UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Traitement du Signal**

Arrêté ministériel : 7 août 2006

Présentée par

Hao GONG

Thèse dirigée par **Michel DESVIGNES**

préparée au sein du **Laboratoire GIPSA-Lab**
dans **l'École Doctorale EEATS**

Segmentation d'Images Couleurs et Multispectrales de la Peau

Thèse soutenue publiquement le **27 juin 2013**,
devant le jury composé de :

Mme. Marinette REVENU
Professeur, ENSICAEN, Rapporteur
M. Johan DEBAYLE
Maître de Conférences, Ecole des Mines de St. Etienne, Rapporteur
Mme. Barbara ROMANIUK
Maître de Conférences, Université de Reims, Examinateur
M. Pierre Yves COULON
Professeur, Grenoble INP, Examinateur
M. Michel DESVIGNES
Professeur, Grenoble INP, Directeur de thèse



Numéro d'ordre : Année 2013

Institut Polytechnique de Grenoble

Laboratoire Grenoble, Images, Parole, Signal, Automatique (GIPSA-Lab)

École Doctorale d'Électronique, Électrotechnique, Automatique et Traitement du Signal (EEATS)

Thèse de l'Université de Grenoble

Présentée en vue d'obtenir le grade de Docteur,
spécialité Traitement du Signal

par

Hao GONG

Segmentation d'Images Couleurs et Multispectrales de la Peau

Thèse soutenue le 27 juin 2013 devant le jury composé de :

Mme.	Marinette REVENU	Professeur, ENSICAEN	Rapporteur
M.	Johan DEBAYLE	Maître de Conférences, ENSMSE	Rapporteur
Mme.	Barbara ROMANIUK	Maître de Conférences, Université de Reims	Examinateur
M.	Pierre-Yves COULON	Professeur, Grenoble INP	Examinateur
M.	Michel DESVIGNES	Professeur, Grenoble INP	Directeur

Laboratoire Grenoble, Images, Parole, Signal, Automatique (GIPSA-Lab)
UMR 5216 CNRS - Grenoble INP - Grenoble Campus
F-38042 Saint Martin d'Hères cedex - France
Tél: +33 (0)4 76 82 71 07 - Fax: +33 (0)4 76 57 47 90

Acknowledgments

First of all, I would like to express my great gratitudes to my thesis advisors, Pr. Michel Desvignes, for his consistent encouragements and support, professional instructions and patience during the last three and a half years. He proposed to me a very interesting Ph.D. subject which crosses several different exciting research domains such as signal/image processing, pattern recognition and dermatology. He is always ready to answer my questions that were poorly expressed. Also, his insightful suggestions and comments inspired me a lot all along this thesis work.

I am very thankful to Pr. Marinette Revenu and Dr. Johan Debayle for taking their precious time to review my thesis manuscript. I am also very grateful to Dr. Barbara Romaniuk and Pr. Pierre-Yves Coulon for having accepted to be examiners of my thesis defense.

I am grateful to Pr. Gang Feng, without whom, I would not have this precious opportunity to pursuit the Ph.D. degree in France.

I want to thank all the members of the department of signal and image at GIPSA lab. Particularly, my thanks go to Dr. Kai Wang for the discussions with him which helped to improve the quality of my work and to Nicolas, Benjamin, Junshi, and Douglas for the daily conversations as colleagues having shared the same office.

I thank Pr. Xiaohui Yuan, my Master thesis advisor at Southeast University, for having persuaded me to do a Ph.D..

I thank all my friends, with whom, I have the best time of my life in this romantic country.

Finally, I would like to dedicate this thesis to my parents for their unconditional support all along the years.

Abstract

Accurate border delineation of pigmented skin lesion (PSL) images is a vital first step in computer-aided diagnosis (CAD) of melanoma. This thesis presents a novel approach of automatic PSL border detection on color and multispectral skin images. We first introduce the concept of energy minimization by graph cuts in terms of maximum *a posteriori* estimation of a Markov random field (MAP-MRF framework). After a brief state of the art in interactive graph-cut based segmentation methods, we study the influence of parameters of the segmentation algorithm on color images. Under this framework, we propose an energy function based on efficient classifiers (support vector machines and random forests) and a feature vector calculated on a local neighborhood. For the segmentation of melanoma, we estimate the concentration maps of skin chromophores, discriminating indices of melanomas from color and multispectral images, and integrate these features in a vector. Finally, we detail a global framework of automatic segmentation of melanoma, which comprises two main stages: automatic selection of "seeds" useful for graph cuts and the selection of discriminating features. This tool is compared favorably to classic graph-cut based segmentation methods in terms of accuracy and robustness.

Keywords: Image segmentation, Graphs, Melanoma, Lesion delineation, Classification.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Motivations

Melanoma is a malignant tumor of melanocytes. Although less common than basal and squamous cell cancers, it is the most fatal skin cancer which causes a considerable number of deaths especially in fair skinned population worldwide [52]. Since treatment for metastatic melanoma is difficult [64], early diagnosis and prompt excision are critical to improve patient survival rate. Several diagnosis systems of melanoma have been defined, such as the ABCDE system [77], a checklist of five parameters (asymmetry, border, color, dimension and evolving), or the 7-points checklist [10], a scoring approach for different features depending on color, shape and texture. Improvements in diagnostic sensitivity and specificity have been reported with the application of dermoscopy [17]. However, the interpretation of dermoscopic images is time-consuming, subjective thus lack of accuracy and reproducibility. It finally relies on the clinical experience of individual dermatologist. Studies have also shown that dermoscopy may actually lower the diagnostic accuracy in the hands of inexperienced dermatologists [86]. Therefore, there is great need to develop computer-aided diagnosis (CAD) systems.

A standard approach in automatic dermoscopic image analysis typically includes four stages: (a) image acquisition; (b) image segmentation; (c) feature extraction and feature selection and (d) lesion classification. The segmentation stage which delineates the borders of pigmented skin lesions (PSLs) is one of the most importance since the border characteristics provide important information for an accurate diagnosis [32]. For instance, lesion dimension, shape, border irregularity are some of important features calculated on the delineated lesion border. However, the implementation of accurate PSLs delineation remains a challenging task due to several reasons: (a) low contrast

between lesion and surrounding skin; (b) variegated pigmentations inside the lesion; (c) fuzzy and irregular lesion boundary and (d) presence of artifacts like hair, bubble and specular reflection. Our work is motivated by the need to offer a robust and efficient segmentation tool for accurate delineation of dermoscopic PSLs images. This thesis investigates and reviews important aspects of automatic lesion segmentation and proposes a novel approach for automatic delineation of pigmented skin lesions based on dermoscopic and spectroscopic skin images under the Graph-Cut/MAP-MRF framework. The developed software has been tested on various dermoscopic lesion images and both qualitative and quantitative studies indicate the efficiency and reliability of the proposed automatic lesion delineation method.

1.2 General Framework: Graph Cuts

Our image segmentation tools are based on graph cuts, which is a rather recent and rapidly expanding field in computer vision. In a first part, we describe the optimization problems in context of computer vision and clarify the relationship between minimum graph-cut and maximum *a posteriori* estimation in Markov random field (MAP-MRF) in context of optimization problem for image segmentation. From the practical point of view, we objectively compare state-of-the-art interactive graph-cut based image segmentations via qualitative segmentation accuracy assessment on natural color images and highlight the strengths and the weakness of each method. We address the optimization of two fundamental parameters in the graph-cut segmentation framework.

1.3 Combining Classification Techniques and Graph Cuts Segmentation

The second part of the thesis is dedicated to innovative methods combining graph-cut based segmentation and classification techniques. We propose in this chapter the formulation of data term (likelihood energy term) of the binary segmentation energy function defined by posterior classification of a classifier, Support Vector Machine or Random Forest. We show how to combine color feature (RGB tri-channel values) and texture feature (LBP code) within a neighborhood template (shape feature) to enhance the discrimination between foreground and background, where the color information alone is less discriminative.

1.4 Application to Melanoma Detection

We detail in Chapter 4 our proposed approaches in application to skin chromophore extraction and melanoma detection. The first application is the estimation of distributions of skin chromophores. It consists of skin chromophore extraction in different color spaces, skin chromophore quantification on regular color images and multi-spectral images within visible wave range. We reviews in detail several existing methods and describe some new extensions. The second application aims at precisely segmenting dermoscopic PSLs images. We develop an automatic seeding approach to build a sufficient training set for the classifier and propose to add chromophore features to the feature configuration of the classifier. In qualitative and quantitative studies, segmentation results are compared to manual delineations and the proposed approach outperforms the state-of-the-art methods.

1.5 Perspectives

We present in the last chapter some interesting perspectives concerning the improvement of our proposed automatic lesion segmentation approach. We first propose to simplify the automatic seed-region selection stage by localizing the lesion region approximately via automatically generated rectangle. At the feature configuration and classification stage, we show the ability of infrared imaging to reveal chromophore properties of deeper skin layer and propose to add multi-spectral PSLs images at certain wave bands (ranges) into the feature configuration.

Chapter 2

Graph Cut Based Segmentation

Contents

In fields of computer vision, graph cuts have been widely employed to efficiently solve low-level computer vision problems such as image restoration or segmentation, stereo vision and many other associated problems that can be formulated as an energy minimization task of pixel labeling. In corporation with the Markov Random Field (MRF), the Bayesian model provides a probabilistic framework where optimal labeling of image pixels can be estimated by maximizing the posterior probability. This MAP-MRF framework was first introduced by Besag [16] in 1974 and later popularized in computer vision by Geman and Geman [44] in 1984 for binary image restoration. Graph cut was first applied in computer vision by Greig et. al [46] in 1989. They showed that exact MAP estimation of a binary image can be reformulated as solving a *min-cut/max-flow* problem via certain energy functions. The energy (object) function is formulated such that minimization of the energy is equivalent to compute a minimum graph cut. Generally in computer vision, an energy function is the sum of two terms corresponding to "data" and "smoothness" constraints that an acceptable solution should satisfy. The former constraint restricts a preferable solution to have a good explain of the observed data while the latter constraint restricts a solution to be spatially smooth. In fact, only energy minimization of binary labeling problems (e.g. binary image denoising, foreground/background image segmentation) can be estimated exactly via graph-cut framework where a global optimum is guaranteed. For these multi-labeling problems (e.g. multi-camera stereo, grayscale image restoration), strong approximate solution of finding global optima can still be converged through iterative versions of min-cut/max-flow algorithms within this powerful framework. In particular, the Graph-Cut/MAP-MRF framework provides researchers a clear and flexible formulation for image segmentation. It provides a convenient interface to encode local segmentation cues, and a set of powerful computational mechanisms to extract global segmentation.

In this chapter, we provide a general overview of the theoretical background of the thesis and the related work. In Section 2.1, we clarify the concept of optimization problems in context of computer vision and formulate an general energy function framework which is the core and basis of the whole work in this thesis. In Section 2.2, we review the standard definitions and notations for graphs and details some basic theorems as well as fundamental algorithms for minimum cut/maximum flow problem. Section 2.3 introduces the general framework for minimization of an energy function via graph cuts

as well as the optimization problem for image segmentation in the probabilistic Markov random field model. In Section 2.4, we briefly review some state-of-the-art interactive image segmentation approaches, and highlight the strengths and the weakness of each method through real data experiments. Finally, in Section 2.5, we discuss briefly on the selection of two fundamental parameters in the framework of graph cut segmentation in order to yield most preferable and robust segmentation results for later comparison of some state-of-the-art techniques as well as the proposed approaches in this work.

2.1 Labeling Problems as Energy Minimization

Existing low-level computer vision problems, such as image segmentation, stereo matching, image restoration, etc. (cf. Figure 2.1) can be formulated as image labeling problems. These problems aim at assigning a label l_p from a label set $\mathcal{L}$ to each pixel p of the observed image $\mathcal{P}$, where the labels represent some local quantity such as disparity [89]. An example of binary image labeling is shown in Figure 2.2. Such labeling problems are naturally represented in terms of energy minimization, where the energy function E has two terms: one term called "data energy" penalizes solutions that are inconsistent with the observed data, whereas the other term called "smoothness energy" encodes smooth constraints of the spatial coherence:

$$E = E_{data} + \lambda \cdot E_{smooth}. \tag{2.1}$$

where the constant λ controls the relative importance of data and smoothness energy. In particular, the data energy has the form:

$$E_{data} = \sum_{p \in \mathcal{P}} D_p(l_p), \tag{2.2}$$

where $D_p(l_p) : \mathcal{L} \to \mathbb{R}$ measures how much assigning label l_p to a pixel p disagrees with the data. Let $\mathcal{N}$ be the set of all neighboring pairs $\{p, q\}$. $\mathcal{N}$ is called the neighborhood system. Given the neighborhood system $\mathcal{N}$, the smoothness energy has the following form:

$$E_{smooth} = \sum_{\{p,q\} \in \mathcal{N}} V_{\{p,q\}}(l_p, l_q), \tag{2.3}$$

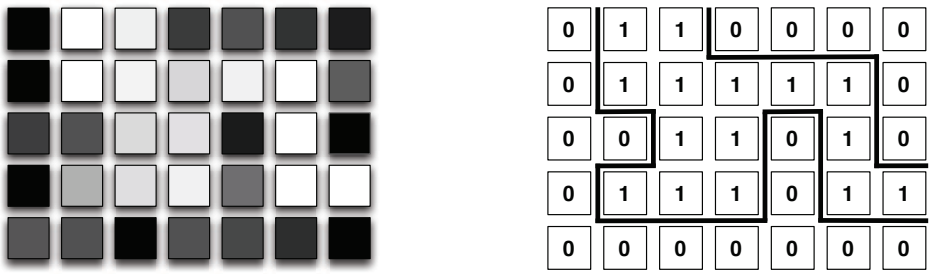
where $V_{\{p,q\}}(l_p, l_q) : \mathcal{L} \times \mathcal{L} \to \mathbb{R}$ is *neighbor interaction* function which encourages spatial coherence by penalizing discontinuities between neighboring pixels. Thus, the smoothness energy E_{smooth} is the sum of neighbor interaction functions for all neighbor pairs. It assigns the labelings which are not smooth a high cost by counting all penalties between neighbor pairs having different labels [94].

(a) Binary image segmentation (b) Stereo matching (c) Image restoration

Figure 2.1: Applications of image labeling problems. (a) Binary image segmentation of "Flower" (Ref. 124084) using classic graph-cut segmentation algorithm [22]; (b) Stereo matching of "Tsukuba" using α-expansion algorithm [59]; (c) Image restoration of "House" using α-β-swap algorithm [26].

Combining Equation (2.1), Equation (2.2) and Equation (2.3) we get the general form of the energy function that has proven to be sufficiently robust and is the most commonly used in computer vision [46]:

$$E(\mathcal{L}) = \sum_{p \in \mathcal{P}} D_p(l_p) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} V_{\{p,q\}}(l_p, l_q). \tag{2.4}$$



(a) A gray image (b) A labeling

Figure 2.2: An example of binary image labeling. A gray image in (a) is a set of pixels $\mathcal{P}$ with observed intensities I_p for each pixel $p \in \mathcal{P}$; A labeling $\mathcal{L}$ in (b) assigns labels $l_p \in \{0, 1\}$ to each pixel $p \in \mathcal{P}$. Such labels can denote object index (image segmentation), depth (stereo matching), original intensity (image restoration), etc. Thick lines show labeling discontinuities between neighboring pixels.

One of the reasons why this general form is so popular is that it can be justified in terms of maximum *a posteriori* estimation (MAP) of a Markov random field (MRF) [16, 44] – more justification on this topic will be provided in Section 2.3.

2.2 Graph Cuts Basics

In this section, we briefly review some standard definitions and notations on graph and fundamental theorems as well as basic algorithms in context of combinatorial optimization in computer vision.

Definition 2.2.1. (Graph). A directed weighted (capacitated) graph $\mathcal{G}$ is an ordered pair $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ is the set of graph nodes (vertices) and $\mathcal{E} \subseteq \{(p,q)|p,q \in \mathcal{V}, p \neq q\}$ is the set of graph edges that connect these nodes. An edge (p,q) of $\mathcal{E}$ is denoted $e_{p,q}$. For each edge $e_{p,q}$ in $\mathcal{G}$, we assign a value $w_{p,q} \in \mathbb{R}^+$ and $w_{p,q}$, named the weight of the edge $e_{p,q}$.

Note that in directed graphs, edge $e_{p,q}$ is distinguished from edge $e_{q,p}$ by its orientation. Directed edge can also be called arc or arrow. In the following we will mainly focus on directed graphs, especially in the presentation of graph flow problems in Section 2.2.1.

Definition 2.2.2. (Cut). A cut is a subset of edges $C \subset \mathcal{E}$ such that two additional distinguished nodes become separated on the induced graph $\mathcal{G}(C) = \langle \mathcal{V}, \mathcal{E} \backslash C \rangle$. The cost (or weight) of a cut is defined as the sum of the weights of the edges that it severs:

$$|C| = \sum_{e_{p,q} \in C} w_{p,q}. \tag{2.5}$$

Definition 2.2.3. (*s*/*t* Cut). An *s*/*t* cut is a special case of graph cut, where the node set $\mathcal{V} = \{s,t\} \cup \mathcal{P}$ contains two special *terminal* nodes, the *source*, s and the *sink*, t, and a set of non-terminal nodes $\mathcal{P}$. Consequently, an *s*/*t* cut is defined as a partition of the nodes in the graph into two disjoint subsets $\mathcal{S}$ and $\mathcal{T}$ meaning — $\mathcal{S} \cup \mathcal{T} = \mathcal{V}$ and $\mathcal{S} \cap \mathcal{T} = \varnothing$ — where $s \in \mathcal{S}$ and $t \in \mathcal{T}$.

Note that in the context of image processing, an edge is called a *t-link* if it connects a non-terminal node in $\mathcal{P}$ with a terminal and an edge is called a *n-link* if it connects two non-terminal nodes. A set of all (directed) n-links is denoted by $\mathcal{N}$. The set of all graph edges $\mathcal{E}$ consists of n-links in $\mathcal{N}$ and t-links $\{(s,p),(p,t)\}$ for non-terminal nodes $p \in \mathcal{P}$. In Figure 2.3 (c) we show a simple example of a graph with the terminals s and t. Here,

each non-terminal nodes (depicted as square) corresponds to a pixel of the 3×3 image in Figure 2.3 (b). Two terminal nodes s (red circle) and t (blue circle) correspond to "foreground" label and "background" label, respectively. Red arrows represent one kind of t-links, (s, p), while blue arrows stand for another kind of t-links, (p, t); n-links (p, q) are depicted as yellow lines. Thickness of arrow or line reflects the edge weight. Figure 2.3 (d) shows an s/t cut (expressed as green dash line), which separates the terminal nodes s and t, resulting a partition of image pixels into "foreground" and "background" (cf. Figure 2.3 (e)).

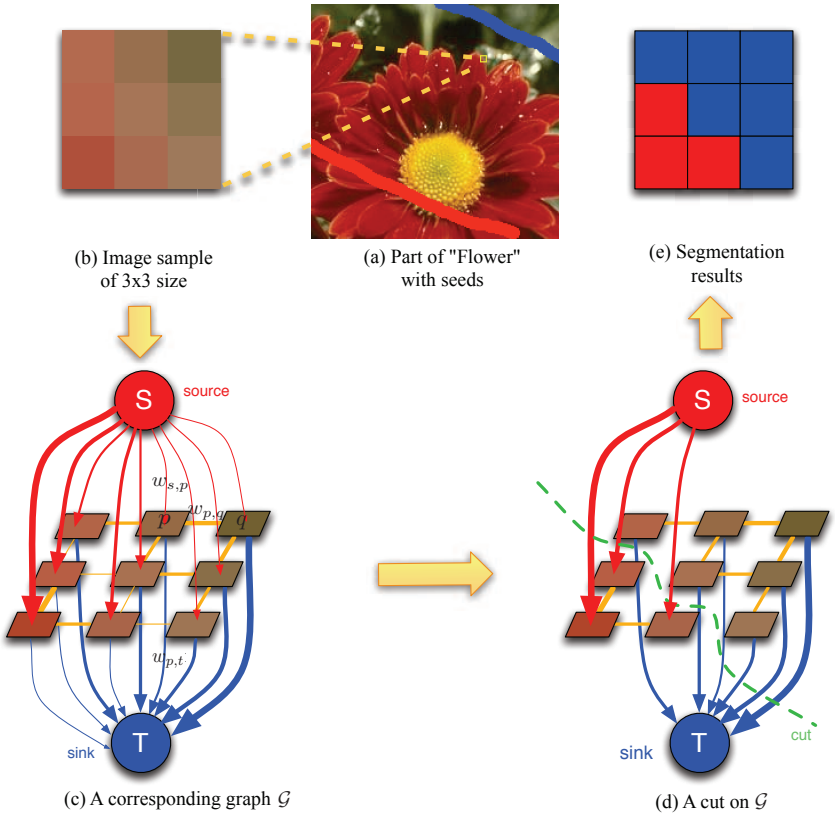


Figure 2.3: A graph-cut based segmentation framework proposed by Boykov and Jolly [22]. (a) Part of "Flower" manually *planted* with seeds (hard constraints). Pixels in red and blue are *a prior* labelings to "foreground" and "background", correspondingly; (b) A sample 3×3 image from (a); (c) Each pixel in (b) corresponds to a non-terminal node in graph. Neighboring nodes (pixels) are interconnected by edges in a regular grid-like fashion (4 neighborhood system). Edge weights are reflected by thickness; (d) *n-links* with lower weight are attractive choices for the minimum cut, which is expressed as green dash line; (e) Global optimal segmentation results can be computed using min-cut/max-flow algorithms [44, 46, 41].

Definition 2.2.4. (*s/t* Minimal Cut). Given a directed weighted graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V} = \{s, t\} \cup \mathcal{P}$. The *s/t* minimal cut problem is to find an *s/t* cut $\hat{C}$ with the smallest cost:

$$\hat{C} = \arg \min_{C \in \mathcal{C}_{\{s,t\}}} |C|. \tag{2.6}$$

where $\mathcal{C}_{\{s,t\}}$ denotes a set of all possible *s/t* cuts.

Note that there are $2^{|\mathcal{V}|-2}$ possible *s/t* cuts in total, making it impossible to find a minimal *s/t* cut in a large graph via enumeration. An alternative solution is to compute the maximum flow in a directed weighted graph according to the *Min-Cut/Max-Flow* theorem [36] given by Theorem 2.2.1 in Section 2.2.1. We will detail the min-cut/max-flow problem in the following section and several polynomial algorithms for this problem will be reviewed briefly. Also notice that we deal with *s/t* cuts solely in this work, thus the *s/t* prefix will be omitted in the following text.

2.2.1 The Min-Cut and Max-Flow Problems

One of the important facts in combinatorial optimization is that the minimum cut problem can be solved by finding a *maximum flow* from the source s to sink t [2, 41]. In graph theory, a graph edge can be interpreted physically as a "pipe" with capacity equal to the edge weight. The *maximum flow* is here depicted intuitively as the maximum "amount of water" that can flow through this edge.

Definition 2.2.5. (Flow). Given a directed weighted graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ with two terminal nodes $\{s, t\} \in \mathcal{V}$, a *flow* in $\mathcal{G}$ is defined as a function $f : \mathcal{E} \to \mathbb{R}_0^+$ assigning each edge $(p, q) \in \mathcal{E}$ a non-negative value $f_{p,q}$ such that:

$$0 \leq f_{p,q} \leq w_{p,q} \quad \forall (p, q) \in \mathcal{E}, \quad (capacity\ constraints) \tag{2.7}$$

$$\sum_{(p,q) \in \mathcal{E}} f_{p,q} = \sum_{(q,p) \in \mathcal{E}} f_{q,p} \quad \forall p \in \mathcal{V} \setminus \{s, t\}. \quad (flow\ conservation\ law) \tag{2.8}$$

Equation (2.7) known as the *capacity constraint* property, indicates that the amount of a flow in an edge cannot be greater than the weight of that edge. Equation (2.8), known as the *flow conservation* property, implies that for each non-terminal node the total amount of incoming and outgoing flow must be the same. The value of a *flow* $|f|$ is defined as the total amount of flow leaving the node s, called the source, or equivalently

the total amount of flow entering the sink t, called the sink:

$$|f| = \sum_{(s,p)\in\mathcal{E}} f_{s,p} - \sum_{(p,s)\in\mathcal{E}} f_{p,s} = \sum_{(p,t)\in\mathcal{E}} f_{p,t} - \sum_{(t,p)\in\mathcal{E}} f_{t,p}. \tag{2.9}$$

We will now focus on the problem of finding a feasible flow of maximum value from the source node s to the sink node t. This is known as a combinatorial optimization problem in which the objective is to maximize f, subject to the constraints given by Equation (2.7) and Equation (2.8). Before we recall some classical algorithms for solving this optimization problem, we first define the *maximum flow problem*.

Definition 2.2.6. (Maximal Flow). Given a directed weighted graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, the maximum flow problem between source node s and sink node t is to find a feasible flow $\hat{f}$ of maximal value:

$$\hat{f} = \arg\max_{f\in\mathbb{F}} \left(\sum_{(s,p)\in\mathcal{E}} f_{s,p} - \sum_{(p,s)\in\mathcal{E}} f_{p,s} = \sum_{(p,t)\in\mathcal{E}} f_{p,t} - \sum_{(t,p)\in\mathcal{E}} f_{t,p} \right). \tag{2.10}$$

where $\mathbb{F}$ is a set of all possible flows in $\mathcal{G}$. Note that a feasible flow of maximum value is not necessarily unique.

To find the maximum flow, Ford and Fulkerson [41] firstly proposed a "augmenting paths" strategy, followed by Edmons-Karp [38], Dinic [37] and Boykov-Kolmogorov [24]. The main idea of this kind of approach is to push flow along non-saturated edges from the source to the sink until the maximum flow in the graph $\mathcal{G}$ is reached. Another kind of popular strategy is so called "push-relabel" method, proposed by Goldberg and Tarjan [45]. Quite different from the "augmenting path" style, they do not maintain a valid flow invariant during the operation, instead, they maintain a labeling of nodes giving a low bound estimate on the distance to the sink along non-saturated edges. During the operation, the flow conservation rule is relaxed and flow is optimistically *pushed* from the source to the sink with positive flow excess allowed at the nodes. Nodes that have a positive flow excess are called *active*. Typically, the *push* operation is applied to *active* nodes based on FIFO selection ("Queue-based Selection Rule") or the largest estimated distance to the sink ("Highest Level Selection Rule"). All these algorithms for min-cut/max-flow problems have polynomial computational complexity (cf. Table 2.1).

Table 2.1: Computational complexity for different min-cut/max-flow algorithms. "Q_PRF" and "H_PRF" denote "Queue-based Selection Rule" and "Highest Level Selection Rule" for *push* operation in "push-relabel" methods.

Algorithms	Computational complexity
Ford-Fulkerson	$\mathcal{O}(\lvert\mathcal{E}\rvert\lvert f\rvert)$
Edmons-Karp	$\mathcal{O}(\lvert\mathcal{V}\rvert\lvert\mathcal{E}\rvert^2)$
Dinic	$\mathcal{O}(\lvert\mathcal{V}\rvert^2\lvert\mathcal{E}\rvert)$
Boykov-Kolmogorov	$\mathcal{O}(\lvert\mathcal{V}\rvert^2\lvert\mathcal{E}\rvert\lvert C\rvert)$
Q_PRF	$\mathcal{O}(\lvert\mathcal{V}\rvert^3)$
H_PRF	$\mathcal{O}(\lvert\mathcal{V}\rvert^2\sqrt{\lvert\mathcal{E}\rvert})$

A well-known theorem proposed by Ford and Fulkerson [41] establishes an equivalence between the maximum flow between two nodes and the minimum cut separating these two nodes.

Theorem 2.2.1. (Min-Cut/Max-Flow Theorem). *For any directed graph the maximal flow value from a source node s to a sink node t is equal to the minimal capacity of all cuts separating s and t.*

Namely, the minimization problem of computing a minimal cut is equivalent to the maximization problem of finding a maximal flow. In fact, the maximum flow value is equal to the cost of the minimum cut (cf. Figure 2.4 (f)). In addition, we intuitively demonstrate that by solving the min-cut/max-flow problem on a graph, we can optimize the energy for image labeling. Consider an example of binary image segmentation in Figure 2.3. The graph corresponds to a 3×3 image. If edge weights are appropriately set based on parameters of an energy like Equation (2.4), a minimum cost cut will correspond to a labeling with the minimum value of this energy.

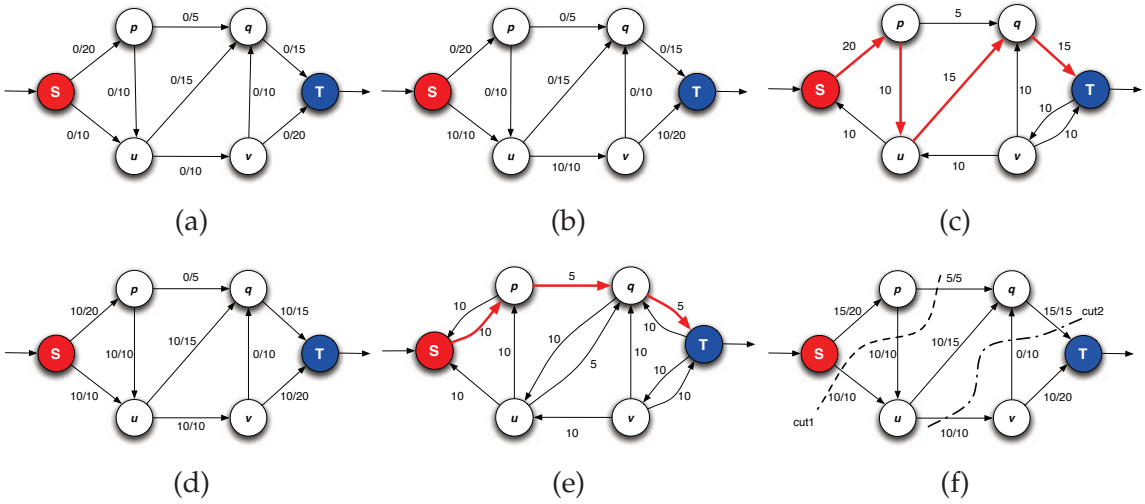


Figure 2.4: Example execution of the augmenting paths strategy proposed by Ford and Fulkerson [41] for min-cut/max-flow computation. (a) Input graph with a zero flow. Each edge is labeled with its flow/weight; (b) Initial graph after pushing 10 flow units along the path (s, u, v, t); (c) Residual graph of (b) with an augmenting path shown in red; (d) Updated graph after pushing 10 flow units along the augmenting path (s, p, u, q, t); (e) Updated residual graph of (d) with a new augmenting path shown in red; (f) Final graph after pushing 5 flow units along the new augmenting path (s, p, q, t). A minimum cut $cut1 = (s, u) \cup (p, u) \cup (p, q)$ while another minimum cut $cut2 = (u, v) \cup (q, t)$.

2.2.2 A Brief Description of Ford & Fulkerson Algorithm

Before the brief description, it is better for us to review the notion of a *residual graph* and *residual capacity*. Given the original graph $\mathcal{G}$ and a flow f, the residual capacity $r_{p,q}$ of an edge $(p, q) \in \mathcal{E}$ is used by algorithm to determine how much flow can pass through the edge (p, q), formally $r_{p,q} = w_{p,q} - f_{p,q}$. An edge is called *saturated* if its residual capacity is zero. A *residual graph* is a directed graph $\mathcal{G}_r$ with the same topology as $\mathcal{G}$ and edge weights reflecting the actual residual capacity with respect to f.

The general idea of the Ford-Fulkerson algorithm can be detailed in pseudo-code as demonstrated in Algorithm 1. An example of the whole process is depicted in Figure 2.4. At the final stage of the augmenting path strategy (cf. Figure 2.4 (f)), there is no way of transmitting more flow through the graph, from source s to sink t, since edges (s, u), (p, u), (p, q) are all saturated, so that the flow shown in Figure 2.4 (f) is the maximum flow. Namely, those edges form a minimum cut $cut1$. The cost of a cut is calculated by sum of the weights of edges that it severs, which is 25 in this example. It equals the amount of flow leaving the source as well as entering the sink. Finally, neither the solution of the minimum cut problem nor the maximum flow problem has to be unique,

several minimum cuts may be possible in the graph, as we can find in Figure 2.4 (f), there are two different forms of minimum cuts sharing one graph.

Algorithm 1 Ford-Fulkerson Algorithm

for each edge $(p, q) \in \mathcal{E}$ **do**
  $f_{p,q} \leftarrow 0$
  $f_{q,p} \leftarrow 0$
end for
while there exits a path P from s to t with no cycles in the residual graph $\mathcal{G}_r$ **do**
  $\Delta \leftarrow \min\{r_{p,q} : (p, q) \in P\}$
  for each edge $(p, q) \in P$ **do**
    $f_{p,q} \leftarrow f_{p,q} + \Delta$
    $f_{q,p} \leftarrow -f_{p,q}$
  end for
end while

2.2.3 A Min-Cut/Max-Flow Algorithm for Vision

All the augmenting path based algorithms reviewed in Section 2.2.1 is based on the breadth-first search (BFS) strategy. In context of graphs in computer vision, operation of BFS strategy can be very expensive thus unpractical since every node (pixel) and every edge in the graph corresponding to the given image will be explored in the worst case. Real-data experiments in computer vision executed by Boykov and Kolmogorov [24] also confirmed that standard augmentation path based algorithms performed poorly due to the rebuilding of a BFS tree on graphs. Subsequently, inspired by Dinic [37], they proposed a new min-cut/max-flow algorithm with a *dynamic tree structure* that proved to be practical for large sparse grids common in computer vision. Specifically, their partially *dynamic* min-cut/max-flow algorithm is a technique with which they could update capacities of certain *t-links* in a graph, and recompute the minimum cut dynamically. They used this method for performing interactive image segmentation (cf. Section 2.4), where the user could improve segmentation results by giving additional segmentation cues (seeds) in an online fashion.

The main idea of the Boykov-Kolmogorov algorithm is illustrated in Figure 2.5. From Figure 2.5, we can clearly observe that it maintains two non-overlapping search trees S and T with roots at the terminal nodes s and t respectively. In tree S all edges from each parent node to its children are non-saturated, while in tree T edges from children to their parents are non-saturated. The nodes that are not in S or T are called *free*. The nodes in the search trees S and T can be either *active* (can *grow* by acquiring new children along non-saturated edges) or *passive*. The algorithm starts by setting all nodes adjacent

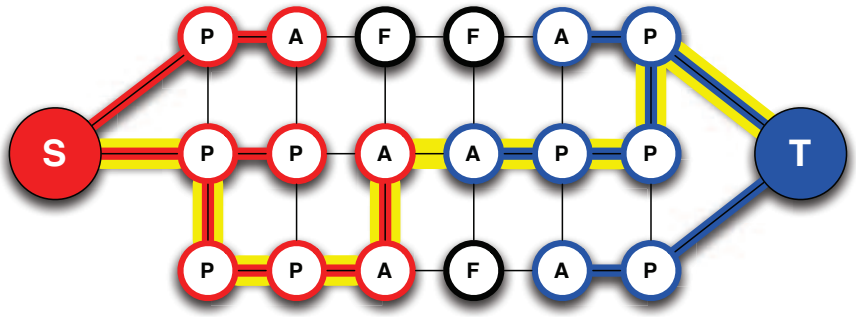


Figure 2.5: A schematic of state-of-the-art min-cut/max-flow algorithm proposed by Boykov and Kolmogorov [24] with the search trees $\mathcal{S}$ (red nodes) and $\mathcal{T}$ (blue nodes) and active (A), passive (P) and free (F) nodes. An augmenting path (bold yellow line) is found when the two dynamic trees touch.

to the terminal nodes as *active*. The three basic stages of the algorithm are as follows:

- "growth" stage: search trees S and T grow until they touch giving an $s \to t$ path
- "augmentation" stage: the found path is augmented, some nodes become "orphans" (edges them to their parents are saturated), search tree(s) break into forest(s),
- "adoption" stage: trees S and T are restored.

Assume there is a directed graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$. As for any augmenting path algorithm, it maintains a flow f and the residual graph G_r (cf. Section 2.2.2). In addition, it keeps the list of all active nodes, A, and all orphans, O. The general framework of the algorithm is presented in Algorithm 2:

Algorithm 2 Boykov-Kolmogorov Algorithm

```
1: S ← s, T ← t, A ← {s, t}, O ← ∅
2: while true do
3:    grow S or T to find an augmenting path P from s to t
4:    if P = ∅ then
5:       terminate
6:    end if
7:    augment on P
8:    adopt O
9: end while
```

Note that Boykov and Kolmogorov's approach does not guarantee a shortest possible path for flow augmentation and thus the strongly polynomial time complexity is lost. In the worst case, time complexity is $\mathcal{O}(|\mathcal{V}|^2|\mathcal{E}||C|)$, which is worse than complexity

of the standard algorithms discussed in Section 2.2.1. However, on typical problem instances in vision, this algorithm significantly outperforms the standard min-cut/max-flow algorithms.

2.3 Energy Minimization via Graph Cuts on MAP-MRF Framework

In this section we first justify the ability of graph cuts to minimize exactly energy functions associated to binary labeling problems in computer vision. Then, the binary image segmentation problem is treated as an optimal configuration of a Markov random field (MRF) which can be solved via maximum *a posteriori estimation* (MAP) according to the Bayesian inference. Some definitions and useful properties of Markov random field will be recalled. Finally, we prove the feasibility of this Graph-Cut/MAP-MRF framework to obtain an optimal solution of binary labeling problem in terms of exact energy minimization.

Given a directed weighted graph $\mathcal{G}(C) = \langle \mathcal{V}, \mathcal{E} \backslash C \rangle$, where $\mathcal{V} = \{s, t\} \cup \mathcal{P}$ and a cut C partitions nodes in this graph into two subsets $\mathcal{S}$ and $\mathcal{T}$. Each non-terminal node p with t-link solely connected to node s (i.e. $p \in \mathcal{S}$) is assigned a label $l_p = 1$ and $l_p = 0$ for node with t-link solely connected to node t (i.e. $p \in \mathcal{T}$). Thus, the cost of cut can be calculated as:

$$|C| = \sum_{p \in \mathcal{V}} w_{s,p} \cdot l_p + \sum_{p \in \mathcal{V}} w_{p,t} \cdot (1 - l_p) + \sum_{(p,q) \in \mathcal{E}} w_{p,q} \cdot (1 - l_q) \cdot l_p. \tag{2.11}$$

Note that we enforce the non-negativity on the value of edge weights since it is possible to compute a minimal cut of a graph if and only if the edge weights of this graph is non-negative.

Rewrite Equation (2.11) by separating constant terms from the terms depending on up to two binary variables, we have the following transformations:

$$|C| = \sum_{p \in \mathcal{V}} w_{p,t} + \sum_{p \in \mathcal{V}} (w_{s,p} - w_{p,t}) \cdot l_p + \sum_{(p,q) \in \mathcal{E}} w_{p,q} \cdot l_p - \sum_{(p,q) \in \mathcal{E}} w_{p,q} \cdot l_p \cdot l_q, \tag{2.12}$$

$$|C| = \sum_{p \in \mathcal{V}} w_{p,t} + \sum_{p \in \mathcal{V}} (w_{s,p} - w_{p,t} + \sum_{q \in \mathcal{V}} w_{p,q}) \cdot l_p - \sum_{(p,q) \in \mathcal{E}} w_{p,q} \cdot l_p \cdot l_q. \tag{2.13}$$

Finally, the weight of cut in this graph can be formulated as a energy function $E(\mathcal{L})$ of binary variable (label) field $l_p \in \mathcal{L} = \{l_1, \cdots, l_n\}$ $(n = |\mathcal{V}|)$:

$$E(\mathcal{L}) = |C| = const + \sum_{p \in \mathcal{V}} \alpha_p \cdot l_p + \sum_{(p,q) \in \mathcal{E}} \beta_{p,q} \cdot l_p \cdot l_q. \tag{2.14}$$

where

$$\begin{cases} const = \sum_{p \in \mathcal{V}} w_{p,t}, \\ \alpha_p = w_{s,p} - w_{p,t} + \sum_{q \in \mathcal{V}} w_{p,q}, \\ \beta_{p,q} = -w_{p,q}. \end{cases} \tag{2.15}$$

Notice that Equation (2.14) differs from Equation (2.4) by a constant term which does not depend on variable l. Thus, we can state that any energy function of a binary labeling problem in computer vision can be minimized with the following characterization:

Theorem 2.3.1. (Exact Energy Minimizations via Graph Cuts). *Any function that can be written as a sum of functions of up to two binary variables $l_p \in \{0, 1\}$ at a time:*

$$E(\mathcal{L}) = const + \sum_{p} E_p(l_p) + \sum_{p,q} E_{p,q}(l_p, l_q) = const + \sum_{p \in \mathcal{V}} \alpha_p \cdot l_p + \sum_{(p,q) \in \mathcal{E}} \beta_{p,q} \cdot l_p \cdot l_q \tag{2.16}$$

can be minimized exactly via graph cuts techniques if and only if $\beta_{p,q} \leq 0$ for all p, q.

