches, Q_1 and Q_2, overlap along their vertical edges (Figure 7(f)); let s and t be the adjacent pixel positions in the overlap region and $Q_1(\cdot)$ and $Q_2(\cdot)$ be the corresponding pixel value at the specific position. Note that the two patches only partially overlap, which means $Q_1 - Q_2 \neq \emptyset$. The following adaptive matching quality cost M proposed by Kwatra et al. (2003) is minimized to cut the overlap region that makes two patches match best:

$$M(s,t,Q_1,Q_2) = \frac{||Q_1(s) - Q_2(s)|| + ||Q_1(t) - Q_2(t)||}{||\nabla^d_{Q_1}(s)|| + ||\nabla^d_{Q_1}(t)|| + ||\nabla^d_{Q_2}(s)|| + ||\nabla^d_{Q_2}(t)||}. \tag{9}$$

Here d is the direction of the gradient and is the same as the edge direction between s and t. $\nabla^d_{Q_1}$ and $\nabla^d_{Q_2}$ are the gradients in the patches along direction d. This adapting matching quality cost penalizes less on seams that cross high-frequency regions than those that cross low-frequency regions. The optimum seam with minimum cost M can be obtained by solving the path-finding problem using graph cuts. Assume that we would like to find a minimum cost patch through the 3×3 overlap region in the graph shown in Figure 7(f). All adjacent pixel nodes are connected with arcs that are labeled with the adaptive matching quality cost $M(s,t,Q_1,Q_2)$. Two extra nodes are included to represent patches Q_1 and Q_2 and are connected to the adjacent nodes with constraint arcs. The cut of the minimum cost in the graph is found by separating node Q_1 and node Q_2. This is a min-cut problem (Ford Jr & Fulkerson, 2015) that is a classical graph cut problem, which has efficient solutions and easy implementation.

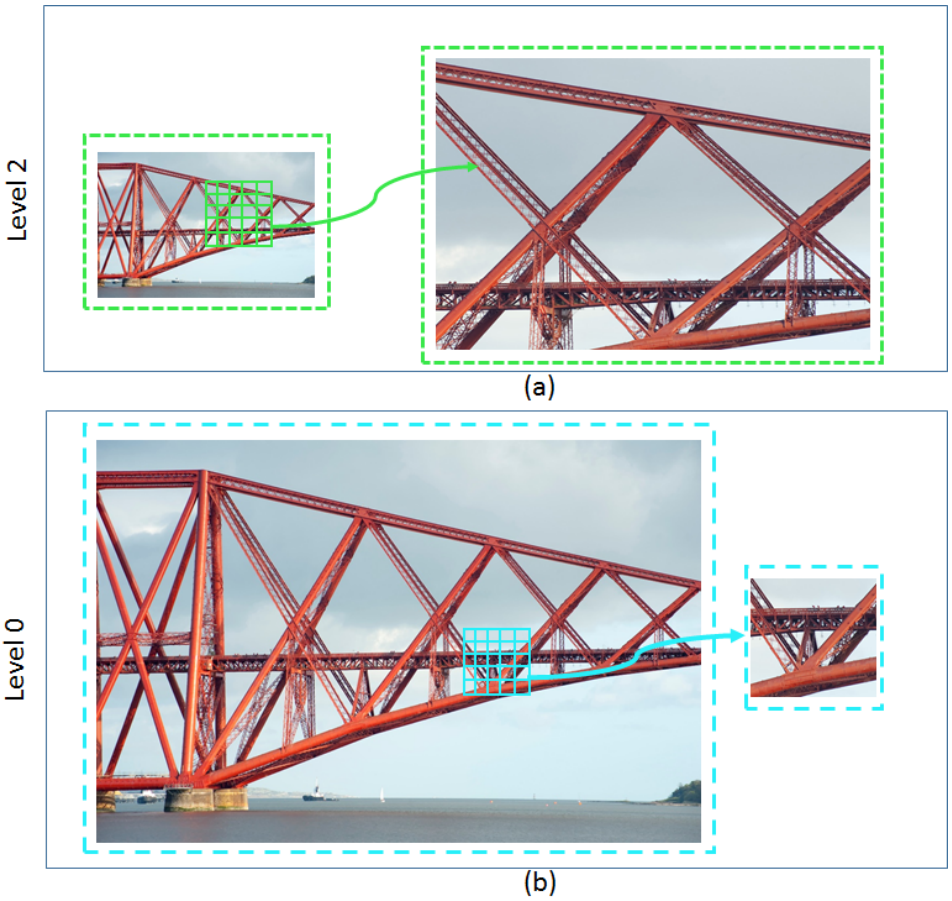

Figure 10: *The use of large patches at a higher pyramid level can preserve more structure information than at a lower pyramid level. (a) A* 5×5 *patch on a bridge picture at* 660×440 *resolution. (b) Same patch is applied to the same picture at* 2200×1464 *resolution.*

3.5 Multiscale solution with dynamic patches

Similar to previous image editing methods, our approach is also implemented in a coarse-to-fine fashion to accelerate convergence and strengthen global consistency. In our approach, the input image is resized into L different resolutions to form an image pyramid. The "search & vote" iterations are conducted at each pyramid level from top to bottom. The patch size is scalable in the "search & vote" process. Additional constraints on the sizes of the patches at each pyramid level are introduced to capture structural and textural information efficiently. As shown in Figure 8, the sizes of patch vary along the vertical direction of the pyramid and adapt automatically to the coarse level of the pyramid.

The constraints we set at each pyramid level limit the max/min size of the patches. Let h_{max} be a user-defined parameter that determines the maximum size of patches allowed in all L pyramid levels. The h_{max} is adjustable according to hardware and application. We use a discrete function $K(\cdot)$ to determine the size of the patches with a given pyramid level l_i. In our approach, $K(\cdot)$ is defined as:

$$K(l_i) = \lfloor \frac{h_{max} - v}{1 + e^{-l_i \times \beta}} \rfloor, \tag{10}$$

where β is a parameter that controls the intervals of different values, and $\lfloor \cdot \rfloor$ is the floor operator. Given that h_i is the size of the ith patch, the size of the patches applied in pyramid level l_i satisfy:

$$\begin{cases} Min(h_1, h_2, \cdots, h_v)_{l_i} \geq K(l_i), & \text{for } l_i > L/2; \\ Max(h_1, h_2, \cdots, h_v)_{l_i} \leq K(l_i), & \text{for } l_i < L/2. \end{cases} \tag{11}$$

Equation 11 constrains the size of the patches that can be applied in coarse level l_i. Large patches are guaranteed to be applied at a higher pyramid level (which means in an image with a low resolution), and small patches must be used at a lower pyramid level (which means in an image with a high resolution).

The form of $K(\cdot)$ can be modified according to real-world applications. Equation 10 is just one of the optional forms. When designing $K(\cdot)$ into different forms, one should consider the monotonicity of $K(\cdot)$. The $K(\cdot)$ we used implies that patch sizes shrink along the pyramid level falls. That is, when $K(\cdot)$ is monotonically increasing, a series of shrinking patches are

applied along the vertical direction of the pyramid. In contrast, when $K(\cdot)$ is monotonically decreasing, a series of enlarging patches are applied at each pyramid level from top to bottom. In this situation, large patches are used at lower pyramid levels.

Our approach limits the patch sizes at each pyramid level with a monotonically increasing $K(\cdot)$. In other words, large patches are applied at higher pyramid levels, because large patches are more effective in capturing structures in coarse images. We compare the structural information captured in two pyramid levels with identical patches, which are represented by the grids in Figure 10. An identical 5×5 patch is applied to the same image at two different scales, 660×440 and 2200×1464. As shown in the figure, a 5×5 patch is able to capture the major structures of the bridge on the image with a lower resolution but the same patch can only include the minor local structure of the bridge in the 2200×1464 image. This example suggests that large patches can capture more information at a higher level than that at a lower pyramid level. Reducing the use of large patches at a lower pyramid level is more economical when considering the computational workload, because the use of large patches at a lower pyramid level has limited enhancement on the structure reconstruction but consumes more computational resources than the use of small patches.

3.6 Evaluations

We verify the performance of our approach by repairing images that contain various types of textures and structures. Our approach is compared with some well-known methods, including ImageMelding (Darabi et al., 2012), and the methods of Wexler et al. (2007) and of Criminisi et al. (2003) and of Iizuka et al. (2017). The program in our experiments is a collection of C++ and Matlab functions, which is run on an Intel Xeon E5-2470 V2 2.40 GHz computer with 8G RAM and an AMD R9 280 graphics card. In the following subsections, the visual effectiveness and run-time performance of our approach are discussed. Besides, two parameter analyses are presented. One discusses the internal and external coherence balanced by w_1 and w_2, and other analyzes the influences of the patch sizes in the vertical direction controlled by $K(\cdot)$. In addition to the subjective visual comparison, a user study is conducted to evaluate the repaired image generated with our approach.

Algorithm 1 Dynamic Patch-based Image Completion

Input: Image I, cavity C, source $S = I - C$, Number of different size patches v, Pyramid level L

Output: Final Image F

Initialize F by filling patches randomly
Compute image pyramid I_{l_i},C_{l_i},$K(l_i)$, $l_i = L, L-1, \ldots, 0$
for each pyramid level l_i **do**
    Define the patch sizes with Equation 11
    repeat
        for All $q \in C$ **do**
            Parallel Search for v different size patches
            Retrieve the patch P that satisfies Equation 8
        end for
        Calculate the minimum cost boundary
        Combine all patches
    until convergence
    Propagate solution to the next level
end for

3.6.1 Effectiveness

Figure 11 presents the repaired results by testing our approach to images that contain complex structures and textures. Our approach is also applied to the task of object removal, and the results are shown in Figure 12. The results show the importance of proper restoration of structures and texture in the overall effect of the repair images. The results generated by the compared methods fail due to the artifacts in the textures and distortions in the structures. When repairing complex structures in the images, even the state-of-the-art ImageMelding (Darabi et al., 2012) fails to generate reasonable structures inside the hole. As shown in Figure 11(d), apparent fractures in the body of the architecture makes the repaired contents unacceptable. Unlike the results of previous methods, our approach successfully preserves structures smoothly with the aid of DPS. As seen in Figure 11(e), the structural information is well captured and properly reproduced. When compared to the deep learning based methods by Iizuka et al. (2017), our results have advantages in image details.

The distance function used in our approach is a combination of the Bhattacharyya distance and the simple SSD. The combined distance and the use of various size patches is helpful to maintain sufficient internal coherence within the hole. Our repaired results remarkably avoid the interfering information in the repair process. The images of the church in Figure 11 are examples. The textured region of the church contains some elements that are easily absorbed into the hole and cause an artifact. These irrelevant information easily interfere the contents of the hole generated by previous methods, and generates unpleasant textures, as shown in Figure 11(d). Our approach successfully avoids the disturbing information and generates images that preserves great coherence with the known region (as shown in Figure 11(f)).

3.6.2 Efficiency and time complexity analysis

Vision problems are commonly formulated into an energy minimization problems, in which the global optimum of the objective function is often difficult or impractical to find. In our approach, the objective function is ensured not to increase in each iteration of the "search & vote" scheme. The energy statistics throughout the iterations of the repair process are presented

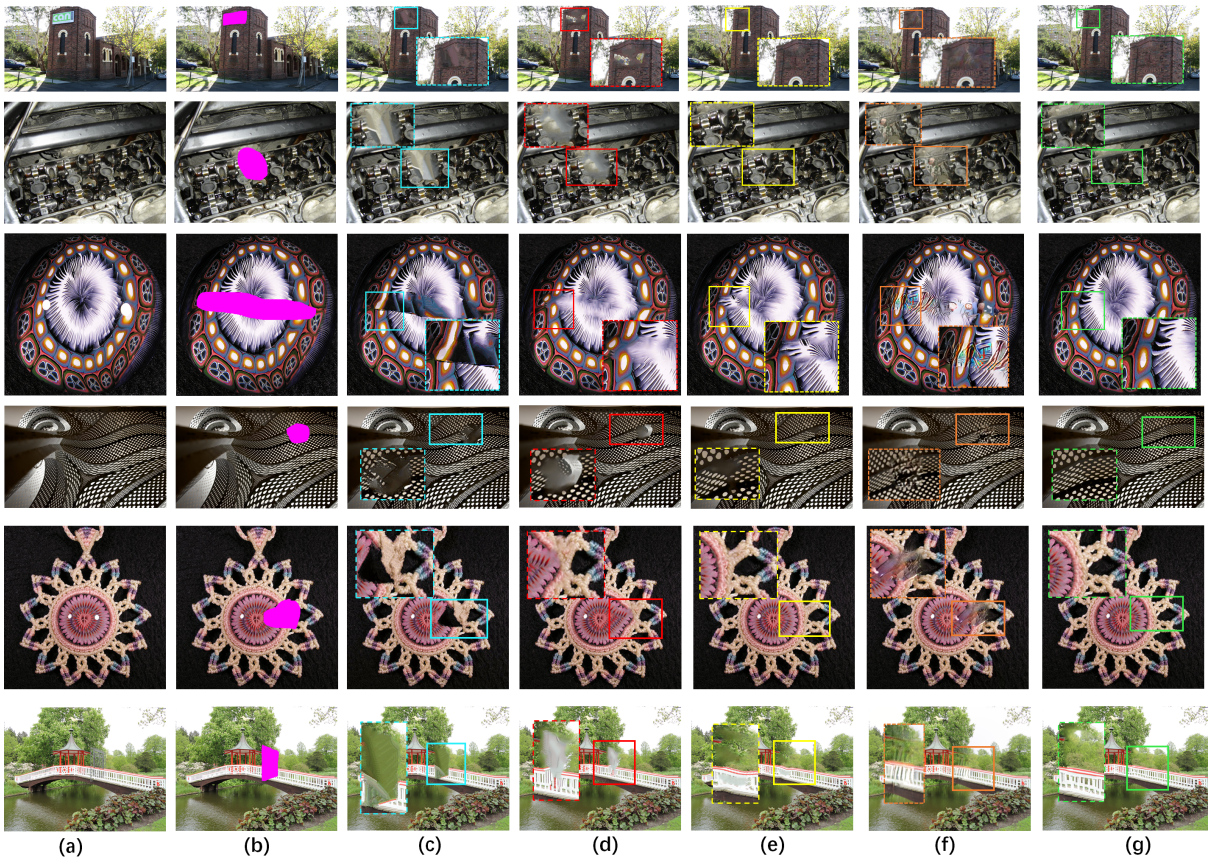

Figure 11: *(a) Original image. (b) Masked image to complete. (c) Results generated with the method of Criminisi et al. (2003). (d) Results generated with the method of Wexler et al. (2007). (e) are the results generated with ImageMelding by Darabi et al. (2012). (f) are the results generated with the method of Iizuka et al. (2017). (g) Results with our approach.*

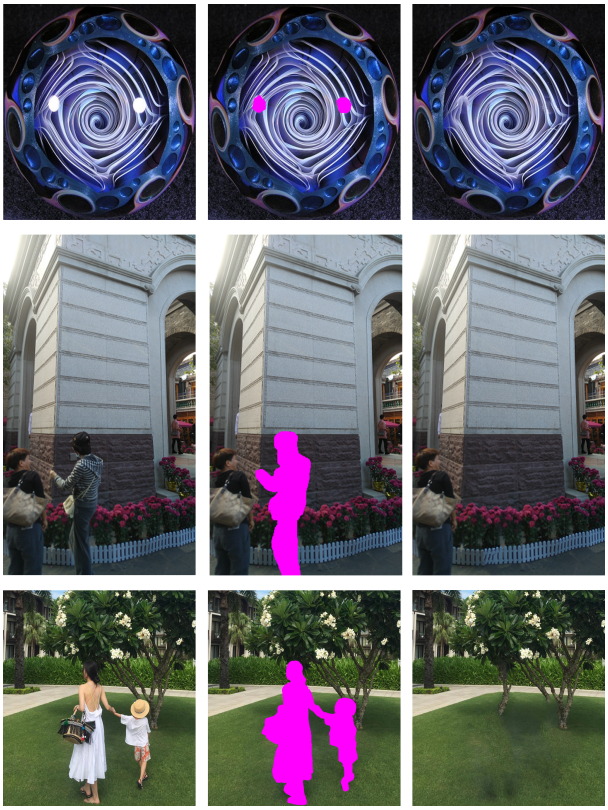

Figure 12: *Our approach performs well on object removal tasks. Images in the left column are the ground truth, and those in the right column are our results. Images in the middle column are masked images.*

Table 1: *Run-time performance of completing the images in Figure 11. With the GPU acceleration, our approach consumes less time than previous methods in general.*

	Church	Engine	Crystal	Net	Heart	Bridge
Image size (pixels)	800×450	800×600	480×480	2000×1333	400×400	730×548
Time Cost (s):						
Criminisi et al. (2003)	2408.69	5001.221	3419.57	19931.94	1151.11	3505.27
Wexler et al. (2007)	1852.84	3847.093	2630.44	15332.26	885.46	2696.36
ImageMelding (Darabi et al., 2012)	822.663	1923.546	1122.996	7118.54	411.110	1251.882
Iizuka et al. (2017)	1.25	2.25	1.75	3.75	1.45	2.33
Ours (without acceleration)	1245.112	2676.238	1699.669	9199.356	531.281	1617.816
Ours (with GPU acceleration)	741.138	1672.649	1011.708	5475.807	316.239	962.986

in Figure 13. Although there are energy inflations in the initialization at each pyramid level, the line chart shows that the energy decreases continuously in each iteration and becomes stable after finite iterations. Although such an approximate scheme cannot guarantee that the objective function arrives at the global optimum, our experimental results are visually pleasing.

The time complexity of our approach depends on various factors such as image size, patch sizes and the number of sizes. We are not able to provide an accurate general conclusion on time complexity. But we can still discuss the time complexity of our approach under a given condition. The two algorithms we used in the two-phases optimization have different time complexities. Here we analyzed the time complexities of approach phase by phase. Supposed that a damaged image contains M pixels, the patch contains $h \times h$ pixels. The time complexity in search phases is $O(h^2 M \log M)$ in search phases with PatchMatch algorithm. Since our approach allows the search of v different size of patches, thus the time complexity of search phases increases to $O(vh^2 M \log M)$. In the vote phases, the graph cut techniques are used and its time complexity depends on the overlap region between patches. For a graph of n nodes, the graph cut has a worst-case computational complexity of $O(n^2)$ and an average of $O(n \log n)$.

From the time complexity analysis, we can see that our approach may have higher time consumption due to the search of various size patches. In our experiment, we programmed our approach that run on CPU only to conduct a fair comparison with other methods. The statistics

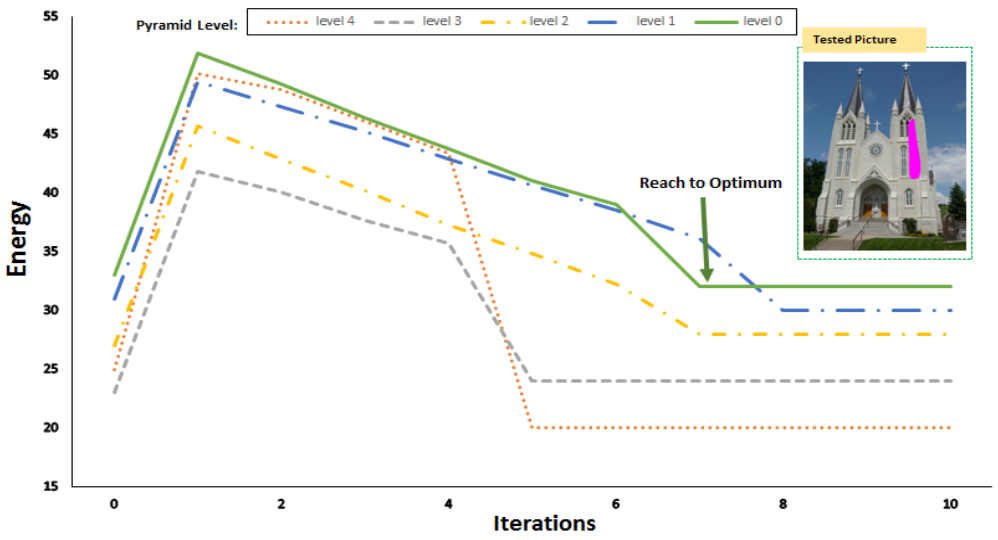

Figure 13: *Energy evaluation. Energy variation of the objective function on a picture of a church is presented. Note that when the result generated by the previous pyramid level propagates to the next pyramid level, the energy inflates. This inflation is caused by the increase in the number of pixels when the resolution increases. The tendency of the energy also provides cues of convergence. When the energy stops decreasing, it suggests that image is complete and has reached to the optimum.*

are presented in Table 1. Our approach is much faster than the method of Wexler et al. (2007) and that of Criminisi et al. (2003). But our approach is slower than the patch-based method of Darabi et al. (2012) due to high computational cost caused by the search for patches of various sizes. Thus, we provide another version of our approach that is implemented with parallel programming techniques. With the acceleration of the GPU, our approach saves approximately 68% of the time cost. Note that our approach can be further accelerated if programmed to maximize the hardware capacity. Also, from Table 1, we can see that the deep learning based method by Iizuka et al. (2017) outperform all the other traditional completion methods and has achieved nearly real-time performance. However, such excellent performance is based on time-consuming training. According to the information reported in the paper (Iizuka et al., 2017), it takes approximately 2 months to train the network.