A constructive proof of this theorem can easily be given by following the procedure that we described previously to calculate the cost of a cut in the given graph. From the theory of combinatorial optimization, it is always possible to find a minimal cut as long as the edge weights are non-negative. Thus, it is straightforward to verify that the minimization of an energy formulated as in Equation (2.16) can be solved with appropriate edge weight settings. Alternatively, it can be proven using the following theorem described in [42] or [60].

Theorem 2.3.2. (Energy Regularity for Graph-representability). *Any function that can be written as a sum of functions of up to two binary variables $l_p \in \{0, 1\}$ at a time:*

$$E(\mathcal{L}) = \sum_{p} E_p(l_p) + \sum_{p,q} E_{p,q}(l_p, l_q) \tag{2.17}$$

can be graph-representable if and only if each pairwise term satisfies the inequality:

$$E_{p,q}(0,0) + E_{p,q}(1,1) \leq E_{p,q}(0,1) + E_{p,q}(1,0) \quad \forall i, j \tag{2.18}$$

This theorem is very useful and important for providing minimization of an energy

function which fulfills the condition of Theorem 2.3.1 a combinatorial optimization tool with power. Kolmogorov and Zabih [60] and Freedman et al. [42] have extended the theorem in application of approximate minimization of energy functions depending on $n \geq 3$ binary variables.

2.3.1 Markov Random Field

Consider a graph $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, with each node p assigned a label $l_p \in \mathcal{L}$. The set of neighbor nodes of a given node p is denoted as $\mathcal{N}_p$. The neighborhood system $\mathcal{N}$ specifies a configuration called *clique* of the given graph.

Definition 2.3.1. (Clique). A clique in an undirected graph $\mathcal{G}$ is a set of nodes such that for every couple of nodes in the set, there exists an edge between the two nodes.

Consider a field of random variables $L = L_1, \cdots, L_n$ with $n = |\mathcal{V}|$. Each variable L_p takes a value l_p from the label set $\mathcal{L}$, which is called an event $L = l$. Here, $l = (l_p)_{p \in \mathcal{V}}$ denotes the specific configuration of the random field L. Let $\Pr(L = l) = \Pr(l)$ denote the probability that the random field L takes the particular configuration l. Also let $\Pr(L_p = l_p) = \Pr(l_p)$ denote the probability that a node p is assigned the label l_p.

Definition 2.3.2. (Markov Random Fields). L is said to be a Markov random field if and only if:

$$\begin{cases} \Pr(l) > 0, \quad \forall l \in \mathcal{F}, \quad (\textit{positivity}) \\ \Pr(l_p | l_{\mathcal{V} \setminus \{p\}}) = \Pr(l_p | l_{\mathcal{N}_p}). \quad (\textit{markovianity}) \end{cases} \tag{2.19}$$

where $\mathcal{F}$ denotes the set of all possible labelings and $l_{\mathcal{N}_p} = \{l_p | p \in \mathcal{N}_p\}$ denotes all labels of nodes in $\mathcal{N}_p$.

Note that the positivity is assumed for some technical reasons and can usually be satisfied in practice. In case that the positivity condition is satisfied, the joint probability of any random field is uniquely determined by its local conditional probabilities [15]. The Markovianity describes the local characteristics of L, that is, only neighboring labels have direct interactions with each other.

When the probability distribution of a random field satisfies the positivity condition mentioned in Definition 2.3.2, it is also referred to as a *Gibbs random field*. Gibbs random field (GRF) is a well known probabilistic model for the description of particles systems in statistical physics.

Definition 2.3.3. (Gibbs Random Fields). L is said to be a Gibbs random field if and only if the probabilities of its configurations $\Pr(l)$ obey a *Gibbs distribution* with the

following form:

$$\Pr(l) = Z^{-1} \cdot \exp(-U(l)), \tag{2.20}$$

where

$$Z = \sum_{l \in \mathcal{F}} \exp(-U(l)), \tag{2.21}$$

is a normalizing constant called the *partition function*, and $U(l)$ is an energy function. The energy

$$U(l) = \sum_{c \in \mathcal{C}} V_c(l). \tag{2.22}$$

is the sum of *clique potentials* $V_c(l)$ over all possible cliques $\mathcal{C}$ and the value of $V_c(l)$ depends on the local configuration of the clique.

An MRF is depicted by its local property (the Markovianity) whereas a GRF is characterized by its global property (the Gibbs distribution). The Hammersley-Clifford theorem recalled in the following establishes the equivalence of these two types of random field.

Theorem 2.3.3. (Hammersley-Clifford Theorem). *L is a Markov random field on $\mathcal{V}$ with respect to $\mathcal{N}$ if and only if $\Pr(L = l)$ obeys a Gibbs distribution.*

2.3.2 Maximum A Posteriori Estimation

Given a binary labeling problem on image $\mathcal{X} = \{x_p | p \in \mathcal{V}\}$, where x_p denotes the pixel-wise property (e.g. pixel-intensity) and $|\mathcal{V}|$ equals the total number of pixels. The problem can be interpreted as the event $L = l$ where $l = (l_p)_{p \in \mathcal{V}}, l_p \in \{0, 1\}$, is a configuration of this binary field L. Generally, the field L is not directly observable and we have to estimate its configuration l based on the observation x. The most popular way to estimate an MRF is to maximize the posterior probability $\Pr(l|x)$ within the Bayesian inference framework. According to the Bayesian rule, the posterior probability $\Pr(l|x)$ can be written as:

$$\Pr(l|x) = \frac{\Pr(x|l) \cdot \Pr(l)}{\Pr(x)}. \tag{2.23}$$

Since $\Pr(x)$ is constant for a given realization x, Equation 2.23 can be rewritten as:

$$\Pr(l|x) \propto \Pr(x|l) \cdot \Pr(l). \tag{2.24}$$

Thus, the maximum *a posteriori* estimation of l is equivalent to the maximization of

the following energy function:

$$\hat{l} = \arg\max_{l \in \mathcal{F}} \Pr(x|l) \cdot \Pr(l). \tag{2.25}$$

2.3.2.1 Prior Model $\Pr(l)$

We first consider an appropriate model for $\Pr(l)$. Suppose that the labels l are special configurations of a Markov random field L. According to the Hammersley-Clifford Theorem (Theorem 2.3.3), the probability $\Pr(l)$ obeys a Gibbs distribution:

$$\Pr(l) \propto \exp\left(-\sum_{c \in \mathcal{C}} V_c(l) \right), \tag{2.26}$$

In order to specify the Markov random field L, we hereby specify the clique potential function $V_c(l)$ as follows:

$$V_c(l) = V_{\{p,q\}}(l_p, l_q) = U_{p,q} \cdot \delta_{l_p \neq l_q}, \tag{2.27}$$

where $U_{p,q}$ is a function independent of the configuration l and δ is the indicator function: $\delta = 1$ for $l_p \neq l_q$ and $\delta = 0$ othewise.

This specification simplifies the MRF by setting its clique potential function to zero for all cliques of size larger than two (i.e., the clique potentials involve only pairwise interaction). And for the cliques of size two, the potential functions are non-zero only if neighbor nodes are assigned different labels. Combining Equation (2.26) with Equation (2.27), we obtain the prior model $\Pr(l)$ formulated as:

$$\Pr(l) \propto \exp\left(-\sum_{p \in \mathcal{V}} \sum_{q \in \mathcal{N}_p} U_{p,q} \cdot \delta_{l_p \neq l_q} \right). \tag{2.28}$$

2.3.2.2 Likelihood Function $\Pr(x|l)$

We now need an appropriate model for the likelihood function $\Pr(x|l)$. Let x_p be the observed data x at pixel (node) p and assume that the conditional probabilities $\Pr(x_p|l_p)$ are mutually independent. This assumption holds, for example, when noise at each pixel p is also independent:

$$\Pr(x|l) = \prod_{p \in \mathcal{V}} \Pr(x_p|l_p) = \exp\left(-\sum_{p \in \mathcal{V}} -\ln(\Pr(x_p|l_p)) \right). \tag{2.29}$$

2.3.2.3 Energy Function

With both prior model $\Pr(l)$ and likelihood function $\Pr(x|l)$ in Equation (2.25) appropriately defined under the MAP-MRF framework, the binary image labeling problem can finally be estimated in terms of maximization of the following energy function:

$$\hat{l} = \arg\max_{l\in\mathcal{F}} \exp\left(-\sum_{p\in\mathcal{V}} -\ln(\Pr(x_p|l_p)) - \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} U_{p,q}\cdot\delta_{l_p\neq l_q}\right), \tag{2.30}$$

Equivalently, this maximization problem is to minimize the following energy function:

$$\hat{l} = \arg\min_{l\in\mathcal{F}} \left(\sum_{p\in\mathcal{V}} -\ln(\Pr(x_p|l_p)) + \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} U_{p,q}\cdot\delta_{l_p\neq l_q}\right). \tag{2.31}$$

Corollary 2.3.1. (Energy Minimization via Minimal Graph Cut on MAP-MRF Framework). *Consider the binary labeling* $\mathcal{L} = \left\{ l_p | l_p \in \{0,1\} \right\}_{p=1}^{|\mathcal{V}|}$ *on a given observation* $\mathcal{X} = \{x_p | p \in \mathcal{V}\}$*, the energy of this labeling problem*

$$E(\mathcal{L}) = \sum_{p\in\mathcal{V}} -\ln(\Pr(x_p|l_p)) + \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} U_{p,q}\cdot\delta_{l_p\neq l_q}. \tag{2.32}$$

can be minimized exactly via computing the minimal cut of a graph with its edge weights appropriately assigned, if and only if the condition $U_{p,q} \in \mathbb{R}^+$ *is satisfied.*

Proof. Firstly, we can formulate the indicator function δ in context of binary labeling (i.e. $l_p, l_q \in \{0,1\}$):

$$\delta_{l_p\neq l_q} = l_p\cdot(1-l_q). \tag{2.33}$$

Putting Equation (2.33) back to Equation (2.32), we have:

$$E(\mathcal{L}) = \sum_{p\in\mathcal{V}} -\ln(\Pr(x_p|l_p)) + \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} U_{p,q}\cdot l_p\cdot(1-l_q), \tag{2.34}$$

$$E(\mathcal{L}) = \sum_{p\in\mathcal{V}} -\ln(\Pr(x_p|l_p)) + \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} U_{p,q}\cdot l_p + \sum_{p\in\mathcal{V}}\sum_{q\in\mathcal{N}_p} -U_{p,q}\cdot l_p\cdot l_q. \tag{2.35}$$

Then, according to Theorem 2.3.1, any function that can be written as a sum of functions of up to two binary variables $l_p \in \{0,1\}$ at a time can be minimized exactly via graph cuts techniques if and only the coefficient of the pairwise energy term of variable-product is non-positive. Thus, we must have $-U_{p,q} \leq 0$, so that $U_{p,q} \in \mathbb{R}^+$, as required. □

2.4 Graph Cuts in Image Segmentation

Image segmentation is an important problem in early vision. It is aimed at highlighting a more meaningful and simplified representation of an original image by locating objects and boundaries in it. Basically, the task of image segmentation can be treated as the process of discrete pixel labeling of an image such that pixels assigned the same label are partitioned into the one "segment" with respect to certain visual properties (e.g. pixel intensity, color, texture, etc.). Various existing approaches for image segmentation, for example, thresholding method, clustering method, region growing or split-and-merge, etc., have been popularized for years due to their simplicity and speed. However, none of these methods promise a robust performance in practice since they do not incorporate any clear form of energy (object/cost) function. A more robust way of segmentation can be formulated as solving an optimization problem of certain energy function. One group of these energy based approaches, for instance, snakes [55], geodesic active contour [28] and level-set based approaches, rely on the continuous energy function via variational optimization techniques. In application to real data segmentation task, however, they are limited due to the local optima property.

Inspired by the successful introduction of graph cut theory from the field of combinatorial optimization to computer vision problem like image restoration [26, 25] and stereo [19, 50, 57], Boykov and Jolly [23] first proposed an interactive object segmentation approach based on optimization of discrete energy function (e.g. energy function defined on the binary label set) via computing a minimal graph cut. The main contribution presented in their work is the appropriate definition of energy function of binary labeling problem under MAP-MRF framework (cf. Section 2.3.1 and Section 2.3.2 for the detail of this framework) as well as the user-interaction interface, namely, the ability to incorporate with manually/automatically set hard constraints or seeds. Despite its simplicity in both energy function formulation and user-manipulation, this graph-cut based interactive segmentation framework epitomizes the best features of combinatorial best features of graph cuts and combinatorial optimization: global optima, practical efficiency, numerical robustness, ability to integrate multiple visual cues and constraints, unrestricted topological properties of segments, and applicability to N-D problems [23]. Throughout this thesis, we focus on binary image segmentation (i.e. foreground/background segmentation) by formulating it as a binary discrete labeling problem and optimize the labeling energy in graph-cut based framework.

In this section, we give a brief review of the interactive graph-cut based segmentation framework proposed by Boykov and Jolly, as well as two extensions that build directly

on this framework: one is called "GrabCut", which interprets regional cues based on Gaussian mixture model (GMM) and adopts an iterative scheme for optimization allowing significant reduction of user interaction; another is named "Lazy Snapping", which interprets regional cues via a clustering technique, the k-MEANS algorithm and speeds up the original segmentation framework based on watershed pre-segmentation. Note that both "GrabCut" and "Lazy Snapping" introduce their novelties by reformulating the data term (likelihood energy) of the energy function to be optimized. We will also adopt this methodology to enhance the state-of-the-art graph cuts segmentation techniques in the following section. At the end of this section, we evaluate all these approaches qualitatively yet objectively through comparative experiments on several natural color images.

2.4.1 Interactive Graph-Cut Based Segmentation Framework

We now address the graph-cut based segmentation framework first outlined by Boykov and Jolly [23]. Given an input image $\mathcal{X} = \{\mathbf{x}_p | p \in \mathcal{V}\}_{p=1}^{|\mathcal{V}|}$, where $\mathbf{x}_p$ denotes the visual cues of a pixel p and $x_p \in \mathbb{R}^+$ for grayscale image while $\mathbf{x}_p \in \mathbb{R}_+^3$ for color image, segmentation of one object from its background in this image can be formulated as a binary labeling problem (cf. Section 2.1). This problem is aimed at assigning each pixel p a unique label $l_p \in \{0, 1\}$ from the label set $\mathcal{L}$, where $l_p = 0$ if p is a "background" pixel and $l_p = 1$ is a "foreground" or "object" pixel. By imposing soft constraints on both "region" and "boundary" properties of the segmentation (the complete labeling "event"), Boykov and Jolly proposed an appropriate definition of energy function introduced in Equation (2.4), where the data term interprets the region property while the smoothness term reveals the boundary property. In addition, by imposing hard constraints via user-painted strokes, a subset of pixels can be pre-labeled either as "foreground" or "background". Subsequently, the reformulated energy function of binary labeling problem can be minimized exactly via graph cuts satisfying both soft and hard constraints.

2.4.1.1 Data Energy

Motivated by the MAP-MRF formulation described in Section 2.3, the data terms in Equation (2.2) were chosen as negative log-likelihoods of the foreground and background gray-level distributions, namely histograms of intensities of pre-labeled pixels:

$$D_p(l_p) = -\log \Pr(x_p | l_p) = -\log h(x_p; l_p) \qquad p \in \mathcal{V}, l_p \in \mathcal{L} \tag{2.36}$$

where $\Pr(x_p|l_p)$ denotes the conditional probability of intensity x_p assigned the label l_p based on the observed data. The value of the conditional probability is estimated from two histograms of intensities of user-labeled pixels $h(x_p; l_p \in \{0,1\})$, one for background and one for foreground. The data energy evaluates how well the gray-level distribution fits into a given intensity model (e.g. histogram) based on the observed data. High-level penalty is given when the label assigned to each pixel is inconsistent with the observed data, and low-level penalty for the preferable labeling, otherwise.

2.4.1.2 Smoothness Energy

Further, the neighbor interaction functions $V_{\{p,q\}}(l_p, l_q)$ in the smoothness term in Equation (2.3) were chosen as:

$$V_{\{p,q\}}(l_p, l_q) = \begin{cases} B_{p.q}, & \text{if } l_p \neq l_q \\ 0, & \text{if } l_p = l_q \end{cases} \tag{2.37}$$

Considering that the smoothness term encourages spatial coherence in region of similar gray level, the coefficient $B_{p,q} \geq 0$ should be interpreted as the penalty for discontinuity between neighboring pixels p and q. Sufficiently, we can set penalty $B_{p,q}$ as a function:

$$B_{p,q} \propto \exp\left(-\frac{(x_p - x_q)^2}{2\sigma^2}\right) \cdot \frac{1}{dist(p,q)} \tag{2.38}$$

where σ is a parameter adjusts the sensitivity of intensity difference between neighboring pixels and $dist(\cdot)$ denotes the Euclidean distance of neighboring pixels.

Combining these two terms together, we have the general energy function $E(\mathcal{L})$ of the binary image labeling problem under the interactive graph-cut segmentation framework:

$$E(\mathcal{L}) = -\sum_{p \in \mathcal{V}} \log h(x_p; l_p) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} \exp\left(-\frac{(x_p - x_q)^2}{2\sigma^2}\right) \cdot \frac{1}{dist(p,q)} \cdot \delta_{l_p \neq l_q} \tag{2.39}$$

where neighborhood system $\mathcal{N}$ contain all unordered pairs $\{p,q\}$ of neighboring pixels. In practice, $\mathcal{N}$ is set as a 8-way connectivity neighborhood system in which neighboring pixels are adjacent either horizontally/vertically or diagonally.

It can be seen from Equation (2.39) that the energy function is a linear combination of the data term and the pairwise term so both factors have to be taken into considerations, with coefficient λ balancing both terms to result an optimal solution. Note that the form of Equation (2.39) and the coefficient $B_{p,q} > 0$ in the smoothness term satisfy

the condition in Corollary 2.3.1, a global optimal solution for this energy function like Equation (2.39) can be found by computing the minimal cut on a graph with appropriate settings of edge weights. An efficient min-cut/max-flow algorithm proposed by Boykov and Kolmogorov [24] (cf. Section 2.2.3) was adopted to accelerate the optimization.

In conclusion, this graph-cut based framework is well suited to image segmentation. As illustrated in Figure 2.3 (d) and (e), a cut separates a given image into two regions, one region including the terminal nodes $\mathcal{S}$ as "foreground labels" and the other including the terminal nodes $\mathcal{T}$ as "background labels", respectively. Red and blue brushes (known as "seeds" or "hard constraints") shown in Figure 2.3 (a) denote the *a priori* labelings of selected pixels as "foreground" and "background" via user interaction. Assume that $\mathcal{O}$ and $\mathcal{B}$ denote the subsets of pixels *a prior* to be a part of "foreground" (pixels in red) and "background" (pixels in blue), corresponding. By satisfying these hard constraints, a minimum cut on a graph can be found with appropriate edge weight settings to generate an optimal segmentation. The edge weights of the graph are given in Table 3.2. In addition, the user-planted seeds are also used to guess the foreground and background intensity models required for the data term.

Table 2.2: Edge weights for maximum *a posteriori* estimation in Boykov-Jolly approach [23].

Edge	Weight	For
$w_{s,p}$	$-\log h(\mathbf{x}_p;0)$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	$+\infty$	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
$w_{p,t}$	$-\log h(\mathbf{x}_p;1)$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	0	$p \in \mathcal{O}$
	$+\infty$	$p \in \mathcal{B}$
$w_{p,q}$	$\lambda \cdot \exp\left(-\frac{(\mathbf{x}_p-\mathbf{x}_q)^2}{2\sigma^2}\right) \cdot \frac{\delta_{l_p \neq l_q}}{dist(p,q)}$	$\{p,q\} \in \mathcal{N}$

2.4.2 Lazy Snapping

An extension of Boykov-Jolly's graph-cut based segmentation approach was proposed by Li et al. [65] in 2004. In this work, they presented an interactive image cutout which makes full advantages of the interactive graph-cut based segmentation framework reviewed in Section 2.4.1: simple and coarse user-specification (also called "seed" or "hard constraint" as mentioned in Section 2.4.1) of "foreground" region or "background" area allows optimal estimation of the segmentation satisfying both soft and hard constraints. In addition, the authors improved the algorithm efficiency for instant feedback of segmentation results via a modified graph-cut segmentation framework on

pre-segmentation. Instead of building a graph on pixel-level of a given image, they corresponded each node of a graph to a pre-segmented region from the watershed segmentation. Since the watershed algorithm they chose declares good characteristics in locating object boundaries whereas preserving small differences inside each small region, the pre-segmentation produces reasonable approximative representation of the given image and improves the speed significantly. We will detail their work in the following text with respect to the augmentation in the Boykov-Jolly's approach.

2.4.2.1 Data Energy

Given an color image $\mathcal{X} = \{\mathbf{x}_p | p \in \mathcal{V}\}_{p=1}^{|\mathcal{V}|}$, where $\mathbf{x}_p = [R_p, G_p, B_p]^\top$ denotes a 3-dimensional vector of RGB intensities of a pixel p. As the increase in dimensions of visual properties makes it difficult to construct reliable histograms in RGB color space, Li et al. [65] proposed to model the foreground/background color distributions by clustering the RGB intensities of user-seeded pixels via the k-Means algorithm [67]. After the same hard constraints assignment as in the Boykov-Jolly segmentation framework, k-Means is employed to cluster colors in foreground seeds $\mathcal{O}$ and background seeds $\mathcal{B}$. The resulting mean colors corresponding to each foreground cluster $C_i^{\mathcal{O}}$ and background cluster $C_j^{\mathcal{B}}$ are denoted as $\mathbf{m}_i^{\mathcal{O}}$ and $\mathbf{m}_j^{\mathcal{B}}$. The k-Means is initialized to have 64 clusters for both foreground and background colors, namely, $\lceil i \rceil = \lceil j \rceil = 64$ in their experiments. For each pixel p, by computing the minimal distance from its color $\mathbf{x}_p$ to foreground clusters as $d_p^{\mathcal{O}} = \min_i ||\mathbf{x}_p - \mathbf{m}_i^{\mathcal{O}}||$, and similarly, $d_p^{\mathcal{B}} = \min_j ||\mathbf{x}_p - \mathbf{m}_j^{\mathcal{B}}||$, they define the data term in Equation (2.2) as:

$$D_p(l_p) = \frac{d_p^{l_p}}{d_p^1 + d_p^0}, \tag{2.40}$$

where $l_p = 1$ for $p \in \mathcal{O}$ and $l_p = 0$ for $p \in \mathcal{B}$. This definition encourages the pixels to be assigned the label with similar colors to foreground or background.

2.4.2.2 Smoothness Energy

Since the smoothness term represents the energy due to the gradient along the object boundary, they defined the neighbor interaction functions $V_{\{p,q\}}(l_p, l_q)$ in the smoothness term in Equation (2.3) as a function the color gradient between pixels p and q:

$$V_{\{p,q\}}(l_p, l_q) = \frac{|l_p - l_q|}{||\mathbf{x}_p - \mathbf{x}_q||^2 + 1}, \tag{2.41}$$

This definition allows the interaction functions $V_{\{p,q\}}(l_p, l_q)$ to give high penalty when two pixels assigned different labels are similar in color, that is, two pixels with similar color are less likely on the segmentation boundary.

Putting Equation (2.40) and Equation (2.41) to Equation (2.4), the complete energy function is defined as:

$$E(\mathcal{L}) = \sum_{p \in \mathcal{V}} \frac{d_p^{l_p}}{d_p^1 + d_p^0} + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} \frac{|l_p - l_q|}{||\mathbf{x}_p - \mathbf{x}_q||^2 + 1}. \tag{2.42}$$

Note that the form of Equation (2.42) and the pairwise term $V_{\{p,q\}}(l_p, l_q)$ satisfy the condition in Theorem 2.3.2, this energy function is graph-representable and can be minimized exactly using min-cut/max-flow algorithm proposed by Boykov and Kolmogorov [24]. The settings of edge weights are given in Table 2.3.

Table 2.3: Edges weights for maximum *a posteriori* estimation in *Lazy Snapping* [65].

Edge	Weight	For
$w_{s,p}$	$\frac{d_p^0}{d_p^1 + d_p^0}$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	$+\infty$	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
$w_{p,t}$	$\frac{d_p^1}{d_p^1 + d_p^0}$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	0	$p \in \mathcal{O}$
	$+\infty$	$p \in \mathcal{B}$
$w_{p,q}$	$\lambda \cdot \frac{\lvert l_p - l_q \rvert}{\lvert\lvert \mathbf{x}_p - \mathbf{x}_q \rvert\rvert^2 + 1}$	$\{p,q\} \in \mathcal{N}$

2.4.3 GrabCut

Another extension of Boykov-Jolly's graph-cut based segmentation approach, named *GrabCut*, was introduced by Rother et al. [78] in 2004. It is aimed at achieving high segmentation performance at the cost of only modest user interaction. The novelty of GrabCut in terms of the formation of energy minimization on the basis of segmentation approach of Boykov and Jolly lies mainly in three aspects as following:

- Color data modeling

  Since it is impractical to model foreground/background color distribution in separated gray-level histogram, an GMM (Gaussian Mixture Model) color model was adopted to define the data term in the energy function (cf. Equation (2.2)). Due to the fact that GMM is a parametric alternative to the non-parametric histogram, it is more flexible and precise in modeling the underlying data to a multimodal

density distribution. Such flexibility and precision allow qualitative performance of color-feature based image segmentation. For a brief view of color data modeling in GrabCut approach, suppose that the pixel value of an input color image in RGB color space or other 3-dimensional color space is denoted by $\mathbf{x}_p \in \mathbb{R}^3$, with the corresponding label $l_p \in \{\text{foreground}(=1), \text{background}(=0)\}$. Given two full-covariance GMMs as descriptions of foreground and background color distribution, each GMM consists of K components (typically $K = 5$), with mean $\boldsymbol{\mu}(\cdot) \in \mathbb{R}^3$ and covariance $\boldsymbol{\Sigma}(\cdot) \in \mathbb{R}^{3\times 3}$ assigning to each pixel p a unique component, $k_p \in \{1, \cdots, K\}$. Now the data term is defined as:

$$\begin{aligned} D_p(l_p, k_p) &= -\log \Pr(\mathbf{x}_p | l_p, k_p) = -\log \pi_p(l_p, k_p) \mathcal{N}(\mathbf{x}_p; \boldsymbol{\mu}(l_p, k_p), \boldsymbol{\Sigma}(l_p, k_p)) \\ &= -\log \pi_p(l_p, k_p) + \frac{1}{2} \log \det \boldsymbol{\Sigma}(l_p, k_p) + \frac{1}{2} [\mathbf{x}_p - \boldsymbol{\mu}(l_p, k_p)]^\top \boldsymbol{\Sigma}(l_p, k_p)^{-1} [\mathbf{x}_p - \boldsymbol{\mu}(l_p, k_p)], \end{aligned} \tag{2.43}$$

where the mixing weights denoted by $\pi(\cdot)$ are non-negative and add up to one.

For smoothness term in the energy function (cf. Equation 2.3), its definition remains unchanged from Equation (2.37) and Equation (2.38):

$$V_{\{p,q\}}(l_p, l_q) = \exp\left(-\frac{||\mathbf{x}_p - \mathbf{x}_q||^2}{2\langle (\mathbf{x}_p - \mathbf{x}_q)^2 \rangle} \right) \cdot \frac{1}{dist(p,q)} \cdot \delta_{l_p \neq l_q}. \tag{2.44}$$

- Iterative energy minimization

  The procedure of iterative energy optimization is aimed at refinement of a binary labeling $\mathcal{L} = \{l_1, l_2, \cdots, l_p | l_p \in \{0, 1\}\}$ for each undefined pixel, which initially locates inside of the user-specified rectangles, marked with green in Figure 2.8 and Figure 2.9. Firstly, initialization is done by labeling the pixels outside of the rectangle (the region outside of the rectangle is denoted as $\mathcal{B}$) with $l = 0$, while the undefined pixels are labeled with $l = 1$. Each undefined pixel is then assigned a GMM components k by simple enumeration and parameters of the color GMM model, $\hat{\pi}(l, k)$, $\hat{\boldsymbol{\mu}}(l, k)$ and $\hat{\boldsymbol{\Sigma}}(l, k)$, are learnt subsequently. Finally, optimal segmentation (updated binary labeling) is obtained by computing the minimum cut on a graph with two additional terminal nodes, foreground s and background t. The edge weights of the graph are given in Table 2.4. The updated binary labeling is adapted to optimize the color GMM parameters in the successive iteration. This whole iteration scheme terminates automatically when the energy function $E(\mathcal{L})$ stops to decrease significantly and converges at least to a local minima.

- Incomplete user interaction

  Due to its iterative optimization scheme, *GrabCut* considerably simplifies the user interaction for algorithm initialization. Instead of providing manual hard labeling to explicitly indicate some pixels must be either part of the "foreground" or "background" (via user-imposed strokes in both Boykov-Jolly and *Lazy Snapping* frameworks), *GrabCut* requires only specification of "background" pixels via roughly enclosure of the intended "foreground" pixels with a rectangle. The incompleteness of prior knowledge on "foreground" specification can be compensated through the iterative minimization allowing tentative labeling of some pixels in the foreground.

Table 2.4: Edges weights for maximum *a posteriori* estimation in *GrabCut* [78].

Edge	Weight	For
$w_{s,p}$	$-\log \hat{\pi}_p(0,k)\mathcal{N}(\mathbf{x}_p;\hat{\boldsymbol{\mu}}(0,k),\hat{\boldsymbol{\Sigma}}(0,k))$	$p \notin \mathcal{B}$
	0	$p \in \mathcal{B}$
$w_{p,t}$	$-\log \hat{\pi}_p(1,k)\mathcal{N}(\mathbf{x}_p;\hat{\boldsymbol{\mu}}(1,k),\hat{\boldsymbol{\Sigma}}(1,k))$	$p \notin \mathcal{B}$
	$+\infty$	$p \in \mathcal{B}$
$w_{p,q}$	$\lambda \cdot \exp\left(-\frac{\|\|\mathbf{x}_p-\mathbf{x}_q\|\|^2}{2\left\langle(\mathbf{x}_p-\mathbf{x}_q)^2\right\rangle}\right)\cdot\frac{\delta_{l_p\neq l_q}}{dist(p,q)}$	$\{p,q\}\in\mathcal{N}$

2.4.4 Application to Natural Color Image Segmentation

We present in this section some segmentation results obtained by previously reviewed approaches, Boykov-Jolly' s approach, *Lazy Snapping* method and *GrabCut* method on natural color images from the Berkeley Segmentation Dataset [35]. We divided the experimental objects into two groups: 10 images for training and 20 images for testing. Each of the image has a size of 481×321 pixels. Several examples of segmentation results are presented from Figure 2.6 to Figure 2.9. To evaluate the performance objectively, we initialize each method appropriately with user-interactions and parameter settings performed according to the principle of equity.

2.4.4.1 Experiment Configuration

- User Interaction

  Since approaches developed based on the interactive graph-cut segmentation framework are very sensitive to the spatial distribution of user-input hard constraints (seeds), it is fair to initialize Boykov-Jolly's approach and Lazy Snapping with exactly the same user interaction (user-input seeds), and the degree of user interaction (amount of seeds) is limited to a moderate level. For GrabCut, the rectangle

imposed manually encloses the object as tightly as possible so that it can include as few background seed as possible in the same image.

- Parameter Setting

  The universal parameter λ for all three approaches is selected as 20 by optimizing the performance against ground truth in the training set of 10 images. Parameter σ of Boykov-Jolly approach is defined as $\sigma = \sqrt{\left\langle (\mathbf{x}_p - \mathbf{x}_q)^2 \right\rangle}$, according to Equation (2.44). Other individual parameters like the number of clusters and the number of GMM components remain the same as described in the related literatures (see Section 2.4.2 and Section 2.4.3). Particularly for GrabCut, we execute the algorithm subject to different number of iterations (number of iteration $= 1, 2, 40$). We terminate the iterative segmentation procedure manually after 40 iteration, and the corresponding results are used as final segmentations for the later comparison.

Note neither pre-processing nor post-processing (e.g. watershed pre-segmentation for Lazy Snapping and border matting for GrabCut) is involved during the experiments, we simply focus on the graph-cut based segmentation mechanism.

2.4.4.2 Results and Discussions

Figure 2.6 and Figure 2.7 present comparatively the results obtained by Boykov-Jolly method and Lazy Snapping. We can observe that Lazy Snapping outperforms Boykov-Jolly method in extracting more accurate object region from the background where the contrast is at a high level. For example, arms of the "Starfish", hind leg of the "Kangaroo", shoes of the "Woman" and the upper left corner of the "Boat" are all mis-segmented as background by Boykov-Jolly method whereas Lazy Snapping performs better segmentations on these regions because it utilizes the clustering method to build a more reasonable model for the color distributions of foreground and background regions. However, the color distribution model that Lazy Snapping adopts is not so "reliable" as mentioned in their literature: performance of Lazy Snapping decreases with the increase of overlaps of foreground and background distributions in color space, especially in the regions of low contrast at the transition from foreground to background. For instance, wall close to the right leg of "Woman" and precipice behind the "Sheep" are mis-segmented as foreground while claws of "Birds", nose of "Plane" are mis-segmented as background. Since the performance of k-Means algorithm that Lazy Snapping uses to cluster colors of foreground/background seeds highly relies on the choice of cluster number and initialization of cluster means, it can be converged to local minimum which results ar-

bitrarily bad clustering for many natural examples due to inappropriate starting means assigned randomly. This weakness can further explain unsatisfied segmentations of images where foreground/background color profiles are quite similar (e.g. "Sheep"). One solution to enhance the performance of Lazy Snapping is to increase the amount of accurate seeds, another is to use an more robust version of *k*-Means, named *k*-Means++ [12], which is guaranteed to find a solution competitive to the optimal clustering. We will detail *k*-Means and *k*-Means++ in Section 4.5.2 and employ *k*-Means++ to enhance Lazy Snapping for later experimental use.

Figure 2.8 and Figure 2.9 demonstrate the results obtain by GrabCut subject to different number of iterations. It can be seen that GrabCut is efficient in the iterative optimization scheme and can converge at least a local minima, often only after 2 iterations. Similar to Lazy Snapping, GrabCut can generate comparative better segmentations than Boykov-Jolly' approach in some simple or moderately difficult images (e.g. "Starfish", "Japanese" and "Boat"). Unfortunately, it fails in the following cases: (i) foreground and background share partial color profiles (e.g. "Kangaroo", "Birds", "Plane"); (ii) color distributions of foreground/background overlap largely in color space (e.g. "Sheep"). Iterative optimization of the data energy defined by GMM color model converges at local minima far from the global optimum. For the former case, we can improve the performance by replacing the user-dragged rectangle with a lasso to increase the accuracy of background color sampling. For the latter case, however, neither accurate user interaction nor precise color model can be useful to achieve a promising segmentation.

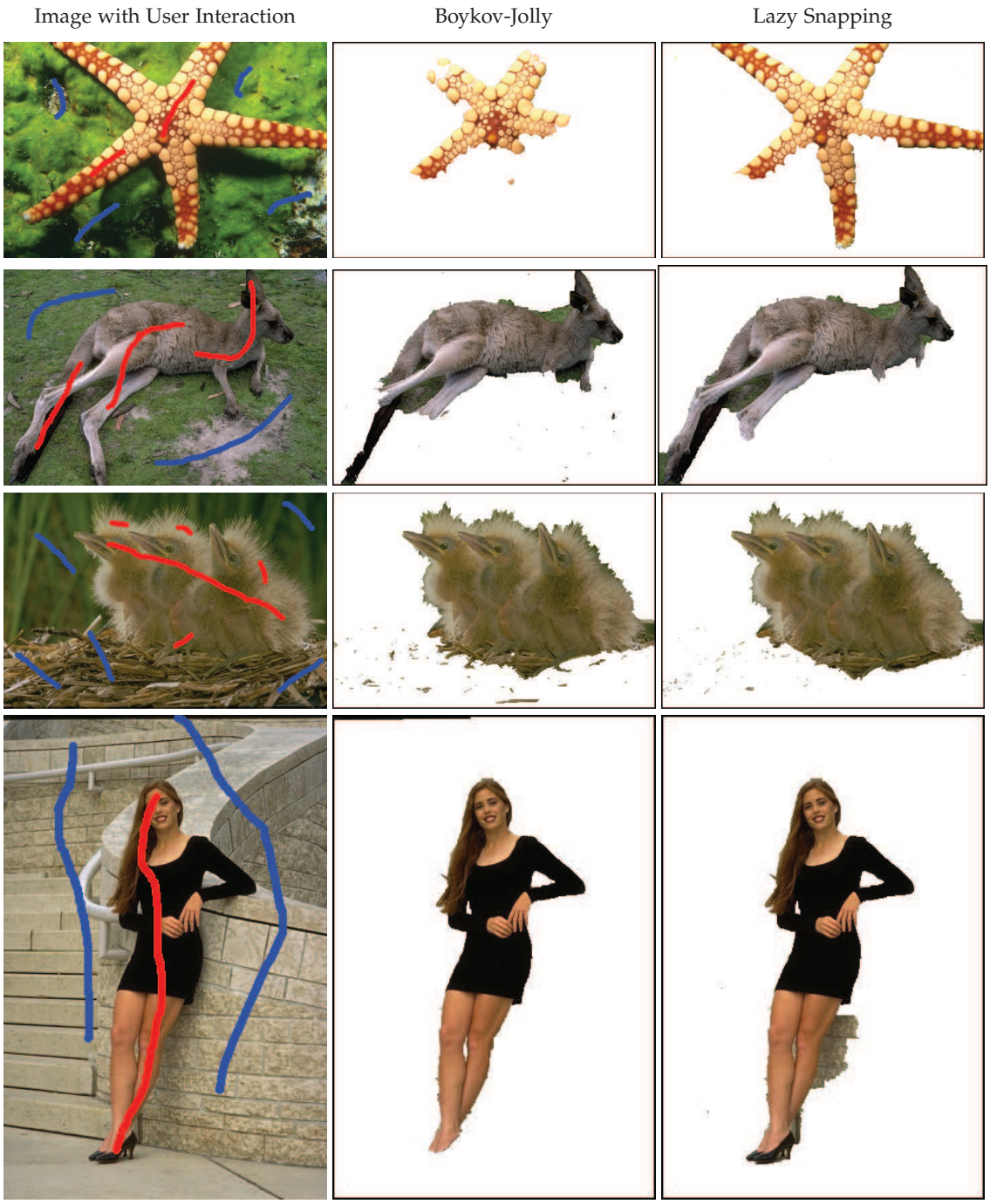


Figure 2.6: Segmentations results by Boykov-Jolly method [23] and *Lazy Snapping* [65] on natural color images from Berkeley Segmentation Dataset [35]. First column: original color images include "Starfish" (Ref. 12003), "Kangaroo" (Ref. 69020), "Birds" (Ref. 163085) and "Lady" (Ref. 388016). Second column: segmentations obtained by Boykov-Jolly method. Last column: segmentations obtained by *Lazy Snapping*.

Figure 2.7: Segmentations results by Boykov-Jolly method [23] and *Lazy Snapping* [65] on natural color images from Berkeley Segmentation Dataset [35]. First column: original color images include "Plane" (Ref. 37073), "Japanese" (Ref. 65019), "Sheep" (Ref. 41025) and "Boat" (Ref. 92059). Second column: segmentations obtained by Boykov-Jolly method. Last column: segmentations obtained by *Lazy Snapping*.

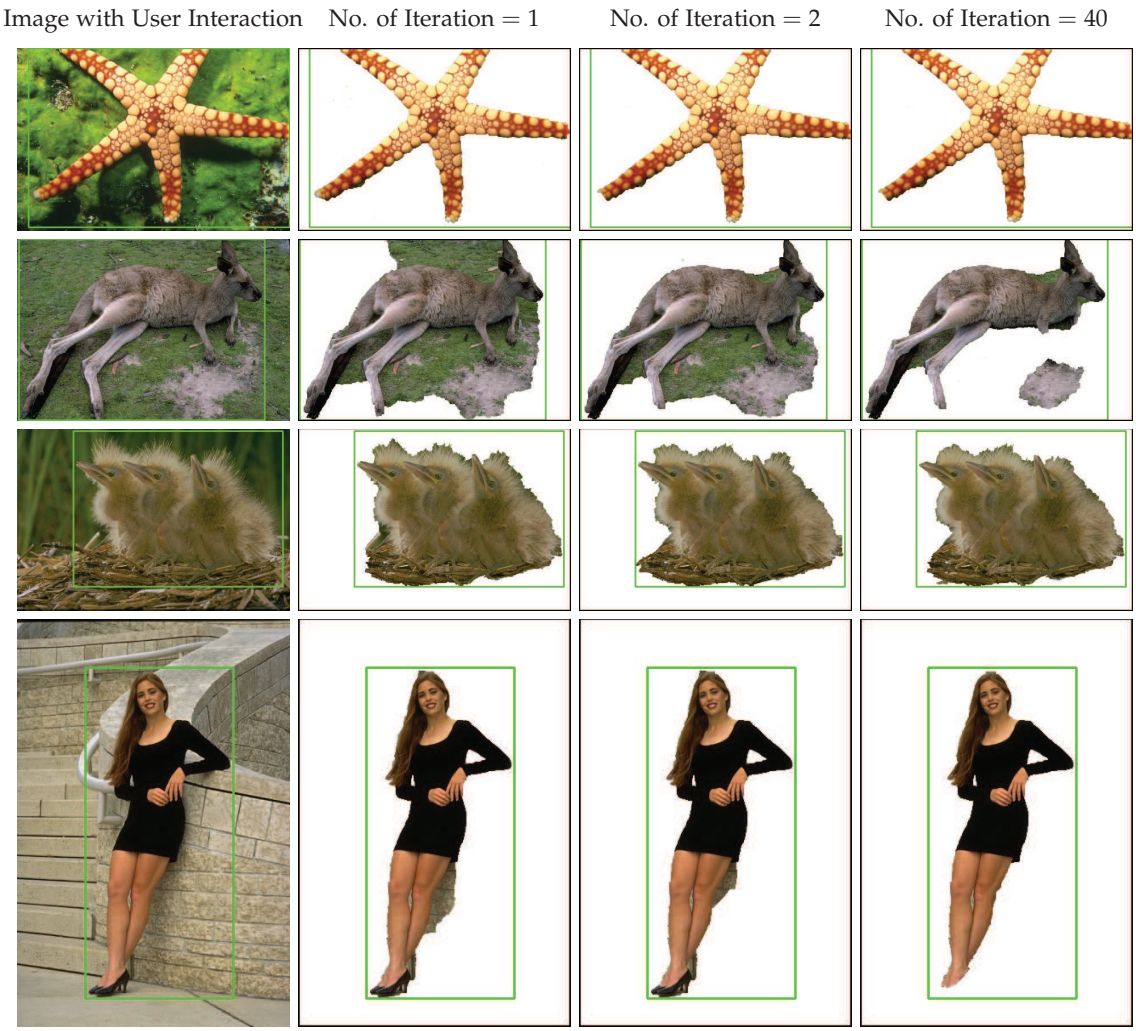


Figure 2.8: *GrabCut* segmentations [78] for increasing number of iterations on natural color images from Berkeley Segmentation Dataset [35]. First column: original color images from Figure 2.6. Second column: segmentations after 1 iteration, third column: segmentations after 2 iterations; last column: segmentations after 40 iterations.

Figure 2.9: *GrabCut* segmentations [78] for increasing number of iterations on natural color images from Berkeley Segmentation Dataset [35]. First column: original color images from Figure 2.7. Second column: segmentations after 1 iteration, third column: segmentations after 2 iterations; last column: segmentations after 40 iterations.

In conclusion, by employing more accurate and robust model for color distribution, Lazy Snapping and GrabCut outperform Boykov-Jolly approach on simple and moderately difficult color images. However, when foreground/background color information is less discriminative, none of these methods can generate satisfying segmentations. In the following chapter, we will focus on the solution to overcome this drawback in corporation with multiple visual cues (e.g. image texture) and propose a novel binary segmentation method under the interactive graph-cut based segmentation framework.

2.5 Parameter Determination for Graph Cut Segmentation

Although graph-cut based segmentation framework proposed by Boykov and Jolly [23] is verified to guarantee a global optima for wide class of energy function [60] and interactive with user-friendly planted *seeds* labeling "foreground" and "background", a fundamental yet unsolved issue of such framework is the parameter selection. Inappropriate choice of parameters can result unsatisfied segmentation, which is obscure for us to compare some state-of-the-art segmentation techniques as well as our proposed approaches (cf. Chapter 3) within this framework.

2.5.1 Selection of Parameter σ

In this section, we first take the parameter σ into consideration. Since our experimental object can never be ideal, unavoidably corrupted by some so-called *image noises* (e.g. Gaussian/Amplifier noise, Salt-and-pepper noise, Shot noise, etc.), it is important to find out the relationship between parameter σ and noise level of the sample image to yield better result in image segmentation. Intuitively, the function presented in Equation (2.38) corresponds to the distribution of noise among neighboring pixels of an image. Thus, σ can be estimated as *image noise*.

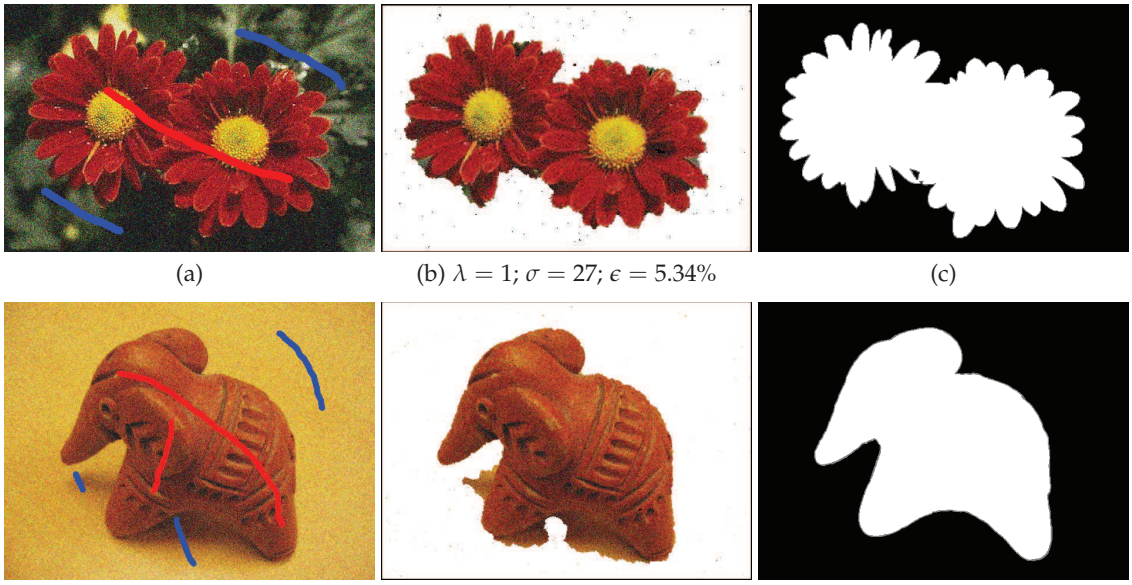
An example of the proposed experiment is demonstrated in Figure 2.10. We add Gaussian noise to a set of natural color images taken from the Berkeley Segmentation Dataset [35] and the MSRC (**M**icro**S**oft **R**esearch **C**ambridge) GrabCut Segmentation Database [20], perform binary image segmentation under this graph-cut segmentation framework and calculate the segmentation error subject to variations of parameter σ and standard deviation of Gaussian noise while parameter λ is fixed on 1 to put the equal emphases on both data and smoothness terms in Equation (2.4). Quantitative evaluation of segmentation performance is measured simply and sufficiently by the segmentation error rate defined as

$$\epsilon = \frac{\text{number of mis-classified pixels}}{\text{number of pixels in classified region of gold truth}} = \frac{FP + FN}{P} \tag{2.45}$$

where *false-positive* $FP = A_A - A_A \cap A_G$, *false-negative* $FN = A_G - A_A \cap A_G$ and *positive* $P = A_G$. A_G denotes the foreground area in gold truth; A_A denotes the segmented foreground area using graph-cut segmentation algorithm proposed by Boykov and Jolly [23]. In addition, we define two more measurements, *false-positive rate* FPR and *false-*

negtative rate FNR:

$$FPR = \frac{A_A - A_A \cap A_G}{A_G} \qquad FNR = \frac{A_G - A_A \cap A_G}{A_G} \tag{2.46}$$



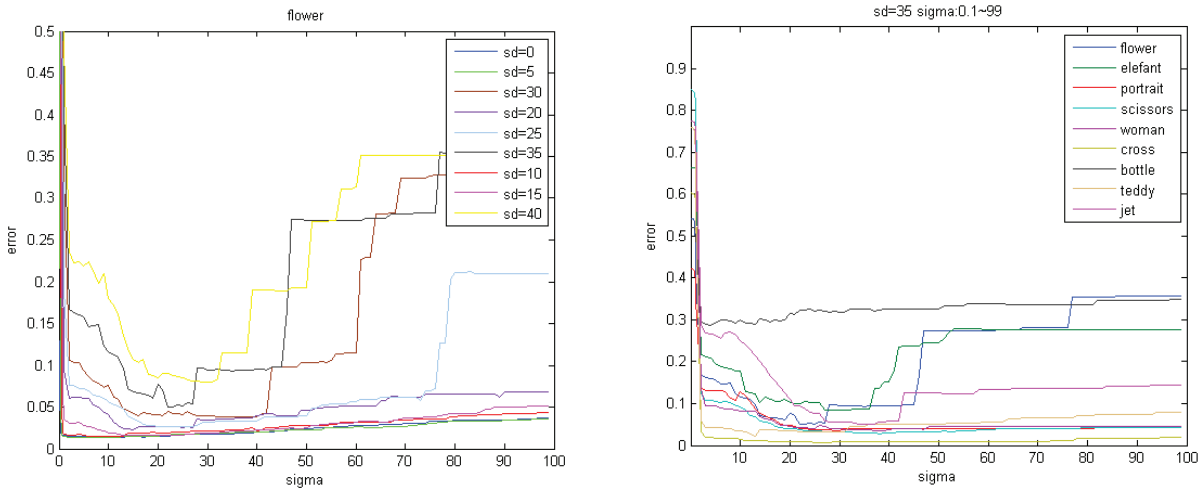
(a) (b) $\lambda = 1; \sigma = 27; \epsilon = 5.34\%$ (c)

(d) (e) $\lambda = 1; \sigma = 35; \epsilon = 12.85\%$ (f)

Figure 2.10: Segmentation results for two natural color images, "Flower" (Ref. 124084) and "Elephant", one from the Berkeley Segmentation Dataset [35] and the latter from the MSRC GrabCut Segmentation Database [20]. (a, d) Two sample images from the test database, corrupted by a gaussian noise with the standard deviation fixed to 35; (b, e) The corresponding segment results evaluated by Equation (2.45), different values of σ chosen to yield the optimal results; (c, f) Gold truth from the test database [35, 20], which assigns binary labels $\mathcal{L} = \{0, 1\}$ as background (black) and foreground (white).

Figure 2.11 (a) illustrates how error of segmentation varies according to the selection of parameter σ in a single object (e.g. "Flower") added by different levels of Gaussian noise, while Figure 2.11 (b) illustrates how error of segmentation varies according to the variation of parameter σ within a set of sample images corrupted by a fixed level of Gaussian noise (e.g. the value of standard deviation of Gaussian noise equals 35 which is very "noisy" for an image in reality). A conclusion can be drawn from Figure 2.11 is that if we set parameter σ close to the value of standard deviation of an additive Gaussian noise, the error of segmentation will decrease to a certain extend, namely, segmentation results can be improved significantly.

More precisely, since σ is closely related to the level of pixel intensity variation of a sample image, it is reasonable to let σ be the average of absolute intensity difference



(a) ϵ-σ for "Flower" with various noise levels

(b) ϵ-σ for 9 benchmark images with a fixed noise level

Figure 2.11: Relationship between parameter σ and noise level of the experimental object.

between two neighboring pixels, p and q [78, 75]:

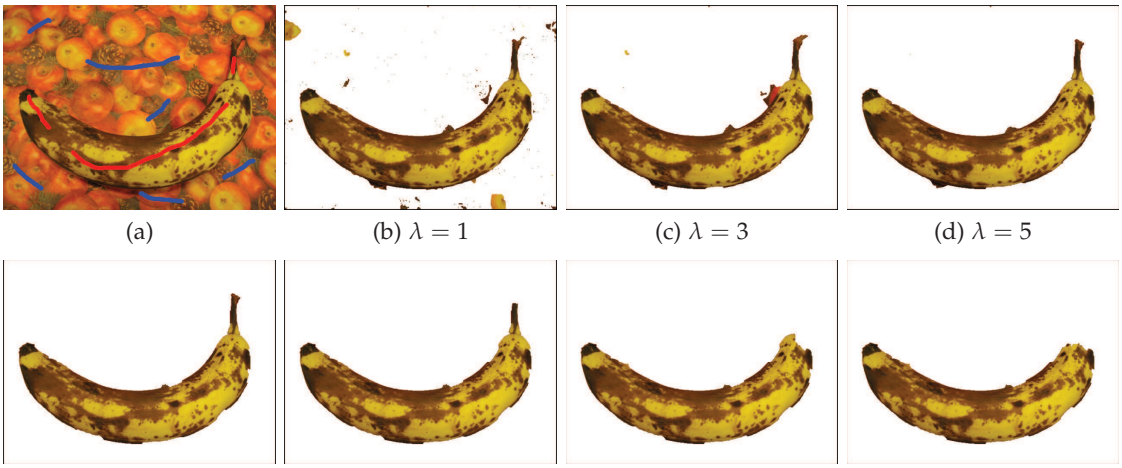
$$\sigma = \sqrt{\left\langle (I_p - I_q)^2 \right\rangle} \tag{2.47}$$

where $\langle \phi \rangle$ denotes the average or expected value of ϕ over a sample image. This choice of σ guarantees the exponential term in Equation (2.38) switches appropriately between high and low contrast thus suitable for the precise adjustment of parameter λ for the optimal segmentation performance.

2.5.2 Selection of Parameter λ

Now we start to address briefly the selection of parameter λ. Figure 2.12 (b-h) shows the segmentation results of Figure 2.12 (a) provided with different values of λ as well as a fixed parameter σ equals 20.

For Figure 2.12 (b, c), over-segmentation occurs when λ is low, while under-segmentation tends to appear under higher value of λ in Figure 2.12 (f-h). In between the over-segmentation and under-segmentation, there is a range of λ, under which results relatively good segmentation (cf. Figure 2.12 (d, e)). All these can be explained by the role which parameter λ plays in the energy function in Equation (2.39). For example, large value of λ encourages the smooth term to be dominant in the whole energy function so that the discontinuity between neighboring pixels is sufficiently or even over penalized, which leads the boundary between foreground and background segments to be too smooth and the background segments may contain parts of the foreground. From the



(a) (b) $\lambda = 1$ (c) $\lambda = 3$ (d) $\lambda = 5$

(e) $\lambda = 10$ (f) $\lambda = 25$ (g) $\lambda = 35$ (h) $\lambda = 45$

Figure 2.12: Manually Tuning of Parameter λ. (a) is the original image "Banana" with manually *planted* seeds; (b, c) are over-segmentations; (d, e) are well segmentations; (f-h) are under-segmentations.

perspective of classification, the *false-negtative rate* of the segmentation result is far larger than that of the *false-positive rate* in under-segmentation while the *false-negtative rate* of the segmentation result is lower than that of the *false-positive rate* in over-segmentation (cf. Table 2.5).

Table 2.5: The error rate ϵ, FNR and FPR of segmentations subject to different λ on "Banana". Ground truth for quantitative assessment is taken from the MSRC GrabCut Segmentation Database [20].

λ	ϵ (%)	FNR (%)	FPR (%)
1	9.65	2.86	6.97
3	5.11	2.92	2.19
5	4.29	3.66	0.63
10	4.73	4.64	0.09
25	5.73	5.64	0.09
35	8.6	8.49	0.11
45	9.58	9.47	0.11

Unlike parameter σ which can be reasonably well determined from a given sample image, parameter λ has to be learnt from each particular class of images, since for each image belong to a particular class, the range of appropriate λ may be different [58]. The most efficient way to estimate λ is so-called *empirical approach* [20, 75]. In [20], Blake et al. learned the parameter λ discriminatively by optimizing performance against ground truth over a training set of 15 images. In [75], Peng and Veksler proposed a novel approach based on binary classification. After the measure of segmentation performances on a training set of 80 images with 10 segmentations each under different values of λ,

segmentations were manually labeled as "good" (positive) or "bad" (negative). Then, a classifier was trained using the AdaBoost algorithm [43] with features based on intensity, texture, gradient direction, and corners.

Chapter 3

Combining Classification Techniques and Graph-Cut Based Segmentation Framework

Contents

Most well-developed interactive segmentation approaches based on graph cuts, for example, *Lazy Snapping* [65] and *GrabCut* [78], have been successful in color image segmentation with relatively high segmentation accuracy and few user-interaction. However, the common drawback of these techniques is the undesirable performance when the foreground object and the background share similar color profiles. For instance, in Figure A.4 (a), color information of pixels around the church-cross part is not distinct enough for color-feature-based image segmentation techniques to discriminate foreground object from the background. Significant decrease of segmentation performance thus occurs in Figure A.4 (b, c, e). More negative examples can be found from Figure 2.6 to Figure 2.9 in Section 2.4. In such cases, texture information is a preferred discriminant.

In this chapter, we propose to formulate the data energy in the energy function considering not only a single pixel but also its neighboring pixels, not only color information but also texture features and utilizing two highly scored classification methods, Support Vector Machine (SVM) and Random Forest (RF), to enhance the discrimination between foreground and background, where the color information is less discriminative. Experimental results on both synthetic and nature images indicate that our approach outperforms the other color-feature-based methods in yielding promising segmentation accuracy.

(a) Initialization with seeds (b) Boykov-Jolly; $\epsilon = 75.81\%$ (c) Lazy Snapping; $\epsilon = 18.43\%$

(d) Initialization with rectangle ROI (e) GrabCut; $\epsilon = 57.82\%$ (f)

Figure 3.1: A negative example for color based graph-cut segmentation techniques. "Cross" monument in the foreground, church and gravestone in the background share the similar color profile which makes color feature less discriminative in their joint regions. (a) Input image with user-planted seeds. This initialization is applied to both Boykov-Jolly algorithm [23] and *Lazy Snapping* algorithm [65]. (b) Results obtained by Boykov-Jolly algorithm. Church is misclassified as "foreground". (c) Results obtained by Lazy Snapping algorithm. Joint regions are misclassified to certain extents. (d) Initialization with a user specified rectangle ROI whose exterior is labeled as "background", while the interior remains unlabeled. This initialization is unique for *GrabCut* algorithm. (e) Results obtained by GrabCut algorithm. Misclassification occurs within the ROI. (f) Manual segmentation as the gold truth.

3.1 Local Binary Pattern for Texture Description

Local binary pattern (LBP) was firstly introduced by Ojala et al. [71] in 1996 as a texture descriptor that showed high discriminative power in texture classification. Later, Ojala et al. [72] extended the original LBP operator to a circular neighborhood of arbitrary radius to describe the multi-scale texture patterns. In addition, *uniform pattern* and rotation-invariant LBP were introduced to improve the discriminative property while decrease the feature labels that a LBP operator produces. Due to the low computational complexity and high discrimination of various types of textures, LBP and its generalizations have been widely used with considerable success in texture classification and segmentation [72, 70], image retrieval [90] and more recently face recognition [1].

The original LBP operator was designed to label pixels of an input image by thresholding a 3×3 neighborhood of each pixel with the center pixel value, binomially weighting the thresholded neighbors and summing them up. Semantic diagram of the whole LBP coding process is illustrated in Figure 3.2. As the neighborhood consists of 8 pixels, a total number of $2^8 = 256$ patterns of texture features can be obtained. Examples of LBP images are shown in Figure 3.6 (b, e). Formally, the original LBP operator is defined as:

$$\mathrm{LBP}(x_c, y_c) = \sum_{n=0}^{7} s(i_n - i_c) 2^n \tag{3.1}$$

where n scans over 8 neighbors surrounding the center pixel c, i_n and i_c denote the gray levels of the neighborhood pixel n and its center pixel c, (x_c, y_c) are the coordinates of the center pixel c. $s(\phi)$ is the thresholding function

$$s(\phi) = \begin{cases} 1, & \text{if } \phi \geq 0 \\ 0, & \text{if } \phi < 0 \end{cases} \tag{3.2}$$

The original LBP operator was subsequently extended to encode a pixel with neighborhood of arbitrary size and sampling points, thus making it more feasible to deal with textures differing in scales [72]. The idea is to align N evenly spaced neighbors n on a circle with radius R around the central pixel c so that

$$i_n = I(x_n, y_n), \quad n = 0, 1, \cdots, N-1 \quad \text{and} \tag{3.3}$$

$$x_n = x_c + R\cos(2\pi p/N), \quad y_n = y_c - R\cos(2\pi p/N) \tag{3.4}$$

where (x_n, y_n) are the coordinates of the neighborhood pixel n and $I(a, b)$ denotes the

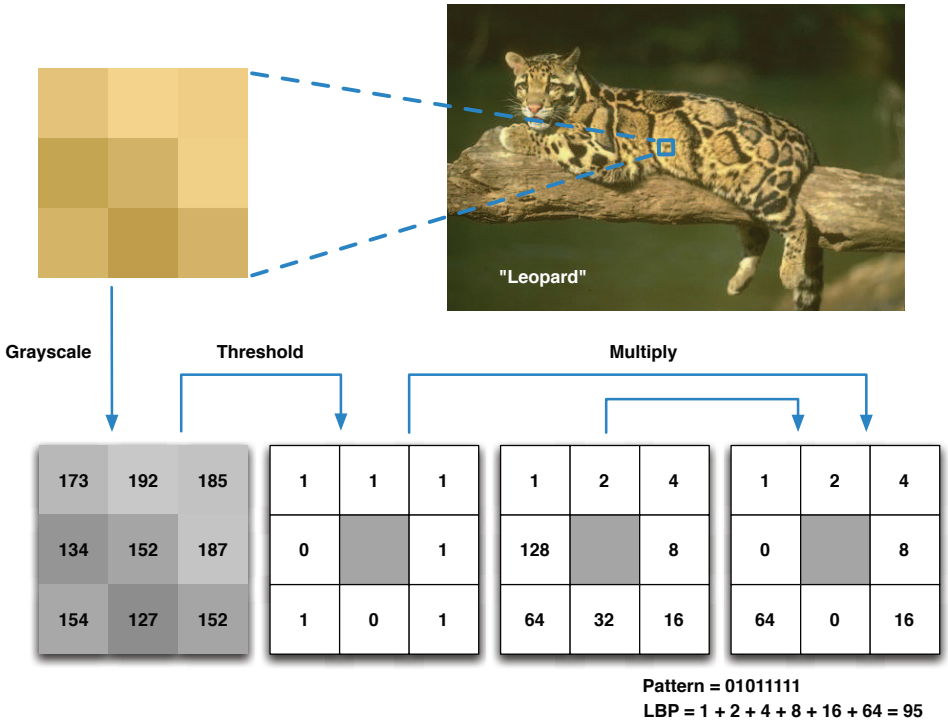


Figure 3.2: Calculation of the original LBP code.

gray level of a pixel at coordinate (a, b). Same as the original LBP operator, this extension LBP operator (denoted by $\text{LBP}_{N,R}$) is defined as

$$\text{LBP}_{N,R}(x_c, y_c) = \sum_{n=0}^{N-1} s(i_n - i_c)2^n \tag{3.5}$$

Figure 3.3 demonstrates three circularly symmetric neighborhood with different value of N and R. Value of a neighbor that is not exactly in the center of a pixel grid is estimated via interpolation.

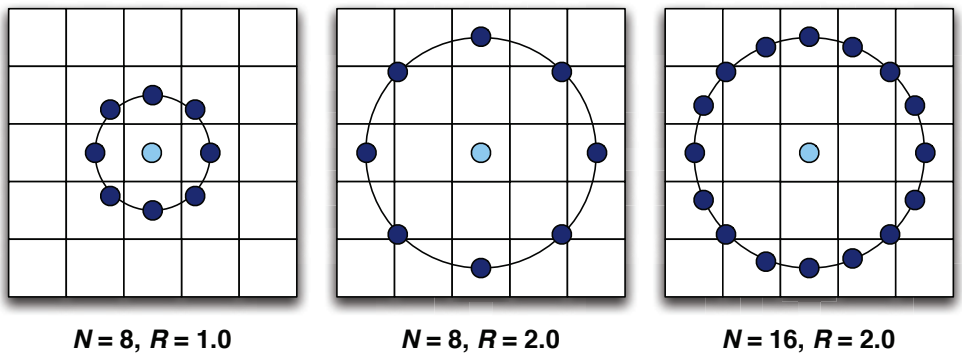


Figure 3.3: LBPs in different circular neighborhoods. Pixel value is obtained via interpolation whenever sampling point that is not in the center of a pixel grid.

3.1.0.1 The uniform LBP

Another extension of the original LBP operator is the use of so-called *uniform patterns* [72]. It was noticed that most of texture features contain a small subset of the total 256 LBP patterns. A LBP pattern is called *uniform* if it contains at most two bitwise transitions from 0 to 1 or vice versa. "11111111" and "00001110" for instance are uniform patterns whereas the pattern "01011111" (the case in Figure 3.2) is *non-uniform*. There are in total $N(N-1)+2$ uniform patterns generated by $\text{LBP}_{N,R}$, which mainly represent primitive micro-features such as line, edge and corner (cf. Figure 3.4). With all the patterns that are not uniform labeled as a unique pattern, the uniform LBP operator (denote by $\text{LBP}^{u2}_{N,R}$) produces $N(N-1)+3$ patterns of texture features. Figure 3.5 (b) shows a $\text{LBP}^{u2}_{8,1}$ histogram with 59 bins.

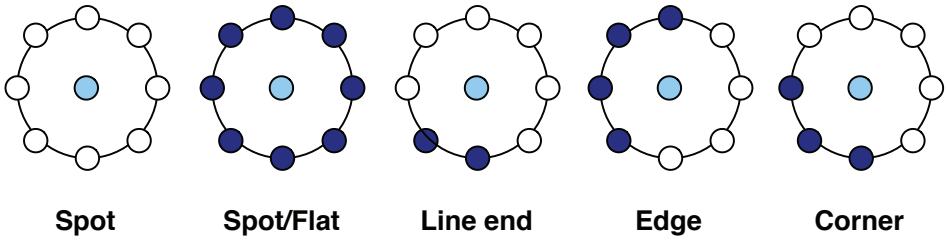


Figure 3.4: Examples of different texture patterns encoded by LBP.

3.1.0.2 The rotation invariant uniform LBP

Inspired by the idea of mapping uniform patterns to the original LBP patterns, a subset of the uniform patterns called *rotation invariant* uniform patterns (*riu*2) can be encoded by rotation to their minimum values. For example, 8-bit uniform pattern, "00001110", "00011100" and "00000111" all map to the minimum pattern "00000111". Compared to $\text{LBP}^{u2}_{N,R}$, a rotation invariant uniform LBP operator (denoted by $\text{LBP}^{riu2}_{N,R}$) produces in total $N+2$ patterns of texture features.

Figure 3.5 shows the robustness of rotation texture discriminant of $\text{LBP}^{riu2}_{N,R}$ compared with $\text{LBP}^{u2}_{N,R}$. Texture image in Figure 3.5 (b) is the rotated version of (a). Observe the $\text{LBP}^{u2}_{N,R}$ and $\text{LBP}^{riu2}_{N,R}$ histograms correspondingly, one can tell that $\text{LBP}^{u2}_{N,R}$ histograms for the two images are markedly irrelevant, while the $\text{LBP}^{riu2}_{N,R}$ histograms are approximately equivalent.

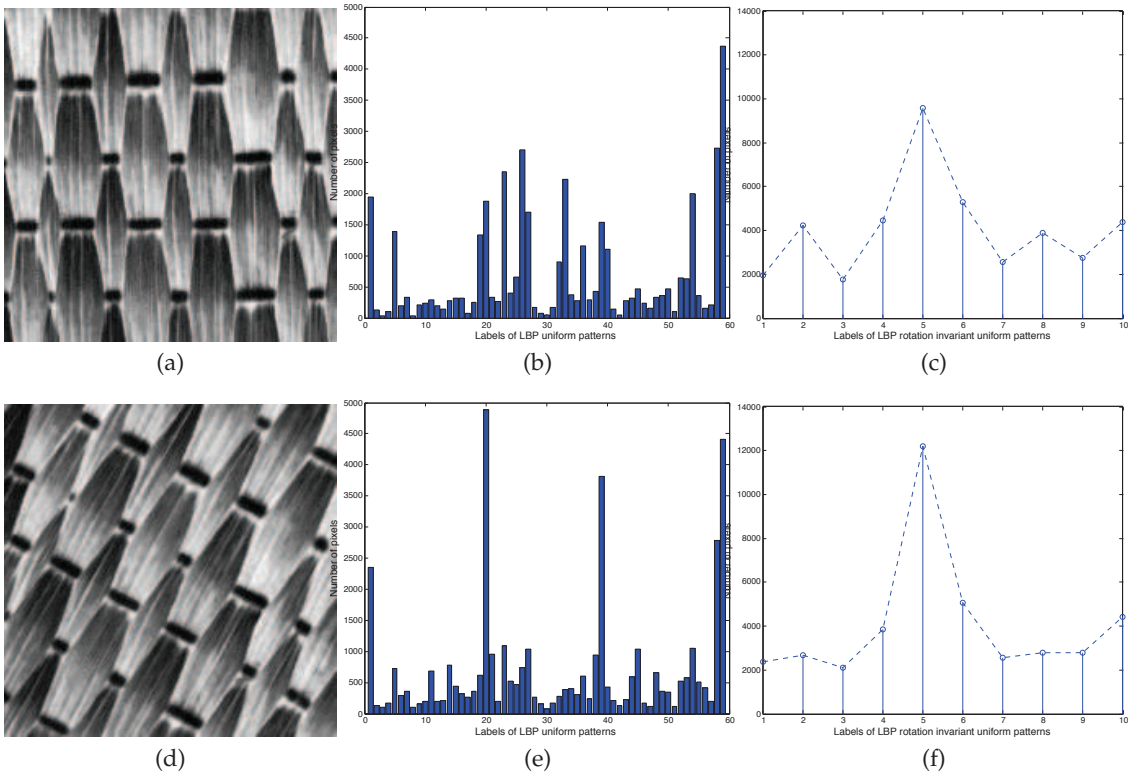


Figure 3.5: First column: texture image with rotation 0° and 30°. Second column: bins 1–59 of the corresponding LBP^{u2} histograms. Third column: bins 1–10 of the corresponding LBP^{riu2} histograms.

3.1.1 Preprocessing with Gaussian Filtering

When LBP is applied to large scale texture features, for instance, the leopard-print shown in Figure 3.6 (a), it is natural to set a large value of the neighborhood radius R to capture large details. However, larger radius comes with greater aliasing effect, which results in noise-sensitive, unreliable and thus less discriminative texture feature description. This is mainly due to the fact that LBP operator thresholds the neighbor with only the center pixel i_c, namely, sampling information at a single pixel position. Applying a Gaussian low-pass filter as preprocessing prior to LBP operation might be an appropriate solution to this dilemma.

Consider a 2-dimensional Gaussian filter, $G(x, y)$, as a zero-mean distribution of two independent variables with equal standard deviations, the size of the filter can be described as the standard deviation σ:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.6}$$

where x and y denote the pixel coordinates. By using Gaussian filter, each sample in the neighborhood can be made to collect intensity information from a larger range defined by σ. Additionally, high-frequency image noise can be filtered efficiently and simultaneously.

Figure 3.6 shows the advantage of using Gaussian filtering to increase the discriminative ability for large-scale texture. LBP code without Gaussian filtering is difficult to interpret (Figure 3.6 (b, c)) while Gaussian filtering with appropriate filter size can improve the discriminative ability a LBP operator (Figure 3.6 (e, f)). Throughout this thesis, the combination of Gaussian filtering and LBP operator is denoted by GLBP.

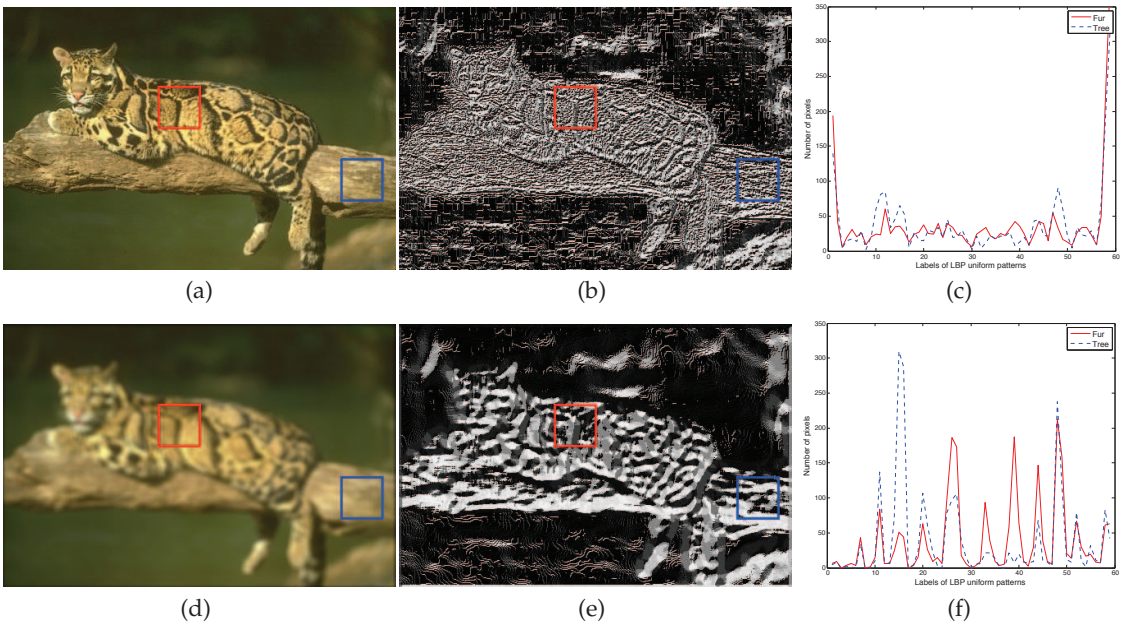


Figure 3.6: Combining gaussian filtering as preprocessing prior to LBP coding. First column: "Leopard" image (Ref. 160068) before (a) and after (b) gaussian blurring. Different type of texture "Leopard-print" and "Tree" marked via red and blue square ROIs respectively. Second column: corresponding original LBP images. Third column: bins 1–59 of the corresponding LBP^{u2} histograms, "Leopard-print" texture features (red solid line) and "Tree" texture features (blue dash line).

3.2 Support Vector Machine

Support Vector Machine (SVM), firstly introduced by Cortes and Vapnik [34], is a supervised learning model originally used for binary classification tasks. Known as a class of kernel based method, SVM is able to create nonlinear classifiers by applying a *Kernel Trick* [3] that maps low-dimensional vectors of input datapoints into a high-dimensional feature space and constructing an optimal hyperplane that separates mapped datapoints into two classes with the distance from this hyperplane to the nearest datapoints of each class maximized. The ability to perform nonlinear classification makes SVM successful in many real-world pattern recognition problems like text categorization [53], handwritten character recognition [81], face detection in images [73]. In the following text, we will review the basic theory of SVM for both linear and nonlinear cases.

3.2.1 Linear Support Vector Classifier

To detail the basic definition and theory of SVM, we will start with the simplest case: binary classification on linear separable data. Let $\mathcal{D} = \{(\mathbf{x}_i, y_i)|\mathbf{x}_i \in \mathbb{R}^p, y_i \in \{-1, +1\}\}_{i=1}^n$ denotes the input training dataset, where $\mathbf{x}_i$ is a sample in p-dimensional vector, y_i is the corresponding class label, the main idea of a SVM classifier is to construct a $(p-1)$-dimensional *separating* hyperplane, $\mathbf{w} \cdot \mathbf{x} + b = 0$, in this p-dimensional space that maximize the margin $2/\|\mathbf{w}\|$ between two *constraint* hyperplanes, $\mathbf{w} \cdot \mathbf{x} + b = +1$ and $\mathbf{w} \cdot \mathbf{x} + b = -1$. Figure 3.7 shows a simple example of 2-dimensional binary classification implemented using SVM. Here, $\mathbf{w}$ is the normal vector of the hyperplane, b is the intercept and $|b|/\|\mathbf{w}\|$ is the perpendicular distance from the *separating* hyperplane to the original. With respect to the description of two *constraint* hyperplanes, sample datapoints that are separated into two classes (red circle denotes class label $y_i = +1$ and blue square denotes class label $y_i = -1$ in Figure 3.7) can be described by

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq +1 \quad \text{for } y_i = +1 \tag{3.7}$$

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \quad \text{for } y_i = -1 \tag{3.8}$$

By combining Equation (3.7) and Equation (3.8), a uniform of constraints can be expressed by one set of inequalities:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0 \quad \forall i \tag{3.9}$$

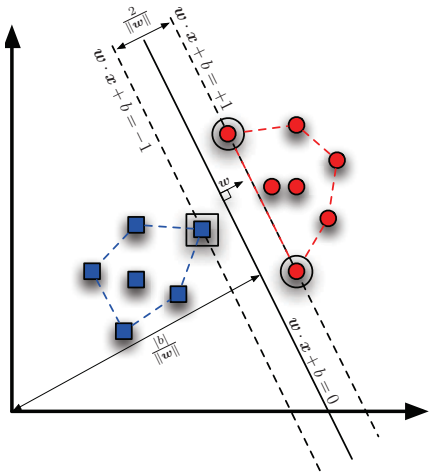


Figure 3.7: An example of SVM classification in 2-dimensional space. Red circle and blue square represent two linearly separable classes. Support vectors (marked by double-borders) are training samples that define the margin (distance between two dash lines) of a hyperplane (solid line).

This constraint prevents sample datapoints from falling within the margin between two constraint hype planes. Some particular datapoints that lie on each of the constraint hyperplane and whose removal would change the current solution, are named *Support Vectors* (indicated in Figure 3.7 by double-borders).

Since the aim of SVM is to find $\mathbf{w}$ and b for the hyperplanes that maximize the classification margin $2/\|\mathbf{w}\|$, the solution for this optimization problem can be formulated as

$$\min_{\mathbf{w},b} \Psi(\mathbf{w}), \quad \Psi(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2, \tag{3.10}$$

subject to constraints formulated in Equation (3.9).

By introducing the Lagrange multipliers α_i, $i = 1, \cdots, n$, Equation (3.9) and Equation (3.10) can be reformulated as:

$$\min_{\mathbf{w},b} \max_{\alpha_i \geq 0} L(\mathbf{w}, b, \boldsymbol{\alpha}), \quad L(\mathbf{w}, b, \boldsymbol{\alpha}) \equiv \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1] \tag{3.11}$$

The optimal solution for Equation (3.11) can be provided by *stationary* Karush-Kuhn-Tucker (KKT) conditions [61] in terms of equality constraints:

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \tag{3.12}$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.13}$$

Substituting Equation (3.12) and Equation (3.13) into Equation (3.11) gives a *Dual* form of $L(\mathbf{w}, b, \boldsymbol{\alpha})$, denoted by $\hat{L}(\boldsymbol{\alpha})$, to be maximized:

$$\hat{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad \text{s.t. } \alpha_i \geq 0 \; \forall i, \; \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.14}$$

This is a quadratic programming optimization problem and the Lagrange multipliers α_i can be optimized by Quadratic Programming (QP) solver. Consider that any input datapoint satisfying Equation (3.13) is a Support Vector, the offset b can be calculated as:

$$b = \frac{1}{N_s} \sum_{i=1}^{N_s} (y_i - \mathbf{w} \cdot \mathbf{x}_i) \tag{3.15}$$

where N_s is number of Support Vectors. Finally, the decision function of SVM is obtained as

$$f_{\text{SVM}}(\mathbf{x}) = \text{sgn}[\mathbf{w} \cdot \mathbf{x} + b] = \text{sgn}\left[\sum_{i=1}^{n} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{x}) + b\right] \tag{3.16}$$

where sgn$[\cdot]$ denotes the signum function.

3.2.1.1 Linear non-separable case

It is worthing noting that in application of classification for real world data, there is often the case that sample dataset is not fully linear separable. A modified maximum margin idea called *Soft Margin* method was then proposed by Cortes and Vapnik [34] that allows misclassified datapoints. To make the linear classifier slightly tolerable with classification error, the constraints in Equation (3.9) can be relaxed by introducing a positive slack variables $\xi_i \geq 0$, $i = 1, \cdots, n$:

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i \geq 0 \quad \forall i \tag{3.17}$$

Also note that the tolerance of error occurrence must be limited as low as possible to prevent overfitting. This can be accomplished by adding a penalty term to the objective function $\Psi(\mathbf{w})$ in Equation (3.10):

$$\min_{\mathbf{w}, b, \boldsymbol{\xi}} \Psi(\mathbf{w}, \boldsymbol{\xi}), \quad \Psi(\mathbf{w}, \boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \tag{3.18}$$

subject to the constraints formulated in Equation (3.17). The soft margin parameter C is specified to control the tradeoff between the slack variable penalty and the margin. The larger value of C, the larger penalty on misclassification of data to a certain class. Consequently, the estimation of optimal hyperplanes (cf. Equarion (3.14)) is same as that

in the linear separable case except that the Lagrange multipliers α_i now have an upper bound of C.