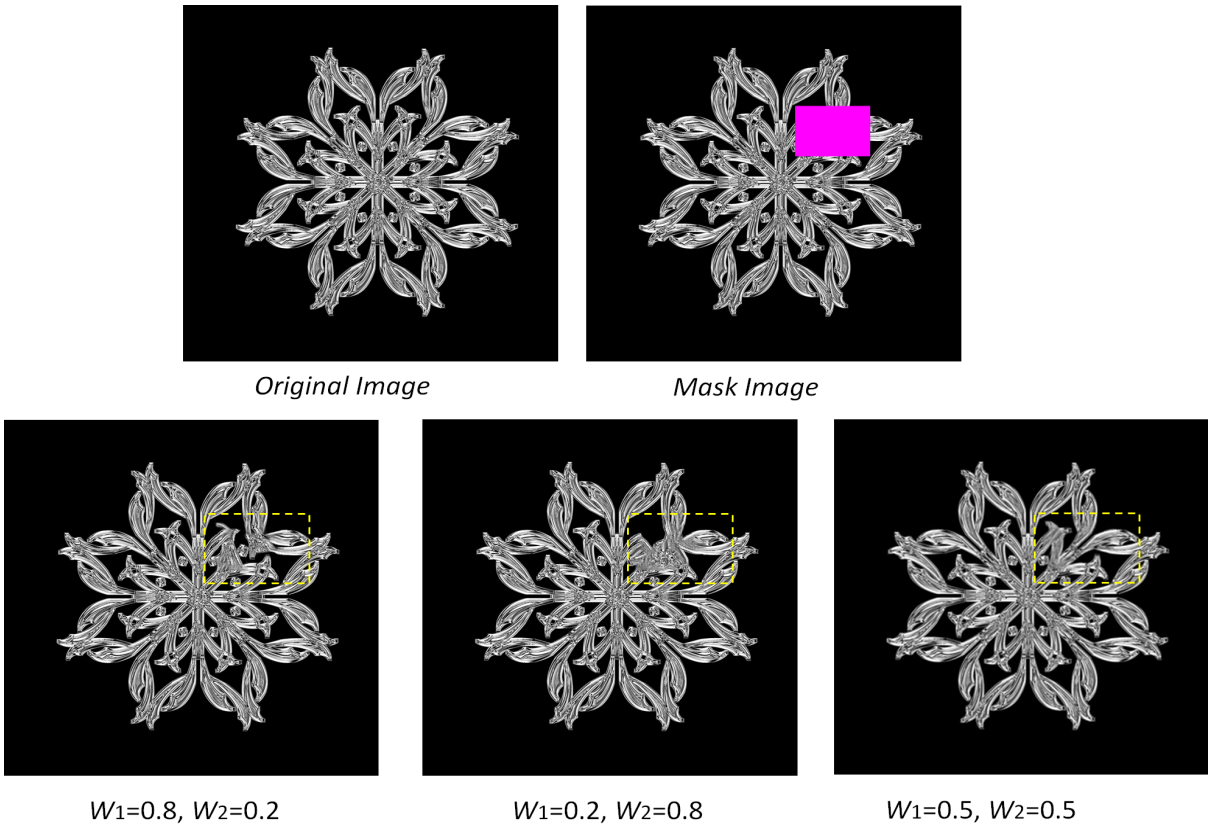

Figure 14: *Results generated by different configurations of* w_1 *and* w_2*. We use the Twenty-Eighty Law as a criterion to control the ratio between* w_1 *and* w_2*. Notice that, when* $w_1 = 0.8$*,* $w_2 = 0.2$*, fracture occurs in the petal of the ice crystal. With the configuration of* $w_1 = 0.2$*,* $w_2 = 0.8$*, the structure is blurred and an artifact occurs.*

3.6.3 External coherence vs internal coherence

The weight factors w_1 and w_2 control the balance between internal and external coherence. Both user-defined coefficients and are adjustable according to the pictures. Figure 14 presents the results generated under various configurations of these weight factors. An objective function with a higher w_1 gives more preference to external coherence, but when w_1 is weighted much larger than w_2, distorted structures and disordered texture may appear, as shown in Figure 14. A larger w_2 gives the objective function a greater emphasis on internal coherence. However, an oversize w_2 may blur contents in the repaired image slightly. Because some patches are reused for too many times. A possible criterion for setting these two weight factors may be the Twenty-Eighty Law. However, a balanced configuration ($w_1 = w_2 = 0.5$) of the weight factors is recommended. As shown in Figure 14, the results generated with this configuration are realistic and has sharp structures.

3.6.4 Enlarging patches vs shrinking patches

An experiment was conducted in which enlarging patches were used in the vertical direction of the image pyramid. The repair results are compared with those generated with our approach using shrinking patches. To use a series of enlarging patches in the vertical direction of the image pyramid, $K(\cdot)$ is set to be a monotonically decreasing function. As illustrated in Figure 15, the repaired contents in the hole are blurred when enlarging patches are used. Some of the textural information is lost, which causes an obvious discontinuity in the structure. In terms of efficiency, the time consumption of the image completion with enlarging patches is 22.82% greater than that with shrinking patches (Enlarging patches, 1783.612 seconds; shrinking patches, 1452.207 seconds). The difference in time performance can be explained with the use of larger patches in low-resolution scales, in which patch searching and filling are much faster. Thus, the use of large patches is more economical at a higher image pyramid level. In addition, the use of large patches to capture more information in the coarse image can provide a sound foundation for completion at the next pyramid level. The intermediate outcomes from two pyramid levels are shown in Figure 15, and it is easy to see a significant difference in visual

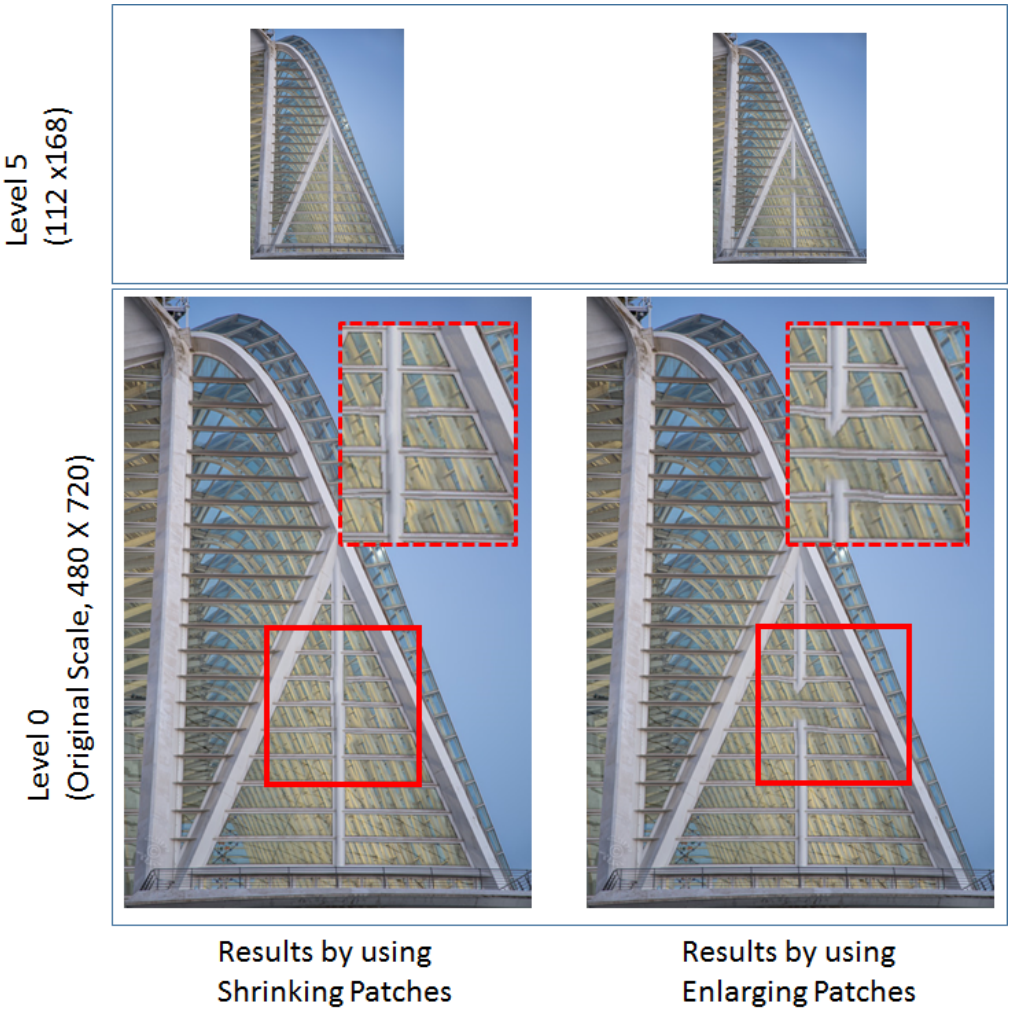

Figure 15: *Comparison of the effect of using enlarging patches and shrinking patches. Pictures on the left are results generated using a series of shrinking patches. Large patches are utilized at a high level of the image pyramid. As shown, most structures are captured at level 5, and provide a sound foundation for the iteration in lower level. The results on the right are generated using a series of enlarging patches. The size of the patches increases along the vertical direction of the image pyramid. Fracture of the structure occurs, and the loss cannot be compensated with the use of large patches at a lower pyramid level.*

effect. The use of large patches at a lower pyramid level cannot compensate for the poor results by the previous iterations and costs extra computational resources. In sum, the use of shrinking patches along the vertical direction of the pyramid is a better choice.

Table 2: *User Study Statistics. The scores difference between our approach and other previous methods are presented in the third column of the table. Note that the lower and upper bounds of the estimated differences do not contain zero.*

Methods	Mean	Standard Deviation	Mean Difference with our approach	95% Confidence Interval	
				Lower Bound	Upper Bound
Our approach with DPS	7.983	0.829	-	-	-
ImageMelding (Darabi et al., 2012)	6.941	0.812	1.041	0.711	1.372
Wexler et al. (2007)	5.500	0.502	2.483	2.153	2.813
Criminisi et al. (2003)	5.230	1.459	2.750	2.419	3.080

3.6.5 User study

To verify whether our approach generates better repaired images than previous methods from the user point of view, a user study was included in our experiments. Thirty subjects were required to rate the texture and structure of the repaired image in a 9-point scale (with nine as the best). Again, we compared our approach with ImageMelding (Darabi et al., 2012), the method of Criminisi et al. (2003), and the method of Wexler et al. (2007). Each repaired image was presented to the subjects, and the rating was conducted without knowledge of the exact technique applied. Our questionnaire contains four sets of test images (16 images in total), and we collected four hundred eighty data samples for analysis. The descriptive statistics of the collected data are shown in Table 2.

From Table 2, ImageMelding (Darabi et al., 2012), the method of Wexler et al. (2007), the method of Criminisi et al. (2003) and our approach are $6.94, 5.50, 5.23$ and 7.98 respectively. An ANalysis Of VAriance (ANOVA) is conducted to verify the presences of a statistically significant difference between the means. The result among these four groups is $F(3, 476) = 213.142, p < 0.001$, in which the F value is the test statistic that reflects the significance. It suggests that a significant difference exists between the means of these four groups. Then in

the post hoc test, the Mean Difference (MD) between our approach and ImageMelding ($MD = 1.042, p < 0.001$), the method of Wexler et al. (2007) ($MD = 2.48, p < 0.001$), the method of Criminisi et al. (2003). ($MD = 2.78, p < 0.001$) are calculated. The results are significant and greater than zero, which suggests that our approach obtains higher scores. From the 95% confidence interval, the result of statistical analysis concludes that the images repaired with our approach are more perceptually pleasing than those with other methods.

3.7 Discussion

In this chapter, we present a patch-based approach for image completion with efficient structure preservation using DPS. Unlike previous methods with fixed-size patches, our approach equipped a DPS, which allows patch sizes to change in both the horizontal and vertical direction of the image pyramid. The introduction of DPS enables the patch-based completion framework to capture structures and textures in various scales economically. The repaired images with our approach do not suffer from structure distortion and texture disorder. Experimental results and user study reflects that our approach outperforms previous methods and that the repaired image is more visually appealing. In the future, we plan to apply our approach to other image-processing tasks, such as satellite image processing and image cloning, which can be formulated into an image completion framework. Our approach may also be improved by optimizing the match propagation path in the patch search process.

4 Color-Consistent Shadow Removal From an Image Completion Prospective

4.1 Introduction

Shadow is a common phenomenon in natural scenes. Shadows usually form on the surface of objects when the light source is occluded. In addition to the illumination cues, shadows can provide valuable information for human perceptions, such as the object shape and the spatial relationship with the ground. However, shadows in images and videos cause extra difficulties in most computer vision tasks such as visual tracking (Cucchiara, Grana, Piccardi, & Prati, 2003), object recognition (Swain & Ballard, 1991), and image segmentation (Klinker, Shafer, & Kanade, 1990). Removal of shadows from images is usually included as a pre-processing step in many computer vision tasks to pursue robustness.

Many methods have been proposed to accomplish the shadow removal task. Previous shadow removal methods can be classified into relighting methods, reintegration methods, and color transfer methods. Relighting methods (Arbel & Hel-Or, 2007; Du, Lin, & Bao, 2005; R. Guo et al., 2013; He, Zhen, Yan, & Ge, 2017) remove shadows by adding a constant factor to the shadow pixels in the log domain. In relighting methods, estimation of the constant factor plays an important role in the final visual effect. Such estimation can be difficult when processing surfaces with irregular texture and non-uniform shadow (e.g., Figure 16(b)). Reintegration methods (Finlayson, Hordley, & Drew, 2002; Finlayson et al., 2006; Finlayson & Fredembach, 2004; M. Xu et al., 2017) generate shadow-free images by nullifying the derivative of the pixels and integrating. These methods depend on accurate shadow edge detection. A wide penumbra area of the shadow and complex color pattern may seriously affect the visual effect of the shadow-free images, as shown in Figure 16(c). Color transfer methods (Shor & Lischinski, 2008; Wu & Tang, 2005; Wu, Tang, Brown, & Shum, 2007) handle the shadow removal task as a color-correction problem. The shadow-free images are achieved by directly transferring a similar color from a lit region to the shadow region. Usually, the color transfer operation was conducted separately in different color channels, assuming that the channels are independent. Nevertheless, in situations in which this assumption is violated, inconsistency in color and unnatural texture may appear, as shown in Figure 16(d). Figure 16(e) shows the shadow-free

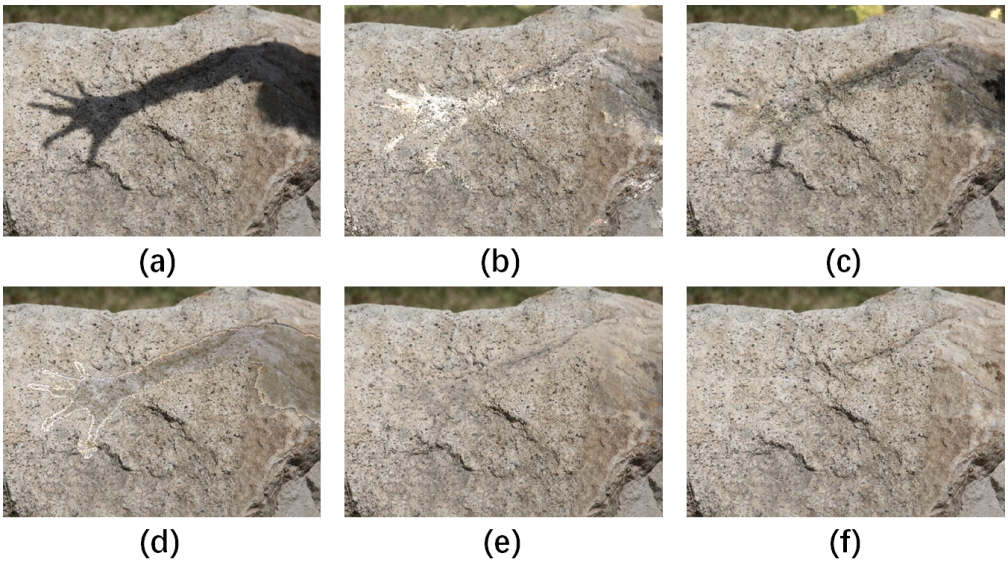

Figure 16: *(a) Original image. Its resolution is 840 × 640. (b) Result with shadow removed with the method of R. R. Guo et al. (2013). Complex color pattern and irregular texture make it difficult to find a fixed factor for shadow removal. (c) Result by Gong and Cosker (2017). Some unnoticeable penumbra regions may seriously affect the results. (d) Result by L. Zhang et al. (2015). Although the shadow region is lightened, the deviation in color makes the lit region visually incompatible with the original part. (e) Our approach. (f) Ground truth.*

image generated by our novel framework proposed in this chapter. The generated result in the shadow region is consistent in both color and illumination, and is completely merged with the lit region.

4.2 Method overview

We consider the shadow on an image as an impairment in the illumination field. Based on this insight, we adapted the shadow removal task as an image completion framework. We first decomposed an image into an illumination field and a reflectance field. To guarantee the color consistency of the shadow-free images, we optimized the reflectance by propagating the color in the lit region to the shadow region while preserving the salient structure. We generated a guiding map for the following patch search phases according to the optimized reflectance field to maximize the use of color information. The guiding map indicates the location of the possible suitable patches. We then developed a patch-based optimization to

repair the illumination of the images. To achieve a consistent illumination effect in the image, we consider both spatial and textural information when recovering the luminance of the target patches. The search for suitable patches is conducted under the guidance of the optimized reflectance field. That is, the search for the suitable patches for a sub-shadowed region will be constrained in a sub-lit region with the same color. After illumination recovery was finished, we recombined the recovered illumination and optimized reflectance to generate the shadow-free image. Figure 17 provides an overview of our approach.

4.3 Shadow information model

According to the image formation equation proposed by Barrow and Tenenbaum (1978), an image $I(x, y)$ can be expressed as a pixel-wise multiplication of an illumination field $L(x, y)$ and a reflectance field $R(x, y)$ as follows:

$$I(x, y) = L(x, y) \cdot R(x, y). \tag{12}$$

Shadow is formed on an image because less light reaches the object surface than the non-shadow area. Thus, we can consider a shadow image as an image with an impaired illumination field. Unlike previous studies (Shor & Lischinski, 2008; Xiao, She, et al., 2013; L. Zhang et al., 2015) that decomposed the illumination field into a sum of direct and indirect illumination, we divided the illumination field into a source area s and a target area t:

$$L(x, y) = \underset{(x,y)\in s}{L^s}(x, y) + \underset{(x,y)\in t}{L^t}(x, y). \tag{13}$$

The source area $L^s(x, y)$ contains pixels that are not occluded in primary light sources. The target area $L^t(x, y)$ is the shadow area of interest. Note that the target area $L^t(x, y)$ can be defined according to a specific application (for example, it may only contain pixels in cast shadow). The partition of $L^s(x, y)$ and $L^t(x, y)$ depends on the shadow detection method. With the partition of the illumination field, we can formulate the shadow removal task into an image completion problem, that is, to repair the target region $L^t(x, y)$ based on the source region $L^s(x, y)$. To properly complete the impaired illumination region $L^t(x, y)$, we designed a patch-based illumination completion optimization that can use the spatial and textural information in

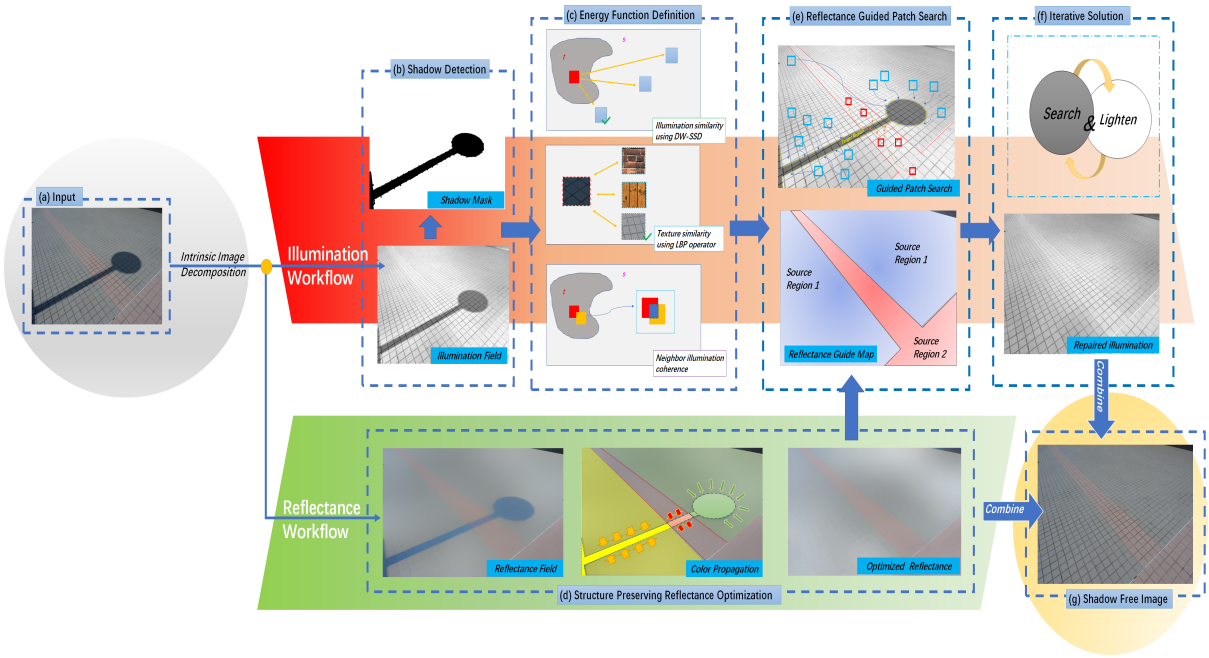

Figure 17: *Overview of our approach. (a) An input image is decomposed into its illumination and reflectance. These two fields will be processed in different workflows. (b) In the illumination workflow, shadow is detected based on its illumination and a shadow mask is generated. The decomposed illumination is then repaired with an energy minimization framework. (c) The definition of our energy function. Our energy function considers illumination similarity, texture similarity, and neighbor coherence simultaneously. (d) In the reflectance workflow, the color information in the lit regions propagates to the connected shadow region. (e) The reflectance is optimized before recovering the illumination to provide a reflectance-guided map for the patch search phases. The search for optimal patches is conducted only in each sub-source region only. (f) The optimum result of our energy function is achieved by "Search and Lighten" iterative optimization. (g) The repaired illumination and optimized reflectance are combined to generate the shadow-free image.*

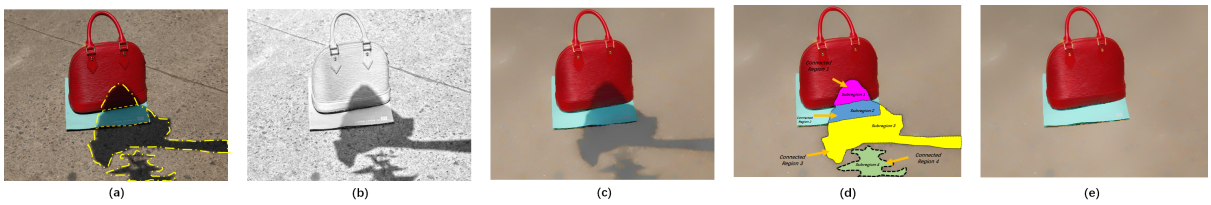

Figure 18: *(a) Original shadow image. The shadow region is divided into a series of subregions by some salient structures. (b) Illumination of shadow image. Most of the texture details are preserved. (c) The reflectance of the shadow image. The mark of the shadow remains, but the image is mainly composed of three color regions. (d) The color propagates from the lit region to the connected shadow region. (e) The result of the optimized reflectance.*

the image.

A precondition to achieve a successful shadow removal result based on this idea is proper intrinsic image decomposition. However, intrinsic image decomposition itself is also a challenging task. Our approach uses the automatic intrinsic image decomposition method described by J. Shen et al. (2013). The decomposition results are high-quality in most situations. However, some cases remain in which that the reflectance is not completely separated from the illumination. The influence of the shadow still exists in the image's reflectance field. A structure-preserving image completion technique is used to eliminate the mark of shadows in the reflectance field.