3.2.2 Nonlinear Support Vector Classifier

Consider a far more complex case depicted in the left hand side of Figure 3.8, where sample dataset is extremely linear non-separable even with relaxation of constraints as in Equation (3.17). However, instead of building linear classifiers, Bernhard *et al.* [21] proposed to create nonlinear classifiers by applying the kernel trick to estimation of optimal hyperplane. The main idea is based on the fact that original input space can always be mapped to some higher-dimensional feature space where the sample dataset is expected to be linearly separable (cf. Figure 3.8 in the right hand side). Soft Margin method can be subsequently used in this transformed high-dimensional feature space in a straightforward way.

Defining a feature mapping $\Phi : \mathbf{x} \to \phi(\mathbf{x})$ and a *kernel function* $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$, the dot products of the input datapoints $\mathbf{x}_i \cdot \mathbf{x}_j$ in the optimization function $\hat{L}(\boldsymbol{\alpha})$ (cf. Equation (3.14)) can be replaced by $k(\mathbf{x}_i, \mathbf{x}_j)$, without computing $\phi(\mathbf{x})$ explicitly:

$$\max_{\boldsymbol{\alpha}} \hat{L}(\boldsymbol{\alpha}), \quad \hat{L}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \quad \text{s.t. } 0 \leq \alpha_i \leq C \ \forall i, \ \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.19}$$

And the decision function of nonlinear SVM can be finally obtained as:

$$f_{\text{SVM}}(\mathbf{x}) = \text{sgn}[\mathbf{w} \cdot \mathbf{x} + b] = \text{sgn}\left[\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b\right] \tag{3.20}$$

There exists several popular kernel functions in SVM:

- polynomial kernel function

$$k(\mathbf{x}_i, \mathbf{x}_j) = (\gamma \mathbf{x}_i \cdot \mathbf{x}_j + \delta)^d \tag{3.21}$$

- sigmoid kernel function

$$k(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma \mathbf{x}_i \cdot \mathbf{x}_j + \delta) \tag{3.22}$$

- gaussian radial basis function (GRBF)

$$k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2} \tag{3.23}$$

where $\gamma > 0, \delta, d$ are parameters defining the kernel's behavior. The GRBF is by far the most popular choice of kernel function in SVM classification due to its ability of localized similarity measurement in Euclidean distance. Throughout this thesis, we adopt this kernel function as a similarity metric for both color and texture features.

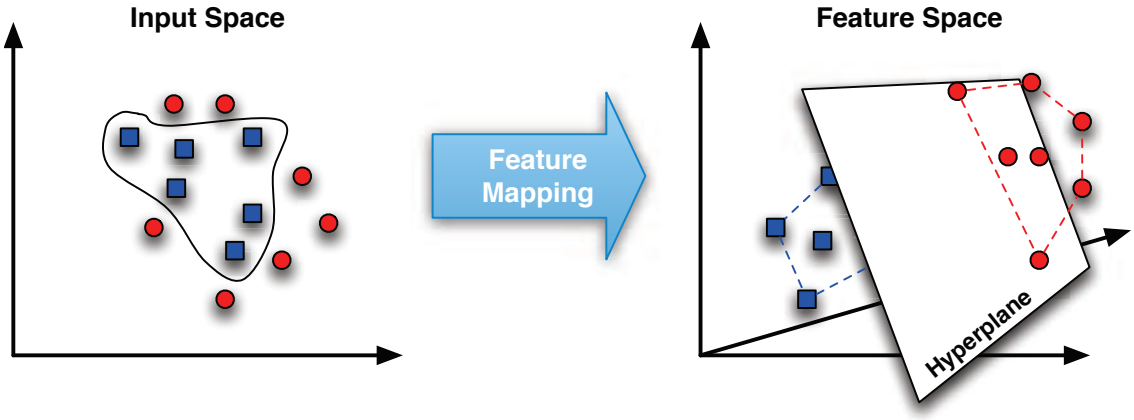


Figure 3.8: An example of kernel trick transforming linearly non-separable training samples from the original 2-dimensional input space to 3-dimensional feature space where samples are possible to be separated linearly by constructing a maximum margin hyperplane.

3.2.3 Probabilistic Output of SVM

To combine the pixelwise binary classification results learned by a classifier (e.g. SVM or RF) and graph-cut based segmentation framework, it is required to use soft classification (posterior class probability, i.e. $\Pr(class|input)$) instead of hard classification (class-label prediction) to determine the data term of the energy function in Equation (2.2). By definition, data energy $D_p(l_p)$ measures how much assigning a label l_p to a pixel p disagrees with the data. This can be interpreted alternatively in scope of binary classification that the data energy $D_{\mathbf{x}_i}(y_i = -1)$ measures the probability of a sample datapoint $\mathbf{x}_i$ assigned an opposite class label, $y_i = +1$. Thus, we can redefine the data energy as:

$$D_{\mathbf{x}_i}(y_i = -1) = \Pr(y_i = +1|\mathbf{x}_i), \quad D_{\mathbf{x}_i}(y_i = +1) = 1 - \Pr(y_i = +1|\mathbf{x}_i) \tag{3.24}$$

However, standard SVMs [93] only predict a class label of any input sample $\mathbf{x}$ by computing a decision function f_{SVM} in Equation (3.20). Platt [76] proposed a method of producing probabilistic output from SVM by fitting a sigmoid model:

$$\Pr(y = +1|\mathbf{x}) \approx P_{A,B}[f(\mathbf{x})] \equiv \frac{1}{1 + \exp[Af(\mathbf{x}) + B]} \tag{3.25}$$

where $f(x)$ is the unthreshold decision function (cf. Equation (3.20)) of SVM,

$$f(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b \tag{3.26}$$

The parameter A and B are determined by solving a regularized maximum likelihood problem:

$$\min - \sum_{i=1}^{n} \Big(t_i \log(p_i) + (1 - t_i) \log(1 - p_i) \Big), \tag{3.27}$$

for

$$p_i = P_{A,B}[f(\mathbf{x}_i)] \quad \text{and} \quad t_i = \begin{cases} \frac{N_+ + 1}{N_+ + 2}, & \text{if } y_i = +1 \\ \frac{1}{N_- + 2}, & \text{if } y_i = -1 \end{cases} \tag{3.28}$$

where N_+ counts the number of positive y_i while N_- negative. It was reported in that an unbiased training set $(f(\mathbf{x}_i), y_i)$ for optimization of Equation (3.27) could be derived by a three-fold cross validation method efficiently.

3.3 Random Forest

Random Forest (RF), systematically introduced by Breiman [27], is a learning ensemble consisting of Breiman's bagging and randomized decision trees proposed by Amit et al. [6]. It has been proven that RF classifiers have several advantages over state-of-the-art classifiers (e.g. SVM) [95]: They are overfit free, highly efficient in computation on large datasets, multi-class capable and resistant to outliers while yielding competitive classification accuracy to SVM classifiers. In addition, the inherent-feature-selection mechanism of a RF classifier provides us with ease of using multiple features (e.g. color, texture) in this thesis to increase the binary pixelwise classification performance significantly.

Let $\mathcal{T} = \{T_1, \cdots, T_N\}$ denotes set of binary decision trees and $\mathcal{D} = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbb{R}^p, y_i \in \{1, \cdots, k\}\}_{i=1}^{n}$ denotes the input training dataset, where p is the dimension of a datapoint $\mathbf{x}_i$ and k is the class label. Randomization of the binary decision tree can be performed by subsampling the entire training dataset using out-of-bag bootstrapping so that each tree is trained with a different subset of the training dataset. Each split node (non-terminal node) of a tree contains a binary decision function that best split the space of input data to be classified while each leaf node (terminal node) is labeled by a posterior class probability $p_t(y_i|\mathbf{x}_t), t = 1, \cdots, N$, for the t^{th} tree.

During the training stage of a binary decision tree, samples from the assigned subset are propagated from a split node to its right or left child nodes based on a suitable decision function in a top-down manner. For each split node, the best decision function

is selected based on a splitting criterion such as Gini impurity $\sum_{i=1}^{n} p_s(y_i|\mathbf{x}_s)$ or Information gain $-\sum_{i=1}^{n} p_s(y_i|\mathbf{x}_s)\log[p_s(y_i|\mathbf{x}_s)]$ with $p_s(y_i|\mathbf{x}_s)$ denoting the posterior probability of sample $\mathbf{x}_s$ in the s^{th} split node belonging to class y_i. The selection process is repeated until a split node receives too few samples or reaches a given depth. For the evaluation, each testing sample is propagated through each tree producing a posterior class probability $p_t(y_i|\mathbf{x})$, $t = 1, \cdots, N$, for the t^{th} tree. Finally, the probabilistic output of a RF classifier can be obtain as a joint distribution:

$$\Pr(y_i|\mathbf{x}) \approx \frac{1}{N}\sum_{t=1}^{N} p_t(y_i|\mathbf{x}) \tag{3.29}$$

as well as the decision function of RF:

$$f_{\mathrm{RF}}(\mathbf{x}) = \arg\max_{y_i} \frac{1}{N}\sum_{t=1}^{N} p_t(y_i|\mathbf{x}) \tag{3.30}$$

3.3.0.1 Performance evaluation between SVM and RF

To evaluate the classification performance comparatively between SVM and RF, four synthetic datasets are generated from both linear and nonlinear function (cf. Figure 3.9). For each dataset, 200 randomized datapoints are used for training while 5×10^5 randomized datapoints are used for testing. Parameters for SVM and RF are estimated using cross validation method to ensure the optimal performance.

Performance of classifier can be evaluated quantitatively by calculating

$$Accuracy = \frac{TP + TN}{P + N} \tag{3.31}$$

Table 3.1: Comparison of classification accuracy between SVM and RF on learning linear and non-linear functions.

Accuracy (%) Function / Classifier	$y = 1.5x$	$y = \cos(10x)$	$y = \tan(10x)$	$y = \sqrt{(\arctan(y/x))^2 - x^2}$
SVM	98.05	93.36	77.20	96.41
RF	96.01	94.27	90.45	94.98

Table 3.1 shows the performances of SVM and RF in case of binary classification. In general, RF classifier yields comparative classification accuracy to SVM classifier except for the non-linear function $y = \tan(10x)$, where RF outperforms SVM in considerable increase of classification accuracy.

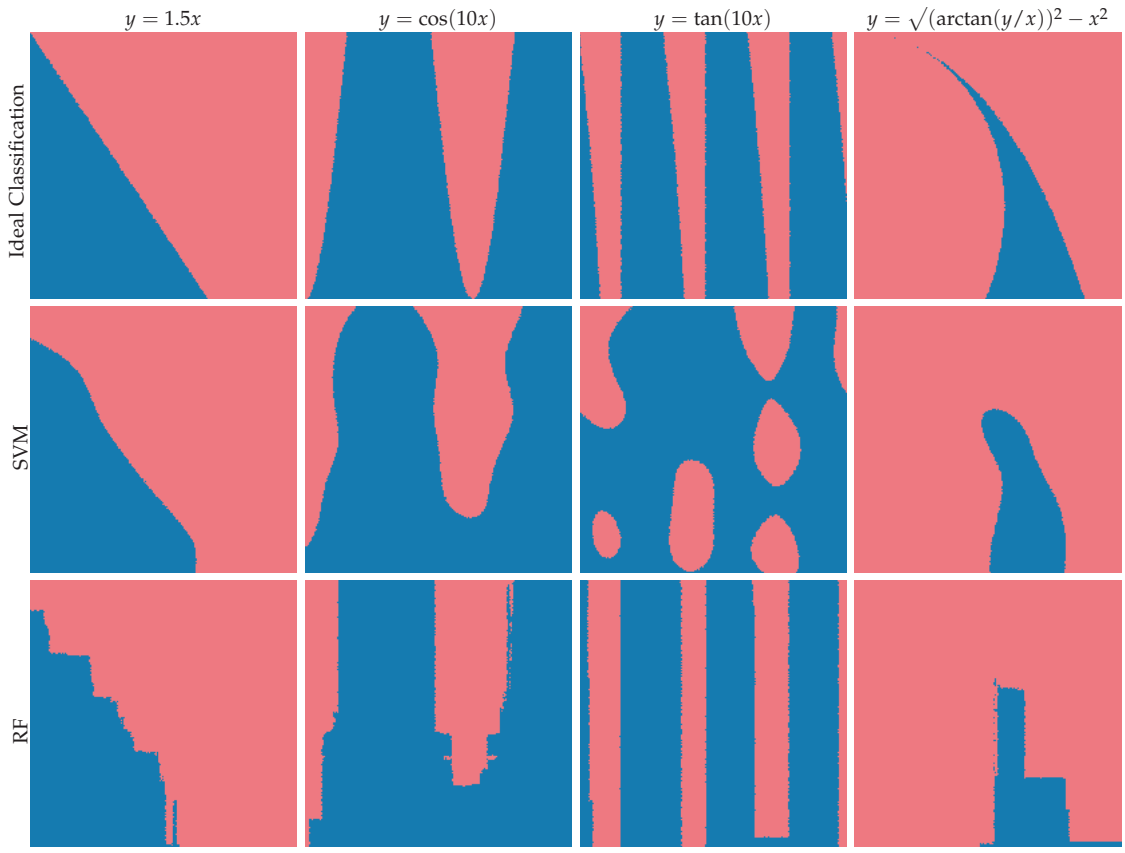


Figure 3.9: Comparison of SVM and RF on synthetic datatsets.

3.4 Graph-Cut based Segmentation using Support Vector Machine and Random Forest

As mentioned at the beginning of this chapter, we propose to combine the original graph-cut segmentation framework with multiple features learned by SVM or RF classifier to improve the segmentation. Input data for the classifier are provided by user-interaction (brushstrokes or a rectangle) assigning a pixel to foreground or background label. Color feature is stored in a 3-dimensional (tri-channel values) vector while LBP texture feature is arbitrary (e.g. original LBP code in 256 bins of histogram, $LBP^{u2}_{8,1}$ code in 59 bins and $LBP^{riu2}_{8,1}$ in 10 bins). To combine both color and texture features, we simply add texture feature and shape parameter as the classical RGB input data. We introduce texture penalties using LBP and shape using a certain neighborhood template, which will be detailed in the following section.

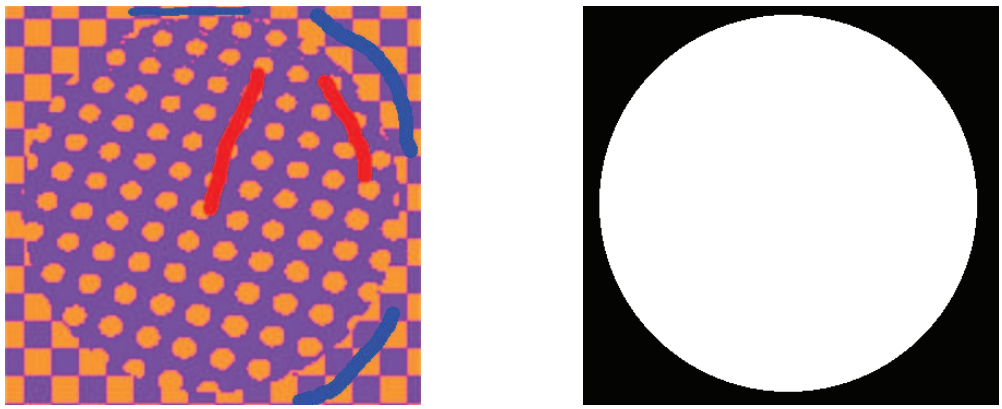
3.4.1 Feature Configuration

3.4.1.1 Neighborhood template for feature modeling

Since texture feature of an image is a local region-based property, it does not make sense to gather texture information from based on a particular pixel without specifying a neighborhood around it. In application to texture classification [72, 70], image retrieval [90], face recognition [1] etc, it is a common approach to divide the input image into non-overlapping small blocks of equal size and modeling the texture feature by calculating the histogram of LBP codes in each block. Possible criticism of this block-histogram based texture feature configuration is arbitrary and non-trivial. In addition, it is likely to cause both aliasing and loss of spatial resolution. Instead of modeling LBP texture feature in histogram block by block, we introduce a neighborhood template (a sliding window) gathering feature information pixel-wisely. During the scanning of this template, LBP code of each pixel in the neighborhood as well as its center pixel is collected in sequence to construct a feature vector. The number of neighbors (size of the window) is fixed to 5 as a compromise of low computational complexity in feature space and high discriminant spatially. Thus, the texture feature configuration using neighborhood template is a 25-dimensional vector.

To demonstrate qualitatively the effectiveness of the proposed feature configuration using 5×5 neighborhood template, we firstly apply it to the segmentation of a synthetic "Texture" image in size of 382×367 shown in Figure 3.10 (a). We can observe that in this "Texture" image, different patterns of texture distinguish the foreground

from the background, whereas color distribution vanishes this distinction, texture only must give better results. Figure 3.11 shows the effectiveness of the 25-dimensional LBP/GLBP texture-shape feature vector (Figure 3.11 (c, d)) compared to the pixel-wise 1-dimensional LBP/GLBP texture feature vector (Figure 3.11 (a, b)). We perform the evaluation task effectively based on both probabilistic outputs of a classifier in terms of probabilistic distribution map (cf. the first row of Figure 3.11) and the corresponding segmentation obtained under the interactive graph-cut based segmentation framework proposed by Boykov and Jolly [23] (cf. Chapter 2, Section 2.4).



(a) Synthetic "Texture" image with seeds (b) The corresponding gold truth

Figure 3.10: An synthetic "Texture" image of 382×367 pixels with *Polka dot* texture in the foreground and *Chessboard* texture in the background, whereas the distribution of color information stays the same.

Recall that energy function of binary labeling problem appropriately defined under MAP-MRF framework can be optimized via computing the minimum cut of a corresponding graph (cf. Chapter 2, Section 2.3). In order to make our multi-feature based segmentation compatible with the interactive graph-cut based segmentation framework, we now reformulate the general energy function in Equation (2.4) with data term given by the probabilistic output of a classifier and smoothness term defined similarly as in Lazy Snapping (cf. Equation (2.41)):

$$E(\mathcal{L}) = -\sum_{p \in \mathcal{V}} \log \Pr(l_p|\mathbf{x}_p) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} \frac{|l_p - l_q|}{||\mathbf{x}_p - \mathbf{x}_q||^2 + 1} \cdot \frac{1}{dist(p,q)}. \tag{3.32}$$

Some notations in Equation (3.32) should be interpreted particularly as follows: $\mathbf{x}_p$ denotes a multi-dimensional feature vector of pixel p. Labeling $\mathcal{L} = \{l_p | l_p \in \{-1, +1\}\}_{p=1}^{|\mathcal{V}|}$ comprises $l_p = -1$ if pixel p is a "background" pixel and $l_p = +1$ if p is a "foreground" or "object" pixel. Also, a set of pixels pre-labeled as "foreground" via user-brushed red strokes is denoted as foreground seeds $\mathcal{O}$ while the set of pixels pre-labeled as "back-

ground" via blue strokes is denoted as background seeds $\mathcal{B}$. $\Pr(l_p|\mathbf{x}_p)$ denotes the posterior probability of labeling a pixel p based on the observation of its features $\mathbf{x}_p$. It is estimated by a classifier whose training data are taken from the seeds (hard constraints) via user-interaction.

General experimental settings are as follows:

- Feature: 1-dimensional LBP/GLBP texture feature vector or 25-dimensional LBP/GLBP texture-shape feature vector. For GLBP, parameter σ is 3 and the kernel size of the gaussian filter is 9×9. This ensures the possibility of GLBP texture feature encoding large-scale texture patterns.

- Classifier: RF classifier is adopted and the parameters of the classifier are estimated using cross-validation method.

- Energy function: Parameter λ is learned using empirical approach and fixed to 10 to ensure the optimal segmentation.

- Graph: A 8-neighborhood adjacent directed graph is constructed with two terminal nodes s and t, respectively corresponds to the "foreground" labels and "background" labels. Note that this graph is hard-constraints free, that is, hard constraints (seeds) are only used for the training of a classifier to estimate the data term of the energy function and it is relaxed during the construction of graph. The edge weights of the graph are given in Table 3.2. Since the quality of final segmentation strongly depends on the probabilistic distribution when hard constraints are not involved in determining the data energy directly (e.g. by assigning pre-labeled pixels with very large penalties), our experiment is able to reveal the discriminative property of the proposed feature configuration.

Table 3.2: Edge weights for maximum *a posteriori* estimation.

Edge	Weight	For
$w_{s,p}$	$-\log \Pr(-1\|\mathbf{x}_p)$	$p \in \mathcal{V}$
$w_{p,t}$	$-\log \Pr(+1\|\mathbf{x}_p)$	$p \in \mathcal{V}$
$w_{p,q}$	$\lambda \cdot \frac{\|l_p - l_q\|}{\|\|\mathbf{x}_p - \mathbf{x}_q\|\|^2 + 1} \cdot \frac{1}{dist(p,q)}$	$\{p,q\} \in \mathcal{N}$

As we can observe in Figure 3.11, the 25-dimensional GLBP texture-shape feature vector outperforms the others in resulting the most discriminative probabilistic distribution. 1-dimensional texture vectors in Figure 3.11 (a, b) fail to model the texture patterns and thus resulting undesirable segmentations (parts of the foreground object are successfully extracted). In addition, Gaussian-blurring preprocess is verified to be a booster

for the texture discriminative ability of LBP (comparing Figure 3.11 (a) with (b) and (c) with (d)).

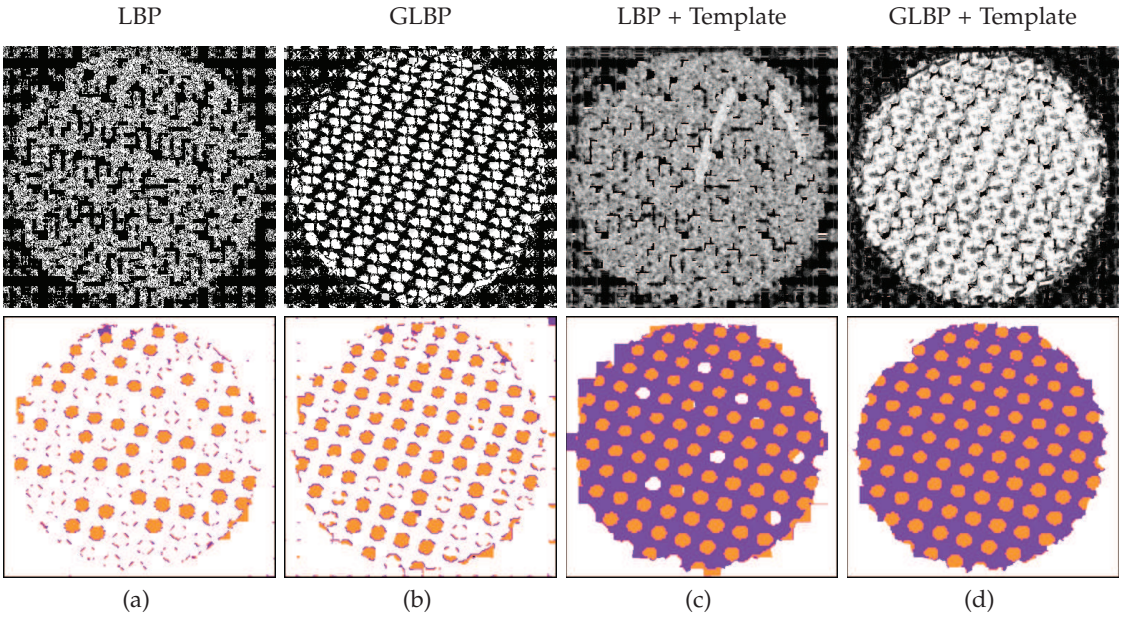


Figure 3.11: Texture feature configuration using neighborhood template improves the graph-cut based segmentation performance on synthetic "Texture" image. RF classifier is applied to learn the features and outputs the probability of each pixel classified to foreground. The corresponding probabilistic distribution maps are shown in the first row while the final segmentations in the graph-cut based framework without hard-constraints are shown in the second row.

3.4.1.2 Fusion of texture/color features

Texture feature is a discriminant in segmentation of the synthetic "Texture" image, however, the contribution of the color feature alone, or the combination of texture and color features is also experimented under same experimental settings but allowing color information in diverse configurations of features: (a) color feature without neighborhood template (3-dimensional color feature vector, each vector consists of tri-channel values), (b) combination of color and texture features (GLBP) without neighborhood template (4-dimensional texture-color feature vector), (c) color feature using neighborhood template (75-dimensional color-shape feature vector) and (d) combination of color and texture features (GLBP) using neighborhood template (100-dimensional texture-color-shape feature vector). Corresponding experimental results are illustrated in Figure 3.12.

Based on Figure 3.12 (a, c), color feature alone is unable to distinguish texture as expected. It results ambiguous probabilistic distribution and false segmentation even strengthened by using neighborhood template (cf. Figure 3.12 (c)). By adding GLBP tex-

ture feature into the feature configuration, segmentation performance can be improved considerably (cf. Figure 3.12 (b, d)). Nevertheless, comparing Figure 3.12 (d) with Figure 3.11 (d), color feature in this particular case degrades the performances of the segmentation. It is somewhat an injurious ingredient and should be excluded in feature configuration.

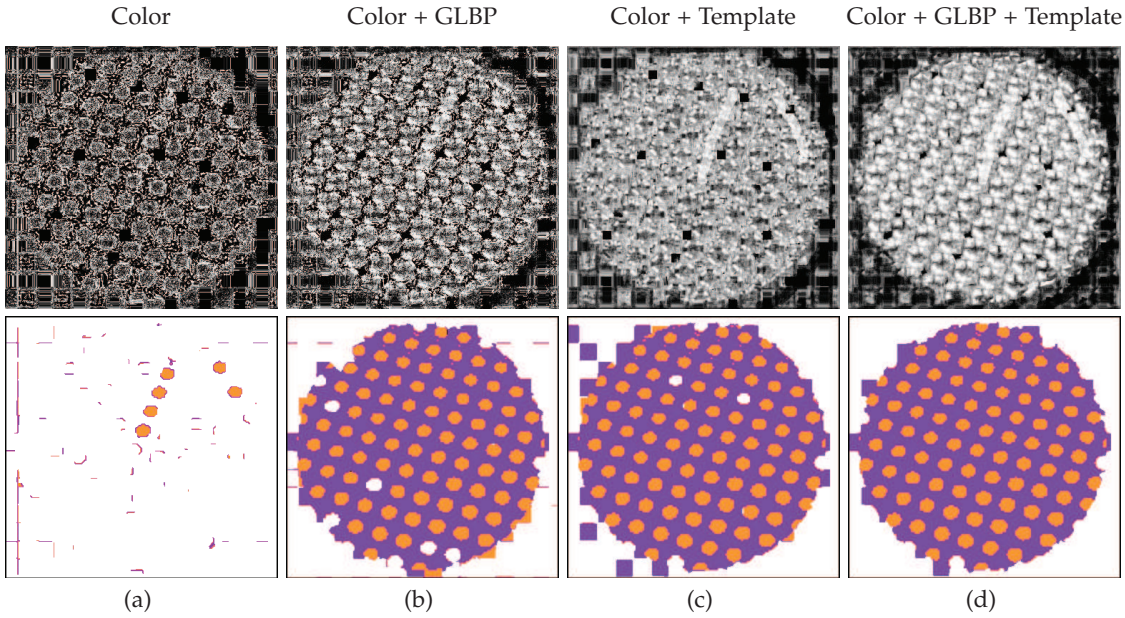


Figure 3.12: Color feature has negative influence on segmentation of the synthetic "Texture" image. First row illustrates probabilistic distribution maps obtained by RF classifier. Second row shows final segmentations in graph-cut framework without hard-constraints.

In conclusion, our proposed neighborhood template for feature modeling can enhance the discriminability of a feature descriptor allowing the classifier to predict classification results with high accuracy. Though neighborhood template was originally designed as a refined alternative of the block-histogram based LBP feature configuration, it also shows excellent generalizability to other local pixel-based properties like color information. In this section, we test our feature configuration through a representative example of the synthetic "Texture" image. The limitation of the example is obvious: texture feature is the only discriminant that can be used to train a classifier and perform the segmentation task in graph-cut based framework. However, for most real-world examples, feature selection is more complicated and it is often the case that we can benefit from the incorporation of multiple features.

In the following section, we will experiment on several natural images whose characteristics depend mainly on pixel color information as well as the texture feature.

3.4.2 Experiments on Natural Color Images

In this section, we perform two groups of experiments, (i) Graph-cut based segmentations using SVM and Random Forest classification learning different combinations of features and (ii) Comparison of color-feature based graph-cut segmentation methods, Boykov-Jolly's approach, Lazy Snapping algorithm and GrabCut algorithm, with our proposed multi-feature based approach, based on natural color images taken from the Berkeley Segmentation Dataset [35] and the MSRC GrabCut Segmentation Database [20]. Images taken from the former dataset are in size of 481×321 pixels while images from the latter are in size of 450×600 pixels. We divided the experimental objects into two image subsets: 40 images for training and 60 images for testing. Quantitative evaluation of segmentation performance on a natural color image is given by the calculation of percentage of mis-classified pixels compared to the corresponding gold truth (cf. Equation (2.45)).

General experimental settings are as follows:

- Feature: In Experiment I, we have 6 different types of feature configurations: (a) 1-dimensional color feature vector ("color"), (b) 4-dimensional LBP texture-color feature vector ("color+LBP"), (c) 4-dimensional GLBP texture-color feature vector ("color+GLBP"), (d) 75-dimensional color-shape feature vector ("color+Temp"), (e) 100-dimensional LBP texture-color-shape feature vector ("color+LBP+Temp") and (f) 100-dimensional GLBP texture-color-shape feature vector ("color+GLBP+Temp"). In Experiment II, only the 100-dimensional texture-color-shape feature vector is taken into consideration. For GLBP, parameter σ is 3 and the kernel size of the gaussian filter is 7×7. This ensures the possibility of GLBP texture feature encoding unpredictable texture patterns without losing too much detail.

- Classifier: SVM and RF classifier are both adopted and the parameters of the classifiers are estimated using cross-validation method. For SVM, kernel function is GRBF, with its parameter γ fixed at 0.5; the soft-margin parameter C is 10. For Random Forest, we train a RF classifier with 100 trees of depth 25.

- Energy function: Parameter λ is selected as 40 by optimizing the performance against ground truth in the training set of 40 images. Other settings of parameters particular for Boykov-Jolly approach, Lazy Snapping and GrabCut are detailed in Section 2.4.4.

- Graph: A 8-neighborhood adjacent directed graph is constructed with two terminal nodes s and t, respectively connected to the "foreground" seeds and "back-

ground" seeds. By satisfying the user-specified hard constraints, a minimum cut on a graph can be found with edge weights appropriately defined can to generate an optimal segmentation. The edge weights of the graph are given in Table 3.3.

Table 3.3: Edges weights for maximum *a posteriori* estimation.

Edge	Weight	For
$w_{s,p}$	$-\log \Pr(-1\|\mathbf{x}_p)$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	$+\infty$	$p \in \mathcal{O}$
	0	$p \in \mathcal{B}$
$w_{p,t}$	$-\log \Pr(+1\|\mathbf{x}_p)$	$p \in \mathcal{V} \setminus \{\mathcal{O} \cup \mathcal{B}\}$
	0	$p \in \mathcal{O}$
	$+\infty$	$p \in \mathcal{B}$
$w_{p,q}$	$\lambda \cdot \frac{\lvert I_p - I_q \rvert}{\lvert\lvert \mathbf{x}_p - \mathbf{x}_q \rvert\rvert^2 + 1} \cdot \frac{1}{dist(p,q)}$	$\{p,q\} \in \mathcal{N}$

Figure 3.13 and Figure 3.14 show the great effectiveness of the color-texture-shape feature configuration on segmentation of 8 representative sample images. By comparing correspondingly (e) with (b), (f) with (c), and (g) with (d) in Figure 3.13 and Figure 3.14, we find that neighborhood template can improve the segmentation performance by enhancing the discrimination of both color and texture features. It allows the LBP code to capture representative texture patterns and helps to gather additional color information from the neighborhood pixels. Enhancement of performance is especially obvious in regions where the color information is less discriminative for the distinction of foreground and background. For instance, hip-leg part in "Leopard", cross-church part in "Cross", gravestones in "Grave", shadow in "Plane", body-precipice part in "Sheep" and plash in "Boat". In addition, by comparing (f) and (g) with (e) in Figure 3.13 and Figure 3.14, we can find that by adding the texture information to the feature configuration, our proposed approach is robust in cases that classifiers trained with color information alone fail to generate distinct soft classifications. Promising segmentations can still be obtained in some difficult cases where foreground/background color profiles are similar (examples are mentioned before), with multi-feature configuration under the proposed segmentation framework. Note that LBP texture feature with Gaussian blurring (GLBP) does not always guarantee higher segmentation accuracy: except for "Leopard" and "Sheep", where strong large-scale texture features like "leopard-print" are excepted to be recognized by GLBP, in other images like "Cross" and "Grave", small-scale or obscure texture structures may be ignored by Gaussian-blurring preprocessing. Table 3.4 states the corresponding segmentation error rates (cf. Equation (2.45) for its definition) subject to different feature configurations on 8 natural color images shown in Figure 3.13 and Figure 3.14. Quantitative evaluation listed in Table 3.4 verifies that feature configuration

combining texture, color and shape information together leads superior segmentation performance. Also, we notice that RF classifier shows competitive learning ability as SVM. In some cases, for example, segmentations on "Cross" and "Grave", RF outperforms SVM in considerably decrease of segmentation error.

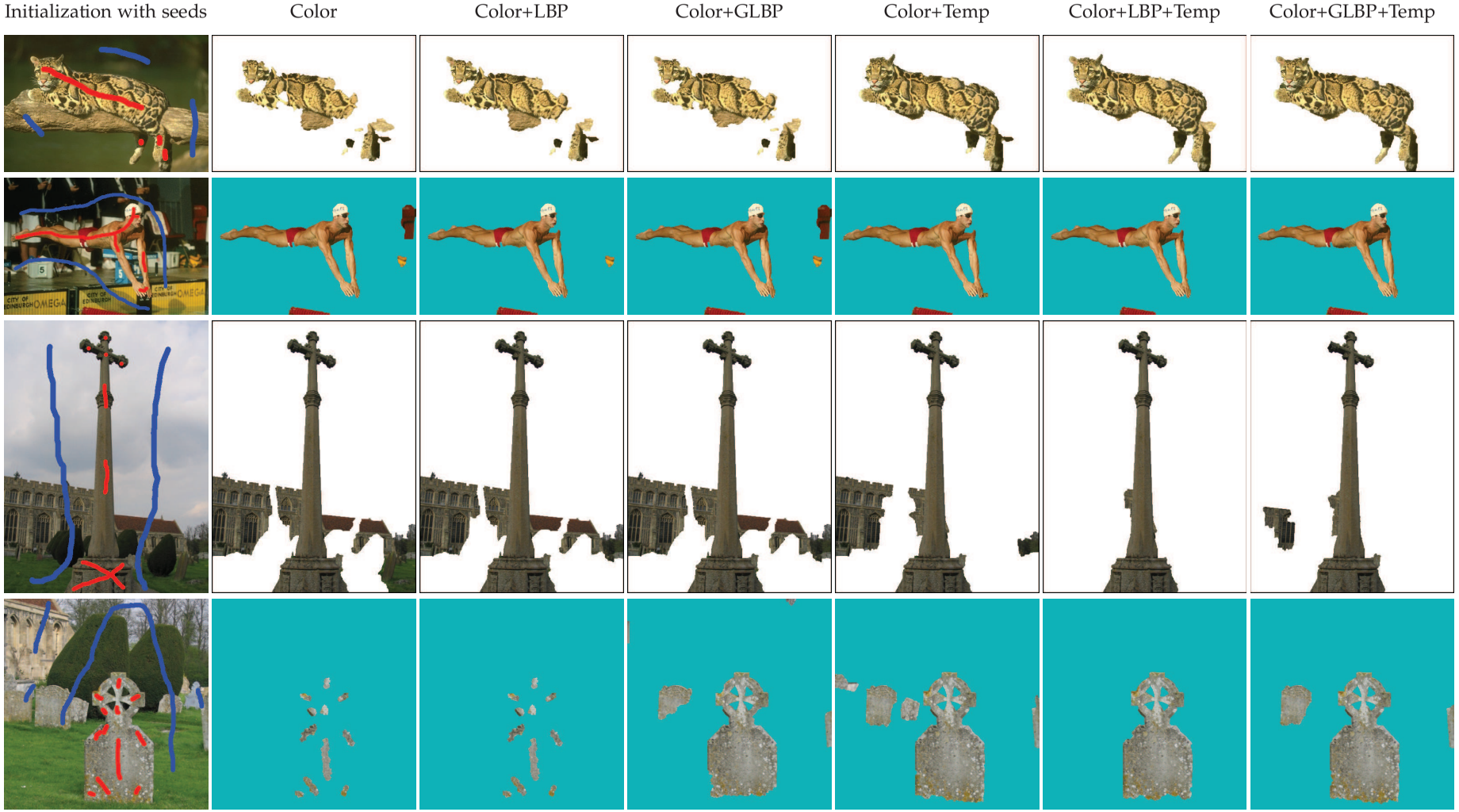


Figure 3.13: Quantitative comparison of different feature combinations on segmentation of natural images from the Berkeley Segmentation Dataset [35] and the MSRC GrabCut Segmentation Database [20].

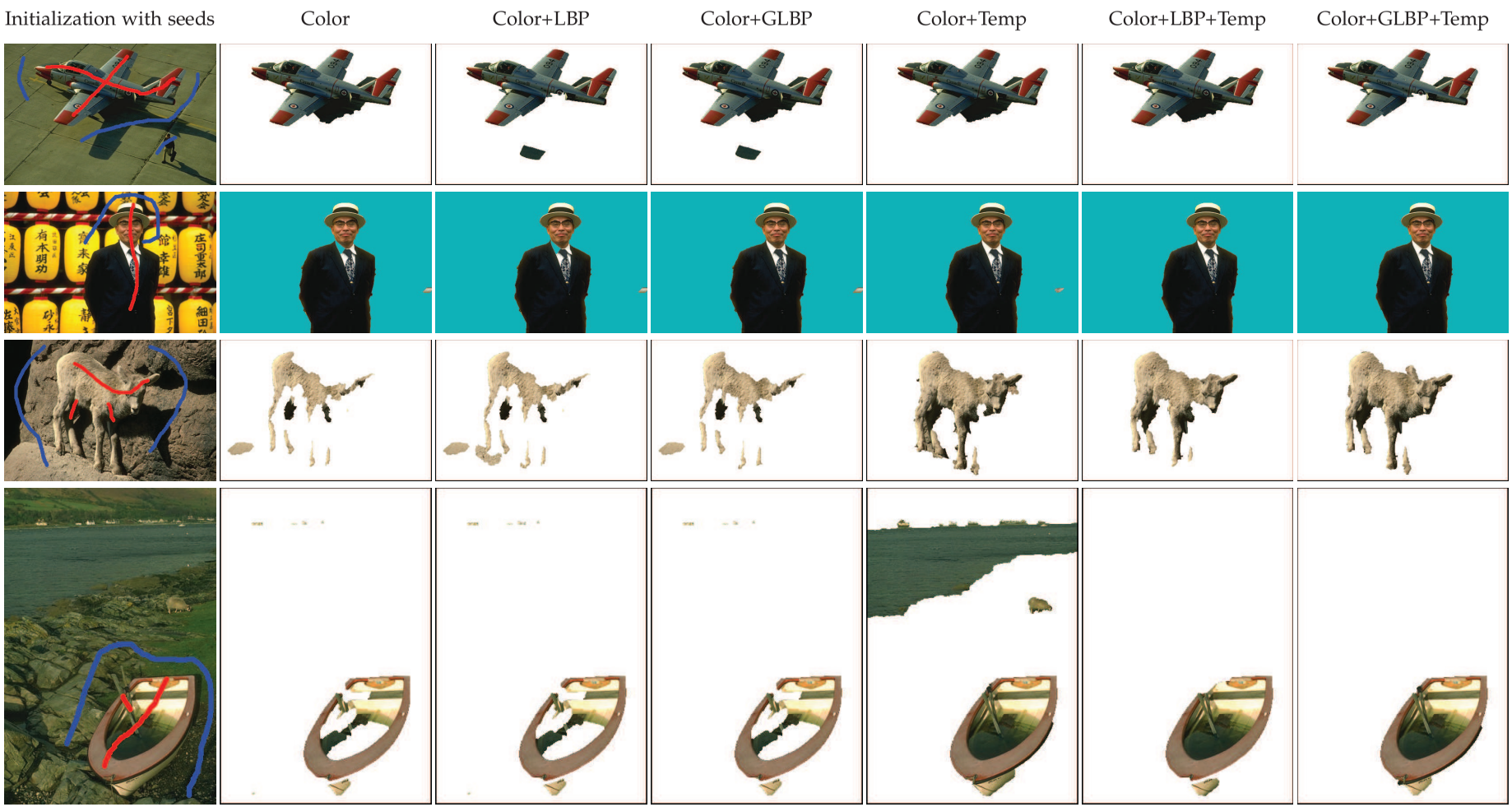


Figure 3.14: Quantitative comparison of different feature combinations on segmentation of natural images from the Berkeley Segmentation Dataset [35].