4.4 Shadow detection

The goal of shadow detection is to label the pixels in the shadow region. Shadow detection is also a challenging task, because various types of shadows may appear in the same scene. The shadow on a surface is a complex interaction of illumination, albedo, and geometry. According to the formation, shadows can be classified into self-shadow and cast shadow. For example, both a cast shadow and a self shadow appear in the image in Figure 19. Self-shadow refers to the shadow cast by the object on itself, and cast shadow is formed by occlusion of the light source by other objects. Even in the cast shadow, the umbra region and penumbra region have different properties. In the umbra regions, the intensity of the shadow is usually uniform, and the texture and color information of the underlying surface may be seriously attenuated in the

umbra region. In the penumbra region, shadow intensity is often non-uniform. To fully explore the information provided in the image, we used the automatic shadow detection network developed by Khan, Bennamoun, Sohel, and Togneri (2016). The designed neural network can automatically learn various features in the image for shadow detection. More specifically, the shadow binary mask M is achieved by a conditional distribution according to:

$$\mathcal{P}(M|I;w) = \frac{1}{Z(w)} \cdot e^{-E(M,I;w)}, \tag{14}$$

where w is the weight vector of the shadow-detection model, and $Z(w)$ is a partition function. The energy function $E(M, I; w)$ is a combination of a unary potential and a pairwise potential. The unary potential estimates the shadow properties both at the regions and at the boundaries, and the pairwise potential estimates the class and spatial transition.

Khan et al. (2016) proved that estimating the probability of Equation 14 via Maximum a Posteriori (MAP) is equivalent to minimizing the following energy function using the "margin rescaled algorithm" (Szummer, Kohli, & Hoiem, 2008):

$$M_{opt} = \operatorname{argmax} \mathcal{P}(M|I;w) = \operatorname{argmin} E(M, I; w). \tag{15}$$

With the minimization of the energy function (Equation 15), we can easily identify the shadow regions in input images. Specifically, the shadow mask we generated with the network is a binary map. Let (x, y) be the pixel locations in the image, and we define the shadow mask as:

$$M(x,y) = \begin{cases} 1, & \text{Shadow Region;} \\ 0, & \text{Lit Area.} \end{cases} \tag{16}$$

Let $\eta(x, y)$ be the attenuate factor that accounts for the attenuation of the illumination by the occluder within the shadowed area. Equation 13 can be rewritten as:

$$L(x,y) = \underset{(x,y)\in s}{L^s}(x,y) + L^{re}(x,y) \cdot M(x,y) \cdot \eta(x,y), \tag{17}$$

where L^{re} is the repaired illumination field achieved via our patch-based recovery method.

4.5 Structure preserving reflectance optimization

The separation of illumination and reflectance fields provides great convenience for shadow removal, because our direct operation in the illumination field can be free from the interfer-

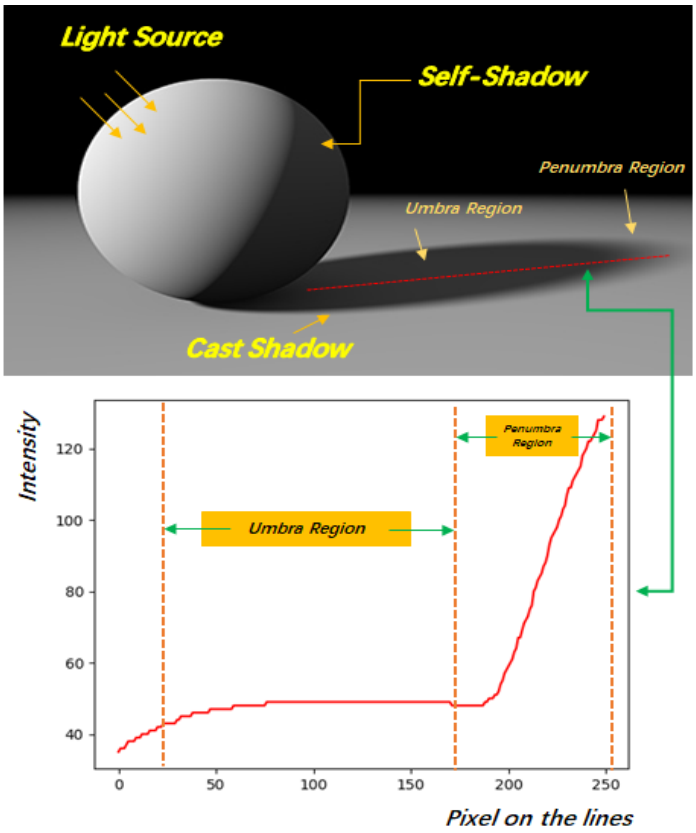

Figure 19: *Different types of shadow. There are two main types of shadow: self-shadow and cast shadow. Within the region of cast shadow, the shadow intensity also has different features. The line chart above shows the shadow intensity changes along the red lines. The intensity is more stable in the umbra region than in the penumbra region. The penumbra region is usually located on the edge near the lit region. Thus, boundary processing is necessary to generate visually pleasing shadow images.*

ence of color difference. However, in some cases that the reflectance field is not completely separated from the illumination field. As shown in Figure 18, although the illumination of the image is properly calculated, some marks remain in the reflectance field. This phenomenon is caused by remaining shadow influence on the reflectance. To develop a general method that can conduct intrinsic image decomposition under any lighting condition and camera setting is difficult. Thus, we prepare color optimization method to handle these marks.

Figure 18(b) shows that most of the texture details are preserved in the illumination field and that the reflectance is actually composed of a series of color segments. The color difference between the shadow region and the lit region still exists. If we directly combined the original

reflectance with the repaired illumination, the final result will have inconsistent color. Thus, the reflectance can also be regarded as a damaged image. However, repairing reflectance is much easier than repairing illumination, because the texture details are preserved in the illumination field, the remaining color information is useless for repair and the shadow region can be considered an empty hole. In fact, the repair of the reflectance is similar to color filling. Our approach first extracts the structure map of the reflectance using Canny operators and divides the shadow region into a series of subregions according to the structure. We then propagate the color in the lit region to the connected sub-shadow region. In the cases in which that any closed region is isolated within the shadow area, the users can assign a reference region or a specific color. To ensure that the structure is well preserved, the completion method of B. Liu, Li, Sheng, Nie, and Wu (2019) is used for reflectance optimization. It uses patches of multiple sizes to capture the structure accurately. The structure-preserving color optimization of the reflectance field is conducted before illumination recovery and provides guidance for the patch searching process.

4.6 Patch-based illumination recovery

Based on the shadow information model we proposed, we formulate the shadow removal process into a completion task in an illumination field. Given the source region in illumination field s and the shadow region t, our object is to complete shadow region t based on the textural and geometric information contained in the image. We pose this task as a patch-based optimization task by minimizing the following energy function:

$$E(L^t, L^s) = \sum_{Q \in t} \underset{P \in s}{min} (\lambda_1 E_1(Q, P) + \lambda_2 E_2(Q, P) \quad + \lambda_3 E_3(Q, Q')), \tag{18}$$

where E_1, E_2, and E_3 are energy terms for luminance, textures, and local coherence, respectively. These terms are defined in the following paragraphs. λ_1, λ_2, and λ_3 are weight factors that control the influence of the energy terms. In our experiments, $\lambda_1 + \lambda_2 + \lambda_3 = 1$. $Q = \mathcal{N}(x_q, y_q)$ is an $h \times h$ patch from the target region with an anchor pixel $Q(x_q, y_q)$ in its center, and $P = f(\mathcal{N}(x_p, y_p))$ is an $h \times h$ patch from the source region with transformation f applied. The center of the source patch P is defined as $P(x_p, y_p)$. All patches are extracted from the illumination field.

E_1 encodes the illumination similarities between the patches from the source region and those from the target regions, as shown in Figure 17(c). We use a Distance Weighted Sum of Squared Difference (DW-SSD) to measure the similarities in illumination. Let us denote $Q(i,j)$ and $P(i,j)$ as the corresponding pixel in patch Q and P, respectively. The energy term can then be expressed as:

$$E_1(Q,P) = \frac{\sum_1^h ||Q(i,j) - P(i,j)||^2}{||x_p - x_q||^2 + ||y_p - y_q||^2}, \tag{19}$$

where (x_q, y_q) are the coordinates of the anchor pixel of patch Q, and (x_p, y_p) are the coordinates for the anchor pixel of patch P. According to our observation, patches that are closed to each other usually share similar light conditions. Source patches that are near the target patches usually provide important cues for shadow removal. Thus, with our DW-SSD metric, preference is given to the patches that are closer to the target region if more than one suitable patch exists.

E_2 constrains the texture similarities between patches. Textural information is the most important cues for finding suitable patches in the source region. To compare the textural differences, we apply the LBP operator (Z. Guo, Zhang, & Zhang, 2010) to both the target patches and the source patches as it is shown in Figure 17(c). Given a pixel (x,y) with patch A that is central in (x_a, y_a), its LBP code of the pixel is obtained through:

$$LBP(A) = \sum_{i=1,j=1}^{h} \psi(A(i,j) - A(x_a, y_a))2^i, \tag{20}$$

where

$$\psi(u) = \begin{cases} 1, & \text{if } u \geq 0; \\ 0, & \text{if } u < 0. \end{cases} \tag{21}$$

Note that we apply the LBP operator directly on patches extracted from the illumination field. Because the LBP operator is not influenced by any monotonic gray-scale transformation, our approach requires no pre-processing with a Laplacian of Gaussian (LoG) operator as Ng, Chen, and Liao (2013), and our descriptor is free from the interference of color difference. We encode every pixel in both the source and target patches using the LBP operator in Equation 20. The texture similarities between patches can then be measured using their differences in LBP value. Thus, the energy term E_2 is defined with:

$$E_2(Q, P) = LBP(Q) - LBP(P). \tag{22}$$

E_3 constrains the consistency in illumination between adjacent patches. Given a target patch Q, its adjacent patch Q' shares common pixels with Q. This energy term is defined as simple SSD between their overlapped region, as shown in Figure 17(c). Let $Q(i, j)$ and $Q'(i, j)$ be the corresponding pixels in patches Q and Q', respectively. The energy term can be written as:

$$E_3(Q, Q') = \sum_{h}^{1} ||Q(i, j) - Q'(i, j)||^2. \tag{23}$$

The introduction of this energy term aims to avoid unnatural artifacts when combining various patches.

4.7 Iterative solution with reflectance-guided patch search

Directly optimization of the energy function (Equation 18) is impractical due to the massive solution space and time-consuming energy term calculation. The energy function is non-convex with lots of local minimum. An iterative solution scheme is usually applied to obtain approximate results. The basic idea of the iterative solution scheme is to initialize each iteration with the results generated with the previous iteration. The energy of the objective function is constrained not to increase in each iteration. Via finite iterations, the energy function can reach a stable status. In our approach, we developed a "Search and Lighten" iterative solution for the shadow removal tasks based on the frameworks of Wexler, Shechtman, and Irani (2004). The iterative solution attempts to reduce the energy of the objective function by repeating operations in two phases: the search phase and the lightening phase. In the search phase, we introduce a reflectance-guided patch search to efficiently locate the suitable patch. In the lightening phase, a local illumination operator is used to transfer the illumination information between matched patches. The shadow-free patches are then blended to reconstruct the target region. Algorithm 2 shows the pseudo-code of our iterative optimization.

Algorithm 2 Patch-based Illumination Recovery

Input: Source region in the illumination L^s , Shadow region in the illumination L^t which contains n subregions $L_1^t, L_2^t, \cdots L_n^t$, Reflectance guided map R' which contains m subregions $R'_1, R'_2, \cdots R'_m$.

Output: Recovered illumination L^{re}

Initialize L^t through filling patches randomly
for each sub shadowed region L_i^t **do**
    Locate the subregion R'_j which satisfy $L_i^t \subset R'_j$
    repeat
        for All $(x, y) \in L_i^t$ **do**
            Parallel Search for patches within the region $\{P|P(x_p, y_p) \in L^s \cap R'_j\}$
            Retrieve the optimal patch $P^* \in L^s$
        end for
        Conducted illumination transfer using Equation 24
        Combined all the patches
    until convergence
end for

4.7.1 Reflectance-guided patch search

An effective method to search for corresponding patches is the PatchMatch (Barnes et al., 2009, 2010), which is a tool for structural image editing using a randomized algorithm. The algorithm begins the search with random matching and continues to seek better approximate nearest-neighbor matches. In the search process, good matches will propagate, and a random search is conducted to avoid being trapped in a local optimum. To adapt the PatchMatch to our task of shadow removal, we develop a reflectance guide patch search based on the PatchMatch. In our task of shadow removal, the original patches in the target area are preserved in the first iteration instead of being replaced by random patches. Using the original patches for initialization can preserve the textural information contained in the target region. Also, the initial matching in the first iteration only considers texture similarities. In addition, to narrow the search space, we take advantage of the reflectance map to guide the search process. In fact, the color information contained in the reflectance is helpful in the search for similar patch. As illustrated in Figure 17(e), the target region can be contained in a monocolour region or divided by several color regions. We assume that patches with similar textures also share similar colors. Thus, the patch search should be conducted in these color regions with higher priority. In the original PatchMatch, the search is conducted in the neighbor of the patch from the previous match and a random location. In our approach, the search is restricted to the same color region with the highest priority (Figure 17(e)).

4.7.2 Lighten phase through local illumination transfer

After finding the corresponding patches in the source region, we must transfer the illumination information from the source patches to the target patches. We employ the local illumination transfer operator designed by L. Zhang et al. (2015). Suppose that source patch P^* is a corresponding suitable patch of the target patch Q. The recovered patch Q^* is obtained with:

$$Q^* = \frac{\beta + 1}{\alpha\beta + 1} P^*, \tag{24}$$

where

$$\begin{aligned}\beta &= \frac{P^* - \theta(Q)}{\alpha\theta(Q) - P^*},\\ \theta(Q) &= \alpha + \gamma Q,\\ \alpha &= \mu(P^*) - \gamma\mu(Q),\\ \gamma &= \frac{\sigma(P^*)}{\sigma(Q)},\end{aligned} \tag{25}$$

where $\mu(\cdot)$ is the mean value of the pixels in the patches, and $\sigma(\cdot)$ is the variance of the pixel value. The illumination recovering operator (Equation 24) is based on the assumption that little variation in illumination exists in a local patch. By applying this operator, the shadow in the target patches can be efficiently removed with the value from the patches with a similar texture.

After the target patches are recovered with the local illumination operator, the recovered patches are combined to form the contents of the target region. Each pixel in the target region may be overlapped by various patches. The recovered pixel value is determined by averaging the overlapping pixels from various patches.

4.8 Shadow boundary processing

The transition between the shadow area and the lit area is critical to generate a successful shadow-free image. An unnatural connection on the shadow boundary will leave traces on the shadow-free images and lead to failure of the task, as shown in Figure 20. Our approach can effectively recover the illumination on the shadow boundary in general. When recovering the pixels on the shadow boundary, the energy term E_1 accounts for the spatial distance, which suggests that matched patches usually intersect with the lit pixels on the boundary. However, in situations with complex and sharp shadow boundaries, or with heavy loss of the texture information, artifacts may appear on the shadow boundaries after the shadow-removal operation. Thus, to achieve seamless connections on the shadow boundary, we use the image quilting technique to generate an optimal seam.

Patch Q_{sb} on the shadow boundary has its anchor pixel within the shadow area; and patch P_{lb} near the shadow boundary has its anchor pixel in the lit area, and these patches overlap along their vertical edges (Figure 20). Note that the two patches are only partially overlap, which means $Q_{sb} - P_{lb} \neq \emptyset$. The overlapping regions are Q^{ov} and P^{ov}. To cut the overlap

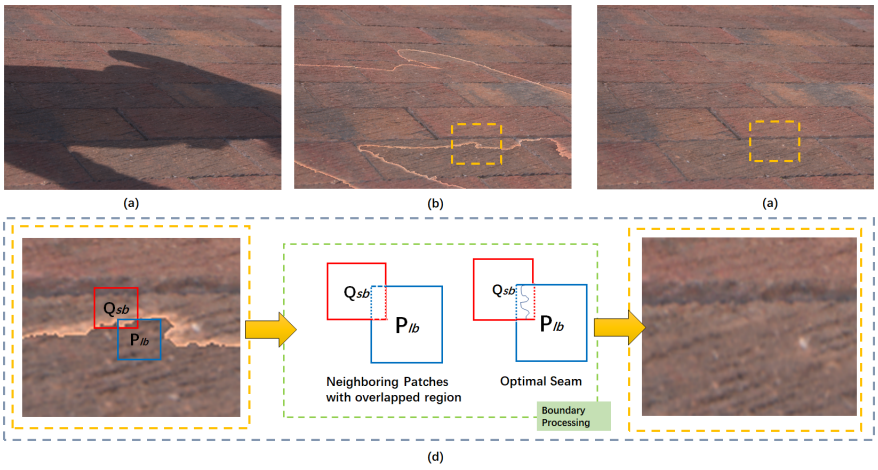

Figure 20: *Shadow boundary processing. (a) Original shadow image. (b) Shadow removed from image without boundary processing. Penumbra region is near the edge of the shadow region and is easily over-lightened. (c) Shadow removed from image by our approach with boundary processing. (d) In our approach, we combine the lightened shadow regions with the original image by calculating the optimal seam with minimum error.*

region to make the two patches match well to each other, we attempt to reduce the overlap error surface $e = ||P^{ov} - Q^{ov}||^2$. This can easily be accomplished using dynamic programming. Let E_{ij} be the cumulative minimum error for all paths; we traverse e_{ij} to find the minimal vertical cut in the surface:

$$E_{ij} = e_{ij} + \min\left(E_{i-1,j-1}, E_{i-1,j}, E_{i-1,j+1}\right). \tag{26}$$

In the end, the minimum value E in the last row will indicate the end of the programming. The optimal path that goes through the overlap error surface can be found by tracing back (Figure 20(d)). The patches that overlap along their horizontal edges can be handled in a similar manner. In situations in which both vertical and horizontal overlaps exist between regions, the minimum paths connect in the middle and hence reduce the overall error.

4.9 Evaluations

To verify the effectiveness of our shadow removal approach, we applied it to pictures that contain various kinds of cast shadow on various scenes and objects. Comparisons between the results with our approach and those with previous methods are presented. The program used in

our experiments is a collection of C++ and Matlab functions run on an Intel i5-4590 3.3 GHz computer with 8G Ram and an AMD R9 280 graphics card. In the following subsections, we discuss the shadow-removal effect of our approach and its run-time performance. A parameter analysis is presented to discuss the influence of each energy term weighted by λ_1, λ_2, and λ_3. In addition to visual comparison, objective evaluation such as Rooted Mean Square Error (RMSE) and a user study are also presented.

4.9.1 Visual comparison

To validate our approach, the test pictures used in our experiments contain various color patterns and complex textures. Figures 21 and 22 are the shadow-free results generated with our approach. The original shadowed images contain complex color patterns. From the results, we can see that color consistency is of vital importance in shadow removal tasks. As for the results generated by L. Zhang et al. (2015) (Figure 21(d)), the inconsistent color that appears in the shadow removed area causes visually unpleasing effects and leads to failure of the removal tasks. The cause of the inconsistent color can be attributed to a failure to properly transfer the illumination between different types of textures. The method of R. Guo et al. (2013) and that of Gong and Cosker (2017) also have similar problems with color consistency. Neither can properly remove the shadow when various types of texture or material appear in the shadow region. Unlike those methods, our approach can preserve the texture information and properly represent color in the shadow area, as shown in Figure 21(e). Most of the textural information is preserved in the illumination field via the intrinsic image decomposition. With independent optimization of reflectance and direct transfer operation on illumination, our results are free from color inconsistency and abnormal texture.

In some challenging cases, multicolor patterns (Figure 21 [the planter]) and irregular texture (Figure 21 [the grass and the sand]) appear in the target region. The difficulty in removing a shadow that contains multicolor patterns with color transfer technique lies in properly matching the color sources and the target regions. If a mismatch appears, the patterns in the shadow region may be contaminated with incorrect colors (Figure 21(d)). Our approach takes advantage of the color connectivity between the lit region and the shadow region to effectively avoid this

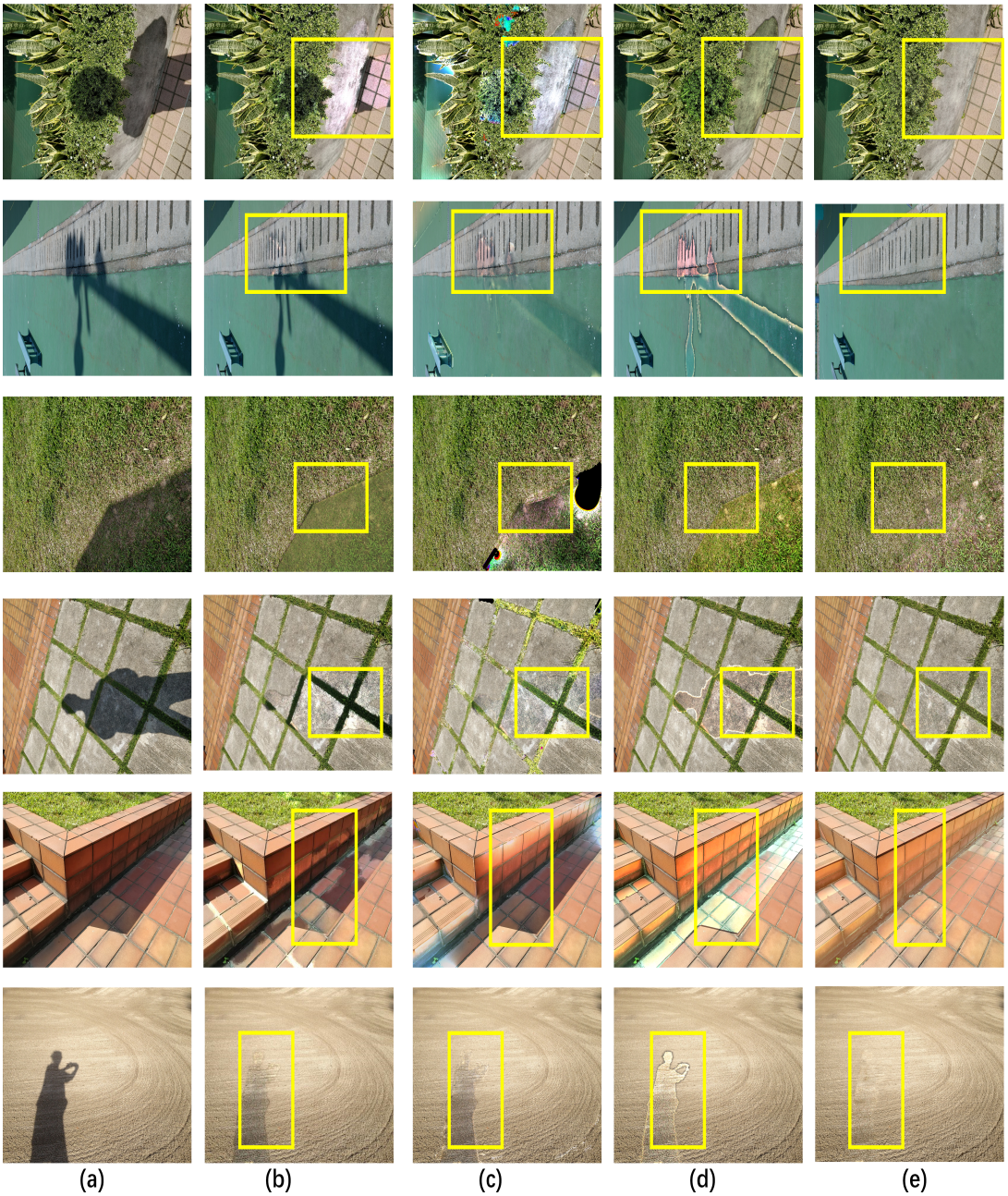

Figure 21: *Visual comparison: (a) Original shadow images. All images have a resolution of* 692×992. *(b) Results generated with the method of R. Guo et al. (2013). (c) Results generated with the method of Gong and Cosker (2017). (d) Results generated with the method of L. Zhang et al. (2015). (e) Results of our approach.*

problem. The challenge in removing shadow that contains irregular texture lies in processing the transitions. The transition in irregular textures may not be properly recognized by methods that depend on accurate shadow edge detection (Finlayson et al., 2006; Gong & Cosker, 2017) as shown in Figure 21(c). With our approach, the adaptive illumination transfer operators allows the removal of non-uniform shadows on each patch independently. The lightened patches are combined to reform the contents in the shadow region by averaging the pixel values. The averaging operation can effectively avoid inconsistency in illumination when handling shadows that contain irregular textures.

Another critical point to accomplish shadow-removal tasks is shadow boundary processing. A successful shadow-free image requires a seamless connection between the shadow region and the lit region. As shown in Figure 22(b), the blurred boundary leaves a trace of the shadow. Also, an abrupt transition in illumination or inconsistent texture on the shadow boundary can also lead to an unpleasant visual effect. By applying the image quilting technique, our results connect the shadow region and the lit region seamlessly (Figure 22(c)).

Our approach can also be applied to remove shadows in aerial remote-sensing images, as shown in Figure 23. Aerial remote-sensing images differ slightly from common pictures taken with consumer-grade equipment. High-resolution aerial remote-sensing images usually contain large amounts of noise, which makes the linear shadow-free method invalid. In addition, buildings and landscapes in the images can be regarded as a complex texture on a surface. As shown in Figure 23, our approach can also handle the complex shadow patterns without distorting the landscape information on the images.

4.9.2 Comparison with the ground truth

One method of quantitative evaluation of the shadow-free image generated with our approach is to compare the shadow-free image with the ground truth. To conduct a fair comparison with other methods, the quantitative comparison is based on the dataset by R. Guo et al. (2013) which contains 156 images photographed in the same scene; once with cast shadow and once without any occlusion in the light sources. Except for the difference in light occlusion, all other conditions, such as camera settings, are the same. We applied our approach to the shadowed

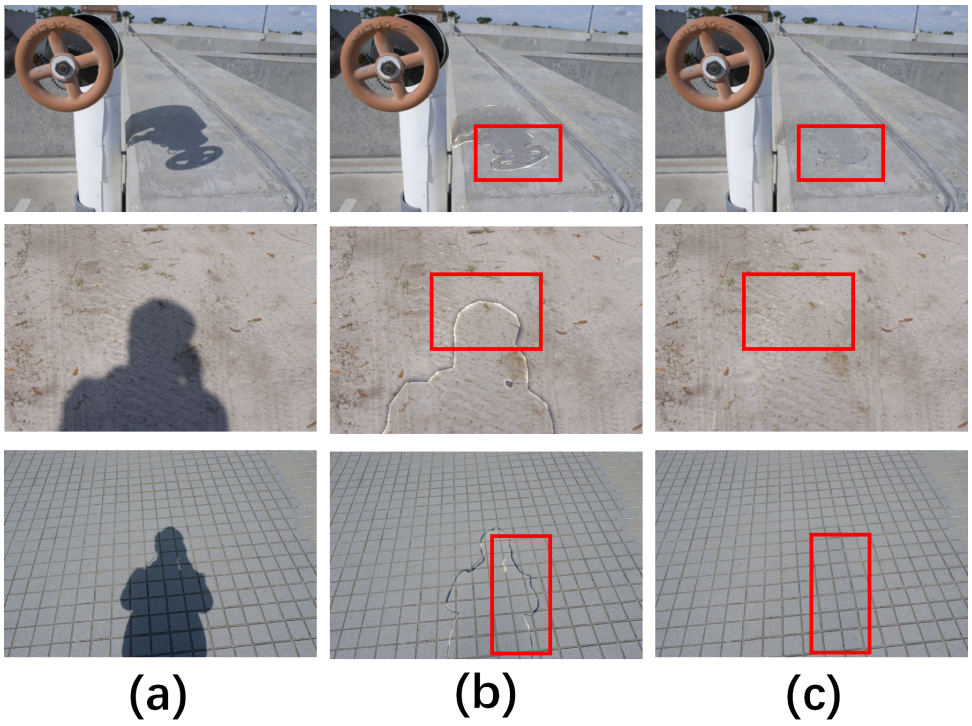

Figure 22: *(a) Original shadow image. (b) Shadow removed from image without boundary processing. Penumbra region is near the edge of the shadow region and is easily over-lightened. (c) Shadow removed from image with our approach with boundary processing. Our approach combines the lightened shadow regions with the original image by calculating the optimal seam with minimum error.*

images and computed the RMSE between the ground truth and the shadow-free images. Figure 24 is a visualization of the RMSE calculation. The RMSE of the previous methods are also computed for comparison. As illustrated in Table 3, our approach generates results with a lower RMSE than previous methods. In other words, the results generated with our approach are closer to the ground truth. It is important to note, however, that a lower RMSE does not generally guarantee a better perceptual effect. The quantitative comparison is included here for completeness. In fact, comparison with the ground truth may not be practical in some applications, in which the ground truth is not available. Our approach focuses mainly on the pursuit of imperceptible shadow-free images.

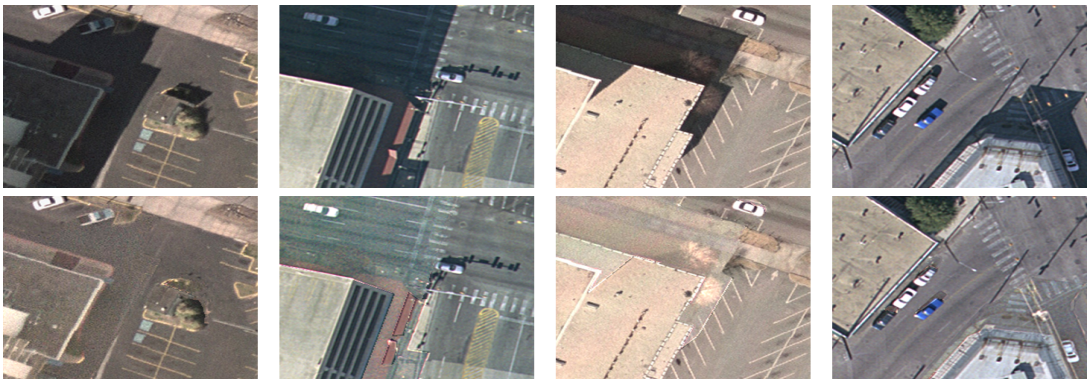

Figure 23: *Shadow removed results of aerial remote-sensing photos. The buildings and objects in aerial images can be treated as complex texture, and our approach can properly remove the shadow without distortion and color inconsistency.*

Figure 24: *Calculation of the RMSE. (a) Original shadowed image. (b) Shadow-removed image generated by our approach. (c) Ground-truth shadow-free image. (d) Squared difference between images (c) and (b). Noted that the difference is multiplied three times for presentation.*

4.9.3 Efficiency

For most shadow removal methods, the required time varies according to the scale of the shadow region. The larger the shadow region, the more pixels that need to be lightened. The time needed for our approach is also related to the size of the whole image. A larger image suggests a larger search space of suitable patches. Our reflectance-guided patch search and parallel programming techniques are designed to handle this problem. As shown in Table 4, the patch-searching phases take up most of the time in the shadow removal process. The patch searching methods for other patch-based shadow removal methods (Gryka, Terry, & Brostow, 2015; L. Zhang et al., 2015) are based mainly on the original PatchMatch (Barnes et al., 2009). As shown in Table 4, our approach has advantages in time consumption, which can be attributed to the informative guidance obtained from the reflectance. As accelerated with the GPU, our

Table 3: *RMSE Between Results of Various Shadow Removal Methods and Ground-Truth Shadow-Free Images*

Methods	Mean RMSE
Ours	33.66
R. Guo et al. (2013)	41.30
Gong and Cosker (2017)	35.36
L. Zhang et al. (2015)	37.44

approach has a further enhancement in executive speed and can save approximately 25% of the executive time compared to the patch-based shadow removal method of L. Zhang et al. (2015).

Table 4: *Time Consumption*

Images in Fig, 21 (692 × 992)	Planter	Playground	Grass	Grid	Stage	Sand
Percentage of the shadow area in the image (%)	26.03	28.21	25.07	26.63	18.88	30.33
Time cost in the steps of our system(s):						
Intrinsic Image Decomposition	309.656	113.866	298.129	304.022	303.17	332.64
Reflectance Reparation	9.58	2.54	8.27	7.80	12.03	6.03
Illumination Recovery:	159.82	100.17	272.39	253.77	282.24	326.34
Shadow Boundary Processing	8.33	2.83	8.27	3.12	10.94	10.97
Total Time of Our System:	487.40	219.42	587.07	568.72	608.39	675.99
Method of R. Guo et al. (2013)	424.98	114.79	665.34	223.85	101.85	21.23
Method of Gong and Cosker (2017)	55.37	23.33	13.11	63.71	25.37	23.09
Method of L. Zhang et al. (2015)	630.05	688.23	1358.92	1323.12	1379.02	1447.91

4.9.4 Parameter analysis

The influence of the energy terms is controlled by the weights λ_1, λ_2, and λ_3, which are user-defined parameters. In our experiments, the weighted parameters are set as $\lambda_1 = 0.2$, $\lambda_2 = 0.6$, and $\lambda_3 = 0.2$. The higher settings of the weight λ_2 gives preference to the textural similarity. Textural information is the most important cue for illumination recovery. In fact, in the first iteration of the patch search phases, initialization is based mainly on textual similarity because of the large difference in intensity between the shadow patches and lit patches. If λ_1 is set relatively higher than λ_2, the search may be trapped in a region of lower intensity. As shown in Figure 25(c), the results generated with our approach have a configuration of $\lambda_1 = 0.6$, $\lambda_2 = 0.2$, and $\lambda_3 = 0.2$. The recovered region is relatively darker than our common configurations. In general, we recommend setting λ_1 lower. It should be noted, however, that a low λ_1 may also weaken the propagation of the good match and the ability to handle soft shadow. λ_3 is used to control the consistency in illumination between overlapping patches. A higher λ_3 can effectively alleviate the unnatural jagged patterns. However, an oversized λ_3 may also blur the contents within the shadow region. As shown in Figure 25(d), our shadow removal approach is applied to the image with a configuration of $\lambda_1 = 0.2$, $\lambda_2 = 0.2$, and $\lambda_3 = 0.6$. The pattern represented in the shadow region is slightly blurred.

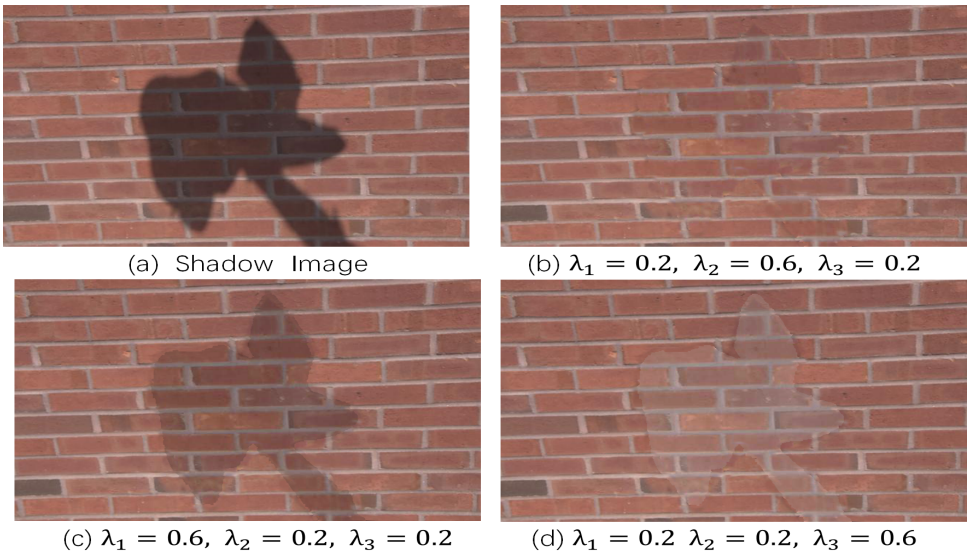

Figure 25: *Results generated with various configurations of λ_1, λ_2, and λ_3. Notice that when $\lambda_1 = 0.6$, $\lambda_2 = 0.2$, and $\lambda_3 = 0,2$, the generated result is relatively darker. With the configuration of $\lambda_1 = 0.2$, $\lambda_2 = 0.2$, and $\lambda_3 = 0,6$, the texture of the generated result is slightly blurred and over-lightened.*

4.9.5 User study

To understand how the results generated with our approach compare to those produced with the previous methods from a user point of view, we conducted a user study. A set of 108 shadow images was selected for user evaluation. The selected images were taken from different scenes and contain various kinds of texture and objects. The selected images are then processed with the shadow removal methods of R. Guo et al. (2013), Gong and Cosker (2017), L. Zhang et al. (2015) and with our approach. The shadow-free results are then presented to the participants without knowledge of the exact shadow removal method applied in the evaluation process. We deployed our questionnaire on a website and recruited 30 participants (16 female and 14 male) from a university via email, WeChat, and Facebook. The participants attended our evaluation online by visiting a link using a desktop computer or tablet. The average age of the participants was 23.4 years. The participants were required to rate the image on a 9-point scale (1 to 9 with 9 the highest quality). The ground-truth shadow-free images were also presented to the participant for reference.

Table 5: *User Study Statistics.*

Methods	Mean	Standard Deviation	Mean Difference with our approach	95% Confidence Interval	
				Lower Bound	Upper Bound
Our approach	7.522	0.283	-	-	-
R. Guo et al. (2013)	3.975	0.508	3.546	3.429	3.663
Gong and Cosker (2017)	5.544	0.445	1.978	1.861	2.095
L. Zhang et al. (2015)	4.97	0.483	2.542	2.425	2.660

Table 5 shows the mean scores of our approach ($M = 7.522$), the method of R. Guo et al. (2013) ($M = 3.975$), the method of Gong and Cosker (2017) ($M = 5.544$), and the method of L. Zhang et al. (2015) ($M = 4.97$). Our approach obtained the highest scores among the four methods. An ANOVA is conducted to verify the statistical significance of the difference between the scores. The F value calculated from ANOVA indicates the significance of the difference. The ANOVA result between the four groups is $F(3, 428) = 4.171, p < 0.05$, which indicates that a significant difference exits between the means of the four groups. To further analyze the difference, we calculated the MD and compared our approach with the method of R. Guo et al. (2013) ($MD = 3.546, p < 0.001$), the method of Gong and Cosker (2017) ($MD = 1.978, p < 0.001$) and the method of L. Zhang et al. (2015) ($MD = 2.542, p < 0.001$) in post hoc tests. The results were all significant and suggest that the participants evaluated the shadow-free images generated with our approach as better than the others. From the 95% confidence interval, it can be concluded that the results with our approach are more visually pleasing than those with previous methods.

4.10 Discussion

This chapter presents a novel shadow removal approach based on a patch-based illumination recovery. Compared to previous shadow removal methods, our approach adapts the shadow removal task to an image completion framework and accomplish the tasks via direct repair in the illumination field using a patch-based recovery. Direct recovery in illumination allows us to maximize the use of textural information and prevent the interference of color differences. Unlike previous methods that use post-processing color correction, our approach includes in-

trinsic image decomposition and optimizes the image reflectance at the beginning to avoid abnormal chromatic differences. The optimized reflectance is used to guide the patch search process, and the search is accelerated via parallel programming. Supported by experiments and the user study, our approach outperforms previous methods and generates more visually pleasing shadow-free results. In the future, we plan to extend our approach to video-editing tasks such as the removal of pedestrian shadows in surveillance videos and vehicle shadows from autonomous driving systems. These tasks require extra optimization in efficiency and robustness. Another possible direction is to adapt our approach to highlight removal tasks in which the light spots in the images can be regarded as a special type of shadow.

5 Explicit Structure Completion via Partial Shape Matching

5.1 Introduction

The shape of an object provides vital information in the process of object recognition. Human beings can easily recognize an object simply according to its shape. In computer vision, an object-recognition system often includes a module for shape matching. In fact, shape matching is a critical task with wide applications in computer vision tasks such as character recognition (You & Tang, 2007), medical image analysis (Heimann & Meinzer, 2009), and robot navigation (Wolter & Latecki, 2004). However, shape matching is not an easy task. For one thing, shape instances generated from the same categories may vary in geometric features due to transformation (such as translation and rotation) or nonlinear deformation (such as noise and occlusion). The difficulty lies in how to extract stable and reliable features from the shape's contours. For another, a matching scheme should be able to preserve the global geometric features while matching the local point features. In sum, a reliable matching scheme contains two components, a smart shape descriptor and an efficient geometric matching method.

Previous studies devoted considerable effort to design smart shape descriptors in an attempt to extract as much information as possible from a contour. From a simple two-dimensional closed curve, there are limited one-dimensional boundary features (tangent angles, Euler Numbers, and curvatures) can be extracted directly for shape matching. Contour curvature is a very important boundary feature for measurement of shape similarities. Previous studies have shown that curvatures provide sufficient discrimination power in shape matching, but there are two problems with direct matching based on curvature. One is that calculation of the curvature requires second-order derivatives, but it is often limited by the angular resolution. The calculation is easily affected by quantization of the curves in digital images. The other problem is that the curvature is a local shape feature and does not preserve global geometric information. Thus, extra techniques are required to achieve geometric matching. Previous researchers attempted to overcome these two problems with more accurate methods of calculating curvatures and deriving various shape descriptors based on curvatures such as *Curvature Flexibility* (C. Xu et al., 2009; Manay et al., 2006) and *Height Function* (Wang et al., 2012).

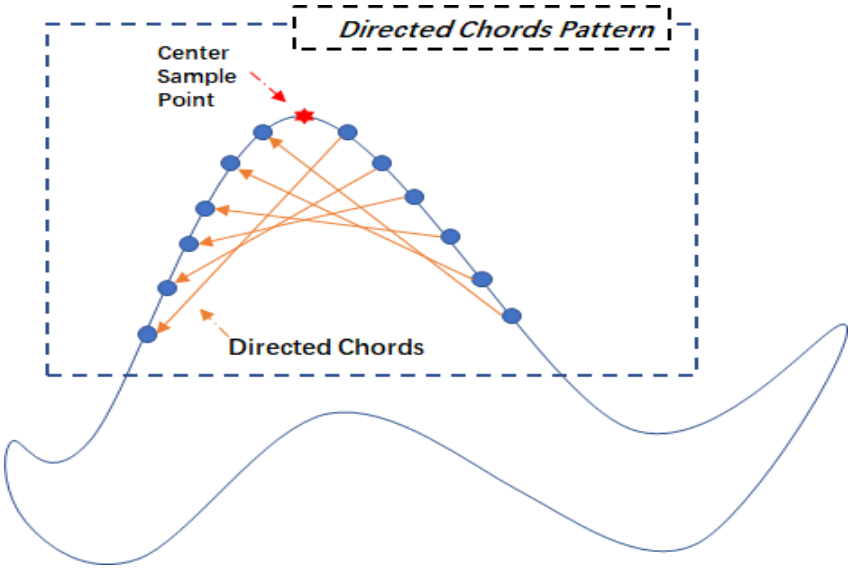

Figure 26: *Example of the Directed Chords Pattern(DCP). The sample point marked with a star is selected as the center. The directed chords are vectors formed by connecting the neighboring sample points. The spatial relationships between the center and the directed chords form the DCP of this point.*

To seek more reliable and effective features, some researchers have attempted to construct shape descriptors based on 2D features. An intrinsic way is to construct chords in the shape. By connecting points on the contour, a series of chords can be easily achieved for analysis. Smith and Jain (1982) used all the chords within a shape to generate a chord distribution for matching shape. However, direct comparison of the global chord distribution may not be efficient, and not all chords are necessary for comparison. We note that some efficient shape descriptors essentially analyze a subset of the chords according to a specific criterion. For example, Shape Context (SC) proposed by Belongie et al. (2002) essentially measures the chord distribution using each sample point on the contours as the origin. The Inner Distance Shape Context (IDSC) introduced by Ling and Jacobs (2007) is an extension of the shape contex by limiting the chords within the shape for statistics. The Beam Angle Statistic (BAS) (Arica & Vural, 2003) records the angles formed by two chords on a connected sequence. Even the traditional "splitting and fit" algorithm (Duda, Hart, & Stork, 2012) uses a series of chords along the contours to simulate the actual curves.

Following the idea to describe shapes by constructing chords, we proposed a novel shape

descriptor called Directed Chords Pattern (DCP) for shape matching. As shown in Figure 26, our DCP shape descriptor takes advantage of a series of directed chords, which are formed sequentially in the neighborhood of a sample point to capture the local geometric features. Our DCP is essentially the spatial relationship between the sample point and the chords in its neighbor. We develop a bipartite graph matching scheme based on our DCP descriptor. To pursue geometric matching and take advantage of global geometric information, we develop a probabilistic model to estimate the transformation between two shapes. We embed probabilities based on our metric of DCP to the Gaussian Mixture Model (GMM). The final shape distance is the weighted sum of both local and global matching. In sum, our work makes the following contributions:

1. *Sensitive to Curvature:* Our DCP provides a unique pattern at high curvature points to provide sufficient discriminating power.

2. *Flexible Computational Burden:* Only chords in the neighbor of a sampled point are used to form the DCP. The chords needed for statistics are flexible according to the application and can be reduced significantly.

3. *Transformation Estimation with DCP:* The probability based on the DCP metric is embedded into a GMM to estimate the transformation between shapes. The DCP metric probability functions provide extra information for the alignment of two shapes.

The remainder of this chapter is arranged as follows. In Section 5.2, the definition of our DCP and its features are presented in detail. Based on the proposed DCP descriptor, the matching scheme that considers both global and local geometric features is presented in Section 5.3. Our shape matching approach based on DCP is tested on benchmark datasets, and the experimental results are provided in Section 5.4. In addition to the benchmark test, we also apply our matching scheme to the task of image repair. We conclude this chapter in Section 5.5.

5.2 Directed chords pattern

Our proposed local shape descriptor is defined on shape silhouettes, the external contour of each input shape (Arica & Vural, 2003; Ling & Jacobs, 2007). The contour of the shape is usually a 2D closed curve. Before a shape descriptor is applied to extract its features, the given silhouette is uniformly sampled and represented as an ordered set of n points:

$$S = \{p_1, p_2, \ldots, p_n\}. \tag{27}$$

Note that the points are sampled in an equidistant manner, which means that $\Delta S_i = p_i - p_{i-1}$ is a constant. The order of points is generated by following the contour in a counter-clockwise direction. Given a point $p_i \in S$, we define a symmetric neighbor with a fixed radius of K as

$$\mathcal{N}_K(p_i) = \mathcal{N}_K^-(p_i) \cup \mathcal{N}_K^+(p_i), \tag{28}$$

where

$$\begin{aligned} \mathcal{N}_K^-(p_i) &= \{p_{i-K}, p_{i-(K-1)}, \cdots p_{i-1}\}, \\ \mathcal{N}_K^+(p_i) &= \{p_{i+1}, p_{i+2}, \cdots p_{i+K}\}. \end{aligned} \tag{29}$$

According to this definition, for every sampled point p_i on shape S, a number of $2\times K$ points are selected symmetrically in its neighborhood to generate its chord pattern. Specifically, K points are selected on the left side of the p_i, and K points are selected on its right side.