Table 3.4: Quantitative evaluation on segmentation of 8 natural images from the Berkeley Segmentation Dataset [35] and the MSRC GrabCut Segmentation Database [20], subject to different feature combination learned by SVM and RF respectively. Error rate ϵ, showing the performance of each evaluated approach, is given by the percent of mis-classified pixels compared to the corresponding gold truth (second columns in Figure 3.15 and Figure 3.16). Value in bold corresponds to the best performance. Cell in color corresponds to the best performance on a particular image.

Input image	Classifier \ Feature	Color	Color + LBP	Color + GLBP	Color+Temp	Color + LBP+Temp	Color + GLBP+Temp
Leopard	SVM	50.99	46.07	42.75	12.86	12.31	**11.84**
	RF	13.32	14.40	14.15	16.17	**13.40**	15.60
Swimmer	SVM	27.24	17.76	26.98	15.40	14.62	**13.10**
	RF	14.17	14.23	**13.99**	20.29	20.94	16.58
Cross	SVM	127.78	107.92	109.20	70.59	**6.53**	23.23
	RF	15.43	12.90	12.50	4.00	**3.99**	4.04
Grave	SVM	84.92	84.91	27.54	33.94	**8.40**	23.48
	RF	7.28	**7.11**	6.67	8.39	8.29	7.61
Plane	SVM	14.60	14.65	14.53	**5.44**	10.19	10.21
	RF	18.97	11.32	22.48	18.73	18.48	**6.78**
Japanese	SVM	17.53	10.08	12.25	**1.66**	2.85	2.82
	RF	2.89	2.87	2.13	1.94	1.63	**1.52**
Sheep	SVM	62.92	66.40	57.63	25.88	27.87	**20.66**
	RF	55.92	48.65	44.64	47.23	45.36	**29.56**
Boat	SVM	36.86	36.84	36.84	144.98	15.11	**9.64**
	RF	**10.65**	10.76	10.84	12.40	12.37	12.40

Based on the results obtained in Experiment I, we decide to compare our proposed approach using 100-dimensional texture-color-shape feature vector (except for the synthetic "Texture" image, we do not add color information into the feature configuration, i.e., 25-dimensional texture-shape feature vector) with other approaches that model color distributions solely: classical graph-cut segmentation method proposed by Boykov and Jolly (Boykov-Jolly), the Lazy Snapping algorithm and the GrabCut algorithm. The segmentations are shown in Figure 3.15 and Figure 3.16 and the corresponding segmentation error rates are listed in Table 3.5. Our proposed approach shows superior performance on segmentation of both artificial and natural data.

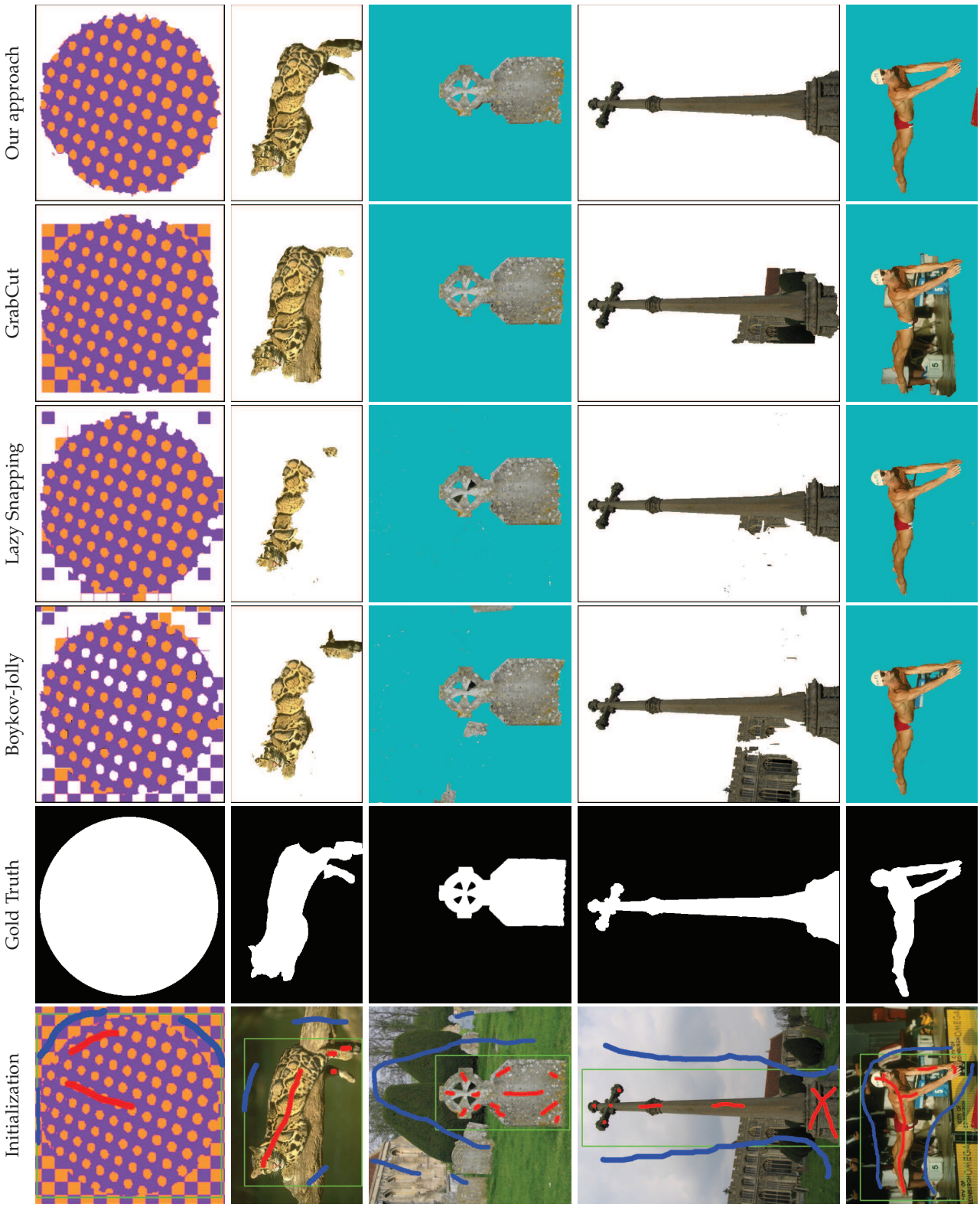


Figure 3.15: Quantitative comparison of our approach against other color-feature-based graph-cut segmentation approaches.

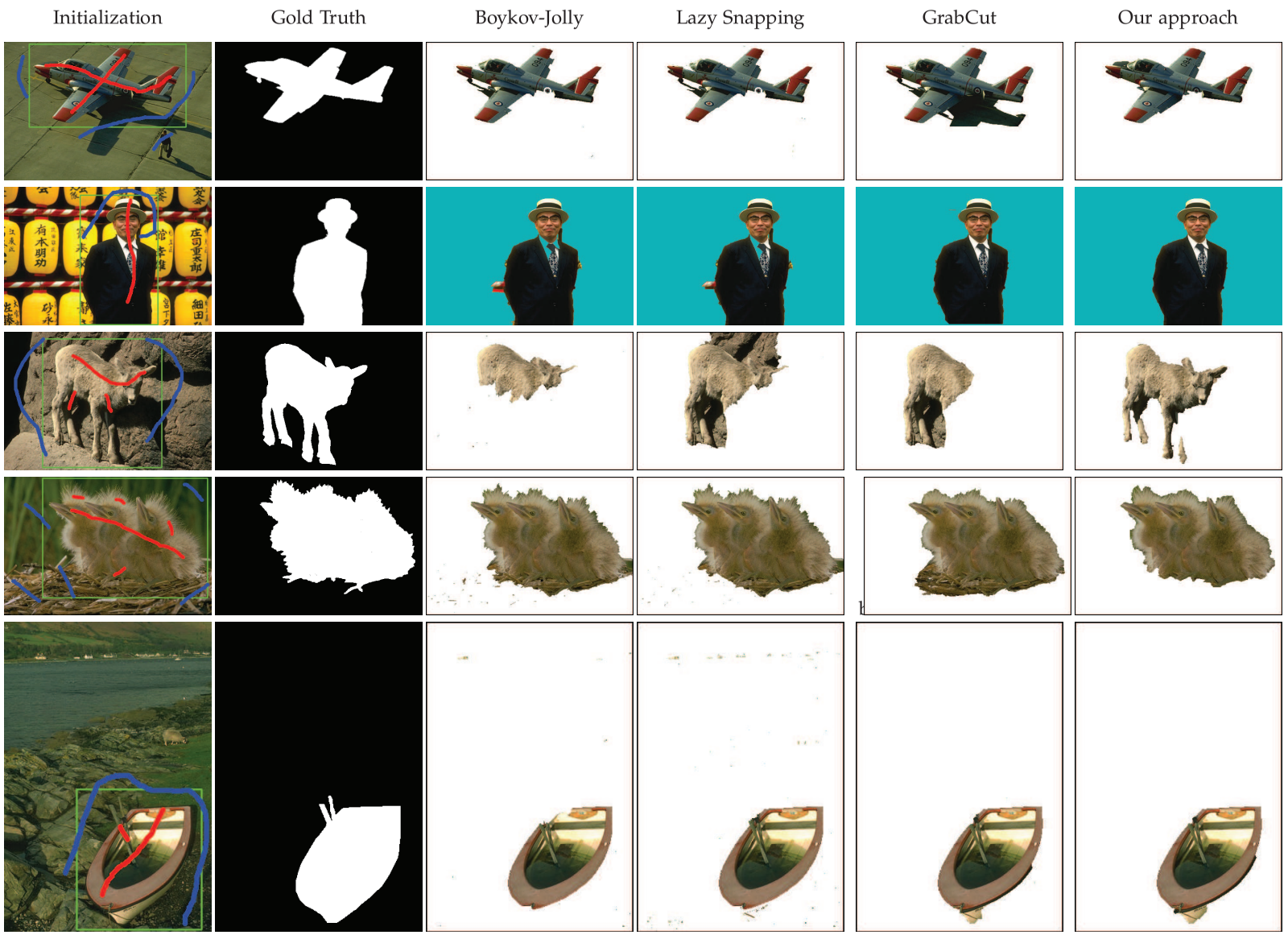


Figure 3.16: Quantitative comparison of our approach against other color-feature-based graph-cut segmentation approaches.

Table 3.5: Quantitative evaluation of Boykov-Jolly, Lazy Snapping, GrabCut and our approach on synthetic texture image and natural color images taken from Berkeley Segmentation Dataset [35] and the MSRC GrabCut Segmentation Database [20]. Value in bold corresponds to the best performance.

Error ϵ (%) Method / Input image	Boykov-Jolly	Lazy Snapping	GrabCut	Our approach
Texture	27.46	11.65	19.46	**3.81**
Leopard	37.22	55.95	47.16	**11.84**
Grave	20.63	10.89	**5.15**	7.61
Cross	75.81	18.43	57.82	**3.99**
Swimmer	16.25	**8.02**	165.20	13.10
Plane	15.59	19.00	38.10	**6.78**
Japanese	6.33	5.29	3.58	**1.52**
Sheep	55.23	70.97	51.43	**20.66**
Birds	19.05	18.64	25.50	**10.82**
Boat	18.60	17.17	10.58	**9.64**

Chapter 4

Application to Melanoma Detection

Contents

Melanoma, named after its high concentration of melanin, is a malignant tumor of melanocytes. Melanocytes are cells that produce a dark chromophore, melanin, which is primarily responsible for the skin color. Melanoma predominantly occurs in skin, but also can be found in other parts of human body that contain melanocytes (i.e. bowel, oral cavity, eye, etc.). As a skin cancer, melanoma is a less frequent type compared to the other types like Basal cell carcinoma and Squamous cell carcinoma [4]. However, it is more likely to spread and most fatal (it causes 75 percent of deaths related to skin cancer [9]) unless diagnosed in the early stage. People who have large numbers of freckles or moles, fair skins with blue eyes, high tendency of sunburns (all these physical signs are particularly common among Caucasians, especially northwestern Europeans living under long-term sunlight exposure) are considered more risky in suffering from melanoma.

Early detection of malignant melanoma when it remains in the epidermis layer of skin can result almost 100 percent of cure rate. Clinical diagnosis of malignant melanoma is mainly based on two different discriminating systems: the ABCDE criteria [77] and the 7-point checklist [10]. For the former system, signs and symptoms of melanoma are summarized by the acronym "ABCDE", which stands for **A**symmetry,

Border irregularity, **C**olor diversification, **D**iameter greater than 6mm and **E**volving over time. For the later system, it provides a scoring diagnostic method for low number of features based on shape, color and texture to conduct simplified epiluminescence microscopy (ELM) pattern analysis.

In dermatologic practice and clinical research, diagnosis of melanoma with human visual inspection is often extremely complicated, thus time-consuming, and somewhat subjective, thus less accurate, even for well-trained dermatologists [18]. In spite that human eyes have the ability to distinguish malignant melanoma from other melanocytic and non-melanocytic pigmented skin lesions (PSLs) when high-contrasting color, texture and other morphological features are observed, we are unable to precisely quantify our feature perception without instrumental means. To obviate the problem of complex and qualitative interpretation by human visual perception, there is of great interest to develop a computer-aided automatic melanoma diagnosis system that can assist the clinical evaluation of dermatologists. This system can be designed based on two subsystems: (a) imaging acquisition (i.e. dermoscopic images and spectroscopic images) on PSLs with surrounding normal skin and (b) PSL image analysis including image segmentation, feature extraction and selection, and PSLs classification. As a first step in melanoma identification, image segmentation is essential to localize the PSL and determine its boundary for the further measurements of features such as maximum diameter, irregular boundary, variegated colors.

4.1 Structure and Optical Property of Human Skin

Human skin is the outmost covering of body and the largest organ in terms of both weight and surface area. It has a very complex structure that consists of various types of components including veins, capillaries, hairs, cells, fibers, etc (as depicted in Figure 4.1). The coexistence of these components makes skin a multi-layered tissue composed of three primary layers: epidermis, dermis and hypodermis (also called Subcutaneous tissue). Each layer is distinct in both functionality and optical property.

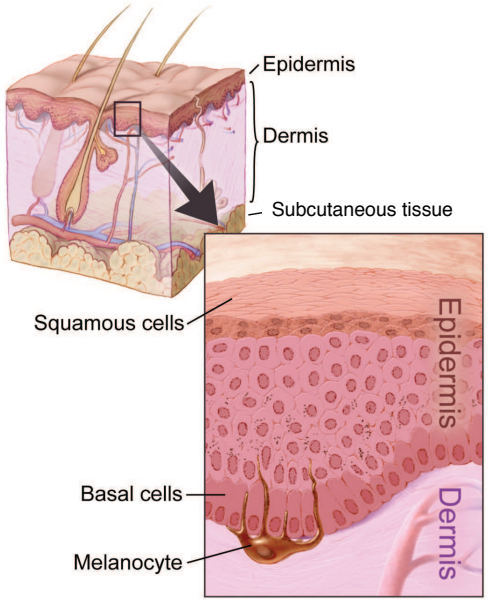


Figure 4.1: Anatomy of human skin. Three primary layers of skin (epidermis, dermis, and hypodermis) are illustrated as well as an inset with a close-up view of the types of cells in the skin (squamous cells, basal cells, and melanocytes). Courtesy of Don Bliss as illustrator of this image.

4.1.1 Skin Chromophores

A chromophore is the part of a molecule that has the ability to absorb light at a certain wavelength with a characteristic efficiency given by the molecular extinction coefficient. Among various types of chromophores in human skin, melanin and hemoglobin (including *oxy*-hemoglobin and *deoxy*-hemoglobin) are most important for understanding the skin color since they absorb light particularly in the visible wavelength range [7, 98].

4.1.1.1 Melanin

The primary determinant of variability in human skin color is the concentration and distribution of the chromophore melanin. Melanin is produced by melanocytes, which can be found in the basal layer of the epidermis (cf. Figure 4.1). Basically, there are two common types of melanin, *eumelanin* and *pheomelanin*, depending on its the chemical structure. Eumelanin characterizes itself in black-to-dark-brown color and usually locates in black hair and retina of eye, while pheomelanin is a yellow-to-reddish-brown chromophore that can be found in red hair and feather. Since eumelanin is the most abundant melanin in human skin, we will refer to the term "melanin" as "eumelanin".

4.1.1.2 Hemoglobin

Hemoglobin, the primary protein constituent of erythrocyte (cf. Figure 4.2), is responsible for transport oxygen throughout body via vessels and capillaries. In general, hemoglobin can be saturated with oxygen molecules (oxy-hemoglobin), or desaturated with oxygen molecules (deoxy-hemoglobin). It mainly distributes in the superficial vascular plexus of the skin (cf. Figure 4.3 (a)) and more than 47% of the hemoglobin is oxy-hemoglobin [8]. Oxy-hemoglobin has a reddish hue while deoxy-hemoglobin has a purplish color.

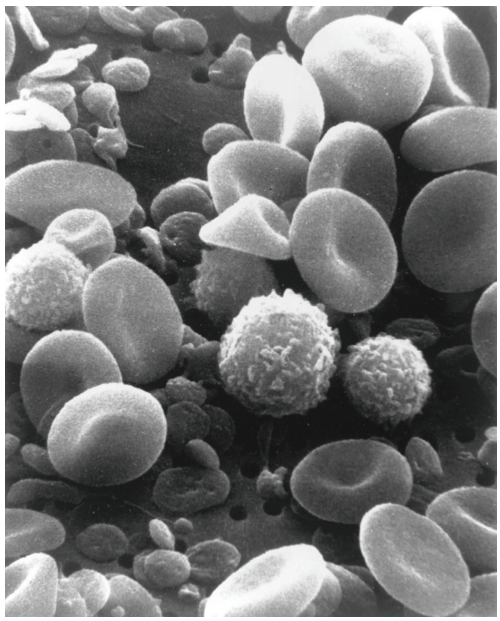


Figure 4.2: A scanning electron microscope image of erythrocytes (red blood cells) from normal circulated human blood. Beside biconcave erythrocytes, this photo also shows small amount of globular leukocytes (white blood cells) and disc-shaped platelets (thrombocytes). Courtesy of Bruce Wetzel and Harry Schaefer as two photographers of this image.

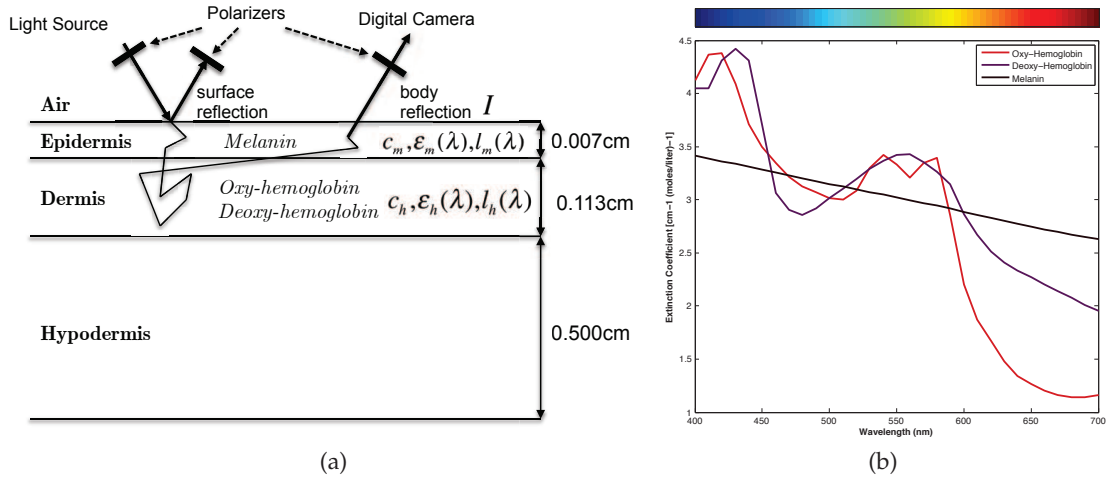
4.1.2 Interaction of Light with Skin

Figure 4.3 (a) shows the schematic model of imaging process of three layered model of skin. Three predominant chromophores found in epidermal and dermal layers are melanin (abbreviated as Mel) and oxy-hemoglobin (abbreviated as HbO2) and deoxy-hemoglobin (abbreviated as Hb) [7]. Based on the *Beer-Lambert law*, the absorbance of this skin model at a wavelength λ can be expressed as

$$\begin{aligned} A(\lambda) &= \log(1/R(\lambda)) \\ &= \epsilon_{\mathrm{HbO_2}}(\lambda) l_{\mathrm{HbO_2}}(\lambda) c_{\mathrm{HbO_2}} + \epsilon_{\mathrm{Hb}}(\lambda) l_{\mathrm{Hb}}(\lambda) c_{\mathrm{Hb}} + \epsilon_{\mathrm{Mel}}(\lambda) l_{\mathrm{Mel}}(\lambda) c_{\mathrm{Mel}} \end{aligned} \tag{4.1}$$

where l is the light penetration depth, c denotes the concentration of the chromophore and ϵ is the extinction coefficient that depends on absorbance spectrum of the chromophore (cf. Figure 4.3 (b) plotted in logarithmic scale).

The extinction coefficient of hemoglobin has local maxima between 542 nm and 577 nm, which provides a convenient wavelength region for the quantification of hemoglobin. The extinction coefficient of melanin has no characteristic maximum in the visible region but demonstrates a monotonic decrease towards larger wavelengths. Particularly, in red region of the spectrum ($>$ 600 nm), the molar absorptivity of melanin is more prominent compared with the other chromophores. Hence, the red region can be used for melanin quantification.



(a) (b)

Figure 4.3: Optics of human skin. (a) Schematic of optical pathway in a 3-layered skin model (scattering is omitted). (b) Estimation of relative molar absorptivity spectra of three predominant chromophores, oxy-hemoglobin, deoxy-hemoglobin and melanin, plotted in logarithmic scale.

4.2 Skin Chromophore Extraction in Different Color Spaces

A variety of familiar cutaneous phenomena, such as tanning after a sunbath, reddening after alcohol assumption are always responses to the quantitative variations of two primitive skin chromophores, epidermal melanin and dermal hemoglobin. Also, many skin diseases (e.g. melanoma) are induced by absence or overmuch of these two chromophores. This makes it necessary and valuable to estimate melanin/hemoglobin content distributions objectively. To avoid ambiguity, the estimation of melanin/hemoglobin content in terms of intuitionistic distribution map is named "skin chromophore extraction" throughout this thesis.

Color is easier to perceive than reflectance spectrum, however, difficult to quantitate objectively. Color spaces, or color models (i.e. RGB, $L^*a^*b^*$, HSV, etc.), provide us a mathematical and geometrical 3-dimensional representation of color information thus allow quantitative measurement of perception of skin color to estimate skin chromophore content.

In this section, we mainly focus on skin chromophore extraction in different color spaces. The primary steps for chromophore content estimation in a skin color image using color information are: (i) to represent image pixels in a suitable color space, (ii) to model the melanin and hemoglobin components using a suitable distribution, (iii) to quantitate the modeled distribution. There exists several relative researches presented in recent literature, including but not limited to:

- Skin pigmentation quantification in $L^*a^*b^*$ space. Chardon et al. [33] proposed a vector presentation for UV-induced tanning reaction and Jung et al. [54] proposed to evaluate erythema and melanin content in port wine stain lesions in $L^*a^*b^*$ space. Three indices of $L^*a^*b^*$ system indicate close but dependent correlation between melanin/hemoglobin contents. Moreover, $L^*a^*b^*$ system has been developed to approximate color perception and not to extract accurate physiological information.

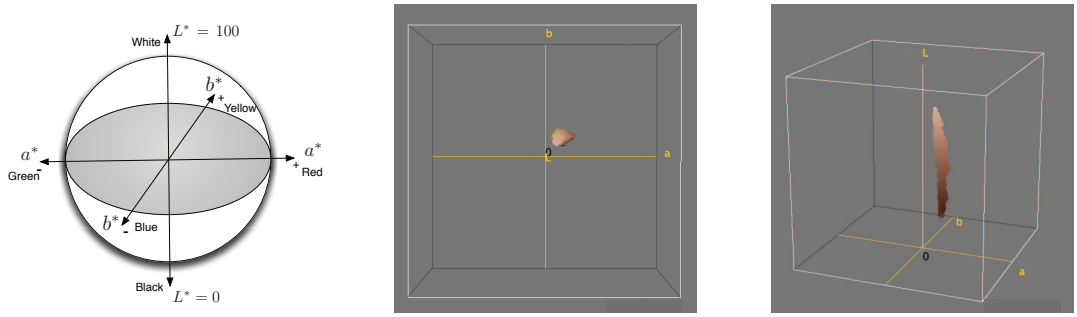
- Skin chromophore decomposition in HSV space. Intuitive color spaces, like HSV, allow convenient measurement of skin color. Kim and Kim [56] proposed a method for skin chromophore decomposition in HSV color space from a single digital image. Though cone-shaped model and radially sorted hue values in HSV color space result in hemoglobin/melanin maps closer to the real distribution of the human subject, it is just a description rather than a true quantitative science without any relation to the optical and biological properties of the skin.

- Separation of spatial distributions of melanin and hemoglobin in RGB space. Principal component analysis (PCA) and independent component analysis (ICA) are applied to extract chromophore distribution in logarithmic RGB space (optical density space) by Tsumura et al. [91]. However, the direction of the color plane extracted by PCA varies as the sample skin region (with different illumination condition) changes. Therefore, their approach fails in producing absolute quantity of the pigments from different samples of a human subject; rather, it produces relative quantities varying from one sample to the other.

To overcome the deficiency of Tsumura's approach in extracting false chromophore distribution of skin sample under uneven illumination condition, we propose to model skin chromophore distribution by fitting a quadratic surface in logarithmic normalized RGB space and flattening it onto a 2-dimensional color plane. Qualitative evaluation based on dermatologic knowledge via comparative experiments indicates that our proposed method extracts more accurate chromophore distribution than other existing approaches.

4.2.1 L*a*b* (CIELAB) Color Space

CIE $L^*a^*b^*$ is the most complete color space specified by the Commission Internationale d'Éclairage (CIE). Unlike the RGB color space, it is designed to represent perceptual uniformity, which is similar to the color sensitivity of human vision system. It consists of three coordinate axes: (i) L^*-axis represents "lightness", which measures the reflected light intensity. Scale of L^* value ranges from 0 indicating the theoretical black to 100 indicating the perfect white; (ii) a^*-axis represents "chromaticity", which measures the color saturation varying from -60 for "green" to +60 for "red"; (iii) b^*-axis also describes "chromaticity" and varies from -60 for "blue" to +60 for "yellow". Three axes are orthogonal to one another (cf. Figure 4.4 (a)).



(a) $L^*a^*b^*$ color space (b) Top view (c) Side view

Figure 4.4: Skin color distribution in $L^*a^*b^*$ color space.

To convert RGB color space to $L^*a^*b^*$ color space, RGB tristimulus coordinates are first converted into device-independent CIE XYZ tristimulus coordinates by a linear transformation:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{4.2}$$

There are numbers of CIE spaces that can be created based on tristimulus values, X, Y and Z, including $L^*a^*b^*$ color space. By nonlinear transformation, $L^*a^*b^*$ system can be defined as:

$$\begin{aligned} L^* &= 116 \times f(Y/Y_n) - 16 \\ a^* &= 500 \times [f(X/X_n) - f(Y/Y_n)] \\ b^* &= 200 \times [f(Y/Y_n) - f(Z/Z_n)] \end{aligned} \tag{4.3}$$

where $f(\phi) = \phi^{1/3}$, for $\phi > 0.008856$ and (X_n, Y_n, Z_n) are X, Y, Z values for the standard white reference.

After conversion from RGB color space into $L^*a^*b^*$ color space, all skin colors fall within a 'banana' shaped volume (cf. Figure 4.4 (b,c)). Chardon et al. [33] proposed to quantitate skin pigmentation based on the experimental fact that skin reddening (erythema) can be represented as a shift on the $L^* - a^*$ whereas variation of melanin content can be represented as a shift on the $L^* - b^*$ plane. The "individual typology angle" or "alpha characteristic angle" α was calculated independently to measure the melanin content:

$$\alpha = \left[\arctan\left(\frac{L^* - 50}{b^*}\right)\right] \times \frac{180}{\pi} \tag{4.4}$$

This angular parameter is inversely proportional to the melanin content and has been empirically shown to correlate well with the color appearance of skin.

In context of characterization of port wine stain (PWS) skin erythema and melanin contents, Jung et al. [54] utilized cross-polarized skin color image to calculate the $L^*a^*b^*$ indices images, where a^* index image was used as an indicator of erythema distribution and the L^* index image the inverse of melanin distribution.

According to the contributions of Chardon and Jung, we propose to measure skin erythema and melanin contents utilizing a^* index image and the combination of L^* and b^* indices images to derive the erythema/melanin distribution maps based on digital

camera images:

$$\begin{aligned} \textbf{Erythema} &= \mathbf{a}^* \\ \textbf{Melanin} &= \frac{\mathbf{b}^*}{\mathbf{L}^*} \end{aligned} \tag{4.5}$$

For melanin content measurement, combination of L^* and b^* indices seems to be a more accurate parameter than L^* or b^* index alone. Since in some particular cases, for example, during application of sub-diastolic pressure with a pressure cuff, Stamatas et al. [85] have shown that increase of hemoglobin content can result in synchronous decrease of values of L^* and b^*.

Except for the digital imaging based approaches mentioned above, there exists several of tristimulus reflectance colorimeters, for example, Chromameter CR 200™(Minolta, Osaka, Japan) and Photovolt™(UMM Electronics, Indianapolis, IN, USA) for commercial use of calculation of $L^*a^*b^*$ values. These contact-instrument based approaches certainly improve the accuracy of skin color measurement in $L^*a^*b^*$ system (X, Y and Z tristimulus values are determined directly from the narrow-band reflectance data, instead of device-dependent RGB tristimulus values of a digital camera image), however, they are limited by (i) small measurement area (maximum 10 mm diameters), (ii) no resolution of spatial resolution and (iii) possible artifacts (e.g. skin blanching or stasis) induced by probe contact pressure.

It is worth noting that neither digital imaging approach nor reflectance colorimetric approach is able to separate individual contributions of chromophores responsible for skin color with high accuracy in $L^*a^*b^*$ system. Changes in content of hemoglobin or melanin can influence all three indices. For instance, Shriver et al. [83] have shown that a^* value can be affected by epidermal melanin content. With high melanin content, the a^* value is not related solely to hemoglobin content due to high absorption of green light by melanin.

4.2.2 RGB Color Space

RGB color space is the most commonly used model for for the sensing, representation, and display of images in electronic systems, such as televisions and computers. It is suitable for color display, however, characteristics of device-dependency and high correlation between channels, make RGB space not suitable for color analysis. Tsumura et al. [91] proposed to extract hemoglobin and melanin information by principal component analysis (PCA) and independent component analysis (ICA) in the logarithmic RGB space (optical density domain) assuming the linearity between contents of chromophores and

observed color signals.

The Tsumura's method to extract hemoglobin and melanin components from a single skin color image is briefly reviewed in this section. As mentioned in Section 4.1, skin is a turbid media with multiple layers. Among various chromophores, melanin and hemoglobin are dominantly contained in the epidermal and dermal layer, respectively. On the basis of the four assumptions made by Tsumura and the Lambert-Beer law [47], skin color can be modeled as a linear combination of the densities of melanin and hemoglobin in the optical density domain of three channels: $-\log(R_{l,m})$, $-\log(G_{l,m})$ and $-\log(B_{l,m})$ (cf. Figure 4.5), where $R_{l,m}$, $G_{l,m}$ and $B_{l,m}$ denote the tri-channel pixel values of the skin color image on the image coordinate (l, m), respectively. It is seen that three densities of skin color are distributed on the two-dimensional plane spanned by two pure color vectors indicating melanin and hemoglobin. The color density vector $\mathbf{c}_{l,m}$ is denoted as

$$\mathbf{c}_{l,m} = [-\log(R_{l,m}), -\log(G_{l,m}), -\log(B_{l,m})]^\top \tag{4.6}$$

According to the skin color model shown in Figure 4.5, the color density vector of skin can be expressed by

$$\mathbf{c}_{l,m} = \mathbf{C}\mathbf{q}_{l,m} + \mathbf{c}(3) \tag{4.7}$$

where $\mathbf{C} = [\mathbf{c}(1), \mathbf{c}(2)]$, $\mathbf{q}_{l,m} = [q_{l,m}(1), q_{l,m}(2)]^\top$. $\mathbf{c}(1)$ and $\mathbf{c}(2)$ denote density vectors of melanin and hemoglobin, $q_{l,m}(1)$ and $q_{l,m}(2)$ are relative quantities of the two chromophores, $\mathbf{c}(3)$ is spatially stationary vector cause by other chromophore and skin structure (i.e. β-carotene and thickness of hypodermis). From Equation (4.7), skin color can be considered as a mixed signal of two independent signals. PCA and ICA are then applied to extract the 2-dimensional plane spanned by $\mathbf{c}(1)$ and $\mathbf{c}(2)$ (note that two principle components $\mathbf{p}(1)$ and $\mathbf{p}(2)$ extracted by PCA will determine the 2-dimension plane spanned by $\mathbf{c}(1)$ and $\mathbf{c}(2)$, as shown in Figure 4.5) and to estimate the quantity vector $\mathbf{q}_{l,m}$, respectively, for getting melanin distribution map and hemoglobin distribution map.

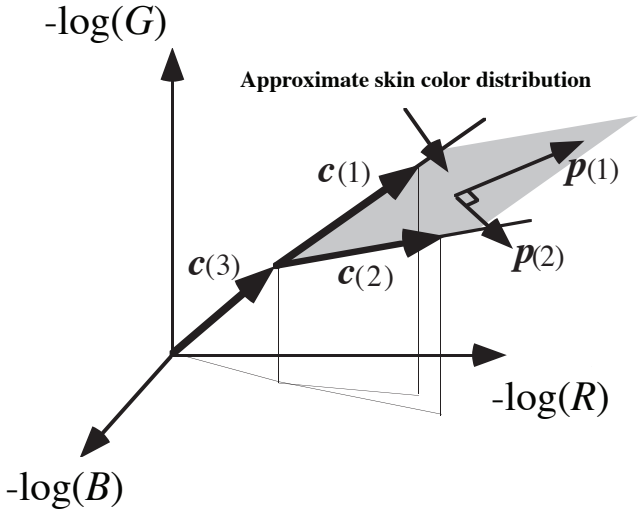
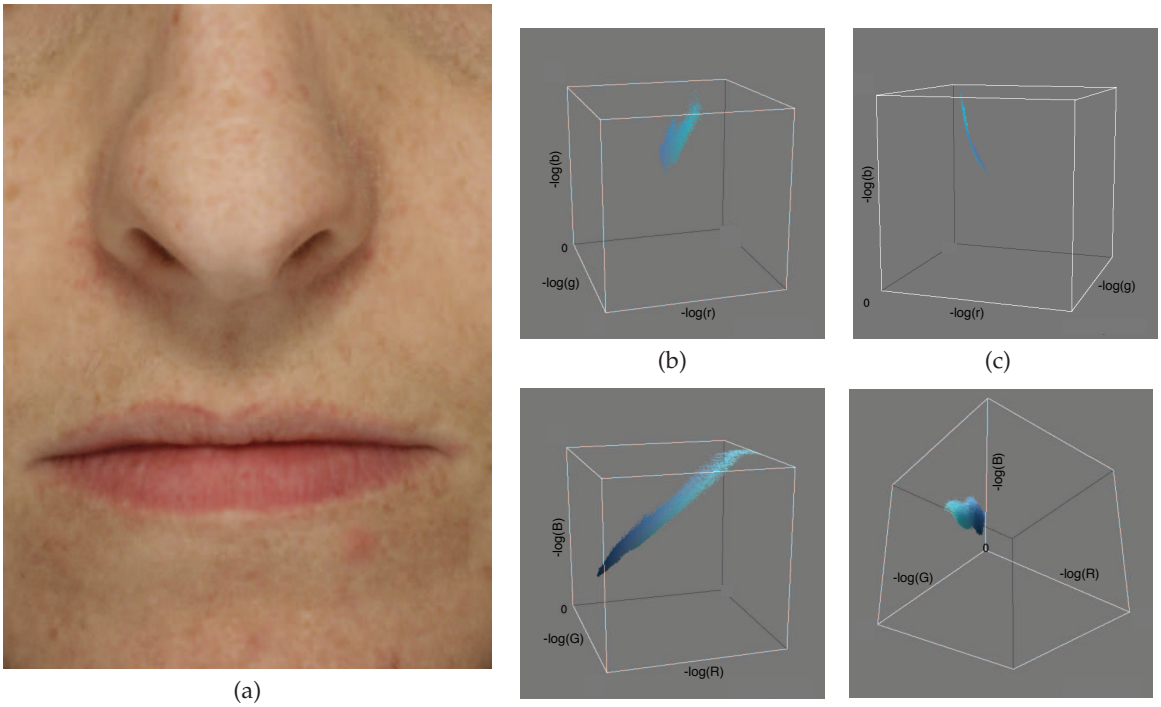


Figure 4.5: A schematic of skin chromophore extraction method proposed by Tsumura et al. [91].

In spite that Tsumura's approach reviewed previously was able to extract relative accurate hemoglobin and melanin distributions on the basis of skin optical property and statistical analysis, its applications are restricted to small region of skin samples (64×64 pixels) where the chromaticity and illumination condition roughly stay invariant. A 2-dimensional color plane approximating the distribution of sampled skin color in optical density domain can be extracted by PCA with high accuracy (99.3%). However, position and direction of the color plane varies according to the changes of local chromaticity and illumination of sampled skin region. For example, if PCA is applied to skin color influenced by shading effect under nose (cf. Figure 4.6 (a)), the extracted color plane will be rotated to the direction of illumination vector $\boldsymbol{1} = [1,1,1]^\top$, which can result false estimation of skin chromophores (cf. Figure 4.12 (a, b)). To eliminate the shading effect, Tsumura et al. [92] proposed a projection technique for shading removal along the illumination vector $\boldsymbol{1}$ and showed promising results particularly in shading skin part. Nevertheless, when the normal of the color plane is tilted largely from the illumination direction, projected color values can be out of the optical density cube. Thus the user has to give different region for the sampling (in the literature, a slide window with size of 100×100 pixels was adopted for sampling task). In addition, Tsumura's approaches have not mentioned how to determine which separation vector corresponds to melanin component and which corresponds to hemoglobin component.

In order to extract hemoglobin and melanin components from a single large-region facial image with complex chromaticity and illumination (cf. Figure 4.6 (a)), it is crucial to model the skin color distribution in a more suitable system as well as intelligent



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

Figure 4.6: Color distribution of (a) large-region facial image of 454×619 pixels in: (b, c) logarithmic Nrgb color space and (e, d) optical density domain.

manner. As we can observe in Figure 4.6 (d, e) that distribution of skin color from large-region facial image no longer lies on a approximate 2-dimension plane, instead, it falls into a complex 3-dimensional volume. In this case, dimensionality reduction by PCA fails in preserving the true structure of the data. Inspired by normalized RGB color space, which is independent of image luminance, we propose to combine optical density with normalized RGB color space to represent skin color so that the new color system holds linearity between quantities of skin chromophores and observed color intensities while being robust to the variation of illumination.