By connecting the points from the the left interval $\mathcal{N}_K^-(p_i)$ and the right interval $\mathcal{N}_K^+(p_i)$, a total number of K^2 of chords can be generated. However, not all chords are necessary. To capture the geometric feature at point p_i, we define a group of sequenced chords, the DCS, as a group of chords with the following definition:

Definition 1. For a point p_i on a sampled contour S and a given radius K, the *Directed Chords Sequence* (DCS) of p_i is defined as

$$\begin{aligned} DCS(p_i) &= \{\overrightarrow{p_u p_v} | v - u = K + 1\}, \\ & p_u \in \mathcal{N}_K^-(p_i), p_v \in \mathcal{N}_K^+(p_i), \end{aligned} \tag{30}$$

where $\overrightarrow{p_u p_v}$ is a directed chord that starts at point p_u and ends at point p_v

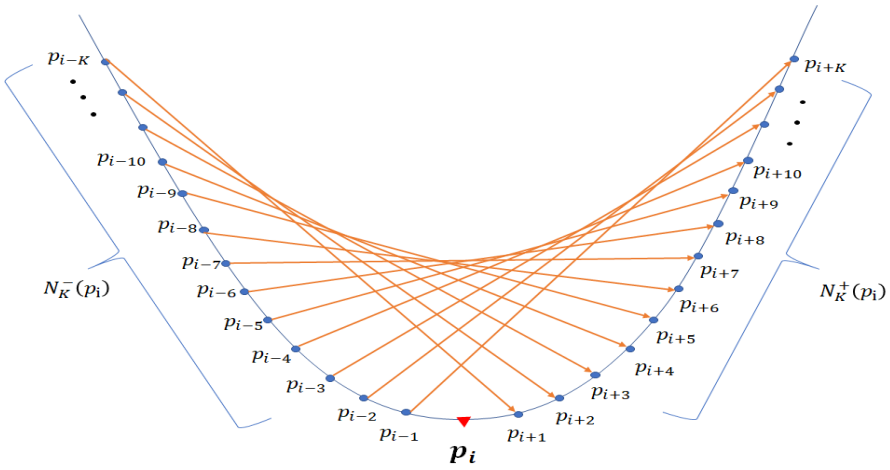

Figure 27: *Directed Chords Sequence (DCS) of p_i is a series of chords constructed by connecting points in set $\mathcal{N}_K^-(p_i)$ and $\mathcal{N}_K^+(p_i)$.*

Figure 27 visualizes the Directed Chords Sequence (DCS) of a point on a parabolic curve. The DCS is a series of vectors that begins from points in $\mathcal{N}_K^-(p_i)$, and ends in points in $\mathcal{N}_K^+(p_i)$ with a fixed interval $K+1$. The interval is set to be larger than the radius of $\mathcal{N}_K(p_i)$ to ensure that p_i is contained in the arch $\widehat{p_u p_v}$

Equipped with the DCS, the geometric features near the central point p_i can be described with the spacial relationship between the central point and the chords contained in DCS. As shown in Figure 28, we developed a coordinate system using the central point p_i as the origin. The spacial relationship between p_i and the chord $\overrightarrow{p_u p_v}$ can be described with the point-to-chord distance and the angle formed between the perpendicular line and the horizontal axis. Let the Euclidean distance between the center point p_i and the chord $\overrightarrow{p_u p_v}$ be $AH_{uv}(p_i)$. $AH_{uv}(p_i)$ is also called the *arch height* at point p_i. Let the angle generated by the perpendicular lines and the horizontal axis be $\theta_{uv}(p_i)$. Note that the range of $\theta_{uv}(p_i)$ is limited to $[-\pi, \pi]$. The two-tuple $(AH_{uv}(p_i), \theta_{uv}(p_i))$ determines the relative position of the chord $\overrightarrow{p_u p_v}$ referring to the origin of central point p_i.

By calculating the relative position between center point p_i and all chords in the DCS, we can obtain a series of two-tuples. With these tuples, we obtain the following definition of the DCP:

Definition 2. For a point p_i on a contour S and the DCS in its symmetric neighbor $\mathcal{N}p_i$, the

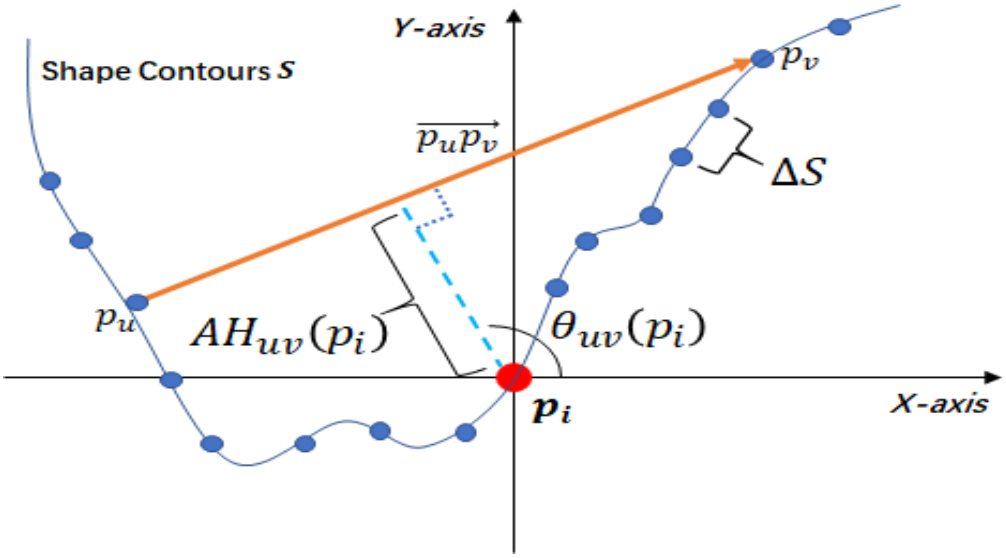

Figure 28: *Spatial relationship between a sampled point and a directed chord. Using the sample point as the origin, a Cartesian coordinate system was developed. The position of a directed chord relative to a sampled point is measured with two parameters: the perpendicular point-to-chord distance* $AH_{uv}(p_i)$ *and the angle between the perpendicular line and the horizontal axis* $\theta_{uv}(p_i)$*. Note that* $AH_{uv}(p_i)$ *is essentially the arch height at point* p_i*.*

Directed Chords Pattern (DCP) of p_i is defined as:

$$DCP(p_i) = \{(AH_{uv}(p_i), \theta_{uv}(p_i)) \, | \overrightarrow{p_u p_v} \in DCS(p_i)\}, \tag{31}$$

where $AH_{uv}(p_i)$ is the arch height component, and the $\theta_{uv}(p_i)$ is the angular component.

If the arch height component $AH_{uv}(p_i)$ and the angular component $\theta_{uv}(p_i)$ are separated from the $DCP(p_i)$ independently, we can easily obtain the Arch Height Pattern (AHP) and the Angular Pattern (AP) of point p_i.

Definition 3. For point p_i on a contour S and its $DCP(p_i)$, its *Arch Height Patern*(AHP) and *Angular Pattern*(AP) are defined as

$$\begin{aligned} AHP(p_i) &= \{AH_{uv}(p_i) \in DCP(p_i)\}, \\ AP(p_i) &= \{\theta_{uv}(p_i) \in DCP(p_i)\}. \end{aligned} \tag{32}$$

These two subpatterns derived from the DCP can provide valuable information near the central point. These two patterns can complement each other in the invariant description of a

contour. The AHP of a point on the contour is invariant to rotation and scales linearly with the object size. The AP is invariant to the size of the object but shifts relative to its orientation.

5.2.1 Features and invariance of the DCP

When applying a local descriptor to a shape, one should consider not only whether the descriptor can properly capture the geometric features at local points, but also whether the description would be stable after certain transformation such as translations, rotation, or a group of affine transformations. We apply our DCP descriptor to a toy example and analyze its components, the AHP and the AP, respectively.

Sensitive to curvature As shown in Figure 29, we use a parabola as a toy example. The parabola is uniformly sampled into 300 points. We select three points on the parabola for explanation: the vertex point p_a and two points that has equal distance from it on either side (Marked as point p_b and point p_c in the Figure). We calculate their DCP with a radius of $K = 40$, which suggests that 40 chords near the central point are used to describe the feature. We use the curvature as a baseline for comparison. In fact, the curvature is one of the effective features used to describe a closed curve on a 2D plane. Many descriptors in previous studies are based on the curvature, such as Curvature Scale Space (CSS) (Mokhtarian & Suomela, 1998), Contour Flexibility (C. Xu et al., 2009) and Height Function (Lin et al., 1992). The three points we selected for discussion have different curvatures.

The DCP of the points is represented by its AHP and AP, respectively. Figure 29(b) shows the DCP of the vertex p_a, which has the highest curvature on the parabola. As shown in the Figure of AHP, the arch height $AH(p_a)$ maximizes at vertex p_a with a sharp climax. The AHP is symmetrically distributed on both sides of vertex p_a. The AP of point p_a is similar to a sigmoid function. The $\theta(p_a)$ jumps significantly at point p_a. The DCP of the point with lower curvature p_b and p_c shows different features. In terms of the AHP, the $AHP(p_b)$ and $AHP(p_c)$ are obviously smoother and more stable. Although they have a similar symmetric pattern as $AHP(p_a)$, the $AHP(p_a)$ has a larger kurtosis than $AHP(p_b)$ and $AHP(p_c)$. In terms of the AP, the $AP(p_b)$ and $AP(p_c)$ are more similar to monotonic linear curves. From

these observations, we notice that the DCP can effectively capture fluctuations in the contour curvature.

Invariance To evaluate the invariance of the DCP, its two components AHP and AP should be considered separately. Invariance to translation is intrinsic to both patterns because both measurements are taken with respect to points on the object in our definition. In terms of scale invariance, the AP is intrinsically scale invariant, as shown in Figure 30. To achieve scale invariance of the AHP, we normalized the arch heights at all points by the average distance between each two neighboring sample points (Figure 28). In terms of rotation invariance, the AHP is not influenced by the rotation but the AP will rotate accordingly. One can achieve a rotation invariant AP though using a relative frame, such as calculating the theta using the tangent vector at each point. In this way, the obtained AP turns with the tangent angle and becomes invariant to rotation. It is worth to note that complete rotation may not be necessary for every application. For one thing, in some cases that reliable or robustly defined relative frame is not available. For another, complete rotation invariance may decrease the discrimination power of the descriptor in some recognition tasks. For example, when distinguishing the number "8" and the infinity sign "∞", rotation invariance would be inappropriate.

Time complexity The time consumption for calculating DCP of a shape depends on the sample rates and the size of the symmetric neighbor. For a point set of size n sampled from on a shape, if a symmetric neighbor of size K is selected, we need to calculate and record nK point-chord positions in total. Notice that part of the directed chords may be shared by some sample points that are closed to each other. Thus, in actual implementation, we can construct the directed chords before we start to record be relative position instead of repeatedly constructing chords in each computation. There are $[n/k]$ directed chords in total in a point set. From the above analysis, we can see that the time complexity of calculating the DCP of a shape is $O(nK)$.

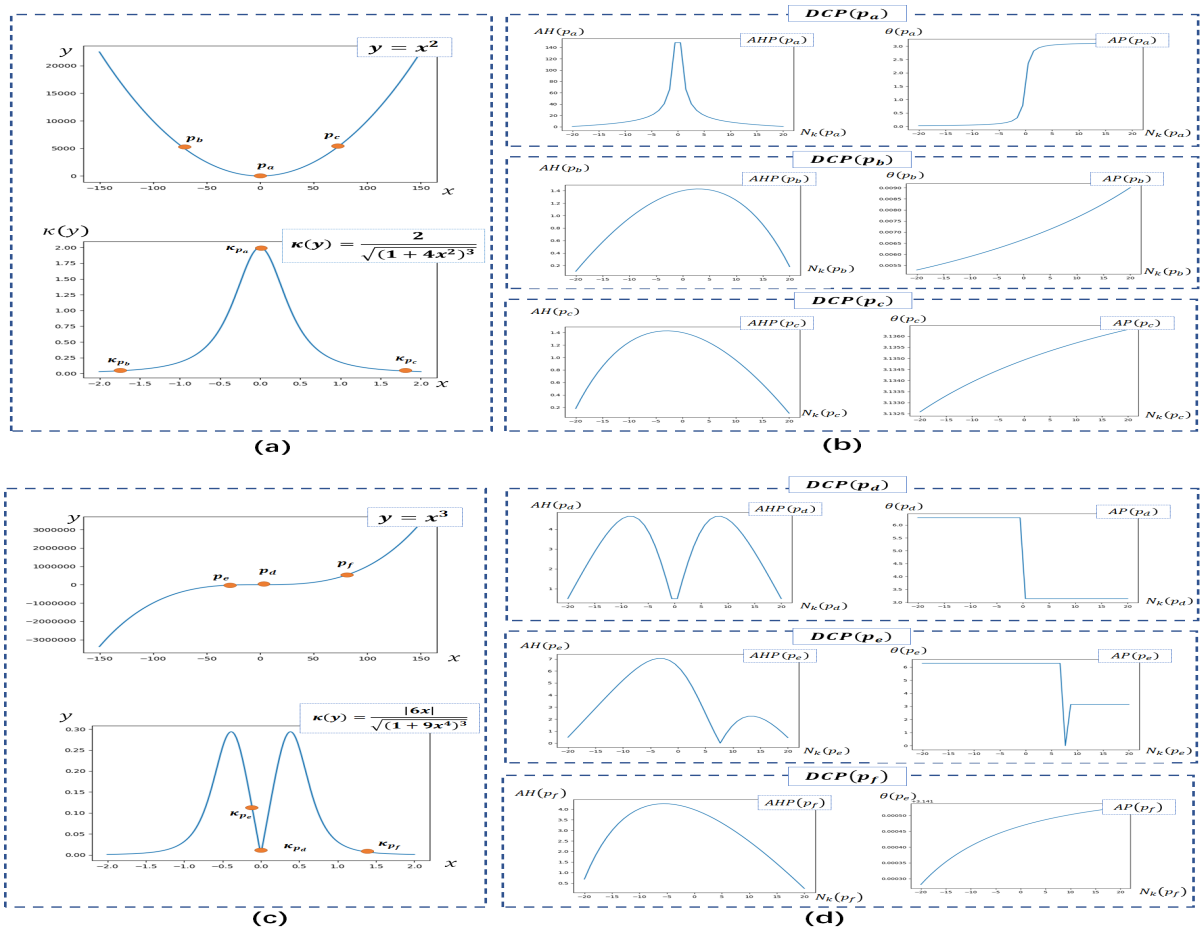

Figure 29: *The Directed Chords Pattern of the points on curves. We provide the DCP of points on curves to demonstrate its discriminating power. (a) A simple parabolic curve* $y = x^2$ *and its curvature function* $\kappa(y)$*. Three points are selected to present their DCP in (b): the vertex* p_a *(which is also the extreme point on the curvature function) and two points* p_b *and* p_c *symmetrically on both side with low curvature. The DCP is significantly different at the point where the curvature changes dramatically. (c) A cubic curves* $y = x^3$ *and its corresponding curvature function* $\kappa(y)$*. Note that the curvature function is presented in the interval that contains the greatest fluctuation. Three points are also selected to present the DCP: the extreme point on curvature function* p_d *where* $\kappa_{p_d} = 0$*, a point* p_e *near the extreme point, and a point* p_f *far away from* p_d*. Their DCP is presented in (d). From the* $DCP(p_e)$*, it can be easily seen that the curvature change at point* p_d *is also captured in the DCP at* p_e*. A comparison of the* $DCP(p_c)$ *and* $DCP(p_f)$ *shows the similarity in curvature between* p_c *and* p_f*.*

5.2.2 Distance of the DCP

Consider a point p_i on the first shape and a point q_j on the second shape. Let $d_{ij} \equiv d_{DCP}(p_i, q_j)$ denote the DCP distance between these two points. We start by defining the distance of the subpattern of the DCP. Similar to previous studies, we use the χ^2 test statistic to evaluate the distance of the AHP and AP:

$$\begin{aligned} d_{AHP}(p_i, q_j) &= \frac{1}{2}\sum_{1\le k}^{K}\frac{[AH_i(p_k) - AH_j(q_k)]^2}{AH_i(p_k) + AH_j(q_k)}, \\ d_{AP}(p_i, q_j) &= \frac{1}{2}\sum_{1\le k}^{K}\frac{[\theta_i(p_k) - \theta_j(q_k)]^2}{\theta_i(p_k) + \theta_j(q_k)}, \end{aligned} \tag{33}$$

where $AH_i(p_k)$ and $AH_j(q_k)$ denote the kth arch height component in their AHP. $\theta_i(p_k)$ and $\theta_j(p_k)$ are the kth angular components in their AP. Because the DCP of each point contains two components, the distance of DCP is defined as the weighted sum of the distance between each corresponding component:

$$d_{DCP}(p_i, q_j) = \omega_1 d_{AHP}(p_i, q_j) + \omega_2 d_{AP}(p_i, q_j), \tag{34}$$

where ω_1 and ω_2 are the weights that control the confidence between two subpatterns that satisfy $\omega_1 + \omega_2 = 1$. The weights can be adjusted according to the practical situation. For example, when scale normalization is not available, the AP could be more reliable than the AHP. In this situation, ω_2 should be set larger than ω_1 to achieve better performance. In our experiments, we set $\omega_1 = \omega_2 = 0.5$.

5.3 Shape matching scheme with directed chords pattern

The DCP describes the geometric features of a point using the chords constructed by its neighboring points. The correspondence between two points can be obtained by calculating the distance of the DCP at each point. Our shape matching scheme is based on comparing the DCP of the sample points set from various shapes.

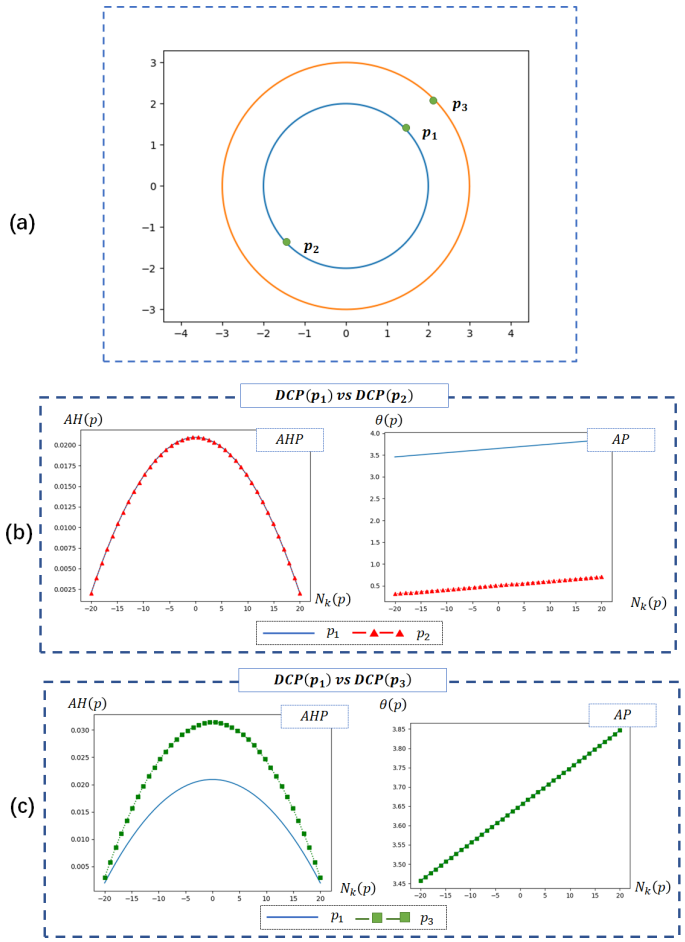

Figure 30: *Invariance of the Directed Chords Pattern. (a) We use the points on circles to illustrate the invariance of DCP.* p_1 *and* p_2 *are two points on the identical circle (which has a radius of 2), and* p_2 *can be regarded as a point transformed from* p_1 *by rotation.* p_3 *is a point on a circle with a radius of* 3*.* p_3 *and* p_1 *have the same relative angular position relative to the center of the circle, and* p_3 *can be regarded as a point transformed from* p_1 *by scaling. (b) Comparison of DCP between* p_1 *and* p_2*.* $AHP(p_1)$ *and* $AHP(p_2)$ *are identical, and the* $AP(p_2)$ *can be regarded as a translated pattern generated from* $AP(p_1)$*, which suggests that the AHP is rotation invariant but the AP is influenced by the rotation. (c) Comparison of DCP between* p_1 *and* p_3*.* $AP(p_1)$ *and* $AP(p_3)$ *are identical.* $AHP(p_1)$ *and* $AHP(p_3)$ *have a similar pattern but* $AHP(p_3)$ *has higher kurtosis. The comparison suggests that the AP is scale invariant but the AHP varies according to scales.*

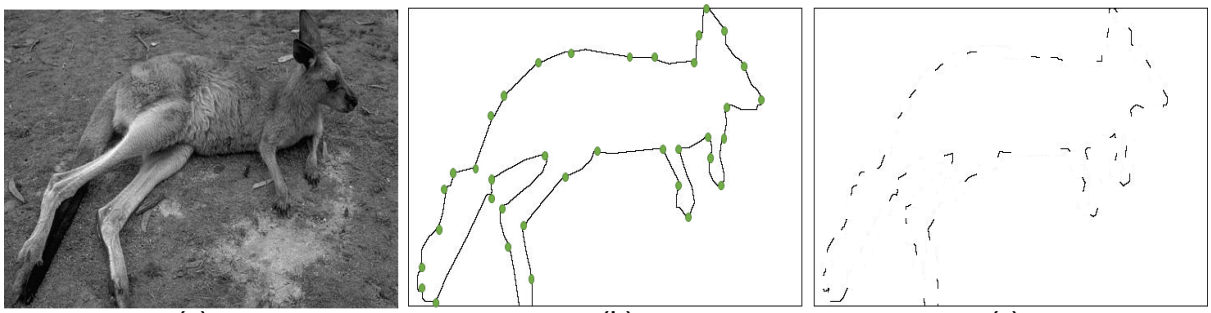

Figure 31: *(a) Kangaroo picture from BSDS300 (Martin et al., 2001). (b) Extracted contour of the kangaroo with high curvature points marked. (c) Kangaroo contour formed by local segments with high curvature points.*

5.3.1 Local features with directed chords pattern

Previous psychological studies (Attneave, 1950; Biederman, 1987) showed that the corners or parts of an object that show a significant change provide highly valuable features when human beings recognize the shape of an object. These corners and parts that contain significant changes are usually those at which contour curvature change significantly. As shown in Figure 31, human beings can still recognize the shape of the kangaroo even though only the segments of the corners remain. We noted in the previous section that the DCP can properly capture fluctuations in contour curvature, so it can be applied to measure the similarity between two shapes.

Given two shape, $S_1 = \{p_i | i = 1 \cdots n\}$ and $S_2 = \{q_j | j = 1 \cdots m\}$, that are sampled at an equal distance along the contour. We calculate their DCP at each sample points with the same radius. The two shapes are matched by minimizing the following energy function:

$$E_{Local}(\psi) = \sum_{i \leq n, j \leq m} d_{DCP}(p_i, q_{\psi(j)}), \tag{35}$$

where ψ is a permutation to ensure that the matching is injective.

Note that we do not strictly require that the number of sample points from two shapes be the same for generality. When comparing closed contours, equal numbers of sampling points ($n = m$) will be preferred. When $n \neq m$, the task becomes a partial shape matching problem.

Minimization of Equation 35 is essentially a Linear Assignment Problem (LAP) that can

be efficiently solved by the Hungarian algorithm (Munkres, 1957) or dynamic programming (Bellman, 1966). The optimum can also be solved in a graph cuts discrete optimization scheme (Schmidt, Toppe, & Cremers, 2009). In the case of partial shape matching, Dynamic Time Warping (DTW) (Keogh & Ratanamahatana, 2005) is also preferred to achieve an optimum match (Michel et al., 2011; Latecki, Megalooikonomou, Wang, & Yu, 2007). However, the selection of optimization techniques does not completely depend on application schemes.

The matching results ψ achieved by direct minimization of Equation 35 ensure that points are matched to the most similar corresponding points based on a local feature discrepancy measure. However, ψ cannot guarantee that the inner distance between points in one shape is preserved in another shape and thus is nongeometric. Figure 32 provides an example in which the identical shape on the right is matched to two different templates on the left. High curvature points on the contours are marked with dots. The template on the left in Fig. 32(a) is a shape constructed by randomly arranging the segments that contain the high curvature points. With the Hungarian algorithm, the corresponding points are properly matched with their local feature. However, the cross of the assignments suggests that global geometry has not been properly restored. The absence of the global geometric information may easily fool the matching system with some local similarities between shapes. In fact, the shape on the right is a part of the heart shape. as shown in Figure 32(b).

One method to consider the distance between the points in both sets is to formulate the task into pairwise matching and often leads to a Quadratic Assignment Problem (QAP) which is known to be *NP*-hard. It can currently be solved with linear (Berg, Berg, & Malik, 2005) or spectral (Leordeanu & Hebert, 2005) relaxations. Another solution is to introduce serialization constraint in matching (Egozi et al., 2010). Scott and Nowak (2006) formulated the matching task into a cyclic order-preserving assignment problem. Dynamic Time Warping (DTW), which is widely applied in speech recognition and partial shape matching, is also an order-preserving matching technique because it was initially developed for matching time series. It can also be applied to closed contour matching but requires alignment of two shapes before comparison (C. Xu et al., 2009). We apply the Fast Dynamic Time Warping (FastDTW) techniques by Salvador and Chan (2007) for partial shape matching.

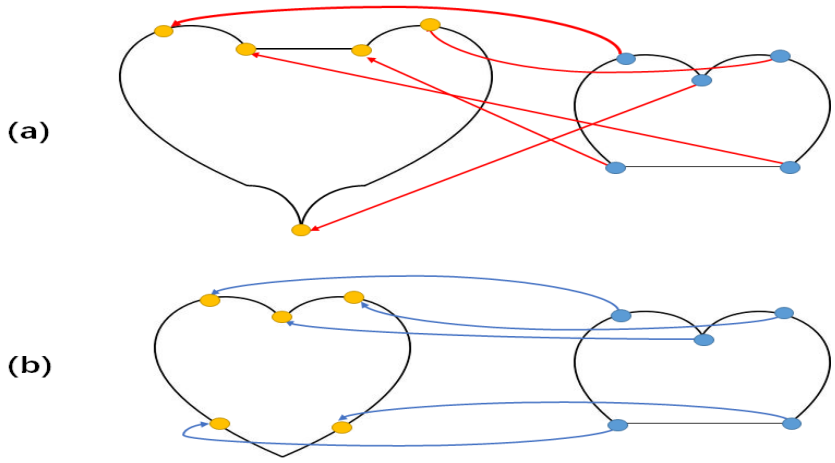

Figure 32: *Geometric matching. We attempt to match an identical shape on the right with two different templates on the left using the selected landmark points. The correspondence between points is shown with arrows. (a) The template on the left is generated by rearranging the landmark points while preserving their local geometric features. The cross of the correspondence arrows suggests that the match is not geometric. (b) The identical shape is matched with a "heart" template. Note that the there are no crosses between correspondent arrows, which suggests a greater affinity between two shapes.*

5.3.2 Global matching with directed chords pattern

Local features can provide sufficient discrimination power between shapes in many circumstances. However, the methods that rely mainly on local features may easily be misguided by some local similarities without considering the global geometric structure. Figure 33 provides difficult pairs of shapes that are easily mismatched. Thus, it is not reliable to completely match with local features only. Global geometric structures also play an important role in more accurate matching. In this section, we try to discuss the global geometric relationship between shapes by estimating a plane transformation.

With a set of correspondence between points sampled from two shapes, we can continue to estimate a plane transformation $\Psi : \mathcal{R}^2 \rightarrow \mathcal{R}^2$ to map points from one shape to the other. In this work, we mainly use a GMM similar to the Coherent Point Drift (CPD) algorithm (Myronenko & Song, 2010) to estimate the transformation. Given two point sets: an anchor set $S_Y = \{p_y | y = 1 \cdots M\}$, and a data set $S_X = \{p_x | x = 1 \cdots N\}$, we wish to estimate a

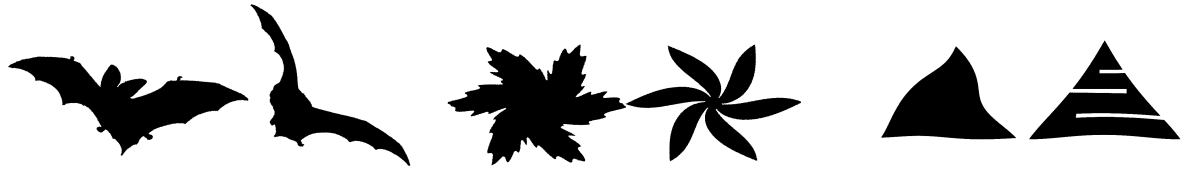

Figure 33: *Three examples that may cause mismatching. These three pairs of shapes are similar but have significant deformation of contours.*

transformation Ψ that minimizes the following energy function:

$$E_{Global}(S_X, \Psi(S_Y)) = \sum_{p_x \in S_X} \sum_{p_y' \in \Psi(S_Y)} ||p_x - p_y'||^2. \tag{36}$$

Energy minimization is achieved with a GMM framework. The anchor set S_Y is treated as the GMM centroid, and S_x is generated by the GMM. In this work, we consider the Euclidean space and the DCP space to measure the similarity simultaneously. Given an arbitrary point p, the GMM probability density function is

$$\eta(p) = \beta \cdot \eta(p|M+1) + (1-\beta) \sum_{y=1}^{M} \eta(y)\eta(p|y), \tag{37}$$

where

$$\eta(p|M+1) = \frac{1}{N}, \tag{38}$$

$$\eta(y) = \frac{1}{M}, \tag{39}$$

$$\eta(p|y) = \eta_{pos} \times \eta_{DCP}, \tag{40}$$

$$\eta_{pos} = \frac{1}{2\pi\sigma_{pos}^2} \cdot e^{-\frac{||p-p_y||^2}{2\sigma_{pos}^2}}, \tag{41}$$

$$\eta_{DCP} = \frac{1}{(2\pi\sigma_{DCP}^2)^K} \cdot e^{-\frac{d_{DCP}(p,p_y)}{2\sigma_{DCP}^2}}. \tag{42}$$

$\eta(p|M+1)$ is an addition uniform distribution that accounts for noise and extra outliers. $\eta(p|y)$ is the probability function of the components in GMM, and each of them is weighted equally with $\frac{1}{M}$. Note that in Equation 42 the probability is based on the DCP distance d_{DCP} instead of the Eculidean distance. Although the DCP distance d_{DCP} has a form of χ^2 statistics and $d_{DCP} \sim \chi^2(K-1)$, we choose the Gaussian probability density function to preserve the

benign properties of the CPD algorithm. When K is large enough, $\frac{(d_{DCP}(p,p_y)-(K-1))}{\sqrt{2(K-1)}} \sim N(0,1)$ follows a standard normal distribution. One can also choose the χ^2 distribution when K is small.

The GMM model can be re-parameterized with a set of parameters (Θ, σ_{pos}). Ψ is the transformation of the point set, and σ_{pos} is the standard deviation of the point position. These parameters are estimated by minimizing the following negative log-likelihood function:

$$E(\Theta, \sigma) = -\sum_{x=1}^{N} log \sum_{y=1}^{M+1} \eta(y)\eta(p_x|y). \tag{43}$$

The correspondence between two points p_y and p_x is defined with the posterior probability $\eta(y|p_x)$ of the GMM centroid given the data from the moving set. According to Bayes' rule, $\eta(y|p_x) = \frac{\eta(y)\eta(p_x|y)}{\eta(p_x)}$.

We apply the Expectation-Maximization (EM) algorithm to determine the minimum of Equation 43. The basic idea of the EM algorithm is to estimate the maximum likelihood via a two-step repetitive iteration: E-steps and M-steps. In E-steps, the algorithm calculates the expectation of the log likelihood function based on the current parameters. In M-steps, the parameters for the next E-step is estimated by maximizing the likelihood function. In our cases, the posterior probability $\eta_{old}(y|p_x)$ is calculated in the E-steps. The new parameters for the next iteration are obtained by minimizing the complete negative log-likelihood function in the M-steps:

$$Q = -\sum_{x=1}^{N}\sum_{y=1}^{M} \eta_{old}(y|p_x) log(\eta_{new}(y)\eta_{new}(p_x|y)). \tag{44}$$

The E-steps and M-steps are conducted alternatively until convergence. To explicitly express the relationship between the transformation Ψ and the above Q-function, Equation 44 can also be rewritten as follows:

$$
\begin{aligned}
Q(\Psi,\Theta) = &-\frac{1}{2\sigma_{pos}^2}\sum_{x=1}^{N}\sum_{y=1}^{M}\eta_{old}(y|p_x)||p_x-\Psi(p_y,\Theta)||^2 \\
&-\frac{1}{2\sigma_{DCP}^2}\sum_{x=1}^{N}\sum_{y=1}^{M}\eta_{old}(y|p_x)d_{DCP}(p_x,p_y) \\
&+(\sum_{x=1}^{N}\sum_{y=1}^{M}\eta_{old}(y|p_x))\cdot log(\sigma_{pos}^2) \\
&+(\sum_{x=1}^{N}\sum_{y=1}^{M}\eta_{old}(y|p_x))\cdot[(K+1)log(2\pi)+log(\sigma_{DCP}^2)]).
\end{aligned}
\tag{45}
$$

Note that the transformation Ψ is not applied to the DCP distance calculation and that the parameters of the DCP are not optimized; thus, the term that contains the DCP distance can be regarded as constant. In fact, the second and fourth terms are independent of Θ and σ_{pos}, which means that they can be ignored. The posterior probability based on the previous estimation in the M-steps can be calculated as :

$$
\eta_{old}(y|p_x) = \frac{e^{-\frac{1}{2}\frac{||px-\Psi(py,\Theta)||^2}{\sigma_{pos}^{old}}}\times e^{-\frac{1}{2}\frac{d_{DCP}(px,py)}{\sigma_{DCP}}}}{\sum_{y=1}^{M} e^{-\frac{1}{2}\frac{||px-\Psi(py,\Theta)||^2}{\sigma_{pos}^{old}}-\frac{1}{2}\frac{d_{DCP}(px,py)}{\sigma_{DCP}}}+c},
\tag{46}
$$

where

$$
c = (2\pi\sigma_{pos}^{old})^2(2\pi\sigma_{DCP})^K\frac{\beta}{1-\beta}\frac{M}{N}.
\tag{47}
$$

Note that the transformation Ψ is only a Euclidean motion and is applied only to the position of the point. The DCP matching probability η_{DCP} uses the distance of DCP with no transformation. By minimizing the Q function (Equation 45), the negative log-likelihood function (Equation 43) decreases until it reaches its local minimum. The complete algorithm is shown in Algorithm 3. Because our approach only enhances the matching probability calculation with the DCP distance, the general procedure is similar to the original CPD but with a different probability calculation in the estimation step.

5.3.3 Matching with local and global DCP features

In this section, we define the shape distance that considers both the local and global features simultaneously. We match two shapes with local DCP features to obtain the correspondence

Algorithm 3 Global Matching using Coherent Point Drift with Directed Chords Pattern

Input: Anchor point set $S_Y = \{p_y | y = 1 \cdots M\}$ and the set of the DCP of each point $\mathbf{DCP}_Y = \{DCP(p_y) | p_y \in S_Y\}$; Data point set $S_X = \{p_x | x = 1 \cdots N\}$ and the set of the DCP of each point $\mathbf{DCP}_X = \{DCP(p_x) | p_x \in S_X\}$

Output: The result of alignment $\mathbf{T} = \Psi(S_Y, \Phi) = S_Y + \mathbf{GW}$; Probability of correspondence $\mathbf{H} = \{\eta\}$

1: Initialization: $\mathbf{W} = \mathbf{0}$, $\sigma^2 = \frac{1}{2MN} \sum_{x,y=1}^{M,N} ||p_x - p_y||^2$

2: Construct kernel matrix $\mathbf{G}$: $g_{ij} = e^{-\frac{1}{2\xi^2}||p_i - p_j||}$, where $i, j \in y$, and $\xi > 0$ is a smoothness parameters

3: Expectation Maximization

4: **while** not converged **do**

5: E-step:

6: Compute the probabilities $\mathbf{H}$ according to Equation 46

7: M-step:

8: Solve $\mathbf{G} + \lambda\sigma^2 d(\mathbf{H1})^{-1}\mathbf{W} = d(\mathbf{H1})^{-1}\mathbf{H}S_X - S_Y$ with the method in (Myronenko & Song, 2010)

9: **end while**

between points and then estimate the transformation as described in Section 5.3.2. After matching, we used a weighted sum of E_{Local} and E_{Global} to measure the shape distance.

Given two shapes that are closed 2D curves, our framework for matching two shapes accounts for both local and global DCP features:

1. Sample the two shapes uniformly at an equal distance along the contour and obtain the sample points of same size $S_1 = p_1, p_2, \cdots, p_n$ and $S_2 = q_1, q_2, \cdots, q_n$

2. Calculate the DCP at each point in both point sets S_1 and S_2.

3. Match the two sample point sets with the calculated DCP with dynamic time warping as described in Section 5.3.1. The correspondence between points is restored.

4. Estimate the transformation Ψ between S_1 and S_2, as described in Section 5.3.2. The initialization of the iteration uses the correspondence restored in local matching.

5. The shape distance between two shapes is determined by:

$$E_{final} = E_{Local} + \gamma E_{global}, \tag{48}$$

where γ is a weighting factor.

5.4 Evaluations

To verify the performance of our proposed approach, we applied our shape-retrieval method based on the DCP to open benchmark databases. A comparison with some well-known shape-retrieval solutions based on other shape descriptors is also presented in this section. The program used in our shape-retrieval experiments is a combination of Python and C++ functions. The program is run on an Intel I5-4590 3.3 GHz computer with 8G ram and an AMD R9 280 graphics card. In the following subsections, we present the experimental results with two benchmark databases. In addition, we apply our DCP descriptor to the task of image repair to evaluate its performance in the practical application.

Table 6: *Benchmark test result on MPEG-7. Our approach is competitive with previous methods.*

Methods	Scores(%)
Our approach	88.67
Shape Contexts (Belongie et al., 2002)	76.51
Contour Flexibility (C. Xu et al., 2009)	89.31
Height Function (Wang et al., 2012)	89.66
Inner Distance Shape Context (Ling & Jacobs, 2007)	85.4
The Method of Michel et al. (2011)	83.4

5.4.1 Experiments on MPEG-7 dataset

One of the most popular datasets used to test shape matching methods is Part B of the MPEG-7-Core Experiment CE-Shape-1 data set (Latecki, Lakamper, & Eckhardt, 2000). The data set contains 1400 binary images of various kinds of objects grouped in 70 categories of 20 images each. The benchmark test is conducted in a way called the Bullseye test. Each shape image in the data set is used as a query and the shapes of the 40 highest scores are retrieved from the dataset to evaluate the retrieval rates. The retrieval rate of the test equals the percentage of correctly matched images (out of $20 \times 70 \times 20 = 28000$).

As shown in Table 6, the retrieval rate of our approach is 88.67%. Figure 34 provides some characteristic results of our full shape matching technique on the MPEG-7 dataset. We also include five previous methods for comparison: Shape Contexts (Belongie et al., 2002), Contour Flexibility (C. Xu et al., 2009), Height Function (Wang et al., 2012), Inner Distance Shape Context (Ling & Jacobs, 2007) and the method of Michel et al. (2011). The comparison results are presented in Table 6, which shows that our approach outperforms other methods based on global chord distribution. Although our approach does not achieve the best scores, our methods have a relatively small distance compared to the other shape descriptor, which relies on detecting shape curvatures.

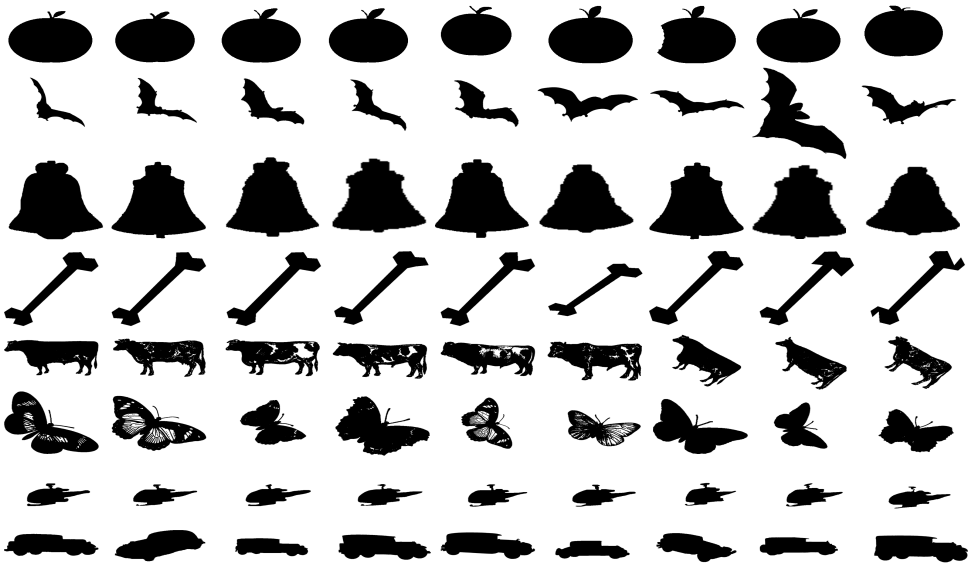

Figure 34: *Characteristic results for shape matching on the MPEG-7 dataset. The images in the first column are used for the query. The rest of each row includes retrieved shapes in order of decreasing shape distance.*

5.4.2 Experiments on gesture dataset

Another benchmark dataset for testing our method is the Gesture dataset by Milios and Petrakis (2000). This dataset contains 17 classes and 980 different hand gesture samples. The queries are shown in Figure 35. The benchmark test designed by Milios and Petrakis (2000) uses human relevance data to evaluate the performance of the shape retrieval. We include four previous methods for comparison: the two dynamic programming methods by Milios and Petrakis (2000), the geometric moments (M. Hu, 1962), and sequential moments (Gupta & Srinath, 1987). Figure 36 shows the precision-recall plots. The recall rate is defined as the ratio of relevant gestures retrieved from the category to which the gesture belongs. The precision rate is the ratio of corresponding gestures retrieved out of all retrievals.

We follow the same procedure for computing the precision-recall rate as that in the paper (Milios & Petrakis, 2000). There are 17 queries in the dataset. For each query, we conduct visual judgment on the retrieve results of our approach. Then for each query, we merged the answers obtained by all the candidate methods and form a database that contains the relevant entries. Although such a method does not allow for absolute judgments such as "method A overlook 10% of the total similar answers in the database.", but it is relatively fair to make

comparison judgment such as " method A returns 10% fewer correct answers than method B". Since visual judgment is used in our experiment, to ensure that we are comparing the methods instead of individual difference of the human judgment, the precision-recall rate shown in figure 36 is the average of the 17 queries. As shown in Figure 36, our approach returns more correct answers than other methods.

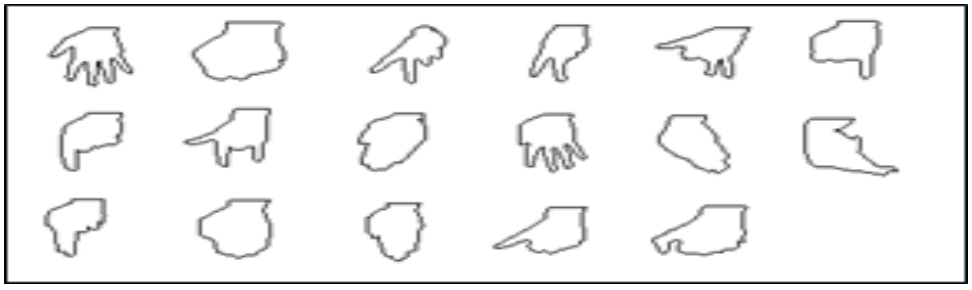

Figure 35: *Queries in the gesture dataset of Milios and Petrakis (2000).*

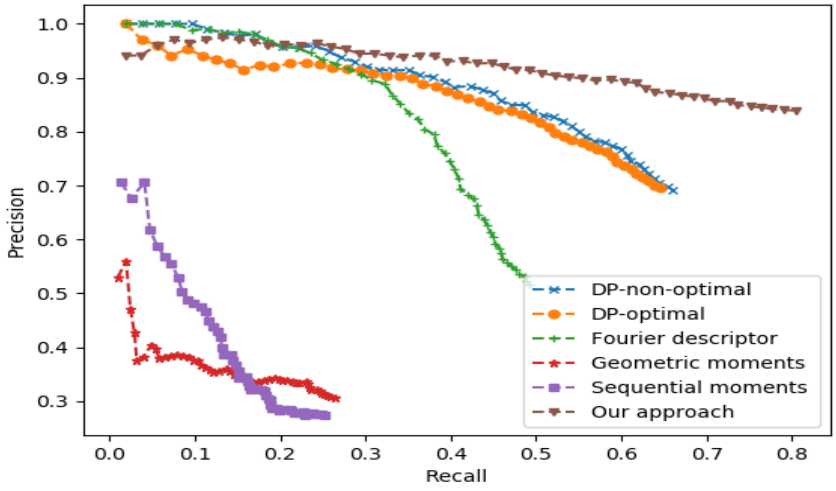

Figure 36: *The precision-recall rate for the retrieval on the gesture dataset. The results generated by our approach outperform those of other methods.*

5.4.3 Application on structure completion in image repair

We apply our shape descriptor to the task of image repair to complete broken structures in images. Digital image repair, also called as image inpainting or image completion in some references, aims to repair a damaged image according to the information in the remaining

region. Given an imapired image I with a cavity C, the image repair task is to infer the contents in the cavity according to the source region $S = I - C$. The image repair task can be divided into two sub-tasks: structure completion and texture synthesis. As noted in a previous study (Bertalmio et al., 2002), the repairing order matters for generation of reasonable contents for the cavity. Structures within the images should be restored prior to textures. Otherwise, structure distortion may appear within in the target region. Structure completion is not an easy task. Previous methods attempted to repair structures either with preset rules or implicit repair. The method by Bertalmio et al. (2002) repaired the structures in the image by allowing the pixels on the structure to propagate according to PDE. J. Sun, Yuan, Jia, and Shum (2005) also had a similar idea and used belief propagation to connect broken structural lines. S. Li and Zhao (2011) matched end points of the broken structure using the color information in the neighbors and designed a set of rules to create more crossing during structure propagation. The Image Melding technique proposed by Darabi et al. (2012) repairs structures with neighbor constraints in pixels in their energy functions. All of these methods are able to handle simple structures such as straight lines or simple curves.