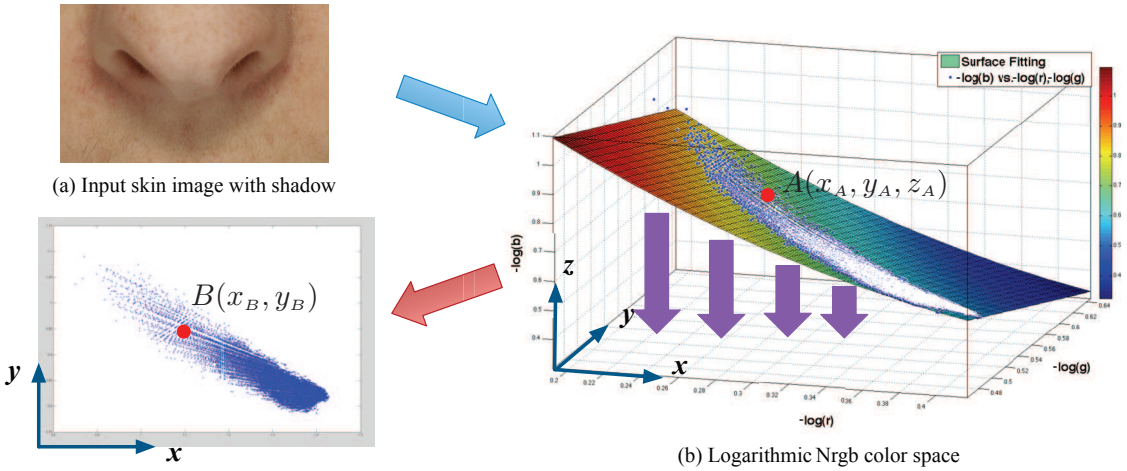
4.2.2.1 Normalized RGB color space

An effective and simple way to eliminate the distortion of skin color image caused by uneven illumination condition (e.g. highlight and shadow) is to get the variations of intensities uniformly across the spectral distribution. The normalized RGB (Nrgb) color space is formulated as:

$$\begin{cases} r = R/(R+G+B) \\ b = B/(R+G+B) \\ g = G/(R+G+B) \end{cases} \tag{4.8}$$

This representation of RGB tristimulus values reduces the sensitivity of the distribution to the color variability by normalization, however, loss of information due to the nonlinear transformation from RGB color space to Nrgb color space leads to noise under low intensity.

Skin color distribution of large-region facial image (cf. Figure 4.6 (a)) in the logarithmic Nrgb color space is demonstrated in Figure 4.6 (b, c). Color information of all pixels of the sample image lies approximately on a 3-dimensional quadratic surface, therefore, it is nature to fit this surface by regression of a second-order polynomial model and flatten (project) it along a certain direction to get a 2-dimensional color plane where skin color distribution can be separated subsequently into two components, hemoglobin and melanin, via independent component analysis (ICA). The whole proposed procedure of skin color distribution modeling by means of *surface fitting and flattening* (cf. Figure 4.7) is denoted by "SF^2 method".



(a) Input skin image with shadow

(b) Logarithmic Nrgb color space

(c) Flattened 2-dimensional plane

Figure 4.7: A schematic of our proposed SF^2 approach based on Tsumura's method.

For simplicity, we firstly formulate the quadratic surface by a quadratic polynomial model defined as:

$$z_i = c_1 x_i^2 + c_2 y_i^2 + c_3 x_i + c_4 y_i + c_5, \quad i = 1, 2, \cdots, n \tag{4.9}$$

which can be compactly written in matrix form as:

$$\mathbf{z}_{n\times 1} = \begin{bmatrix} \mathbf{x}\circ\mathbf{x} & \mathbf{y}\circ\mathbf{y} & \mathbf{x} & \mathbf{y} & \boldsymbol{1} \end{bmatrix}_{n\times 5} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{bmatrix}_{5\times 1} \tag{4.10}$$

Also, as shown in Figure 4.7 (b), the assignments for three coordinate axes, x, y and z are $\log r$, $\log g$ and $\log b$, respectively. According to the system of linear equations in Equation (4.10), constant coefficients, $c_1, c_2, \cdots, c_5$, can be estimated through Moore-Penrose pseudoinverse that minimize the the Sum of Squared Error (SSE):

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \end{bmatrix}_{5\times 1} = \begin{bmatrix} (\log\mathbf{r})\circ(\log\mathbf{r}) & (\log\mathbf{g})\circ(\log\mathbf{g}) & (-\log\mathbf{r}) & (-\log\mathbf{g}) & \boldsymbol{1} \end{bmatrix}^{+}_{5\times n} (-\log\mathbf{g})_{n\times 1} \tag{4.11}$$

where **r**, **g** and **b** are the normalized images of red, green and blue channel, respectively. n is the number of total pixels of the input image.

After the determination of the mathematical model of skin color distribution in the logarithmic Nrgb color space via surface-fitting, we propose to flatten the fitted quadratic surface vertically along the direction of negative z-axis. Suppose an arbitrary point $A(x_A, y_A, z_A)$ lying on the surface (cf. Figure 4.7 (b)), the corresponding point $B(x_B, y_B)$ in the flattened 2-dimensional plane (cf. Figure 4.7 (c)) can be calculated as:

$$\begin{aligned} x_B &= dist_{geodesic}\Big(A(x_A, y_A, z_A), P_1(0, y_A, c_2 y_A^2 + c_4 y_A + c_5)\Big) \\ y_B &= dist_{geodesic}\Big(A(x_A, y_A, z_A), P_2(x_A, 0, c_1 x_A^2 + c_3 x_A + c_5)\Big) \end{aligned} \tag{4.12}$$

where P_1 is the intersection point of curve $u_1: \ z = c_1x^2 + c_3x + c_2y_A^2 + c_4y_A + c_5$ and y-axis, P_2 is the intersection point of curve $u_2: \ z = c_2y^2 + c_4y + c_1x_A^2 + c_3x_A + c_5$ and x-axis, distance function $dist_{geodesic}(p, q)$ computes the geodesic distance between two points p and q on a Riemannian manifold M.

4.2.2.2 Geodesic distance

By definition, the shortest curve on a Riemannian manifold M that connects two given points is the geodesic. Given a continuously differentiable curve $u : [a, b] \to M$ and its two terminal points $p, q \in M$ with $p = u(a)$ and $q = u(b)$, the arc length $L(u)$ of u can be defined as:

$$L(u) = \int_a^b |u'(t)| \mathrm{d}t \tag{4.13}$$

If the tangent vector at each point along this curve stays constant, that is,

$$\nabla_{u'} u' = 0 \tag{4.14}$$

then we can call the curve a *geodesic* and the geodesic distance between p and q can be defined as:

$$dist_{geodesic}(p, q) = \int_a^b |u'(t)| \mathrm{d}t \quad \forall t \in [a, b], \nabla_{u'(t)} u'(t) \equiv 0 \tag{4.15}$$

Combining Equation (4.15) and Equation (4.12), the projected point $B(x_B, y_B)$ in the flattened 2-dimensional plane from an arbitrary point $A(x_A, y_A, z_A)$ in the quadratic surface can be determined as:

$$\begin{aligned} x_B &= \int_0^{x_A} \sqrt{(2c_1 x + c_3)^2 + 1}\, \mathrm{d}x \\ y_B &= \int_0^{y_A} \sqrt{(2c_2 y + c_4)^2 + 1}\, \mathrm{d}y \end{aligned} \tag{4.16}$$

Finally, we apply independent component analysis (ICA) to the skin color distribution on the flattened 2-dimensional plane to estimate the distributions of two skin chromophores, $\mathbf{Q}(1) = [q_{i,j}(1)]_{m \times n}$ and $\mathbf{Q}(2) = [q_{i,j}(2)]_{m \times n}$, on a $m \times n$ skin color image. Here, $q_{i,j}(1)$ and $q_{i,j}(2)$ are two elements of the quantity vector $\mathbf{q}_{i,j} = [q_{i,j}(1), q_{i,j}(2)]^\top$ in Equation (4.7). One main problem which is not mentioned in Tsumura's work is that it is difficult to determine if $\mathbf{Q}(1)$ indicates the hemoglobin content distribution and $\mathbf{Q}(2)$ for melanin content distribution, or vice versa, without *a prior* any relevant information. To distinguish the estimated chromophore content distributions, the Pearson correlation coefficient (PCC) is introduced to measure the correlation (linear dependency) between $\mathbf{Q}(1)$ or $\mathbf{Q}(2)$ and a^* index image of CIE $L^*a^*b^*$ color space, since a^* index indicates

close correlation with hemoglobin content, as mentioned in Section 4.2.1.

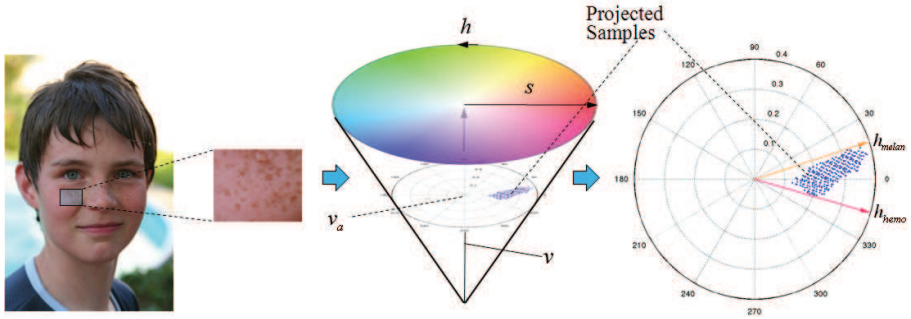
$$r_k = \frac{\sum_m \sum_n (\mathbf{Q}_{ij}(k) - \overline{\mathbf{Q}(k)})(\mathbf{a^*}_{ij} - \overline{\mathbf{a^*}})}{\sqrt{\left(\sum_m \sum_n (\mathbf{Q}_{ij}(k) - \overline{\mathbf{Q}(k)})^2\right)\left(\sum_m \sum_n (\mathbf{a^*}_{ij} - \overline{\mathbf{a^*}})^2\right)}} \quad k = 1, 2 \tag{4.17}$$

If $r_1 > r_2$, then $\mathbf{Q}(1)$ corresponds to the distribution of hemoglobin content and $\mathbf{Q}(2)$ to the distribution of melanin content. Otherwise, $\mathbf{Q}(1)$ corresponds to the distribution of melanin content and $\mathbf{Q}(2)$ to the distribution of hemoglobin content.

4.2.3 HSV Color Space

In this section, we briefly review the Kim's method [56] for skin chromophore extraction in HSV color space. One visualization model of the HSV is cone. In this representation, *hue* is depicted as the color wheel of a 3-dimensional cone; *saturation* is represented by the distance from the center of a circular cross-section of the cone, and *value* is the distance from the apex of the cone representing the brightness (cf. Figure 4.8 (b)). The user-selected sample region $\mathbf{S}$, from the body reflection image, is shown in Figure 4.8 (a). RGB color values (R_i, G_i, B_i) of the pixels $p_i \in \mathbf{S}$ are mapped to HSV color space (h_i, s_i, v_i). By averaging v, the third component of HSV color space, we can find the color plane (i.e. a particular circular cross-section of the HSV color cone) (h, s, v_a), where $v_a = \sum_{i=1}^{n} v_i/n$. It can be assumed that this color plane contains the two primitive components, indicating melanin and hemoglobin, based on the following facts: the color plane perpendicular to the *value* axis contains all hue and saturation informations; color planes from different human subjects or from varying sample regions are different only in their brightness. Then, the sample colors are projected along vector $\mathbf{p} = [h_i, s_i, v_i]^\top$ on to the color plane, as shown in Figure 4.8. Hemoglobin and melanin components are then extracted successively by projecting the colors onto the color plane to two independent axes. Since HSV color plane, well sorts out its hue values radically around its center and sample color data is distributed nearly in the same area of red to green hues, it is intuitionistic for the user to perceive where the independent color vectors are, without counting on a statistical method such as ICA; this can be procedurally done simply by radially searching the hue values, that encompass the color samples, on the color plane projected, as shown in Figure 4.8 (c). In addition, based on the fact that melanin contributes the yellow-brown color of skin and hemoglobin contributes the violet-red color, it is reasonable that color vector with maximum *hue* value stands for hemoglobin component while color vector with minimum *hue* value represents melanin. By assigning

each pixel with its corresponding color vector magnitude, we can obtain the estimated hemoglobin/melanin distribution map.

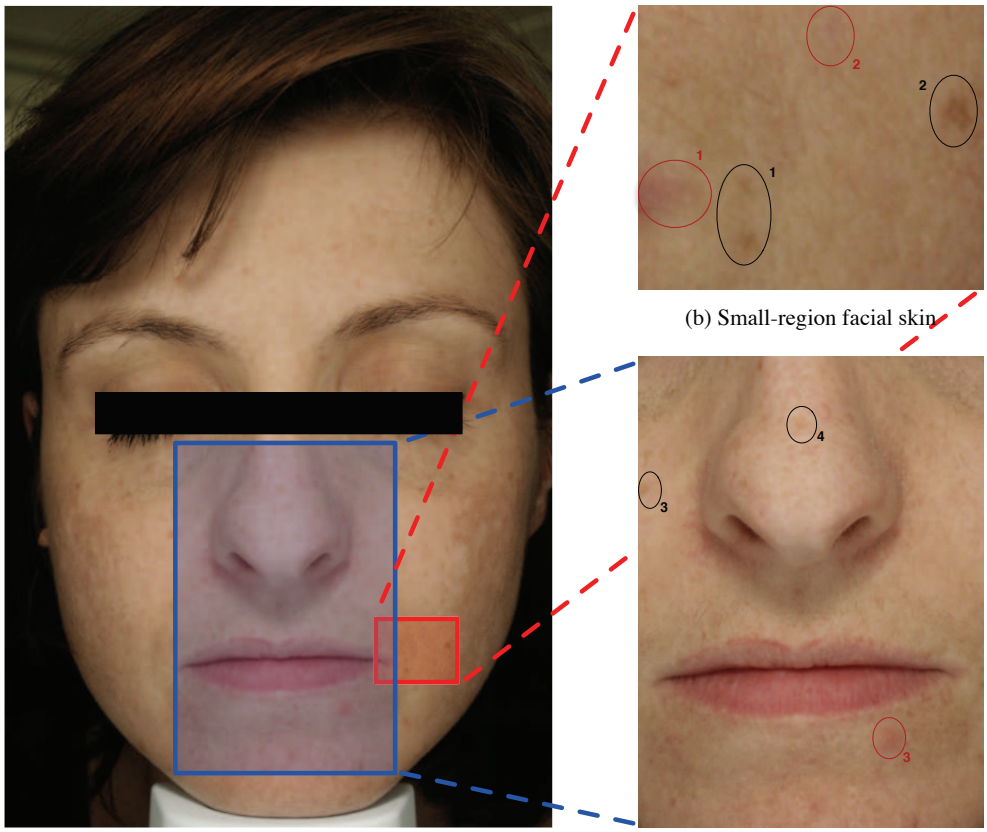


(a) Sampled skin region (b) HSV color space (c) Projection on circular cross section

Figure 4.8: A schematic of skin chromophore extraction method proposed by Kim and Kim [56].

4.2.4 Experimental Results and Discussion

Skin chromophore extraction performances of the techniques mentioned in Section 4.2 are evaluated successively on two groups of experimental data (cf. Figure 4.9): one is a 313×257 resolution image of small-region facial skin (Figure 4.9 (b)) which contains pimples, lips and freckles; another is a 454×619 resolution image of large-region facial skin (Figure 4.9 (c)) which contains not only freckles and pimples, but also highlighted area (e.g. nose bridge), as well as shaded areas within and around the nose part (e.g. the nostril, the in-between part of nose and lips, foot of the nose bridge, etc.). Both of the two images are sampled from a 778×1167 high-resolution image of entire facial skin (Figure 4.9 (a)) where surface reflection that always produces highlight is reduced by cross-polarization filter.



(a) Entire facial skin

(b) Small-region facial skin

(c) Large-region facial skin

Figure 4.9: Experimental inputs for skin chromophore extraction in different color spaces: (a) a cross-polarized facial skin image of 778×1167 pixels, (b) a small region (313×257 pixels) of the entire facial skin which contains lips (red circle 1), a pimple (red circle 2) and two groups of freckles (black circle 1 and 2), (c) a large region (454×619) of the entire facial skin which contains highlighted/shading area, a pimple (red circle 3) and two groups of freckles (black circle 3 and 4). Black indicates high concentration of melanin while red for high concentration of hemoglobin.

One qualitative way to evaluate the results of skin chromophore extraction is to manually observe if freckles or pimples/lips are successfully separated. Generally speaking, freckles will mainly appear in melanin component while pimples/lips appear mainly in hemoglobin component. Figure 4.10 presents extraction results on small-region facial skin (cf. Figure 4.9 (b)). Here, the proposed method in $L^*a^*b^*$ color space gives false estimation on both melanin and hemoglobin extraction. Black circle 1 and 2 in Figure 4.10 (a) are supposed to have lower contents of hemoglobin while red circle 1 and 2 in Figure 4.10 (b) should contain no melanin at all. Kim's method outperforms the $L^*a^*b^*$ method by extracting relatively better melanin distribution map (though the melanin concentrations marked by black circle 1 and 2 in Figure 4.10 (d) are supposed to be higher than their surroundings), it also fails in estimation the hemoglobin distribution (Figure 4.10

(a) and (c) are almost identical). On the other hand, both Tsumura's method and our proposed SF^2 method obtain comparatively good extractions of skin chromophores as they are able to estimate hemoglobin/melanin distributions much closer to the distributions of pimples/freckles.

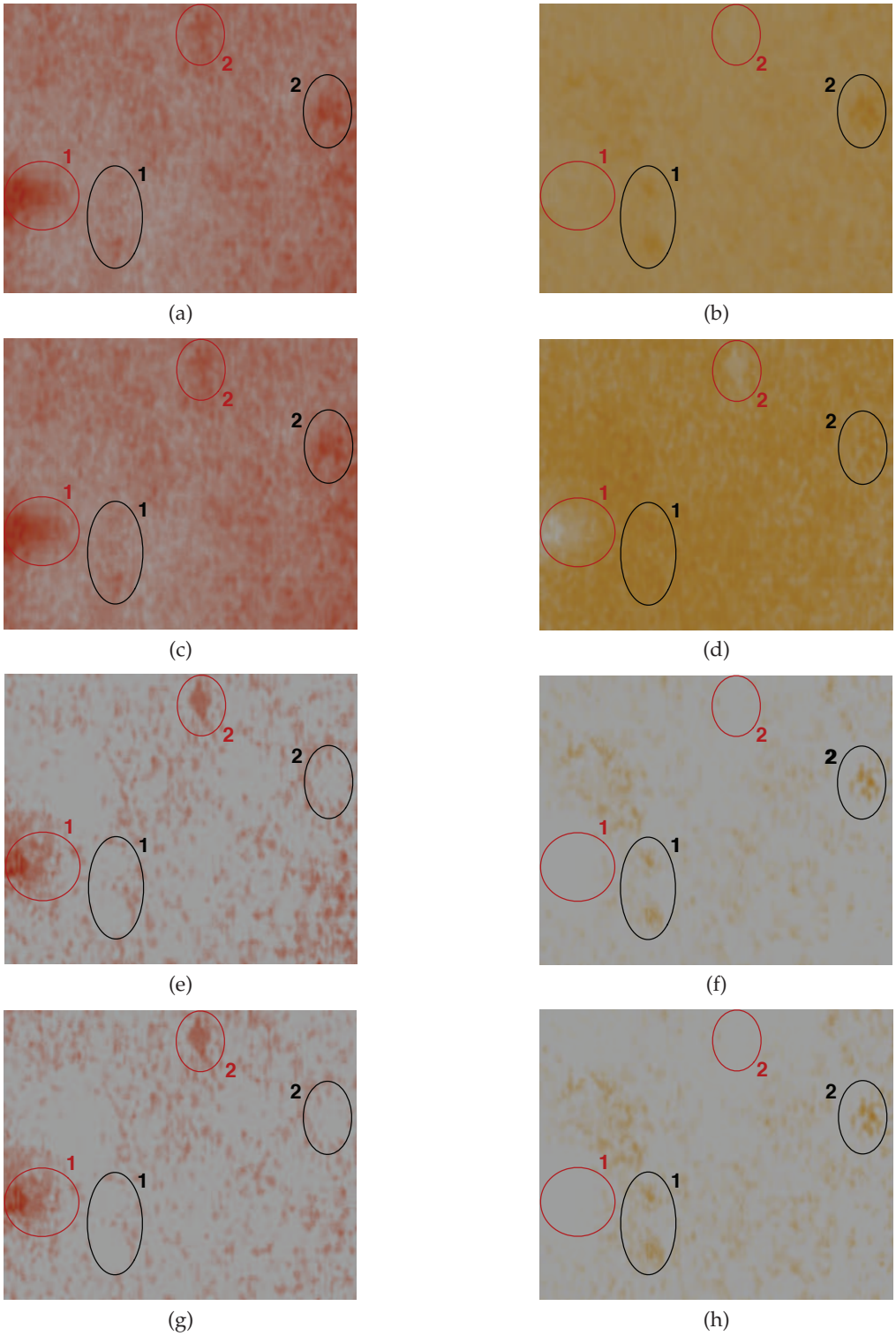


Figure 4.10: Skin chromophore extraction results of small-region facial skin by our proposed method in $L^*a^*b^*$ color space: (a), (b); Kim's method: (c), (d); Tsumura's method: (e), (f) and our proposed SF^2 method: (g), (h). From left to right: hemoglobin distribution map and melanin distribution map.

Figure 4.11 and Figure 4.12 show the extraction results on large-region facial skin (cf. Figure 4.9 (c)). Firstly, we evaluate the performance of each method by verifying that whether the extracted chromophore distributions match the actual distribution of pimples and freckles or not. In Figure 4.11 (b), $L^*a^*b^*$ based method fails to exact accurate melanin distribution where black circle 4 is supposed to indicate a region with relatively higher content of melanin while the area marked by red circle 3 should have no melanin at all. Kim's method obtains slightly better estimation of melanin distribution (cf. Figure 4.11 (d)) than the former method. However, the contrasts between interior and exterior of circles are lack of discrimination. Tsumura's method in this case gives the worst estimation on melanin distribution (cf. Figure 4.12 (b)) since no evidence of melanin contents can be observed within black circle 3 or 4, which conflicts with the fact that regions marked by black circles are freckle with high content of melanin. In addition, observe the highlighted noise bridge part where black circle 4 locates. In the original image (Figure 4.9 (b)), this part appears pale with a large quantity of dark-brown freckles and pores and a small quantity of red pimples, which is supposed to contains more melanin than hemoglobin. However, falsely estimated by Tsumura's method, hemoglobin content is relatively high (cf. Figure 4.12 (a)), compared to the melanin content (almost nothing in Figure 4.12 (b)). Best results of chromophore extraction are given by the proposed SF^2 approach. One can easily indicate the locations of pimples and freckles according to the hemoglobin/melanin distribution maps in Figure 4.12 (c) and (d).

Secondly, we evaluate the performance of each method by verifying whether they are resilient to the shading effect at shaded parts including the in-between part of nose and lips, as well as the foot of the nose bridge. Both $L^*a^*b^*$ based method and Kim's approach result in inaccurate estimation of hemoglobin distribution. In Figure 4.11 (a) and (c), hemoglobin contents of the in-between part of nose and lips and the foot of nose bridge are higher than that of the surrounding area, which is supposed to be virtually constant. Tsumura's method gives inaccurate estimation of the melanin distribution due to the shading effect. In Figure 4.12 (b), melanin contents of the in-between part of nose and lips and the foot of nose bridge are remarkably higher than their surrounding skin area, which is supposed to have equivalent content of melanin. Compared to the former three methods, our proposed SF^2 method results in more accurate extraction of hemoglobin/melanin distributions in Figure 4.12 (c) and (d), where the shading caused by nose can not be observed.

Finally, we compare the extraction results of each method at another type of shaded area, the nostrils. Due to its very low intensity, color distributions in RGB/HSV/$L^*a^*b^*$ color space are contaminated by noise, especially in the normalized RGB color space.

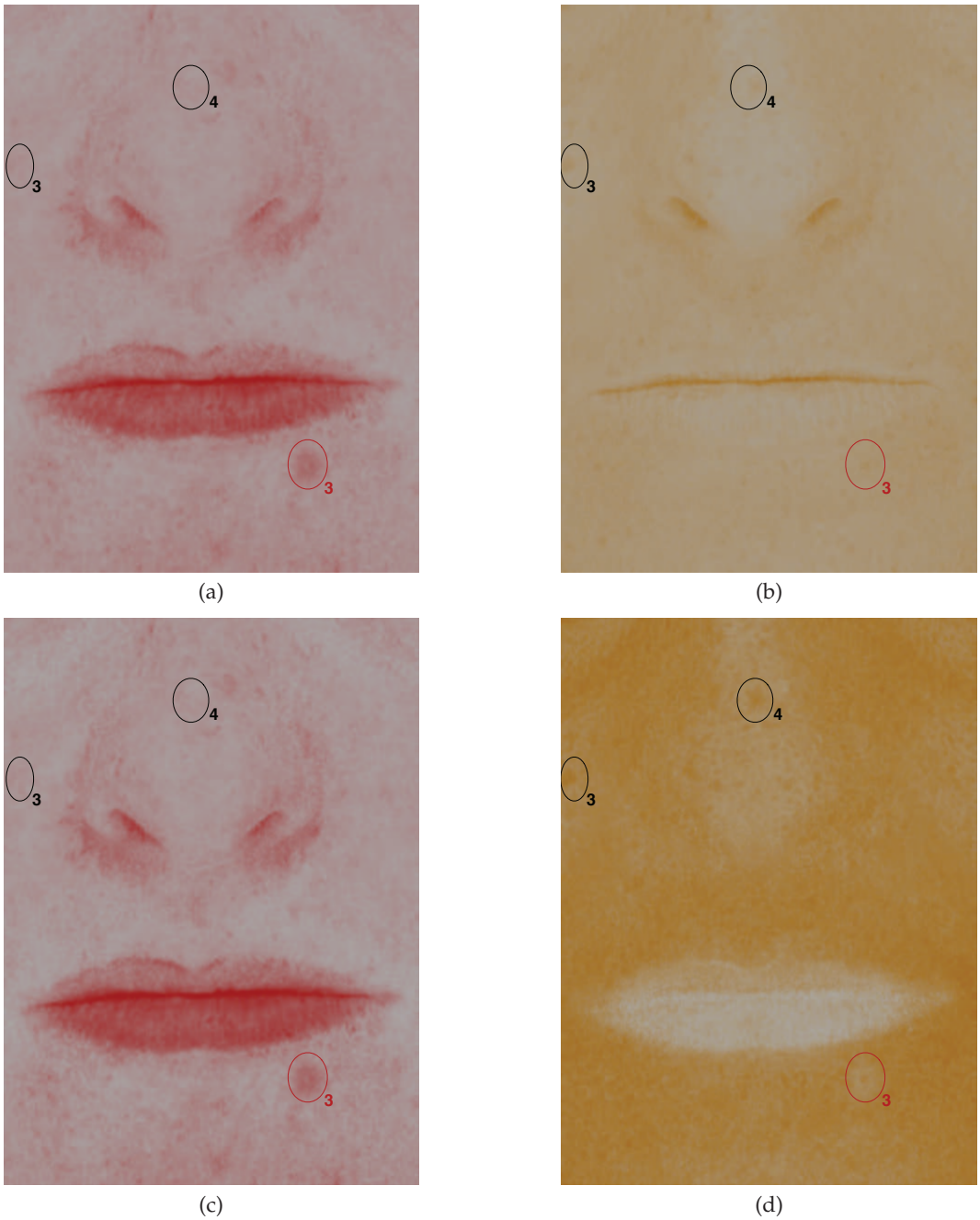


Figure 4.11: Skin chromophore extraction results of large-region facial skin by our proposed method in $L^*a^*b^*$ color space: (a), (b) and Kim's method: (c), (d). From left to right: hemoglobin distribution map and melanin distribution map.

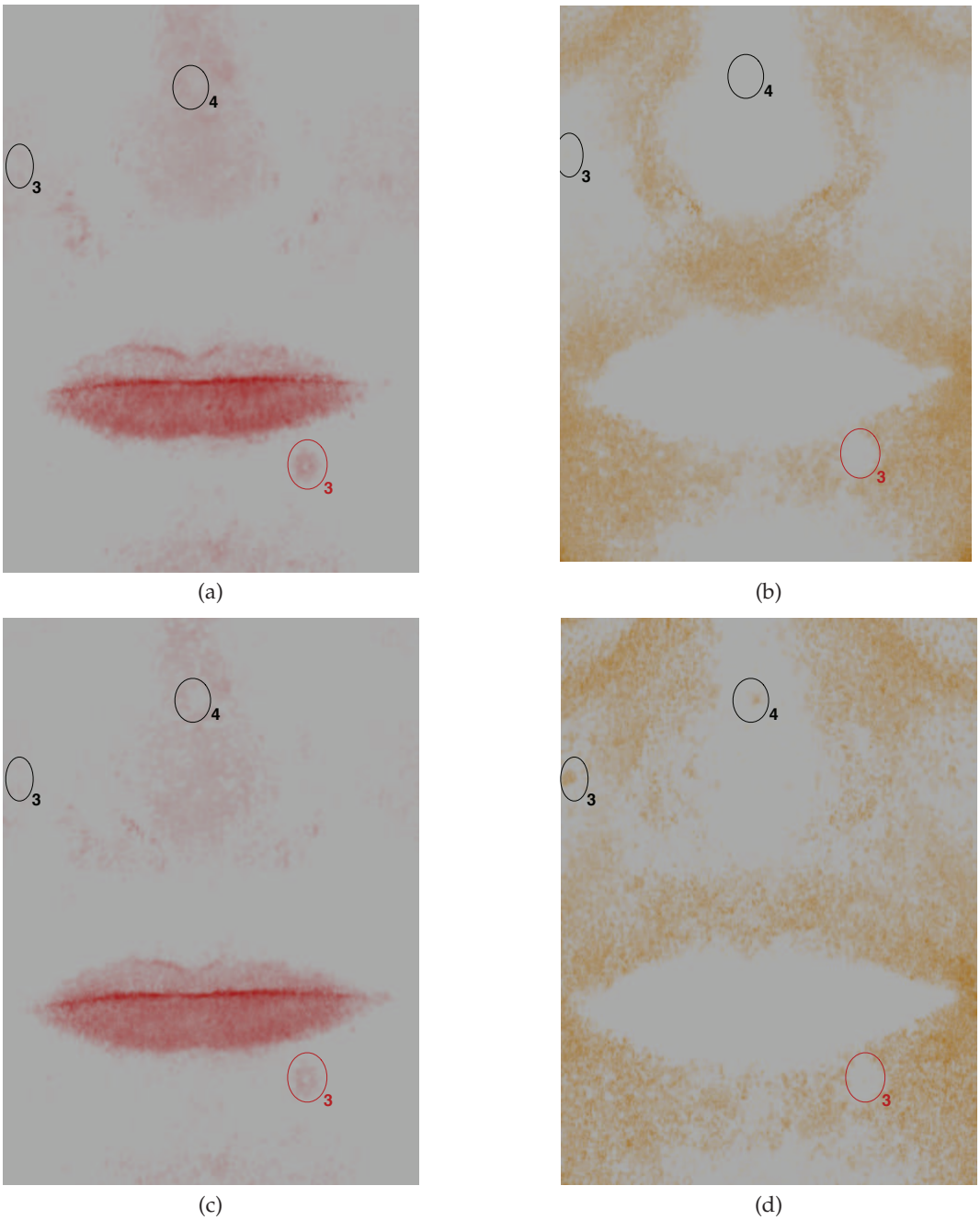


Figure 4.12: Skin chromophore extraction results of large-region facial skin by Tsumura's method: (a), (b) and our proposed SF^2 method: (c), (d). From left to right: hemoglobin distribution map and melanin distribution map.

Thus, none of the four methods can estimate the chromophore distribution correctly. Since we are lack of *a prior* knowledge about the chromophore concentration of nostrils, a possible assumption that nostrils contain more hemoglobin than melanin due to the fact there exist plenty of blood vessels which contain high content of hemoglobin will be reasonable. Based on this, the proposed SF^2 method gives estimation of chromophore distribution close to the assumption.

In conclusion, our proposed SF^2 method outperforms the other three methods in more accurate extraction of hemoglobin/melanin distributions, especially at the highlighted and shaded parts of the facial skin image.

Essentially, both Kim's method and Tsumura's method adopt the same methodology to perform skin chromophore extraction, which contains three main steps: (a) skin data modeling; (b) data dimensionality reduction; (c) independent feature extraction. At the first step, Kim's method adopts an intuitive color model, HSV color model, making the next two steps easier to be executed accurately without counting on any statistical techniques, such as PCA and ICA. Due to insufficient knowledge of optical skin properties, however, this method can give incorrect description of skin chromophore concentration (cf. Figure 4.10 (c, d)) and thus cannot be employed as an aide for medical diagnosis. Tsumura's method, on the other hand, is based on optical density model of human skin. Results obtained by this technique in Figure 4.10 (e) and (f) agree quite well with the physiological knowledge. However, it performs incorrect chromophore extraction when applied to process the large-region facial skin image (cf. Figure 4.12 (a, b)). Large-region facial skin (cf. Figure 4.6 (a) or Figure 4.9 (c)) data in optical density domain falls into a complex 3-dimensional volume (cf. Figure 4.6 (d, e)), which makes PCA fail in preserving the true structure of the data and thus results in incorrect extraction. To overcome the drawback of Tsumura's approach in extracting false chromophore distribution of skin sample under uneven illumination condition, we propose to model skin chromophore distribution by fitting a quadratic surface in logarithmic normalized RGB space and flattening it onto a 2-dimensional color plane. Qualitative evaluation based on dermatologic knowledge via comparative experiments indicates that our proposed SF^2 method extracts more accurate chromophore distributions than Kim's and Tsumura's approaches.

4.3 Skin Chromophore Quantification on Regular Color Images with Physical Model

The focus of our work in this section is to estimate hemoglobin and melanin quantitatively based on regular color imaging of skin. We propose and compare four different approaches using erythema/melanin indices, source separation algorithms (ICA and NMF) and model-fitting. Both qualitative and quantitative evaluations indicate that model-fitting approach outperforms the other three approaches.

4.3.1 Erythema Index and Melanin Index

In practice, quantification of skin chromophores can be characterized by the two parameters named 'Erythema Index' (EI) and 'Melanin Index' (MI) (or 'Pigmentation Index'), which hold excellent linearity with hemoglobin concentration and melanin concentration respectively. Unlike color coordinates such as $L^*a^*b^*$, the EI and MI are not indicators for evaluating 'color' but indices for quantifying the amounts of hemoglobin and melanin. To derive EI and MI, there exists several narrow-band reflectance instruments for commercial use: Mexameter™(Courage & Khazaka GmbH, Cologne, Germany), the DermaSpectrometer™(Cortex Technology, Hadsund, Denmark) and the Erythema/Melanin Meter™(DiaStron Ltd, Andover, U.K.), etc. All these instruments utilize the same basic approach, taking the log of ratios of reflectance within two or three selected wavebands in the visible and infrared. These instrument-based approaches lead a significant step forward in the quantification of the chromophores responsible for skin color. However, as contact-type instruments, their applications are limited by the size of measuring probe. It is difficult to readily evaluate an object larger or smaller than the measurement area (approximately 10 mm diameter maximum). Also, probe pressure on skin surface by direct contact can lead to artifacts such as blanching, which consequently results in bad quantification of skin chromophores. Based on the theories of absorbance of skin model (cf. Section 4.1), Takiwaki et al. [97] proposed a simple method to derive EI and MI images from ordinary RGB images. The equations for calculating EI and MI are written as follows:

$$\begin{aligned} \mathbf{EI} &= \log_{10}(1/\mathbf{R}_{\text{green}}) - \log_{10}(1/\mathbf{R}_{\text{red}}) \\ \mathbf{MI} &= \log_{10}(1/\mathbf{R}_{\text{red}}) \end{aligned} \tag{4.18}$$

where $\mathbf{R}_{\text{red, green}} = \mathbf{S}_{\text{red, green}}/W_{\text{red, green}}$. $\mathbf{R}_{\text{red, green}}$ are the calibrated red and green reflectance images of the sample under study, $\mathbf{S}_{\text{red, green}}$ are the acquired red and green

color images of the sample and $W_{\text{red, green}}$ are the average red and green values supposed to be nearly equivalent in the white standard. In fact, this method is a simplified application of algorithm proposed by Stamatas [85] within visible spectrum range.

4.3.2 Independent Component Analysis Based Approach

Hyvärinen et al. [49] proposed several techniques for Independent Component Analysis (ICA) for decomposing multivariate data into independent components using a function of non-Gaussianity. One of the most important contribution of their related works is the FastICA algorithm. We have applied the FastICA for our problem of skin color decomposition to quantify hemoglobin and melanin.

The general approach of ICA can be formulated as following: assume that we have a sequence of observed data $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$ which can be arranged in the rows of the matrix $\mathbf{X}$. If each data $\mathbf{x}_k$ is the linear combination of the source data (independent components) $\mathbf{s}_k$ then the observation matrix $\mathbf{X}$ can be represented as a product of two matrices

$$\mathbf{X} = \mathbf{AS} \tag{4.19}$$

where $\mathbf{A}$ is a mixing matrix and $\mathbf{S}$ is a matrix of source data. The task of ICA is to determine the mixing matrix $\mathbf{A}$ and the matrix of independent source data $\mathbf{S}$ given the observation matrix $\mathbf{X}$.

In our particular case, the application of ICA is based on independency of hemoglobin and melanin, as well as the absorbance of the multi-layered model described in Section 4.1. Here, Equation (4.1) can be modified in form of Equation (4.19) (deoxy-hemoglobin is omitted):

$$\begin{bmatrix} \log(1/\mathbf{r}(\lambda_R)) \\ \log(1/\mathbf{r}(\lambda_G)) \\ \log(1/\mathbf{r}(\lambda_B)) \end{bmatrix} = \begin{bmatrix} \epsilon_{\text{HbO}_2}(\lambda_R) & \epsilon_{\text{Mel}}(\lambda_R) \\ \epsilon_{\text{HbO}_2}(\lambda_G) & \epsilon_{\text{Mel}}(\lambda_G) \\ \epsilon_{\text{HbO}_2}(\lambda_B) & \epsilon_{\text{Mel}}(\lambda_B) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{\text{HbO}_2} \\ \mathbf{c}_{\text{Mel}} \end{bmatrix} \tag{4.20}$$

where $\mathbf{r}(\lambda_{R,G,B})$ are the calibrated reflectance images of red, green and blue channel, respectively. $\epsilon_{\text{HbO}_2,\text{Mel}}(\lambda_{R,G,B})$ are extinction coefficients of hemoglobin and melanin at 'red band', 'green band' and 'blue band', respectively. $\mathbf{c}_{\text{HbO}_2,\text{Mel}}$ denote concentration distribution maps of hemoglobin and melanin. In Equation (4.20), we notice that the number of reflectance images (three) in observation matrix is greater than the expected number of independent components (two). Therefore, principle component analysis (PCA) is applied to reduce the dimensionality of the observation matrix and *prewhiten* the data so that the task of finding the mixing matrix reduces to the task of estimating a square orthogonal matrix.

The observation data is linearly transformed by PCA such that

$$\tilde{\mathbf{X}} = \mathbf{MX} = \mathbf{MAS} = \tilde{\mathbf{A}}\mathbf{S} \tag{4.21}$$

where $\mathbf{M}$ is the whitening matrix calculated by eigen-value decomposition (EVD) of the covariance matrix $E\{\mathbf{X}\mathbf{X}^\top\}$ so that $E\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\top\}$=$\mathbf{I}$; $\tilde{\mathbf{A}}$=$\mathbf{MA}$ is a new orthogonal mixing matrix that can be seen from

$$E\{\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\top\} = \tilde{\mathbf{A}}E\{\mathbf{S}\mathbf{S}^\top\}\tilde{\mathbf{A}}^\top = \tilde{\mathbf{A}}\tilde{\mathbf{A}}^\top = \mathbf{I} \tag{4.22}$$

The problem is now reduced to the estimation of the orthogonal mixing matrix $\tilde{\mathbf{A}}$. The columns of the matrix $\tilde{\mathbf{A}}$ are denoted by $\tilde{\mathbf{a}}_i$ and the i^{th} source component can be computed using $\tilde{\mathbf{X}}$ by the following equation,

$$\mathbf{s}_i = \tilde{\mathbf{a}}_i^\top \tilde{\mathbf{X}} = \mathbf{w}^\top \tilde{\mathbf{X}} \tag{4.23}$$

The FastICA algorithm for one unit as described in Equation (4.23) is

$$\mathbf{w}^+ = E\{\tilde{\mathbf{X}}g(\mathbf{w}^\top\tilde{\mathbf{X}})\} - E\{g'(\mathbf{w}^\top\tilde{\mathbf{X}})\}\mathbf{w} \tag{4.24}$$

$$\mathbf{w}_{\text{new}} = \mathbf{w}^+ / \|\mathbf{w}^+\| \tag{4.25}$$

The one-unit algorithm can be extended to the estimation of the whole ICA transformation $\mathbf{S}$=$\mathbf{W}^\top\tilde{\mathbf{X}}$ using the Gram Schmidt deflation approach. This step of decorrelation ensures that we do not obtain the same independent component.