For most previous methods, the structure completion relies mainly on the geometric structures of the broken lines near the cavity, and the task is similar to curve-fitting. However, in some cases, a simple connection between broken structural lines with a smooth curve is insufficient to generate reasonable contents within the target areas. Figure 37(b) provides challenging examples of repairing the limbs of a horse. It is obvious that it is impractical to repair the contours of the horse limbs simply according to the geometric features (such as curvatures). To generate the limbs of the horse, we must enable the program to recognize the object it is matching. Thus, prior object knowledge is necessary. To repair a specific object requires prior object information. Here, instead of developing a general image-repair system, we present a system to repair horses in the image with priors from the Weizmann Horse Database (Borenstein & Ullman, 2002, 2004).

The first step to repair the broken contour of an object is to recognize the shape according to the remaining contours. This recognition task can be converted to the task of partial shape matching. Figure 37 presents the repaired horse contours generated by our system. We use

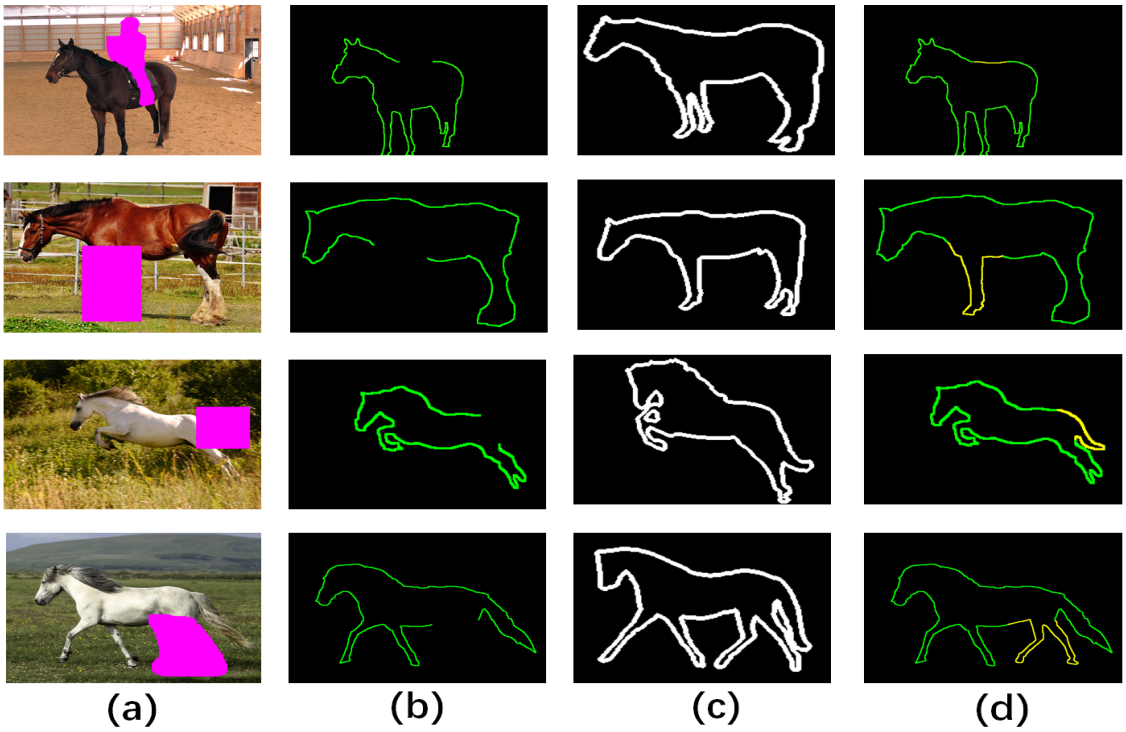

Figure 37: *Contour completion results with our approach using DCP. (a) Images of horses with holes. The target area is masked with magenta. (b) Impaired contours extracted from the images in (a) are used as input for shape matching. (c) Matched shapes retrieved from the Weizemann Horse Database. (d) Completed structures generated by our system. By modifying matched shapes with the estimated transformation, we repaired the missing structure in the target contours with the segments from the matched shapes.*

the broken contours as input and match them with the shapes in the database. After obtaining the object shape information, the missing structure is reconstructed by applying the estimated transformation Ψ on the shape template. After the structures are completed, the remaining region within the cavity is filled with texture synthesis techniques that use information from the color template. Figure 38 provides some comparison between our repaired images and those generated by previous methods. It is obvious that the introduction of object information can effectively avoid structure distortion and unnatural patterns.

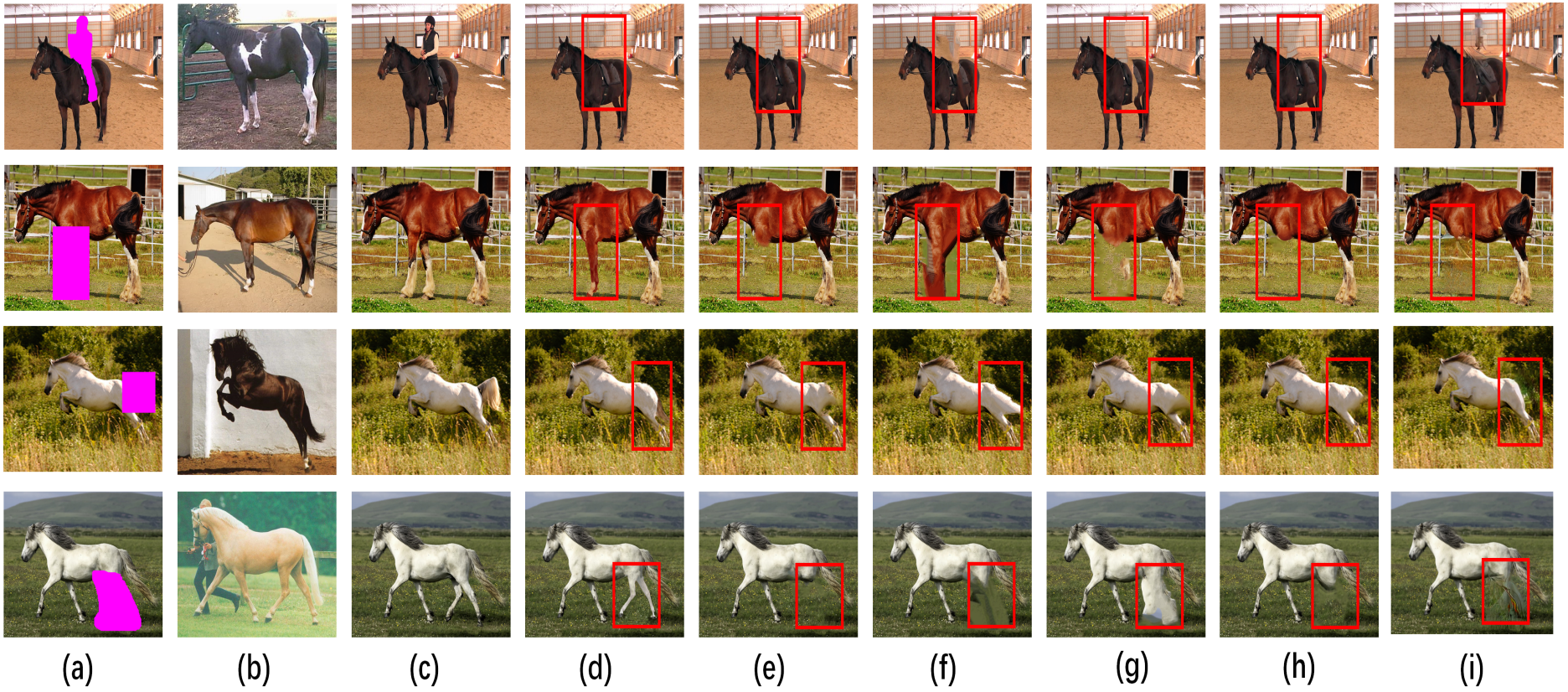

Figure 38: *Comparison with previous methods. (a) Masked image. (b) Color images of the matched shapes from the Weizmann Horse Dataset. (c) Ground truth. (d) Repaired results generated by our approach. The contours of the horses are properly reconstructed. (e) Results with Photoshop Content Aware Filling. (f) Results from the method of Bertalmio et al. (2002). (g) Results from the method of Wexler et al. (2004) (h) Results by Image Melding (Darabi et al., 2012). (i) Results from the method of Iizuka et al. (2017).*

5.5 Discussion

In this chapter, we present a novel shape descriptor called DCP, which uses the point-chord relationship to extract the geometric structure of the points on a contour. The DCP has significant discrimination power and can capture fluctuations in the contour curvatures of the shape. The DCP descriptor takes advantage of the distance and the angular position of chords to feature a point. It is invariant to translation and scaling, and can be easily made rotation-invariant according to specific requirements. Based on the designed DCP descriptor, we develop a graph-matching scheme to match the local features of the sample points between two shapes. To make our matching preserve the geometric structures within shapes, we estimate the transformation between shapes using a GMM model equipped with a DCP measure. Experiments on benchmark datasets suggest that our shape matching approach is competitive with previous methods. To further verify the performance of our DCP shape descriptor, we provide an application for demonstration: an object-aware image repair program. The repaired results show the effectiveness of our matching scheme based on the DCP.

Although the DCP shape descriptor has shown sufficient discriminating power in the benchmark test and actual application, its complete characteristics are waiting for exploration and it still has room for further improvement. One problem that worths for exploration is the influence of sample methods on the robustness of DCP. In our experiments, all the contours are uniformly sampled. The relationship between the distribution of the sampling points and the DCP still needs careful inspection. In addition, the performance of DCP under different image details still needs further verification. Although our shape descriptor is invariant to certain transformation, whether the DCP can provide sufficient discriminating power in low-resolution images remains unknown. In the image with low resolution, the object contour can be very noisy.

The DCP shape descriptor may have bad performance when the shape contours are noisy. Because the DCP shape descriptor extracts shape features at each sample point by connecting neighboring points to construct chords. If the sampled points have serious deviations from the actual contour position, the constructed chords may be seriously biased and cause false signals

in DCP. Noises in the contours can be a serious barrier for the application of most contour-based shape descriptor. Although our approach performs well on the queries with noisy boundary in the benchmark test on MPEG-7, corresponding solutions for handling noises on the contour can be important for the DCP to be applied in a real-world situation.

In the future, we will try to develop a solid theoretical foundation for the DCP. One important property of the DCP is its sensitivity to contour curvature. It seems that there are some links between the contour curvature and the discriminating power of DCP. The DCP may be considered as a certain representation of curvature. H. Liu et al. (2008) provided a unified definition of curvature for curves. We will seek for possibility of expressing the DCP with this definition or developing a compatible theoretical framework.

6 Discussion and Conclusions

In this thesis, we study image repair with enhanced structure completion ability. We reconstruct structures in images with implicit and explicit enhancement. The implicit enhancement in structure completion is achieved by applying patches of various sizes in the image repair process. During the coarse-to-fine image repair, the sizes of patches used to complete the cavity can vary to capture the structure at different scales. Unlike previous methods that prefer to use large patches to capture more information, we add constraints upon the patch sizes to achieve a balance between performance and efficiency. In addition, we explore the potential versatility of the patch-based image completion framework. We adapt the image completion method with implicit enhancement to the shadow removal task. This adaptation is based on the insight that a shadow image can be considered as an image with damaged illumination. In other words, the cavity exists only in the illuminance field. Following this idea, we repair the illuminance of the shadow image with a patch-based method and combine it with the optimized reflectance of the image to obtain shadow-free images.

The explicit enhancement of structure completion is achieved via partial shape matching with our novel designed shape descriptor DCP. One type of important structure in an images is the contour of objects. To repair the deficit contour of an object, we need to equip the completion system with prior knowledge and enable the computer system to recognize the object. We used the DCP descriptor to retrieve shape via partial shape matching. Our DCP describes shapes with the spatial relationship between a local point and the chords distributed in its neighbor. The DCP is invariant to transformation and is highly correlated to contour curvatures. Equipped with DCP, we can retrieve the object shape from the database according to the impaired object contours in the image. Completion of the object contours is achieved by transforming the retrieved contours and combining them with the impaired contours.

6.1 Impact of this thesis

This thesis contains material from previous publications during my PhD program. The most important contributions of this thesis include:

Dynamic Patch System Unlike the traditional patch-based method in which used large patches of fixed size are used to include more information, we use patches of various sizes to capture the structural information in the completion process. Because structures vary in scale, patches of various size are retrieved simultaneously, and suitable patches are selected via a competitive mechanism. In addition, we allow the size of patches to vary according to the coarse level to maximize the information retrieved by the patches and reduce unnecessary computation. (Chapter 3. Work appeared in **CGI2017** (B. Liu, Li, Sheng, & Wu, 2017; B. Liu et al., 2019))

Shadow Removal via Image Completion We adapt our image completion techniques to the task of shadow removal task. Based on the idea that shadows in images can be regarded as holes in their illumination fields, we formulate the shadow removal task into an illumination recovery task. By dividing the image into its reflectance and illumination, we can directly repair the illumination without interference of the color information. The damaged illumination is repaired with a patch-based framework from image completion. The reflectance is optimized with our image completion approach with DPS. The generated shadow-free images have better color consistency than those generated with other techniques.

Efficient Partial Shape Matching We propose a novel shape descriptor called DCP for matching and recognition of 2D shapes. A series of chords are constructed by connecting sample points with a fixed interval, and each sample point is encoded via the spatial position related to chords. The proposed descriptor is sensitive to curvature and provides strong signals at points where contour curvature changes, and it is invariant to geometric transformation such as translation, rotation, and scaling. Corresponding points on similar shapes will have similar DCP, and we achieve correspondence between points by matching their DCP using dynamic time warping. Given the point correspondence, we estimate the transformation between shapes with a probabilistic model that accounts for both distances in Euclidean and DCP. We test our shape matching approach on open datasets, and the results demonstrate the effectiveness of our approach. We also apply our approach to the task of image repair. The image repair system based on our DCP can properly restore object contours with a prior collected database.

6.2 Limitation and future directions

I believe this work, especially the proposed Dynamic Patch System (DPS) and the DCP shape descriptor, are fundamentally novel. I hope that the systems and concepts presented in this thesis can be beneficial for studies of image processing and computer vision. Although this work makes improvements in image completion and goes beyond the previous framework, many difficulties and challenges still await. In particular, there is potential to improve our current work from the following aspects:

- **Improvement of implicit structure completion**

    1. Support superpixel patches and patches with arbitrary shape. The patches used in the completion are usually square. When capturing irregular patterns with square patches, partial loss of information is inevitable. Superpixel patches or deformed patches group pixels according to certain criteria, and the shapes of the patches are not limited to squares. They may help to capture patterns of arbitrary shapes and provide a foundation to implement completion at a higher level.
    2. Optimization of the match propagation paths. The original PatchMatch algorithm propagates the good matches between patches from the top left to the bottom right sequentially. It may be more efficient and flexible if match propagation is allowed to be conducted in multiple directions with different priorities.

- **Improvement of explicit structure completion**

    Generation of missing contents in the damaged object according to the template object requires further study:

    1. Content consistency within objects. In our completion framework, some unique missing contents of an object are generated by combining the target object and the matched part from the template object. However, differences may exist in image resolution, color, and textures. The recovered object requires consistency in the pre-existing content and repaired contents

2. Interaction between objects and background. In some situations, the objects are partially blocked and mixed with the image's background. When repairing these object contours, a smooth and natural transition between objects and the background is required.

In addition, explicit structure repair relies on recognition of the object. Thus, improving the recognition accuracy can help to reconstruct a better structure in the final completion results. We suggest that recognition can be improved from the following aspects:

1. Database collection for object recognition. The database provides prior knowledge for our completion approach to recognize the object in the impaired image. The ability of recognition heavily depends on the object information stored within the database. We suggest two directions for expansion of the database. One is to increase the number of object categories, which would be helpful for more general recognition tasks. The other direction is to create a specific database for certain recognition tasks, such as vehicles and pedestrians. The database should try to include various views or poses of the objects.
2. Handling multiple objects. In our completion approach, we repair a damaged image with a single target. However, in many situations, the damaged area covers several objects. Repair of these images may require separation in the recognizing phase and proper relocation of the different objects in the repair phase.
3. Consider contents similarity cues. Our shape matching approach takes advantage of the object contours only. Consideration of the object contents can help to improve the matching results.
4. Introduce internal structure recognition. Some internal structures within the objects (such as closed-curves within an object) may provide useful information and extra discrimination power in recognition.

Although The DCP shape descriptor has shown discriminating power in our experiments, there are rooms for improvement and problems for investigation:

1. Develop suitable sampling methods. The relationship between the sample point distribution and the discriminating power of DCP needs further investigation. Corresponding sampling methods should be developed accordingly.

2. Design corresponding noise suppressing approach. Noises on the contour may impair the discriminating power of DCP and cause false signals. An effective smoothing method is necessary for DCP to be put into practical use.

3. Compatibility with the curvature definition. The DCP is sensitive to contour curvature. It seems that there are certain connections between them. The DCP may be compatible with the definition of curvature or may be a certain representation of curvature. Corresponding theory on curvature may be used to explain the discriminating power of DCP.

- **Application of our techniques**

  The techniques we develop in this thesis can be adapted to other image and video processing tasks. For example, our shadow removal approach may be extended to the shadow removal tasks in the video, such as removing cast shadows in surveillance videos. In addition to shadow removal, other image processing tasks can also benefit from our image completion approach if they can be formulated into the patch-based completion framework. For instance, object cloning refers to the task of reproducing a given object in a specific location in an image. This task can be regarded as a completion task with already-known contents.

  The development of our approach derives several techniques for recognition and matching that can also be applied to computer vision tasks. For instance, our DCP descriptor can be applied to gesture and pose recognition. The benchmark test of our DCP descriptor shows that our shape descriptor can be used to develope gesture detection and human pose estimation systems.

References

Arbel, E., & Hel-Or, H. (2007). Texture-preserving shadow removal in color images containing curved surfaces. In *IEEE Conference on Computer Vision and Pattern Recognition 2007* (pp. 1–8).

Arias, P., Facciolo, G., Caselles, V., & Sapiro, G. (2011). A variational framework for exemplar-based image inpainting. *International Journal of Computer Vision*, *93*(3), 319–347.

Arica, N., & Vural, F. T. Y. (2003). BAS: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters*, *24*(9-10), 1627–1639.

Attneave, F. (1950). Dimensions of similarity. *American Journal of Psychology*, *63*(4), 516-556.

Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., & Verdera, J. (2001). Filling-in by joint interpolation of vector fields and gray levels. *IEEE Transactions on Image Processing*, *10*(8), 1200–1211.

Barnes, C., Shechtman, E., Finkelstein, A., & Goldman, D. (2009). Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics*, *28*(3), 24:1–24:11.

Barnes, C., Shechtman, E., Goldman, D. B., & Finkelstein, A. (2010). The generalized patchmatch correspondence algorithm. In *European Conference on Computer Vision 2010* (pp. 29–43).

Barrow, H., & Tenenbaum, J. (1978). Recovering intrinsic scene characteristics from images. In A. Hanson (Ed.), *Computer Vision Systems* (Vol. 2, pp. 1–25). Academic Press.

Bauckhage, C., & Tsotsos, J. K. (2005). Bounding box splitting for robust shape classification. In *IEEE International Conference on Image Processing 2005* (Vol. 2, pp. II–478).

Bellman, R. (1966). Dynamic programming. *Science*, *153*(3731), 34-37.

Belongie, S. J., Malik, J., & Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(4), 509–522.

Berg, A. C., Berg, T. L., & Malik, J. (2005). Shape matching and object recognition using low distortion correspondences. In *IEEE Conference on Computer Vision and Pattern Recognition 2005* (Vol. 1, pp. 26–33).

Bertalmio, M., Sapiro, G., Caselles, V., & Ballester, C. (2002). Image inpainting. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (pp. 417–424).

Bertalmio, M., Vese, L., Sapiro, G., & Osher, S. (2003). Simultaneous structure and texture image inpainting. *IEEE Transactions on Image Processing*, *12*(8), 882–889.

Biederman, I. (1987). Recognition-by-components: a theory of human image understanding. *Psychological Review*, *94*(2), 115-147.

Borenstein, E., & Ullman, S. (2002). Class-specific, top-down segmentation. In *European Conference on Computer Vision 2002* (pp. 109–122).

Borenstein, E., & Ullman, S. (2004). Learning to segment. In *European Conference on Computer Vision 2004* (pp. 315–328).

Bousseau, A., Paris, S., & Durand, F. (2009). User-assisted intrinsic images. *ACM Transactions on Graphics*, *28*(5), 130:1–130:10.

Brendt, W. (2011). Inpainting by joint optimization of linear combinations of exemplars. *IEEE Signal Processing Letters*, *18*(1), 75-78.

Bugeau, A., Bertalmio, M., Caselles, V., & Sapiro, G. (2010). A comprehensive framework for image inpainting. *IEEE Transactions on Image Processing*, *19*(10), 2634–2645.

Burt, P. J., & Adelson, E. H. (1983). A multiresolution spline with application to image mosaics. *ACM Transactions on Graphics*, *2*(4), 217–236.

Celebi, M. E., & Aslandogan, Y. A. (2005). A comparative study of three moment-based shape descriptors. In *International Conference on Information Technology: Coding and Computing - Volume II* (Vol. 1, pp. 788–793).

Chakrabarti, K., Ortega-Binderberger, M., Porkaew, K., & Mehrotra, S. (2000). Similar shape retrieval in mars. In *IEEE International Conference on Multimedia and Expo 2000* (Vol. 2, pp. 709–712).

Chan, T. F., & Shen, J. (2001). Nontexture inpainting by curvature-driven diffusions. *Journal*

of Visual Communication and Image Representation, *12*(4), 436–449.

Chen, T., Cheng, M.-M., Tan, P., Shamir, A., & Hu, S.-M. (2009). Sketch2photo: Internet image montage. *ACM Transactions on Graphics*, *28*(5), 124:1–124:10.