4.3.3 Non-Negative Matrix Factorization Based Approach

Non-negative matrix factorization (NMF) suggested by Lee and Seung [63] is a useful method of decomposition of multivariate data. The method explicitly enforces the non-negativity constraint on the values of the source data as well as the mixing quantities of the source data forming the mixed data.

Compared to the FastICA algorithm, NMF has two main advantages in application to our problem. First, non-negativity constraint on the source data prevents meaningless negative value of chromophore concentration. Second, no constrain on the orthogonality of the source data allows dependency between skin chromophores, which is closer to the reality.

To extend NMF in our application, we simply employ the same linear mixture model as described in Equation (4.19) and Equation (4.20). The non-negativity constraint is enforced on the source data and the mixing matrix as $\mathbf{S} \geq 0$ and $\mathbf{A} \geq 0$ respectively. Therefore, the problem can be formulated as a maximum-likelihood problem with least squares solution as:

$$\begin{aligned} \mathbf{A}_{ML}, \mathbf{S}_{ML} &= \arg\max_{\mathbf{A},\mathbf{S}} p(\mathbf{X}|\mathbf{A}, \mathbf{S}) && (4.26) \\ \Rightarrow F &= \arg\min_{\mathbf{A},\mathbf{S}} \|\mathbf{X} - \mathbf{AS}\|^2 && (4.27) \\ & \quad \textit{Subject to} : \mathbf{A} \geq 0, \mathbf{S} \geq 0 \end{aligned}$$

In the maximum likelihood optimization, the negative log-likelihood of F is minimized i.e. $\log \|\mathbf{X} - \mathbf{AS}\|$ is computed at each iteration. Here, $\|\cdot\|$ is the Euclidean norm. The updates of $\mathbf{A}$ and $\mathbf{S}$ can be performed under the 'multiplicative update rule' in forms as:

$$\mathbf{A} \leftarrow \mathbf{A}\frac{\mathbf{XS}^\top}{\mathbf{ASS}^\top}, \quad \mathbf{S} \leftarrow \mathbf{S}\frac{\mathbf{A}^\top\mathbf{X}}{\mathbf{A}^\top\mathbf{AS}} \tag{4.28}$$

This rule ensures the non-negative properties of the optimal solutions, $\mathbf{A}_{ML}$ and $\mathbf{S}_{ML}$ if the initial matrices $\mathbf{A}_{\text{initial}}$ and $\mathbf{S}_{\text{initial}}$ are strictly positive. The initialization of the source matrix $\mathbf{S}$ is given by EI and MI and the mixing matrix $\mathbf{A}$ can be initialized using least squares estimation with a single constraint as given in Equation (4.29).

$$\begin{aligned} \mathbf{S}_{\text{initial}} &= \begin{bmatrix} \mathbf{EI} \\ \mathbf{MI} \end{bmatrix} && (4.29) \\ & \arg\min_{\mathbf{A}_{\text{initial}}} \|\mathbf{X} - \mathbf{A}_{\text{initial}}\mathbf{S}_{\text{initial}}\|^2 && (4.30) \\ & \textit{Subject to} : \mathbf{A}_{\text{initial}} \geq 0 \end{aligned}$$

4.3.4 Model-Fitting Based Approach

Source separation based approaches, like ICA and NMF, give us a statistical tool to quantify skin hemoglobin and melanin if the mixing matrix $\mathbf{A}$ is unknown.

In this section, we employ a more accurate model which includes oxy-hemoglobin and deoxy-hemoglobin based on the oxygen-saturation of hemoglobin. So that Equation (4.19) can be rewritten as

$$\begin{bmatrix} \log(1/\mathbf{r}(\lambda_R)) \\ \log(1/\mathbf{r}(\lambda_G)) \\ \log(1/\mathbf{r}(\lambda_B)) \end{bmatrix} = \begin{bmatrix} \epsilon_{\text{HbO}_2}(\lambda_R) & \epsilon_{\text{Hb}}(\lambda_R) & \epsilon_{\text{Mel}}(\lambda_R) \\ \epsilon_{\text{HbO}_2}(\lambda_G) & \epsilon_{\text{Hb}}(\lambda_G) & \epsilon_{\text{Mel}}(\lambda_G) \\ \epsilon_{\text{HbO}_2}(\lambda_B) & \epsilon_{\text{Hb}}(\lambda_B) & \epsilon_{\text{Mel}}(\lambda_B) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{\text{HbO}_2} \\ \mathbf{c}_{\text{Hb}} \\ \mathbf{c}_{\text{Mel}} \end{bmatrix} \tag{4.31}$$

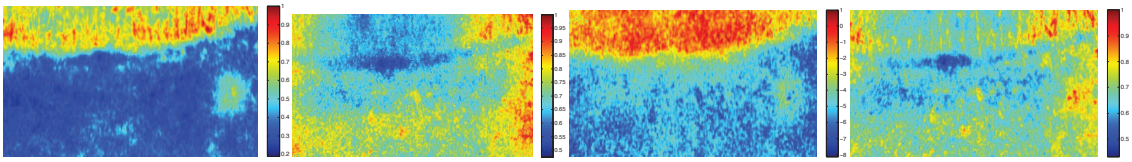
where the mixing matrix $\mathbf{A}$ is approximated using tabulated extinction coefficients of three predominant chromophores [82, 88], $\epsilon_{\mathrm{HbO_2}}(\lambda)$, $\epsilon_{\mathrm{Hb}}(\lambda)$ and $\epsilon_{\mathrm{Mel}}(\lambda)$. λ_R, λ_G and λ_B are selected at 600 nm, 540 nm and 440 nm. Now the problem is simply to fit this model by solving a system of linear equations. Since the number of equations is equal to the number of unknowns in this linear system, we can obtain the solutions as

$$\mathbf{S} = \mathbf{A}_{\mathrm{tabulate}}^{-1}\mathbf{X} \tag{4.32}$$

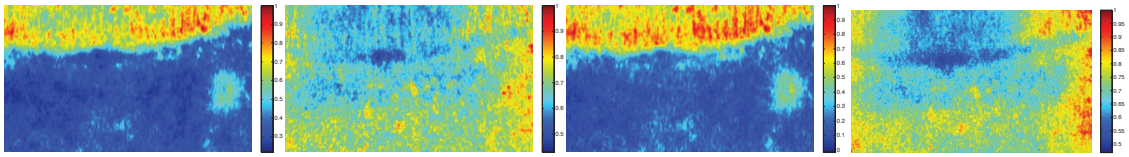
4.3.5 Experimental Results and Discussion

In this section, we compare our NMF based and model-fitting methods to Takiwak's and ICA based methods. Firstly, we evaluate qualitatively the performances of these methods using a 'lip-pimple' image (cf. Figure 4.13 (a)) and a 'melanocytic nevus' image (cf. Figure 4.14 (a)). Based on the dermatologic knowledge that (i) lip or pimple has higher concentration of hemoglobin and lower concentration of melanin, (ii) increase of melanin content and decrease of hemoglobin content are responsible for the dark color of melanocytic nevus, one can see that model-fitting method outperforms the other three approaches in extracting relatively accurate concentration cartographies of hemoglobin and melanin. ICA based method gives poor quantitative estimations due to some unrealistic negative values of independent components. Takiwaki's method and NMF based approach give similar results though less accurate compared with model-fitting method. For example, Takiwaki's method overestimates hemoglobin concentration in nevus (cf. Figure 4.14 (b)) and NMF based method overestimates melanin concentration in lip (cf. Figure 4.13 (g)).

(a) Lip and Pimple

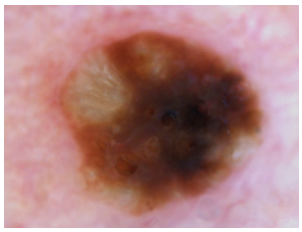


(b) EI (c) MI (d) H-ICA (e) M-ICA

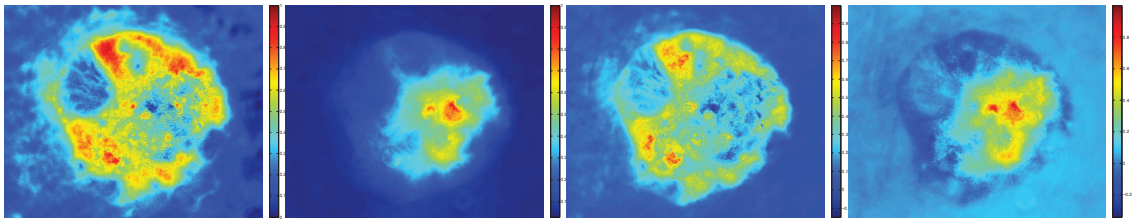


(f) H-NMF (g) M-NMF (h) H-MF (i) M-MF

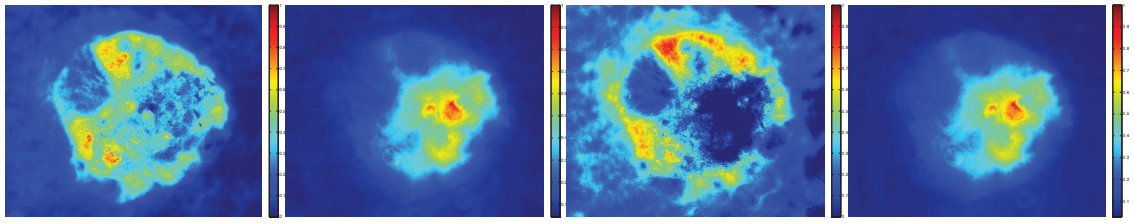
Figure 4.13: Comparison of hemoglobin and melanin concentration cartographies on 'lip-pimple'. 'H': hemoglobin. 'M': melanin. 'MF': model-fitting.



(a) Melanocytic Nevus



(b) EI (c) MI (d) H-ICA (e) M-ICA



(f) H-NMF (g) M-NMF (h) H-MF (i) M-MF

Figure 4.14: Comparison of hemoglobin and melanin concentration cartographies on 'melanocytic nevus'.

In context of melanoma detection, a precise and robust segmentation of pigmented skin lesion is required to discriminate tumor cell boundary and the surrounding tissue. Thus, the accuracy of the different quantification methods can be evaluated by the accuracy of the segmentation of melanoma, measured by Dice similarity coefficient (DSC), false negative ratio (FNR) and false positive ratio (FPR). To define the three metrics, let SG denote the result of graph-cut segmentation method, GT denote the result of manual segmentation by expert dermatologist (the ground truth). Both SG and GT are binary images such that all the pixels inside the boundary are labeled as 1 and all the others have label 0. The metrics are calculated as follows (FNR and FPR were firstly introduced in Section 2.5.1 and calculated in Equation (2.46) as two criteria for quantitative evaluation of Boykov-Jolly proposed graph-cut segmentation algorithm on the natural color image under different value of λ, which will be redefined in the context of PSLs segmentation):

- Dice similarity coefficient (DSC)

  This metric measures how the pixels classified as lesion by the graph-cut based segmentation agree with the pixels classified as lesion by the expert dermatologist (similarity between graph-cut segmentation and manual segmentation):

$$DSC(SG, GT) = \frac{2 \cdot \#(SG \cap GT)}{\#(SG) + \#(GT)}. \tag{4.33}$$

- False negative rate (FNR)

  This metric measures the rate of pixels classified as skin by the graph-cut segmentation that were not classified as skin by the expert dermatologist: (rate of lesion pixels incorrectly identified as skin):

$$FNR(SG, GT) = \frac{\#(\overline{SG} \cap GT)}{\#(GT)}. \tag{4.34}$$

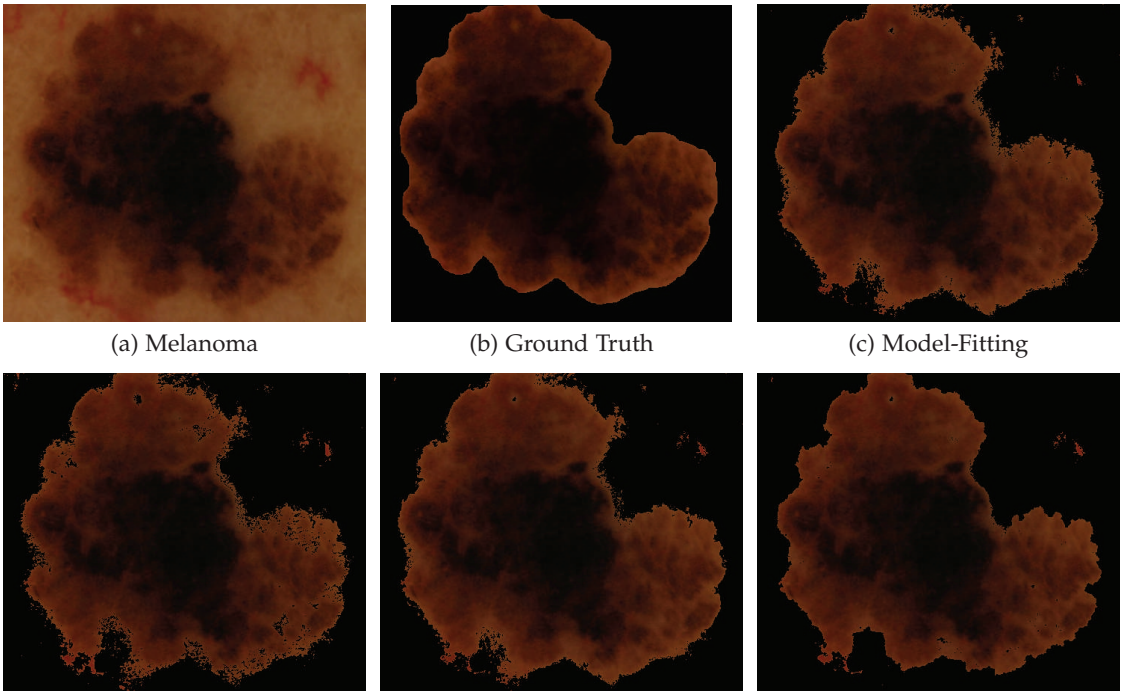
- False positive rate (FPR)

  This metric measures the rate of pixels classified as lesion by the graph-cut segmentation that were not classified as lesion by the expert dermatologist: (rate of skin pixels incorrectly identified as lesion):

$$FPR(SG, GT) = \frac{\#(SG \cap \overline{GT})}{\#(GT)}. \tag{4.35}$$

We use one representative 'Melanoma' image (cf. Figure 4.15 (a)) from the total

evaluated 30 melanoma lesions to present the evaluation task. These 'Melanoma' images are 938×872 pixels. The ground truth (cf. Figure 4.15 (b)) is obtained by manual segmentation of one expert dermatologist. First, we compare a classical graph-cut segmentation approach proposed by Boykov and Jolly (Boykov-Jolly) with a modified graph-cut segmentation approach on 5-level features: skin color feature (RGB tri-channel values) and skin chromophore feature (concentrations of epidermal melanin and dermal hemoglobin). It will be demonstrated below that by combining skin chromophore feature and color feature, graph-cut segmentation gives better results. Second, we compare the modified graph-cut segmentations based on skin chromophore features obtained by different quantification methods. Table 4.1 shows the average accuracy of segmentation based on each proposed feature-configuration. In Table 4.1, we can observe how model-fitting method achieves better results. The DSC is increased to 0.982 while both FNR and FPR decrease. NMF based method and Takiwaki's method give similar results, which is exactly the same as for the qualitative evaluation.



(a) Melanoma (b) Ground Truth (c) Model-Fitting

(d) NMF (e) Takiwaki (f) Boykov-Jolly

Figure 4.15: Quantitative evaluation of skin chromophore quantification approaches via comparison of segmentation results on regular 'Melanoma' color image.

As a conclusion, we propose and compare four different quantitative estimation approaches on skin color image using erythema/melanin indices, source separation algorithms (ICA and NMF) and model-fitting. By means of two comparative experiments based on dermatologic knowledge and graph-cut segmentation, model-fitting approach

Table 4.1: Comparison of average segmentation accuracy, quantified by Dice similarity coefficient, false negative rate and false positive rate. Borders of melanomas manually drawn by 1 expert dermatologist are considered as the ground-truth reference for accuracy assessment. Value in bold corresponds to the best performance.

Criterion \ Method	Model-Fitting	NMF	Takiwaki	ICA	Classic GC
DSC	**0.982**	0.967	0.971	0.954	0.965
FNR	0.013	0.044	**0.012**	0.059	0.061
FPR	**0.023**	0.024	0.027	0.031	0.028

obtains more accurate quantitative estimation of skin hemoglobin and melanin.

4.4 Skin Chromophore Quantification on Multi-Spectral Images

The interest of working with multi-spectral images is to have more accurate information on skin properties than those obtained on conventional cameras. Integration of a multi-spectral light will allow: (a) Reveal to the physicians the properties of non-visible skin characteristics (e.g. hemoglobin, melanin, collagen, etc.); (b) Observation of the cutaneous in-depth layers. Several methods based on spectrum in visible wavelength range have already been presented in the literatures [97, 85]. The main idea of these methods is to select specific spectral bands in the data to extract information on skin chromophores.

In this section, we propose to extract hemoglobin and melanin components considering the whole spectrum. Multi-spectral images acquired in visible wavelength region (e.g. 400-700 nm) are input into a mathematical optical skin model that considers the contributions from different chromophores in the epidermis and dermis skin layers. Through two different algorithms, non-negative factorization and least-squares based model fitting, we can quantify the concentrations of hemoglobin and melanin in a given area of skin lesion and surrounding heathy skin. To evaluate quantitatively, we apply graph-cut segmentation to both inputs and outputs of the proposed algorithms. Experimental results indicate that model-fitting approach obtains more accurate quantitative estimation than NMF.

4.4.1 Multi-Spectral Image Acquisition and Pre-processing

As shown in Figure 4.16, the multi-spectral imaging system (*SpectraCam*™, Newtone Technologies, Lyon, France) used in this work contains a liquid crystal tunable filter (LCTF) (VariSpec™, Model VIS2, Cambridge Research & Instrumentation, Inc., Boston,

MA) fitted in front of a PCO SensiCam Model 370 KL camera with 1168 × 1036 pixels on a progressive scan CCD image sensor (PCO Computer Optics, 93309 Kelheim, Germany) and a linear polarizing filter. The LCTF has a nominal bandwidth of 30 nm and a nominal accuracy of the selected peak wavelength of 4 nm. This allows to select about 80 significantly different tuning positions in the range from 400 nm to 720 nm. A linearly polarized light source was used with its polarization plane positioned vertical to the polarization plane of the camera polarizer. Thus, artifacts due to specular reflection were eliminated. The multi-spectral images were acquired over the whole melanoma area and the surrounding healthy skin in the wavelength range 400-700 nm with the scanning step 10 nm.

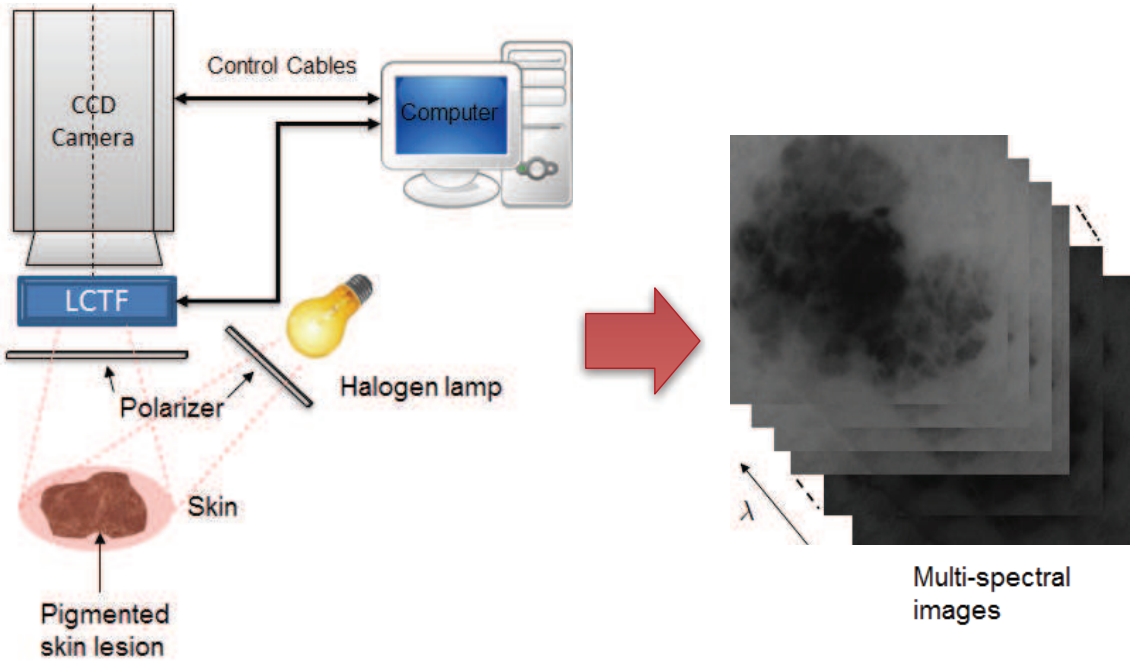


Figure 4.16: A schematic representation of liquid crystal tunable filter (LCTF) based multi-spectral imaging system.

4.4.1.1 Image Calibration

Proper calibration is essential in quantitative imaging (i.e. quantification of skin chromophores) to validate the reproducibility of spectral reflectance of skin. Inhomogeneities of illumination can be removed by correcting the acquired multi-spectral images $\mathbf{S}(\lambda)$ at each wavelength λ with corresponding ones of white reflectance standard (defined at 100% at all wavelength), $\mathbf{S}_{\text{ref}}(\lambda)$ and the dark current $\mathbf{D}$:

$$\mathbf{R}(\lambda) = \frac{\mathbf{S}(\lambda) - \mathbf{D}}{\mathbf{S}_{\text{ref}}(\lambda) - \mathbf{D}} \tag{4.36}$$

where $\mathbf{R}(\lambda)$ is the calibrated reflectance image at wavelength λ.

4.4.2 Different Approaches of Skin Chromophore Quantification on Multi-spectral Images

4.4.2.1 Erythema/Melanin Indices

Erythema is a kind of skin condition characterized by redness or rash, which can be explained by the concentration of hemoglobin. Conventionally, Erythema Index (EI) and Melanin Index (MI) provide the linear interval data scale for hemoglobin (type of hemoglobin not specified) and eumelanin. According to the molar absorptivity spectra of skin chromophores depicted in Figure 4.3 (b), melanin (eumelanin) absorbs incident light strongly within the wavelength range 620-650 nm, where the extinction coefficient of oxy-hemoglobin decreases sharply into its minimum. On the other hand, oxy-hemoglobin absorbs incident light prominently within the wavelength range 540-570 nm where melanin (eumelanin) slightly decreases. Based on the characteristics of oxy-hemoglobin and melanin absorbances in visible wavelength region, we propose a simple method to derive EI and MI images from multi-spectral images. The equations for calculating EI and MI are written as follows:

$$\mathbf{EI} = \log_{10}\left(1/\mathbf{R}(\lambda_1)\right) - \log_{10}\left(1/\mathbf{R}(\lambda_2)\right) \tag{4.37}$$

$$\mathbf{MI} = \log_{10}\left(1/\mathbf{R}(\lambda_2)\right) \tag{4.38}$$

where $\mathbf{R}(\lambda_i)$ is the calibrated reflectance image of the sample under study at wavelength λ_i, λ_1 is set at a wavelength range of 540-570 nm, and λ_2 is at 620-650 nm.

The limitation of the proposed EI/MI approach is that it lacks information on distinction of skin chromophores. For instance, extinction coefficient of melanin at 555 nm is higher than at 635 nm, which may allow a positive EI indicating certain contents of hemoglobin for a pigmented skin area with no hemoglobin at all. Also, deoxy-hemoglobin has a comparatively high extinction coefficient with that of melanin at 620 nm, which may result in positive MI for the blood stasis area with no melanin at all.

4.4.2.2 Non-Negative Matrix Factorization

As mentioned in Section 4.3.3, the non-negative matrix factorization (NMF) provides us a efficient statistical tool for decomposition of multivariate data. To make NMF applicable in our skin chromophore quantification on multi-spectral images, we simply redefine the linear mixture model described in Equation (4.19). Here, $\mathbf{X}$ is the observation data

matrix of m rows and n columns, with each row representing the absorbance image vector at each wavelength. $\mathbf{S}$ is a source data matrix of l rows and n columns, with l being the number of sources to be extracted. The matrix $\mathbf{A}$ is known as the mixing matrix with m rows and l columns. The columns of which represent the mixing values of each source component. According to the absorbance of the multi-layered model described in Section 4.1, Equation (4.1) can be modified in form of Equation (4.19) (deoxy-hemoglobin is omitted):

$$\begin{bmatrix} \log(1/\mathbf{R}(\lambda_1)) \\ \vdots \\ \log(1/\mathbf{R}(\lambda_m)) \end{bmatrix}_{m\times n} = \begin{bmatrix} \epsilon_{\mathrm{HbO_2}}(\lambda_1) & \epsilon_{\mathrm{Mel}}(\lambda_1) \\ \vdots & \vdots \\ \epsilon_{\mathrm{HbO_2}}(\lambda_m) & \epsilon_{\mathrm{Mel}}(\lambda_m) \end{bmatrix}_{m\times 2} \begin{bmatrix} \mathbf{c}_{\mathrm{HbO_2}} \\ \mathbf{c}_{\mathrm{Mel}} \end{bmatrix}_{2\times n} \tag{4.39}$$

where $\mathbf{R}(\lambda_i)$ is the calibrated reflectance image at wavelength λ_i. $\epsilon_{\mathrm{HbO_2,Mel}}(\lambda_i)$ are extinction coefficients of hemoglobin and melanin at wavelength λ_i, respectively. $\boldsymbol{c}_{\mathrm{HbO_2,Mel}}$ denote concentration distribution maps of hemoglobin and melanin. The dimension m is 26 in our experiment since we use multi-spectral images over 26 wavelengths sampled between 450-700 nm.

4.4.2.3 Model Fitting Approach

Similar to the model-fitting method described in Section 4.3.4, we introduce a more accurate model which includes oxy-hemoglobin and deoxy-hemoglobin based on the oxygen-saturation of hemoglobin in case that the mixing matrix is provided with tabulated extinction coefficients of these three chromophores. Thus, Equation (4.19) can be rewritten as

$$\begin{bmatrix} \log(1/\mathbf{R}(\lambda_1)) \\ \vdots \\ \log(1/\mathbf{R}(\lambda_m)) \end{bmatrix} = \begin{bmatrix} \epsilon_{\mathrm{HbO_2}}(\lambda_1) & \epsilon_{\mathrm{Hb}}(\lambda_1) & \epsilon_{\mathrm{Mel}}(\lambda_1) \\ \vdots & \vdots & \vdots \\ \epsilon_{\mathrm{HbO_2}}(\lambda_m) & \epsilon_{\mathrm{Hb}}(\lambda_m) & \epsilon_{\mathrm{Mel}}(\lambda_m) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{\mathrm{HbO_2}} \\ \mathbf{c}_{\mathrm{Hb}} \\ \mathbf{c}_{\mathrm{Mel}} \end{bmatrix} \tag{4.40}$$

where the mixing matrix $\mathbf{A}$ is approximated using tabulated extinction coefficients of three predominant chromophores [82, 88], $\epsilon_{\mathrm{HbO_2}}(\lambda_i)$, $\epsilon_{\mathrm{Hb}}(\lambda_i)$ and $\epsilon_{\mathrm{Mel}}(\lambda_i)$ at wavelength λ_i. Now the problem is simply to fit this model by solving a system of linear equations. Solutions of this overdetermined system can be obtained using least-squares estimation

with a single constraint as given in Equation (4.41):

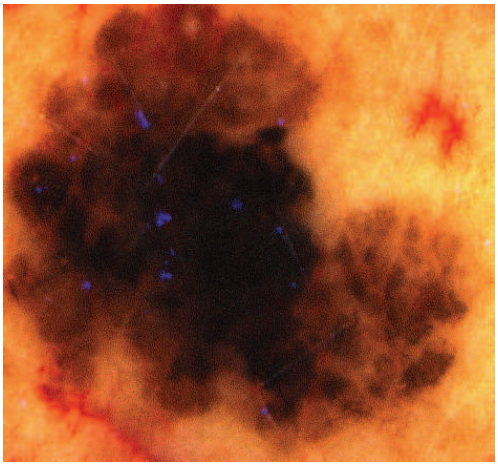
$$\begin{aligned} &\arg\min_{\mathbf{A}_{\text{tabulated}}} \|\mathbf{X} - \mathbf{A}_{\text{tabulated}}\mathbf{S}\|^2 \\ &\textit{Subject to} : \mathbf{S} \geq 0 \end{aligned} \tag{4.41}$$

4.4.3 Experimental Results and Discussion

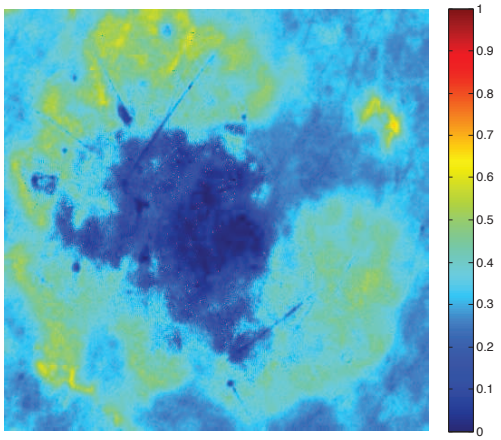
In this section, we compare the proposed algorithms using acquired multi-spectral images of melanoma (469×436 pixels) at 26 wavelength sampled equally from 450 nm to 700 nm. Based on the dermatologic knowledge that (i) vasculature contains higher concentration of hemoglobin and lower concentration of melanin, (ii) increase of melanin content and decrease of hemoglobin content are responsible for the dark color of melanoma, one can see that model-fitting method outperforms NMF based method in extracting relatively accurate concentration cartographies of hemoglobin and melanin. For example, NMF based method overestimates the hemoglobin concentration within the central melanoma area (cf. Figure 4.17 (b)) and underestimates the melanin concentration on the near-border melanoma area (cf. Figure 4.17 (c)).

In context of melanoma detection, a precise and robust segmentation of pigmented skin lesion is required to discriminate tumor cell boundary and the surrounding tissue. Thus, the accuracy of the different quantification methods can be evaluated by the accuracy of the segmentation of melanoma, measured by three metrics mentioned in Section 4.3.5, the Dice similarity coefficient (DSC), false negative ratio (FNR) and false positive ratio (FPR). We use extracted melanin images (cf. Figure 4.17 (c, e)) as well as the 26 acquired multi-spectral images to perform the graph-cut segmentation. The manual input seed map (cf. Figure 4.18 (a)) and the manual segmented ground truth (cf. Figure 4.18 (b)) are both obtained from one expert dermatologist. First, we compare a classic graph-cut segmentation on reconstructed color image (cf. Figure 4.17 (a)) with a modified approach on 26-level images (26 multi-spectral images). It will be demonstrated below that using multi-spectral images, performance of segmentation can be enhanced considerably. Second, we compare the segmentation results on 4-level images (reconstructed color image + melanin image) with 27-level images (multi-spectral images + melanin image). Table 4.2 shows the average accuracy of segmentation based on each proposed feature-configuration. In Table 4.2, we can observe that how model-fitting method achieves better results compared with NMF based method. The DSC is increased to 0.965 while both FNR and FPR decrease.

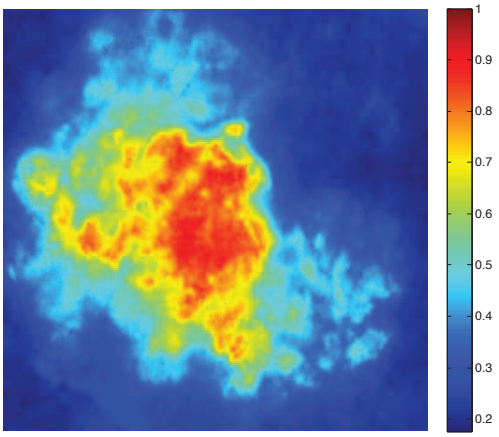
As a conclusion, we propose and compare two different quantitative estimation ap-



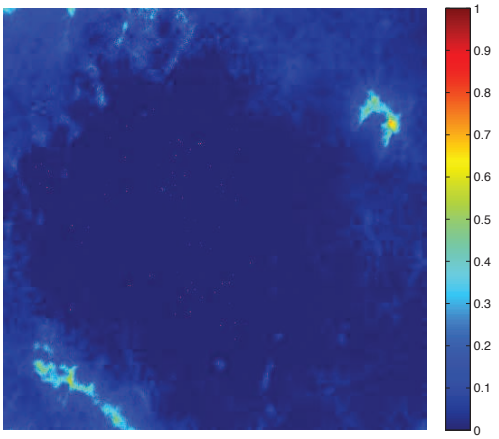
(a) Reconstructed Color Image



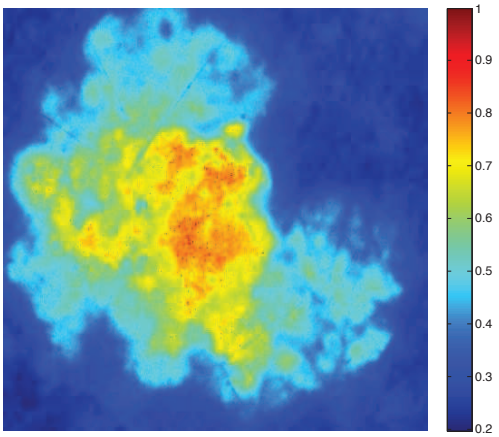
(b) Hemoglobin by NMF



(c) Melanin by NMF

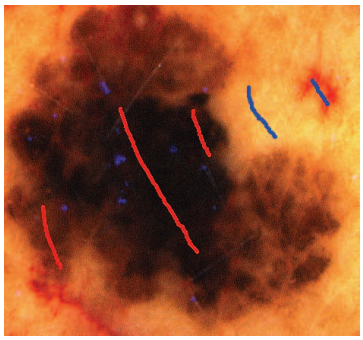


(d) Hemoglobin by MF

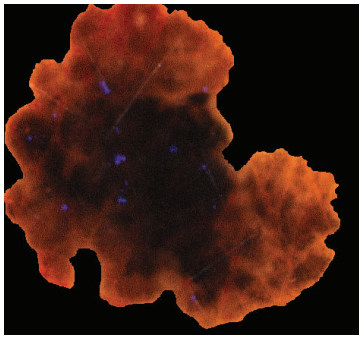


(e) Melanin by MF

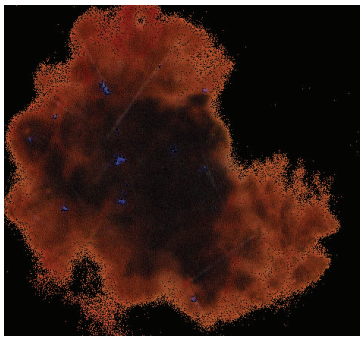
Figure 4.17: Comparison of hemoglobin and melanin concentration cartographies on 'Melanoma'. 'MF': model-fitting method.



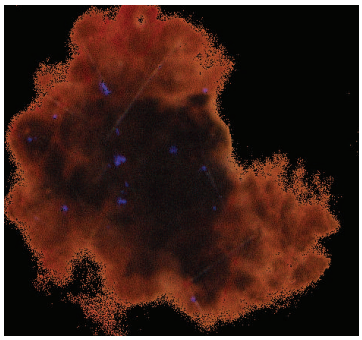
(a) Seed Map



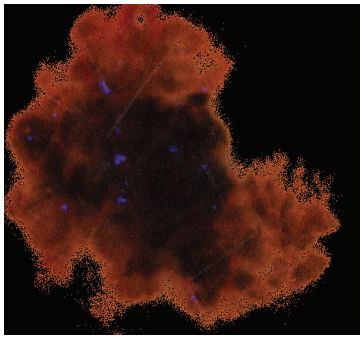
(b) Ground Truth



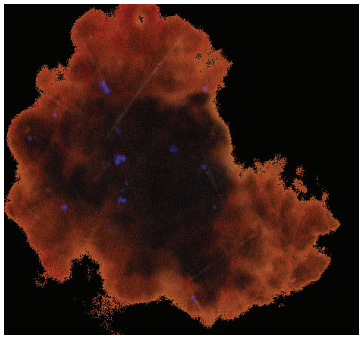
(c) Classic GC



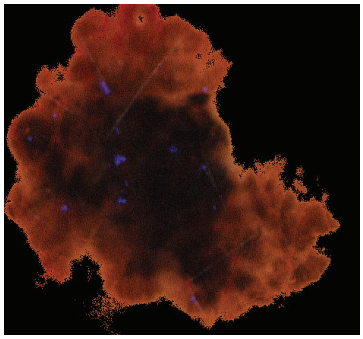
(d) '4-level' GC with NMF



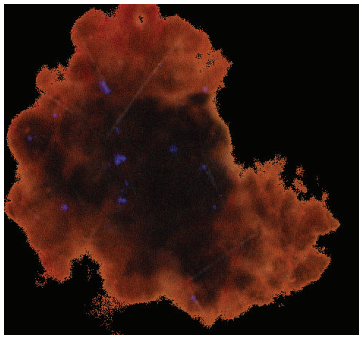
(e) '4-level' GC with MF



(e) '26-level' GC



(f) '27-level' GC with NMF



(g) '27-level' GC with MF

Figure 4.18: Quantitative evaluation of skin chromophore quantification approaches via comparison of segmentation results on multi-spectral 'Melanoma' images. 'GC': graph-cut.

Table 4.2: Comparison of average segmentation accuracy, quantified by Dice similarity coefficient, false negative rate and false positive rate. Borders of melanomas manually drawn by 1 expert dermatologist are considered as the ground-truth reference for accuracy assessment. Value in bold corresponds to the best performance.

Criterion \ Method	Classic GC	4-level NMF	4-level MF	26-level GC	27-level NMF	27-level MF
DSC	0.943	0.950	0.954	0.962	0.963	**0.965**
FNR	0.026	0.011	0.009	**0.008**	**0.008**	**0.008**
FPR	0.091	0.092	0.085	0.068	0.067	**0.065**

proaches on multi-spectral skin images using NMF and model fitting. By means of two comparative experiments based on dermatologic knowledge and graph-cut segmentation, we show that model-fitting approach obtains more accurate quantitative estimation of skin hemoglobin and melanin. In future work, scattering and penetration depth will be taken into account in skin optics model.

4.5 Pigmented Skin Lesion Segmentation

Objective and accurate delineation of pigmented skin lesions (PSLs) without prior knowledge provided by expert dermatologists is an ultimately ideal first step in computer-aided skin cancer (melanoma) diagnosis system. Since objectivity and accuracy of a segmentation approach are a prerequisite for good feature extraction of PSLs while automation (unsupervised framework) is highly efficient, extensively applicable (requires no expertise in dermatology), thus commercially attractive, automatic PSLs segmentation techniques have increased in popularity, especially in cooperation with dermatoscopy.

Dermatoscopy, also known as dermoscopy or epiluminescence microscopy, is a non-invasive diagnostic technique involved in the examination of skin lesions with a dermatoscope, which is typically a hand-held optical magnifier equipped with a non-polarized light source, a transparent plate and a liquid medium between the instrument and the skin, and allows inspection of skin lesions. It has been reported that compared to visual diagnosis of melanoma, diagnostic accuracy by dermoscopy is significantly increased from 20% to 30% [13]. This improvement in diagnostic accuracy, however, is seen primarily with experts who have specialized training in dermoscopy, or when the user applies specific diagnostic algorithms that are often not practical in the clinical setting. In the expert-diagnosis approach, reproducibility remains a problem since even experienced dermatologists do not always show a reliable intra- and inter-operator agreement in delineation of lesion border. For user-applied specific diagnosis approach, supervised segmentation methods require *a prior* knowledge input by analysts, such as

samples of lesion and skin pixels (e.g. initializations of Boykov-Jolly method [23] and Lazy Snapping [65]), or an approximation of lesion area (e.g. initialization of GrabCut method [78]). In addition, user-specified initialization should be provided on each particular image being analyzed and the performance of this kind of approach also relies on the initial values. Thus, although supervised PSLs segmentation methods may be very effective, they are too time-consuming to be practically applicable in real-time clinical examination.