Criminisi, A., Perez, P., & Toyama, K. (2003). Object removal by exemplar-based inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition 2003* (Vol. 2, pp. 721–728).

Cucchiara, R., Grana, C., Piccardi, M., & Prati, A. (2003). Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(10), 1337–1342.

Darabi, S., Shechtman, E., Barnes, C., Goldman, D. B., & Sen, P. (2012). Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transaction on Graphics*, *31*(4), 82–1.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *39*, 1–38.

Dolhansky, B., & Canton Ferrer, C. (2018). Eye in-painting with exemplar generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018* (pp. 7902–7911).

Donoser, M., Riemenschneider, H., & Bischof, H. (2009). Efficient partial shape matching of outer contours. In *Asian Conference on Computer Vision 2009* (pp. 281–292).

Drori, I., Cohenor, D., & Yeshurun, H. (2003). Fragment-based image completion. *ACM Transactions on Graphics*, *22*(3), 303–312.

Du, Z., Lin, H., & Bao, H. (2005). Shadow removal in gradient domain. In M. Kamel & A. Campilho (Eds.), *International Conference Image Analysis and Recognition 2005, Lecture Notes in Computer Science* (Vol. 3656, pp. 107–115). Springer Berlin Heidelberg.

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons.

Efros, A. A., & Freeman, W. T. (2001). Image quilting for texture synthesis and transfer. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive*

Techniques (pp. 341–346).

Efros, A. A., & Leung, T. K. (1999). Texture synthesis by non-parametric sampling. In *IEEE International Conference on Computer Vision 1999* (Vol. 2, pp. 1033–1038).

Egozi, A., Keller, Y., & Guterman, H. (2010). Improving shape retrieval by spectral matching and meta similarity. *IEEE Transactions on Image Processing*, *19*(5), 1319–1327.

Finlayson, G. D., Drew, M. S., & Lu, C. (2004). Intrinsic images by entropy minimization. In *European Conference on Computer Vision 2004* (pp. 582–595).

Finlayson, G. D., & Fredembach, C. (2004). Fast re-integration of shadow free images. In *Twelth Color Imaging Conference: Color Science and Engineering Systems, Technologies, and Applications* (Vol. 2004, pp. 117–122).

Finlayson, G. D., Hordley, S. D., & Drew, M. S. (2002). Removing shadows from images. In *Seventh European Conference on Computer Vision-Part IV* (pp. 823–836).

Finlayson, G. D., Hordley, S. D., Lu, C., & Drew, M. S. (2006). On the removal of shadows from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(1), 59–68.

Ford Jr, L. R., & Fulkerson, D. R. (2015). *Flows in networks*. Princeton university press.

Fredembach, C., & Finlayson, G. (2006). Simple shadow removal. In *the 18th International Conference on Pattern Recognition* (Vol. 1, pp. 832–835).

Gong, H., & Cosker, D. (2017). User-assisted image shadow removal. *Image and Vision Computing*, *62*, 19–27.

Gonzalez, R. C., & Woods, R. E. (2006). *Digital Image Processing (3rd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Gracias, N., Mahoor, M., Negahdaripour, S., & Gleason, A. (2009). Fast image blending using watersheds and graph cuts. *Image and Vision Computing*, *27*(5), 597–607.

Gryka, M., Terry, M., & Brostow, G. J. (2015). Learning to remove soft shadows. *ACM Transactions on Graphics*, *34*(5), 153.

Guo, R., Dai, Q., & Hoiem, D. (2013). Paired regions for shadow detection and removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(12), 2956–2967.

Guo, Z., Zhang, L., & Zhang, D. (2010). A completed modeling of local binary pattern operator

for texture classification. *IEEE Transactions on Image Processing*, *19*(6), 1657–1663.

Gupta, L., & Srinath, M. D. (1987). Contour sequence moments for the classification of closed planar shapes. *Pattern Recognition*, *20*(3), 267–272.

Han, J. H., & Poston, T. (2001). Chord-to-point distance accumulation and planar curvature: a new approach to discrete curvature. *Pattern Recognition Letters*, *22*(10), 1133–1144.

He, K., Zhen, R., Yan, J., & Ge, Y. (2017). Single-image shadow removal using 3d intensity surface modeling. *IEEE Transactions on Image Processing*, *26*(12), 6046-6060.

Heimann, T., & Meinzer, H.-P. (2009). Statistical shape models for 3d medical image segmentation: a review. *Medical Image Analysis*, *13*(4), 543–563.

Hu, M. (1962). Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory*, *8*(2), 179–187.

Hu, R.-X., Jia, W., Ling, H., Zhao, Y., & Gui, J. (2014). Angular pattern and binary angular pattern for shape retrieval. *IEEE Transactions on Image Processing*, *23*(3), 1118–1127.

Iizuka, S., Simo-Serra, E., & Ishikawa, H. (2017). Globally and locally consistent image completion. *ACM Transactions on Graphics*, *36*(4), 107.

Jalba, A. C., Wilkinson, M. H., & Roerdink, J. B. (2006). Shape representation and recognition through morphological curvature scale spaces. *IEEE Transactions on Image Processing*, *15*(2), 331–341.

Jia, J., & Tang, C.-K. (2003). Image repairing: Robust image synthesis by adaptive ND tensor voting. In *IEEE Conference on Computer Vision and Pattern Recognition 2003* (Vol. 1, pp. I–I).

Kanizsa, G. (1985). Seeing and thinking. *Acta Psychologica*, *59*(1), 23–33.

Keogh, E. J., & Ratanamahatana, C. A. (2005). Exact indexing of dynamic time warping. *Knowledge and Information Systems*, *7*(3), 358–386.

Khan, S. H., Bennamoun, M., Sohel, F., & Togneri, R. (2016). Automatic shadow detection and removal from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *38*(3), 431–446.

Klinker, G. J., Shafer, S. A., & Kanade, T. (1990). A physical approach to color image understanding. *International Journal of Computer Vision*, *4*(1), 7–38.

Kopf, J., Fu, C.-W., Cohen-Or, D., Deussen, O., Lischinski, D., & Wong, T.-T. (2007). Solid texture synthesis from $2D$ exemplars. *ACM Transactions on Graphics*, *26*(3), 2.

Kwatra, V., Schödl, A., Essa, I., Turk, G., & Bobick, A. (2003). Graphcut textures: image and video synthesis using graph cuts. *ACM Transactions on Graphics*, *22*(3), 277–286.

Kwok, T.-H., Sheung, H., & Wang, C. C. (2010). Fast query for exemplar-based image completion. *IEEE Transactions on Image Processing*, *19*(12), 3106–3115.

Lalonde, J.-F., Efros, A. A., & Narasimhan, S. G. (2010). Detecting ground shadows in outdoor consumer photographs. In *European Conference on Computer Vision 2010* (pp. 322–335).

Land, E. H., & McCann, J. J. (1971). Lightness and retinex theory. *Journal of the Optical Society of America*, *61*(1), 1–11.

Latecki, L. J., Lakamper, R., & Eckhardt, T. (2000). Shape descriptors for non-rigid shapes with a single closed contour. In *IEEE Conference on Computer Vision and Pattern Recognition 2000* (Vol. 1, pp. 424–429).

Latecki, L. J., Megalooikonomou, V., Wang, Q., & Yu, D. (2007). An elastic partial shape matching technique. *Pattern Recognition*, *40*(11), 3069–3080.

Lefebvre, S., & Hoppe, H. (2005). Parallel controllable texture synthesis. *ACM Transactions on Graphics*, *24*(3), 777–786.

Leordeanu, M., & Hebert, M. (2005). A spectral technique for correspondence problems using pairwise constraints. In *Tenth IEEE International Conference on Computer Vision* (Vol. 1, pp. 1482–1489).

Levin, A., Zomet, A., & Weiss, Y. (2003). Learning how to inpaint from global image statistics. In *IEEE International Conference on Computer Vision 2003* (Vol. 1, p. 305-312).

Li, S., & Zhao, M. (2011). Image inpainting with salient structure completion and texture propagation. *Pattern Recognition Letters*, *32*(9), 1256–1266.

Li, Y., Liu, S., Yang, J., & Yang, M.-H. (2017). Generative face completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017* (pp. 3911–3919).

Liang, L., Liu, C., Xu, Y.-Q., Guo, B., & Shum, H.-Y. (2001). Real-time texture synthesis by

patch-based sampling. *ACM Transactions on Graphics*, *20*(3), 127–150.

Lin, Y., Dou, J., & Wang, H. (1992). Contour shape description based on an arch height function. *Pattern Recognition*, *25*(1), 17–23.

Ling, H., & Jacobs, D. W. (2007). Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(2), 286–299.

Liu, B., Li, P., Sheng, B., Nie, Y., & Wu, E. (2019). Structure-preserving image completion with multi-level dynamic patches. *The Visual Computer*, *35*, 85–98.

Liu, B., Li, P., Sheng, B., & Wu, E. (2017). Image completion with dynamic patches. In *Proceedings of the Computer Graphics International Conference 2017* (pp. 3:1–3:4).

Liu, F., & Gleicher, M. (2008). Texture-consistent shadow removal. In *European Conference on Computer Vision 2008* (pp. 437–450).

Liu, H., Latecki, L. J., & Liu, W. (2008). A unified curvature definition for regular, polygonal, and digital planar curves. *International Journal of Computer Vision*, *80*(1), 104–124.

Liu, H.-C., & Srinath, M. D. (1990). Partial shape classification using contour matching in distance transformation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12*(11), 1072–1079.

Manay, S., Cremers, D., Hong, B. W., Jr Yezzi, A. J., & Soatto, S. (2006). Integral invariants for shape matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(10), 1602-1618.

Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision 2001* (Vol. 2, pp. 416–423).

Masnou, S., & Morel, J. M. (1998). Level lines based disocclusion. In *International Conference on Image Processing 1998* (Vol. 3, pp. 259–263).

Michel, D., Oikonomidis, I., & Argyros, A. A. (2011). Scale invariant and deformation tolerant partial shape matching. *Image and Vision Computing*, *29*(7), 459–469.

Milios, E. E., & Petrakis, E. G. M. (2000). Shape retrieval based on dynamic programming. *IEEE Transactions on Image Processing*, *9*(1), 141–147.

Mohan, A., Tumblin, J., & Choudhury, P. (2007). Editing soft shadows in a digital photograph. *IEEE Computer Graphics and Applications*, *27*(2), 23–31.

Mokhtarian, F., & Mackworth, A. K. (1992). A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*(8), 789–805.

Mokhtarian, F., & Suomela, R. (1998). Robust image corner detection through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*(12), 1376–1381.

Mukundan, R. (2004). A new class of rotational invariants using discrete orthogonal moments. In *The 6th IASTED International Conference on Signal and Image Processing* (pp. 84–88).

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, *5*(1), 32-38.

Myronenko, A., & Song, X. B. (2010). Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(12), 2262–2275.

Ng, H.-F., Chen, I.-C., & Liao, H.-Y. (2013). An illumination invariant image descriptor for color image matching. *Scientometrics*, *25*(1), 306–311.

Nill, N. B., & Bouzas, B. H. (1992). Objective image quality measure derived from digital image power spectra. *Optical Engineering*, *31*(4), 813–825.

Palmer, S. E. (1999). *Vision science: Photons to phenomenology*. Cambridge, Massachusetts: MIT press.

Panagopoulos, A., Wang, C., Samaras, D., & Paragios, N. (2011). Illumination estimation and cast shadow detection through a higher-order graphical model. In *IEEE Conference on Computer Vision and Pattern Recognition 2011* (pp. 673–680).

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., & Efros, A. A. (2016). Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition 2016* (pp. 2536–2544).

Pessoa, L., Thompson, E., & Noe, A. (1998). Finding out about filling-in: a guide to perceptual completion for visual science and the philosophy of perception. *Behavioral and Brain*

Sciences, *21*(06), 723–748.

Qu, L., Tian, J., He, S., Tang, Y., & Lau, R. W. (2017). Deshadownet: A multi-context embedding deep network for shadow removal. In *IEEE Conference on Computer Vision and Pattern Recognition 2017* (pp. 4067–4075).

Ren, J. S., Xu, L., Yan, Q., & Sun, W. (2015). Shepard convolutional neural networks. In *Advances in Neural Information Processing Systems* (pp. 901–909).

Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, *60*, 259–268.

Salvador, S., & Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, *11*(5), 561–580.

Schmidt, F. R., Farin, D., & Cremers, D. (2007). Fast matching of planar shapes in sub-cubic runtime. In *IEEE International Conference on Computer Vision 2007* (pp. 1–6).

Schmidt, F. R., Toppe, E., & Cremers, D. (2009). Efficient planar graph cuts with applications in computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition 2009* (p. 351-356).

Scott, C., & Nowak, R. D. (2006). Robust contour matching via the order-preserving assignment problem. *IEEE Transactions on Image Processing*, *15*(7), 1831–1838.

Shen, J., Jin, X., Zhou, C., & Wang, C. C. (2007). Gradient based image completion by solving the poisson equation. *Computers & Graphics*, *31*(1), 119–126.

Shen, J., Yang, X., Li, X., & Jia, Y. (2013). Intrinsic image decomposition using optimization and user scribbles. *IEEE Transactions on Cybernetics*, *43*(2), 425–436.

Shen, L., Tan, P., & Lin, S. (2008). Intrinsic image decomposition with non-local texture cues. In *IEEE Conference on Computer Vision and Pattern Recognition 2008* (pp. 1–7).

Shor, Y., & Lischinski, D. (2008). The shadow meets the mask: Pyramid-based shadow removal. *Computer Graphics Forum*, *27*(2), 577–586.

Shusterman, E., & Feder, M. (1994). Image compression via improved quadtree decomposition algorithms. *IEEE Transactions on Image Processing*, *3*(2), 207–215.

Smith, S. P., & Jain, A. K. (1982). Chord distributions for shape matching. *Computer Graphics and Image Processing*, *20*(3), 259–271.

Sun, J., Yuan, L., Jia, J., & Shum, H.-Y. (2005). Image completion with structure propagation. *ACM Transactions on Graphics*, *24*(3), 861–868.

Sun, Q., Ma, L., Joon Oh, S., Van Gool, L., Schiele, B., & Fritz, M. (2018). Natural and effective obfuscation by head inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018* (pp. 5050–5059).

Sunkavalli, K., Johnson, M. K., Matusik, W., & Pfister, H. (2010). Multi-scale image harmonization. *ACM Transactions on Graphics*, *29*(4), 125:1–125:10.

Swain, M. J., & Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision*, *7*(1), 11–32.

Szummer, M., Kohli, P., & Hoiem, D. (2008). Learning CRFs using graph cuts. In D. Forsyth, P. Torr, & A. Zisserman (Eds.), *European Conference on Computer Vision 2008* (Vol. 5303, pp. 582–595). Springer Berlin Heidelberg.

Tappen, M. F., Freeman, W. T., & Adelson, E. H. (2003). Recovering intrinsic images from a single image. In *Advances in Neural Information Processing Systems* (pp. 1367–1374).

Telea, A. (2004). An image inpainting technique based on the fast marching method. *Journal of Graphics Tools*, *9*(1), 23–34.

Tschumperle, D. (2006). Fast anisotropic smoothing of multi-valued images using curvature-preserving pde's. *International Journal of Computer Vision*, *68*(1), 65–82.

Veltkamp, R. C., & Hagedoorn, M. (2001). State of the art in shape matching. In M. S. Lew (Ed.), *Principles of Visual Information Retrieval. Advances in Pattern Recognition* (pp. 87–119). Springer.

Wang, J., Bai, X., You, X., Liu, W., & Latecki, L. J. (2012). Shape matching and classification using height functions. *Pattern Recognition Letters*, *33*(2), 134–143.

Weickert, J. (1996). Theoretical foundations of anisotropic diffusion in image processing. In *Theoretical Foundations of Computer Vision* (p. 221-236). Springer Vienna.

Weickert, J. (1999). Coherence-enhancing diffusion filtering. *International Journal of Computer Vision*, *31*, 111–127.

Weiss, Y. (2001). Deriving intrinsic images from image sequences. In *IEEE International Conference on Computer Vision 2001* (Vol. 2, pp. 68–75).

Wexler, Y., Shechtman, E., & Irani, M. (2004). Space-time video completion. In *IEEE Conference on Computer Vision and Pattern Recognition 2004* (Vol. 1, pp. I–I).

Wexler, Y., Shechtman, E., & Irani, M. (2007). Space-time completion of video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(3), 463–476.

Whitaker, R. T. (2000). A level-set approach to image blending. *IEEE Transactions on Image Processing*, *9*(11), 1849–1861.

Wolter, D., & Latecki, L. J. (2004). Shape matching for robot mapping. In *Pacific Rim International Conference on Artificial Intelligence 2004* (pp. 693–702).

Wu, T.-P., & Tang, C.-K. (2005). A bayesian approach for shadow extraction from a single image. In *IEEE International Conference on Computer Vision 2005* (Vol. 1, pp. 480–487).

Wu, T.-P., Tang, C.-K., Brown, M. S., & Shum, H.-Y. (2007). Natural shadow matting. *ACM Transactions on Graphics*, *26*(2), 1–21.

Xiao, C., She, R., Xiao, D., & Ma, K.-L. (2013). Fast shadow removal using adaptive multi-scale illumination transfer. *Computer Graphics Forum*, *32*(8), 207–218.

Xiao, C., Xiao, D., Zhang, L., & Chen, L. (2013). Efficient shadow removal using subregion matching illumination transfer. *Computer Graphics Forum*, *32*(7), 421–430.

Xie, J., Xu, L., & Chen, E. (2012). Image denoising and inpainting with deep neural networks. In *Advances in neural information processing systems* (pp. 341–349).

Xu, C., Liu, J., & Tang, X. (2009). 2d shape matching by contour flexibility. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(1), 180–186.

Xu, M., Zhu, J., Lv, P., Zhou, B., Tappen, M. F., & Ji, R. (2017). Learning-based shadow recognition and removal from monochromatic natural images. *IEEE Transactions on Image Processing*, *26*(12), 5811-5824.

Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., & Li, H. (2017). High-resolution image inpainting using multi-scale neural patch synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017* (pp. 6721–6729).

Yang, M., Kidiyo, K., & Ronsin, J. (2008). Shape matching and object recognition using chord contexts. In *2008 International Conference Visualisation* (pp. 63–69).

Yang, X., Bai, X., Latecki, L. J., & Tu, Z. (2008). Improving shape retrieval by learning graph transduction. In *European Conference on Computer Vision 2008* (pp. 788–801).

Yang, X., Koknar-Tezel, S., & Latecki, L. J. (2009). Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition 2009* (pp. 357–364).

Yeh, R. A., Chen, C., Yian Lim, T., Schwing, A. G., Hasegawa-Johnson, M., & Do, M. N. (2017). Semantic image inpainting with deep generative models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017* (pp. 5485–5493).

You, X., & Tang, Y. Y. (2007). Wavelet-based approach to character skeleton. *IEEE Transactions on Image Processing*, *16*(5), 1220–1231.

Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., & Huang, T. S. (2018). Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018* (pp. 5505–5514).

Yue, H., Yang, J., Sun, X., Wu, F., & Hou, C. (2017). Contrast enhancement based on intrinsic image decomposition. *IEEE Transactions on Image Processing*, *26*(8), 3981-3994.

Zhang, D., & Lu, G. (2001). A comparative study on shape retrieval using fourier descriptors with different shape signatures. In *International Conference on Intelligent Multimedia and Distance Education 2001* (pp. 1–9).

Zhang, L., Zhang, Q., & Xiao, C. (2015). Shadow remover: image shadow removal based on illumination recovering optimization. *IEEE Transactions on Image Processing*, *24*(11), 4623–4636.

Zhu, J., Samuel, K. G., Masood, S. Z., & Tappen, M. F. (2010). Learning to recognize shadows in monochromatic natural images. In *IEEE Conference on Computer Vision and Pattern Recognition 2010* (pp. 223–230).

Appendix A: Glossary

This list contains definitions and brief explanations of terms that are important to understand the contents in this thesis. We provide this list to readers for easy access to the newly defined concepts and professional terms in this thesis.

Coherent Point Drift A point set registration algorithm proposed by Myronenko and Song (2010) that considers the alignment problem of two point sets as a probability density estimation. In this algorithm, one point set represents the Gaussian Mixture Model (GMM) centroid, and the other represents the data points. The algorithm forces the GMM centroids to move coherently as a group, maximizing the likelihood and find the posterior probabilities of centroids.

Directed Chord Pattern (DCP): A novel shape descriptor proposed in this study. It extracts the features of a shape at each sampling point by constructing a series of directed chords in the point neighbor. The feature of the shape at each point is described with the relative position between the point and the directed chords. The relative positions are recorded by the point-to-chord distance and the angle formed by the perpendicular line and the horizontal axis.

Dynamic Patch System (DPS): A system we developed for image completion tasks. The system enables parallel search for multiple size patches and selection of a suitable one with a competitive mechanism. The system enables patch sizes to be changed in the multi-scale image representation to balance the computational workload.

Expectation-Maximization (EM) algorithm: An iterative computation of maximum-likelihood estimates in which the observations can be viewed as incomplete data. Each iteration of the algorithm consists of an Expectation step followed by a Maximization step (Dempster, Laird, & Rubin, 1977). It is commonly used to find maximum likelihood parameters of a statistical model.

Image Completion: An image-editing task. Given an image with a hole, the goal is to generate contents with in the hole according to the remaining parts of the image.

Image Pyramid: A type of multi-scale signal representation in which an image is subjected to repeated smoothing and subsampling. It is a collection of decreasing resolution images arranged in the shape of a pyramid (Gonzalez & Woods, 2006).

Intrinsic Images: A mid-level description of an image proposed by Barrow and Tenenbaum (1978). Given an input image, its intrinsic images are a family of images that contain intrinsic characteristics in registration with the corresponding input image. Each intrinsic image contains, in addition to the value of the characteristic at each point, explicit indications of boundaries due to discontinuities in the value or gradient. Examples of intrinsic images include surface reflectance, distance, surface orientation, incident illumination, and transparency.

Illumination and Reflectance: A kind of intrinsic image decomposition proposed by Barrow and Tenenbaum (1978). Assuming that the image intensity is calibrated to give the reflected flux density at the corresponding scene point, the reflected flux density is the product of the integrated incident illumination and the reflectance (albedo) at a surface element. This model illustrates that an image can be viewed as a product of an illumination image and a reflectance image.

Shape: The external contours of an object. In computer vision tasks, shapes in digital images are restored as pixels and can be considered as finite point sets.

Shape Descriptor: A shape descriptor is a set of numbers calculated to extract the features of a given shape. It attempts to quantify a shape in a manner that is coherent with human perception (Veltkamp & Hagedoorn, 2001).

Shape Matching: The task of shape matching is to match two arbitrary shapes and measure the similarity (metric) between them. When two shapes are represented in finite point sets, the shape matching task is to find a correspondence between two point sets.

Shadow Removal: An image editing task. A shadow is a dark area in an image in which the light from a light source is partially or completely blocked. The task of shadow removal aims

to transform the dark area into a condition in which sufficient light is received.

Visual Completion: The perceptual filling in of parts of objects that are hidden from viewed is called visual completion (Palmer, 1999).