Due to the fact mentioned above, in recent years, there has been increasing interest in developing automatic segmentation techniques for dermoscopic images of PSLs from the surrounding normal skin. The resulting boundary delineation not only provides a basis for calculation of important clinical features such as color, texture and especially the properties of lesion border (e.g. size, shape and irregularity), but it is also crucial for extraction of discriminating dermoscopic features such as atypical pigment networks and radial streaming [69]. Most common automatic segmentation algorithms rely on techniques based on intensity histogram thresholding [96, 84, 29, 80], for instance, Xu et al. [96] proposed a heuristic method using double thresholds to automatically segment lesion from skin through a few selected border points. Silveira et al. [84] recently developed an automatic adaptive thresholding (AT) approach that segments an image by comparing color of each pixel with a threshold. Other approaches includes PDE based approaches, edge-based approaches [79, 39], clustering [68, 100] and region-based techniques [84, 51, 31]. For the last type of approaches, Celebi et al. [31] recommended a fast and unsupervised approach based on the statistical region merging algorithm (SRM). SRM was proven to be a robust segmentation algorithm for segmentation of color images. Iyatomi et al. [51] introduced a dermatologist-like tumor area extraction algorithm. This segmentation algorithm first performs an initial segmentation by finding high-frequency components and then thresholding using the Otsu method. Isolated areas are separated and small regions are merged until they are of sufficient size. Afterwards, subset of regions which is considered to belong to the lesion is selected via elaborate rules. Finally, to mimic dermatologists' tendency to conservatively segment the lesion, the border is slightly expanded by a region growing approach. It is worth noting that all these automatic PSLs segmentation approaches reviewed above are on the basis of color information/pixel intensity solely. Although researchers have also attempted to explore different color spaces to enhance the segmentation performance, color information alone is insufficient for adequate and reliable automated segmentation of PSLs [100]. Properties of PSLs and skin conditions present large variabilities among dermoscopic images, ranging from homogeneously distributed color (cf. Figure 4.20 (a))

to non-uniform and variegated pigmentation (cf. Figure 4.32 (a)), from well-defined (cf. Figure 4.20 (a) and Figure 4.32 (a)) to smoothly transited edges (cf. Figure 4.27 (a) and Figure 4.29 (a)), from high contrast (cf. Figure 4.30 (a)) to low contrast (cf. Figure 4.34 (a)), from sizes smaller than a pin's head to diameters larger than 15 mm. Therefore, the implementation of an accurate automatic segmentation method for pigmented skin lesions in dermoscopy images remains a challenge. In addition, irregular shapes (cf. Figure 4.31 (a)), specular reflection (cf. Figure 4.28 (a)) and the presence of dark heavy hairs along with oil bubbles (cf. Figure 4.34 (a) and Figure 4.35 (a)) occlude some useful information of lesions required to perform an accurate segmentation. For instance, thresholding-based methods achieve undesirable results when contrast is low between lesion and skin; edge-based approaches perform poorly when boundary of the lesion is not well defined. Smooth transition between skin and lesion allows the presence of gaps in edges where the contour may leak. Another difficulty is due to the edge pixels that do not belong to real lesion boundary. They can be the results of artifacts such as hair, specular reflection or even irregularity of the skin texture and they may prevent contour to converge to the lesion boundary; region-based approaches have difficulties when lesion or skin regions are textured, which often leads to over-segmentation. In conclusion, existing automatic PSLs segmentation approaches do not necessarily perform well for specific problems of dermoscopic PSLs delineation and it appears unlikely that one particular method could outperform all other segmentation methods for any lesion [101].

In this section, we propose a novel framework of fully automatic dermoscopic PSLs image segmentation based on the multi-feature graph cuts segmentation approach introduced in Chapter 3. Under this framework, clustering and thinning approaches are co-operating with basic mathematical morphological pre/post-processing to accurately localize seed regions without user intervention or supervision. Subsequently, both color intensities and extracted chromophores of an automatically binary classified pixel within the seed regions are collected via the neighborhood template to construct a multi-dimensional feature vector. Learned by a nonlinear SVM classifier is trained on seeded regions. Soft classification on lesion and skin pixels in terms of posterior probability is adopted to properly define the likelihood energy of binary labeling problem, which can be globally optimized by minimization of graph cuts. In the following text, we denote the proposed *Automatic Pigmented Skin Lesion Segmentation* framework as the *APS* technique.

4.5.1 Methodology

The proposed APS framework is characterized in the following two main stages where it introduces novelties in the state-of-the-art approaches designed particularly for PSLs image segmentation:

1. Auto-Seeding: seeds are used to build a training set for binary classification of foreground lesion and background skin. In addition, they provide hard constraints for the construction of graph. The performance of final segmentation depends on the soft classification: seeds must be really either background skin or foreground lesion. In this context, we propose an unsupervised approach for selection of lesion (object) seed regions based on *k*-MEANS++ clustering and skeleton pruning [14]. Clustering based on color characteristics of typical lesions provides the coarse pre-segmentation that roughly locates the lesion area. Subsequent skeletonization processing with efficient redundant skeleton branches pruning extracts stroke-like lesion seed regions. The resulted lesion seed distributions lie on axes between lesion center and vertices of the polygon simplified by Discrete Curve Evolution (DCE) [62]. This kind of distribution ensures diversity of training samples from definite inner lesion pixels to possible peripheral lesion pixels. For background skin seeds, we propose a simple approach based on the fact that most of skin lesions approximately locate in the center of the dermoscope during the acquisition process. A frame of certain thickness is added to the border of an acquired dermoscopic PSL image and pixels within the frame region are labeled as "skin". However, in cases of large scale skin lesions that cover the border of the dermoscopic image, we left pixels of overlaps between lesion cluster and skin border unlabeled. In addition, basic morphological operations like erosion, closing and dilation are served as pre- and post-processing of the proposed seed generation approach. In the following text, we denote the entire coarse-to-fine automatic seed-generation scheme as the *Auto-Seeding* approach.

2. Feature configuration: the proposed PSLs image segmentation procedure is designed to incorporate with multiple visual features of pixels in a given dermoscopic PSL image. According to the research presented in Section 3.4, Chapter 3, our proposed segmentation algorithm under the Graph-Cut/MAP-MRF framework is based on binary classification. A nonlinear SVM classifier is served as a probabilistic interface responsible for passing the posterior classification probabilities to the likelihood energy function of a binary segmentation problem. To train a classifier within one image, the training samples are automatically chosen

via the proposed Auto-Seeding approach. Basically, color information of seeds is sufficient to train a classifier to estimate the distribution of lesion and skin pixels. With the generic neighborhood template, we can improve the segmentation by adding additional features like texture to the feature vector. Moreover, intrinsic properties of skin lesions objectively interpreted can be collected to enhance the discriminability of our algorithm. Here we propose to combine other intrinsic characteristics of pigmented skin lesion related to skin chromophores with its color information to construct a powerful feature vector. We expect this kind of feature configuration to yield more accurate PSLs delineations. Same as the feature configuration described in Section 3.4, color information is interpreted in RGB color space and texture feature is encoded using GLBP. As the newly introduced feature, we choose the concentrations of two predominant chromophores of human skin, epidermal melanin and dermal hemoglobin, quantified by the proposed model fitting approach (cf. Section 4.3.4).

A flowchart of the proposed APS framework is depicted in Figure 4.19.

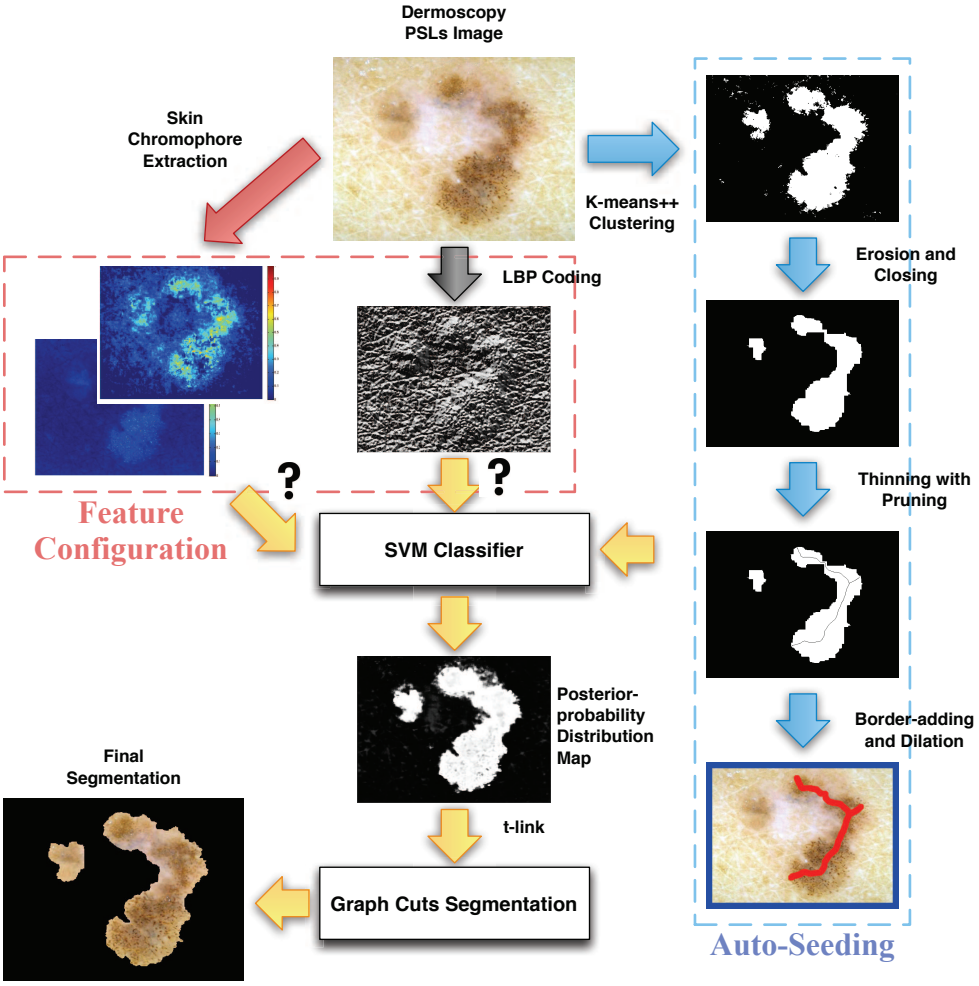


Figure 4.19: A schematic of our proposed **A**utomatic **P**igmented Skin Lesions **S**egmentation (APS) approach which is particularly suitable for dermoscopic images.

4.5.2 Automatic Hard Constraint Generation using Clustering and Morphological Processing

4.5.2.1 *K*-Means Clustering Algorithm and Its Augmentation

Clustering literally means grouping a set of data instances into subsets in such a manner that instances in the same group (called *cluster*) are more similar to each other than to those in the different groups (clusters). As a fundamental problem in data mining, clustering technique is also widely used in machine learning, pattern recognition, and image analysis as a statistical data analysis tool. In context of machine learning, clustering can be referred as a term "automatic classification", since classification is mostly employed as a supervised learning method while clustering for unsupervised learning method, except that selection of appropriate clustering algorithms and parameter set-

tings (e.g. number of expected clusters, initial condition, distance function, etc.) depend on the individual data set and intended use of the results. Although clustering and classification both aim to partition set of observations into different categories (clusters or classes), they differ from each other in the fact that principle of clustering is intrinsic, which results in descriptive assessments, while that of classification is extrinsic (incorporation with reference set of observations and their corresponding classes), which results in predictive assessments.

Currently, there exists numbers of clustering algorithms as the notion of "cluster" cannot be precisely defined [40]. A simple but clear way of categorization of these clustering algorithms is based on their distinct "cluster models". For instance, typical cluster models includes connectivity models (e.g. hierarchical clustering), centroid models (e.g. k-Means algorithm), distribution models (e.g. Expectation-maximization algorithm), density models (e.g. DBSCAN) and so on. Among various clustering algorithms, k-Means is the most widely used technique in fields of machine learning as well as image analysis for its simplicity and practical observed speed. Originally devised by MacQueen [67] in 1967, k-Means is an unsupervised clustering algorithm whose goal is to solve an optimization problem: find multiple cluster centers and assign each observation to its nearest cluster center, depending on the minimum squared Euclidean distance in between. Solving this problem is theoretically NP-hard, even with just two clusters [5]. Fortunately, an efficient heuristic algorithm proposed by Lloyd [66] is able to find an approximate solution by iteratively refining the clustering encoded by cluster centers. The basic idea of Lloyd's algorithm is to find a clustering structure (Voronoi diagram) that minimize the Sum of Squared Error (SSE), which measures the distance of each instance to its representative values. Problem of global optimization of SSE may be solved by exhaustively enumerating all partitions, which is extremely time consuming, or by giving an approximate solution using heuristics, which is capable to be converged quickly to a local minimum. The latter opinion, adopted by Lloyd's algorithm, is the most common intuitive and frequently used alternative. That is the reason why Lloyd's algorithm is served as the standard algorithm for k-Means clustering.

For a brief view of standard k-Means algorithm (Lloyd's algorithm), suppose that a set of n input observations are given as a d-dimensional real vector $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\} \subset \mathbb{R}^d$, the Lloyd's algorithm partitions input data into a clustering set of k $(k < n)$ clusters, $\mathcal{C} = \{C_1, C_2, \cdots, C_k\}$ with a set of corresponding means, $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_k\} \subset \mathbb{R}^d$,

so as to minimize the objective (potential) function:

$$\phi(\mathcal{X}, \mathcal{M}) = \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in C_i} ||\mathbf{x}_j - \mathbf{m}_i||^2 \tag{4.42}$$
$$\text{where} \quad \mathbf{m}_i = \frac{\sum_{\mathbf{x}_j \in C_i} \mathbf{x}_j}{|C_i|}.$$

Here, the potential (distortion) ϕ is defined as total summation of the distances of data points from its cluster center. To minimize ϕ, Lloyd's algorithm adopts an iterative refinement technique (cf. Algorithm 3): starting with a set of randomly chosen initial centers, one repeatedly assigns each input point to its nearest center, and then recomputes the centers given the point assignment. This local search, called Lloyd's iteration, continues until the solution does not change between two consecutive rounds.

Algorithm 3 k-MEANS (Lloyd's) Algorithm

Require: k: number of expected clusters, $\{\mathbf{x}_i\}_{i=1}^{n}$: sample datapoints, $\{C_j^{(0)}\}_{j=1}^{k}$: optional user-supplied initial clustering, t_{max}: optional maximum iteration number.
Ensure: $\{C_j\}_{j=1}^{k}$: final clustering

1: Initialize k clusters: $C_1^{(0)}, \cdots, C_k^{(0)}$ either randomly or given by user-specified cluster-labeling. $t \leftarrow 0$.
2: Calculate centers for initial k clusters as:

$$\mathbf{m}_j^{(0)} = \frac{\sum_{\mathbf{x}_i \in C_j^{(0)}} \mathbf{x}_i}{|C_j^{(0)}|}.$$

3: **while** $\{\mathbf{m}_j^{(t)}\}_{j=1}^{k} \neq \{\mathbf{m}_j^{(t-1)}\}_{j=1}^{k}$ $(t \geq 1)$ or $t < t_{max}$ **do**
4: Label each datapoint $\mathbf{x}_i$ based on cluster centers $\mathbf{m}_j^{(t)}$ as:

$$l^{(t+1)}(\mathbf{x}_i) = \arg\min_j ||\mathbf{x}_i - \mathbf{m}_j^{(t)}||^2.$$

5: Update the clustering as:

$$C_j^{(t+1)} \leftarrow \{\mathbf{x}_i : l^{(t+1)}(\mathbf{x}_i) = j\}.$$

6: Re-calculating the centers for all updated clusters:

$$\mathbf{m}_j^{(t+1)} = \frac{\sum_{\mathbf{x}_i \in C_j^{(t+1)}} \mathbf{x}_i}{|C_j^{(t+1)}|}.$$

7: **end while**

In computer vision and image segmentation, k-MEANS often serves as a preprocess-

Algorithm 4 *k*-MEANS++ (Arthur's) Algorithm

Require: k: number of expected clusters, $\{\mathbf{x}_i\}_{i=1}^{n}$: sample datapoints.
Ensure: $\{C_j\}_{j=1}^{k}$: final clustering

1: Initialize one cluster center $\mathbf{m}_1^{(0)}$ uniformly at random from $\mathcal{X}$. $\mathcal{M} \leftarrow \{\mathbf{m}_1^{(0)}\}$.
2: **while** $j < k$ **do**
3: $\quad j \leftarrow j + 1$.
4: $\quad$ Selecting $\mathbf{m}_j^{(0)} = \mathbf{x}' \in \mathcal{X}$ with probability $\frac{D(\mathbf{x}')^2}{\sum_{\mathbf{x} \in \mathcal{X}} D(\mathbf{x})^2}$.
5: $\quad \mathcal{M} \leftarrow \mathcal{M} \cup \{\mathbf{m}_j^{(0)}\}$.
6: **end while**
7: Proceed as the standard *k*-MEANS (Algorithm 3) from Step 3 to Step 7.

ing step for other algorithms, for example, to find a starting configuration. In our case, it helps to produce a primary yet coarse segmentation before we apply the effective skeleton proposed by Bai et al [14].

Nevertheless, despite its popularity for general clustering, *k*-MEANS suffers from two major shortcomings: (a) the running time can be exponential in the worst case; (b) the final solution is locally optimal, it can be very far away from the global optimum, even under repeated random initializations (cf. Figure 4.20 (d), Figure 4.21 (d) and Figure 4.22 (d)). Therefore, an augmented version named *k*-MEANS++ was proposed by Arthur and Vassilvitskii [12]. They proposed to improve the initialization procedure of *k*-MEANS by choosing random starting centers with very specific probabilities. *k*-MEANS++ is proven to $O(\log k)$-competitive with the optimal clustering. The general framework of *k*-MEANS++ is presented in Algorithm 4.

4.5.2.2 Comparison of Accuracy between *k*-Means and *k*-Means++

In order to evaluate the performance of *k*-MEANS++ against *k*-MEANS in the context of pre-segmentation of dermoscopic PSLs image, we tested each method on 3 dermoscopy images, including 1 dysplastic nevus (cf. Figure 4.20 (a)), 1 melanoma (cf. Figure 4.21 (a)) and 1 basal cell carcinoma (cf. Figure 4.22 (a)). As both methods are randomly initialized (for *k*-MEANS++ only the first cluster center is randomly selected), we ran multiple trials (attempt $= 1, 10, 80$ and 100) for each case. Here, the internal evaluation of clustering performance is measured by the calculating the value of the potential ϕ, while the external evaluation is quantified in terms of Jaccard index, a metric for similarity measurement between two datasets. In the context of segmentation of skin lesions, the Jaccard index measures how the pixels classified as lesion by clustering agree with the pixels classified as lesion by the expert dermatologist (manual segmentation served as the ground truth, cf. Figure 4.20 (b), Figure 4.21 (b) and Figure 4.22 (b)). Given the

following definitions:

TP: true positive, lesion pixels correctly classified as lesion,

FP: false positive, skin pixels incorrectly identified as lesion,

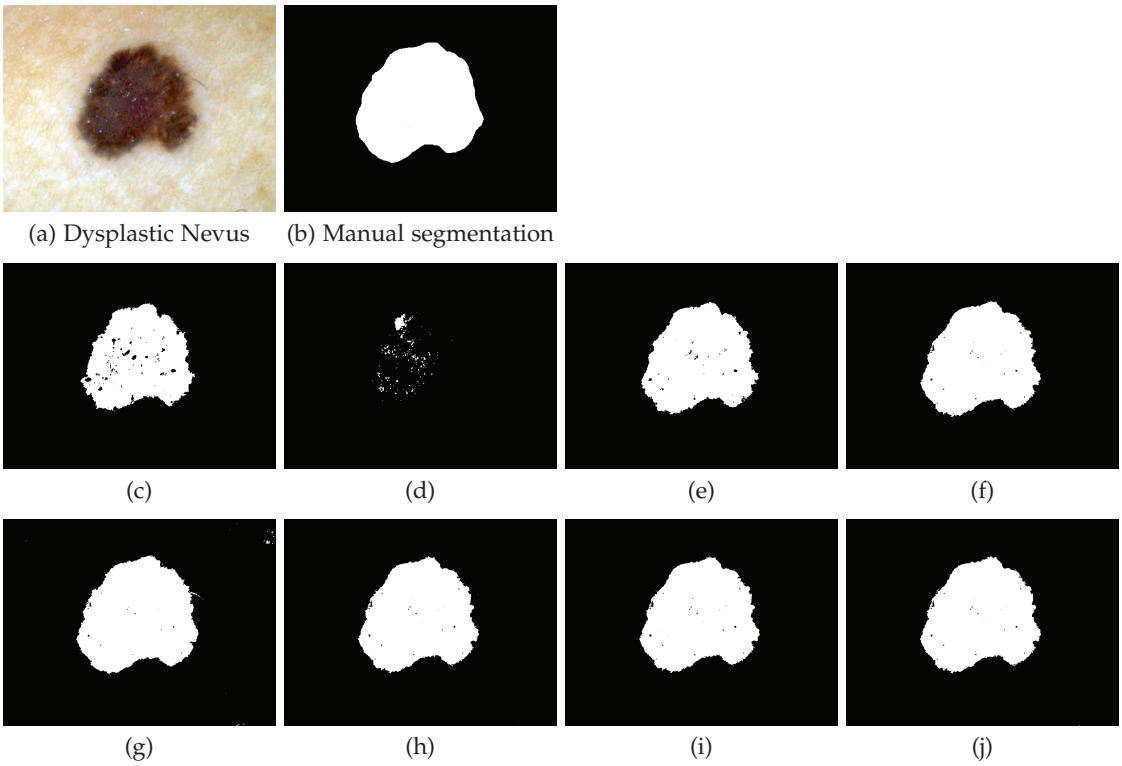
FN: false negative, lesion pixels incorrectly identified as skin,

the Jaccard index is given by:

$$J(SG, GT) = \frac{\#(SG \cap GT)}{\#(SG \cup GT)} = \frac{\#TP}{\#TP + \#FP + \#FN}. \tag{4.43}$$

where the definitions of SG and GT are given Section 4.3.5.

Clustering-based segmentation results by k-Means and k-Means++ are shown in Figure 4.20 through Figure 4.22, along with the internal/external evaluations displayed in Table 4.3 through Table 4.4. Based on both qualitative and quantitative evaluations, we can observe that k-Means++ substantially outperformed k-Means in terms of clustering accuracy.

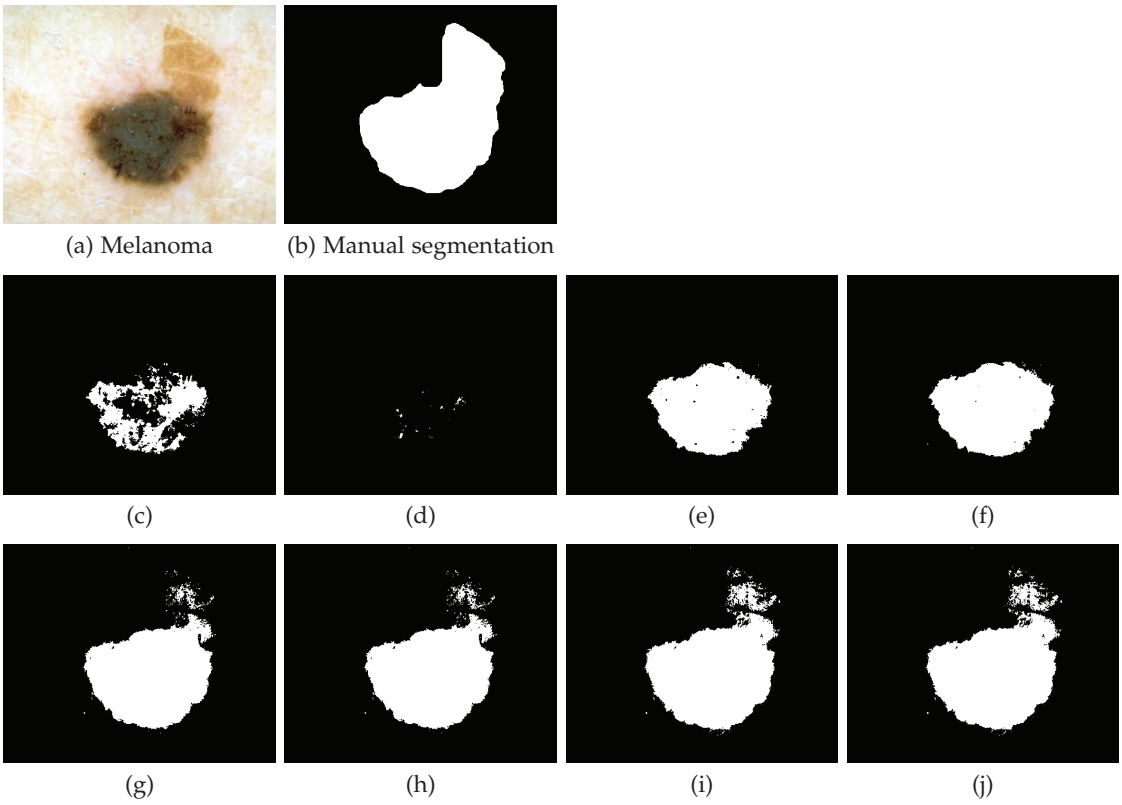


(a) Dysplastic Nevus (b) Manual segmentation

(c) (d) (e) (f)

(g) (h) (i) (j)

Figure 4.20: Comparison between clustering results of k-Means algorithm and k-Means++ algorithm on dysplastic nevus (Ref. Fclo10) subject to increasing number of attempts (first column: attempt = 1, second column: attempts = 10, third column: attempts = 80, last column: attempts = 100).



(a) Melanoma (b) Manual segmentation

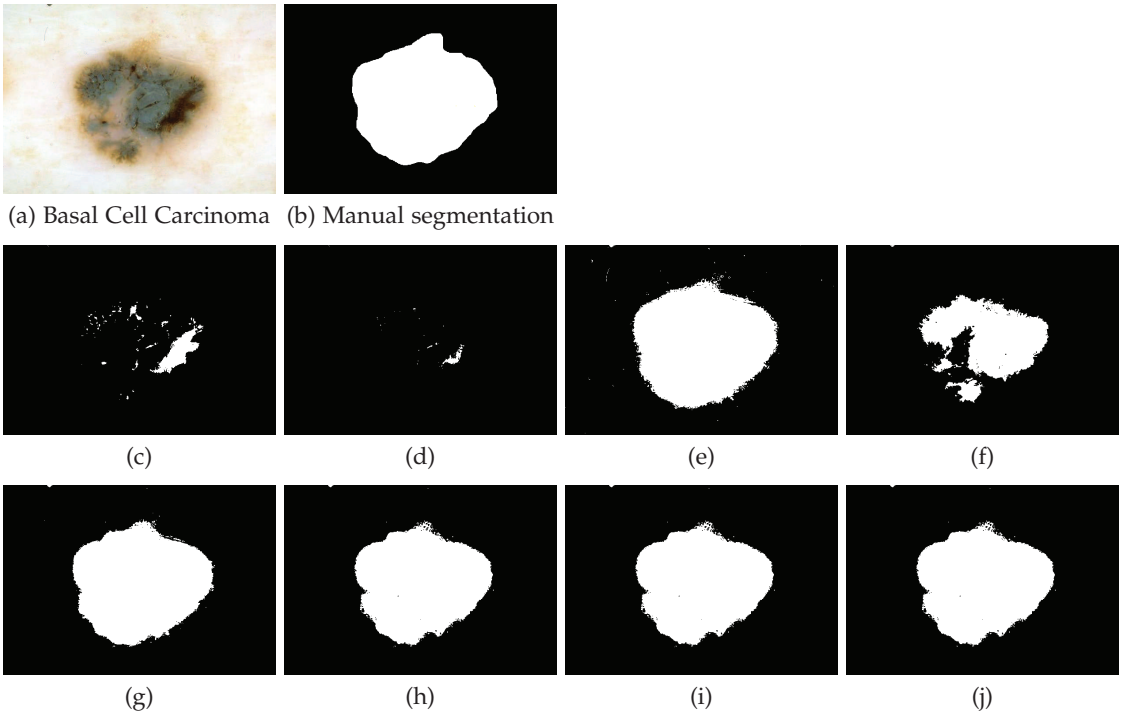
(c) (d) (e) (f)

(g) (h) (i) (j)

Figure 4.21: Comparison between clustering results of k-Means algorithm and k-Means++ algorithm on melanoma (Ref. Nblo06) subject to increasing number of attempts (first column: attempt = 1, second column: attempts = 10, third column: attempts = 80, last column: attempts = 100).

Table 4.3: Internal evaluation of k-Means algorithm and k-Means++ algorithm on 3 dermoscopic PSLs images (Dysplastic nevus in Figure 4.20 (a), melanoma in Figure 4.21 (a) and basal cell carcinoma in Figure 4.22 (a)) subject to different attempts. For both algorithms, k-Means and k-Means++, we list the actual value of potential ϕ. Additionally, for k-Means++, percentage of improvement over k-Means is presented: $\Delta = 100\% \cdot (1 - \frac{k-\text{Means++}}{k-\text{Means}})$.

Input	Approach \ Attempt	1	10	80	100
Fclo10	k-Means	$3.8496 \cdot 10^4$	$1.0534 \cdot 10^4$	$6.7350 \cdot 10^3$	$4.4481 \cdot 10^3$
	k-Means++	$2.9667 \cdot 10^3$	$1.7060 \cdot 10^3$	$1.6749 \cdot 10^3$	$1.6522 \cdot 10^3$
	Δ (%)	92.29	83.80	75.13	62.85
Nblo06	k-Means	$3.8766 \cdot 10^4$	$1.1121 \cdot 10^4$	$6.5221 \cdot 10^3$	$3.9642 \cdot 10^3$
	k-Means++	$1.7394 \cdot 10^3$	$1.7394 \cdot 10^3$	$1.7103 \cdot 10^3$	$1.7103 \cdot 10^3$
	Δ (%)	95.51	84.36	73.78	56.86
Nhlo11	k-Means	$3.9974 \cdot 10^4$	$1.1783 \cdot 10^4$	$7.2221 \cdot 10^3$	$4.6626 \cdot 10^3$
	k-Means++	$1.5972 \cdot 10^3$	$1.4104 \cdot 10^3$	$1.4104 \cdot 10^3$	$1.4102 \cdot 10^3$
	Δ (%)	96.00	88.03	80.47	69.75



(a) Basal Cell Carcinoma (b) Manual segmentation

(c) (d) (e) (f)

(g) (h) (i) (j)

Figure 4.22: Comparison between clustering results of *k*-Means algorithm and *k*-Means++ algorithm on basal cell carcinoma (Ref. Nhlo11) subject to increasing number of attempts (first column: attempt = 1, second column: attempts = 10, third column: attempts = 80, last column: attempts = 100).

Table 4.4: External evaluation of *k*-Means algorithm and *k*-Means++ algorithm through segmentation of 3 dermoscopic PSLs images (cf. Figure 4.20 through Figure 4.22) subject to different attempts, quantified by Jaccard index (%). Borders of PSLs manually drawn by 1 expert dermatologist are considered as the ground-truth reference for accuracy assessment.

Input	Jaccard Index (%) Attempt / Approach	1	10	80	100
Fclo10	*k*-Means	73.92	2.57	81.90	87.38
	k-Means++	91.69	88.24	87.62	88.66
Nblo06	*k*-Means	27.01	0.41	53.42	56.14
	k-Means++	69.78	69.78	75.61	75.61
Nhlo11	*k*-Means	7.78	0.99	93.06	51.85
	k-Means++	91.42	86.20	86.20	86.17

4.5.2.3 Skeletonization with Efficient Pruning

In order to efficiently sample pixels of a PSL image from the lesion area, we propose to generate seeds from the *k*-MEANS++ clustering results based on skeletonization (or thinning) algorithms. The idea is to automatically extract stroke-like regions (seeds) in an intelligent way to mimic the supervised seeding process: try to label the most *representative* pixels of large spatial dispersion but limited within the possible lesion area as "lesion" seeds while the amount of seeds are limited. Skeletonization (or thinning) suits well for this kind of problem. By definition, skeletonization is the act of identifying the skeleton of an object, which is presumed to represent the shape of the object in a relatively small number of pixels. Skeletonization is usually one step in preparing an image for further processing. The nature of the subsequent steps often dictates the properties needed of the skeleton. The majority of existing thinning algorithms are based on the iterative morphological approach firstly proposed by Steniford [87] in 1983. Also, to eliminate possible artifacts in the resulting skeleton, for example, the redundant or spurious skeleton branches, he proposed a pre-processing including smoothing and acute angle emphasis. The Steniford's algorithm is followed by Zhang-Suen algorithm [99], Holt's algorithm [48], and concluded by J. R. Parker [74], who suggested a effective skeletonization framework: Stentiford's preprocessing scheme feeding images into Zhang-Suen's basic algorithm, with Holt's staircase removal as a post-processor. However, these skeletonization algorithms still suffer from the problem of being sensitive to an object's boundary deformation (cf. Figure 4.23 (e-h)): little noise or a variation of the boundary often generates redundant skeleton branches that may seriously disturb the topology of the skeleton's graph. To overcome this problem, Bai et al. [14] in 2005 proposed a skeleton pruning algorithm based on contour partitioning using Discrete Curve Evolution (DCE). It is proven to be stable on shape variation and accurate in generating skeletons without spurious branches (cf. Figure 4.23 (a-d)). Considering the irregularity of the skin pigmented lesion, Bai's skeletonization method is a better alternative than other skeletonization algorithms (e.g. Parker's method), as shown in Figure 4.24.

4.5.2.4 Pre-processing and Post-processing of Auto-Seeding

Before we apply clustering to pre-segment the original PSLs images, it is necessary to eliminate artifacts like heavy dark hairs which occlude the visual cues of lesion or skin. The images were filtered with the morphological closing filter using a disk as structuring element (diameter=10 pixels) and subsequently with a median filter (3×3 neighborhood). After clustering, mathematical morphological erosion is then applied

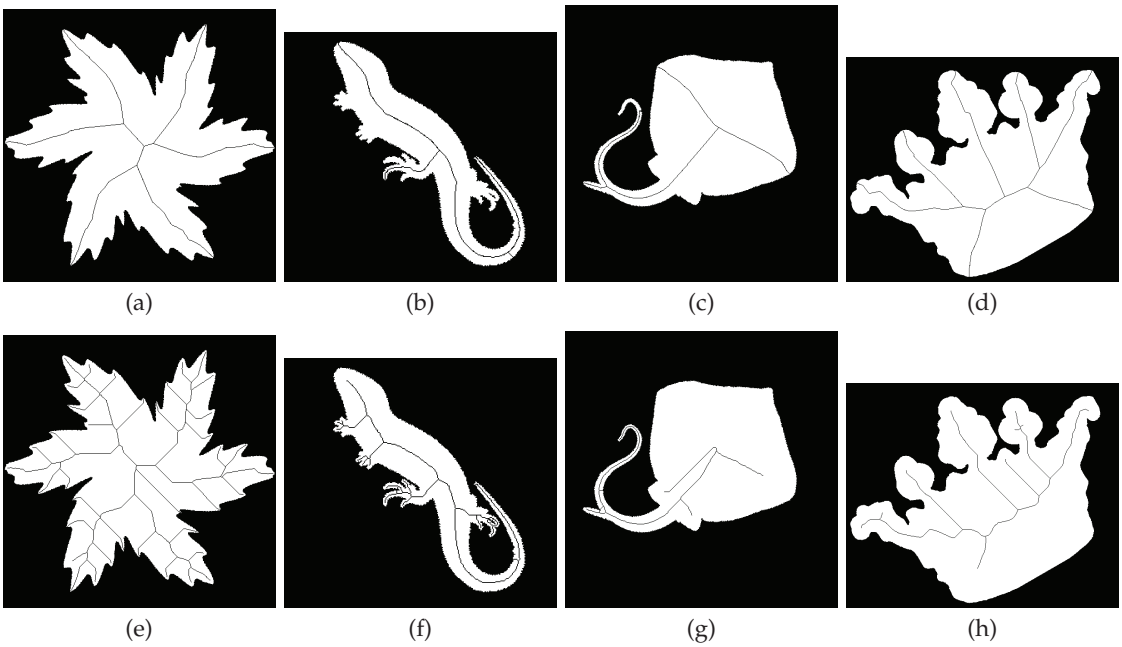


Figure 4.23: Comparison between skeletonizing (thinning) results obtained by Bai's skeleton pruning method (first row) and Parker's thinning approach (second row) on Mpeg 7 shape database.

to the binary mask obtained from the binary clustering ("lesion" and "skin" clusters) to eliminate small isolated regions and closing operation is used to fill the holes. In our experiment, erosion is applied using a disk of diameter $d_{erode} = 15$ pixels and closing using a disk of diameter $d_{close} = \lfloor \frac{1}{6} \cdot \sqrt{\sum_p I_p} \rfloor$ with I_p denoting the pixel value of the binary mask image. The selection of size of structural element for erosion and closing seems to be a safe choice since it allows refinement of the clustering results without deforming the original topology of the "lesion" cluster. Finally, the largest connected component in the binary image representing the inner region of skin lesion is ready to be skeletonized into *seeds*. In the last step of the Auto-Seeding stage, morphological dilation is adopted to generate the final stroke-like "lesion" seeds from the extracted skeletons (painted in red). In order to prevent seeds from exceeding the lesion area marked via clustering, a slightly smaller disk element of size $d_{dilation} = 10$ pixels is used for dilation. For "skin" seeds, we simply add a frame on the border of the original PSLs images with thickness of 20 pixels and paint the frame in blue color. Results of the proposed Auto-Seeding approach on 60 dermoscopic PSLs images are present in Figure 4.25 and Figure 4.26.

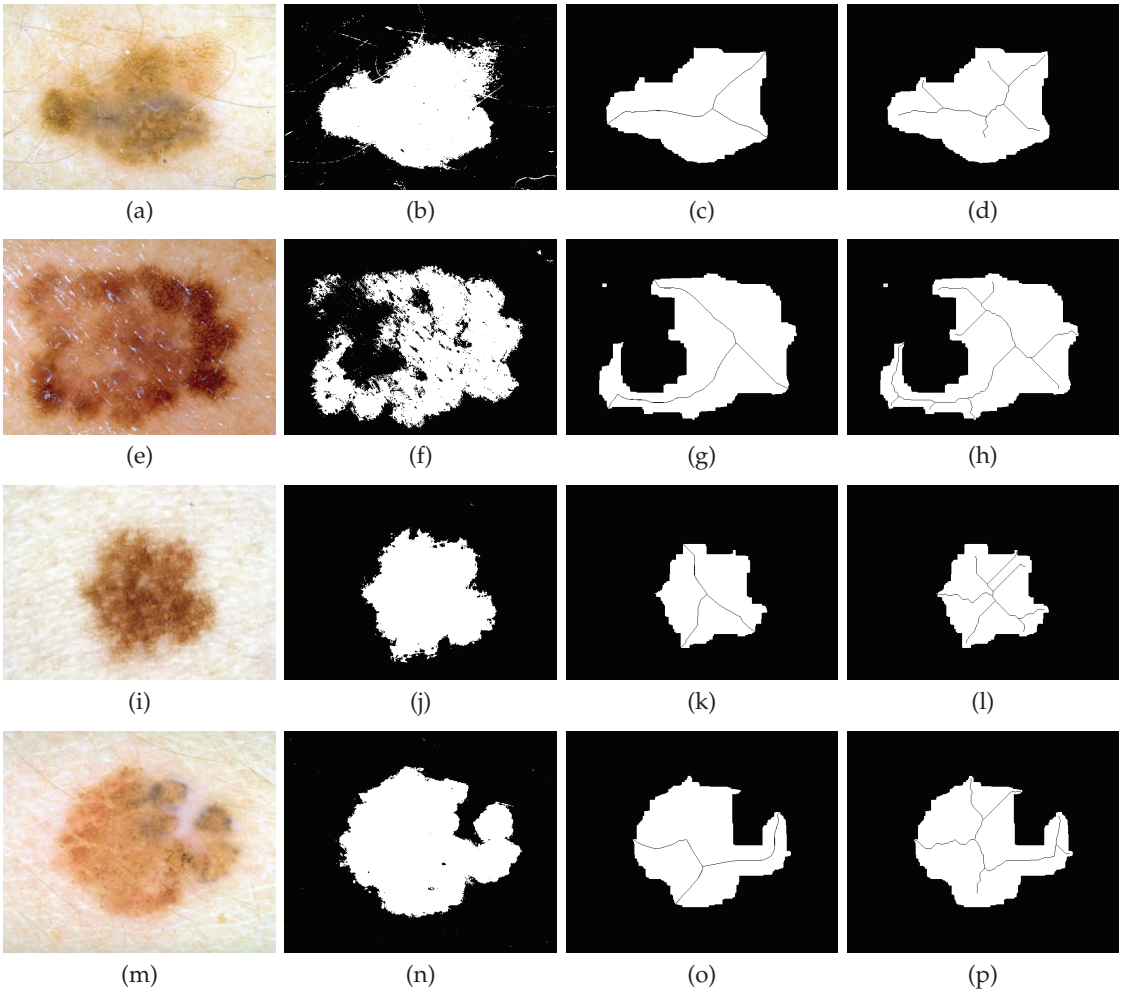


Figure 4.24: Comparison between skeletonizing (thinning) results obtained by Bai's skeleton pruning method (third row) and Parker's thinning approach (last row) on melanomas (a), (c) and dysplastic nevus (i), (m). Second row shows the clustering results obtained by k-MEANS++ algorithm with 1 attempt.

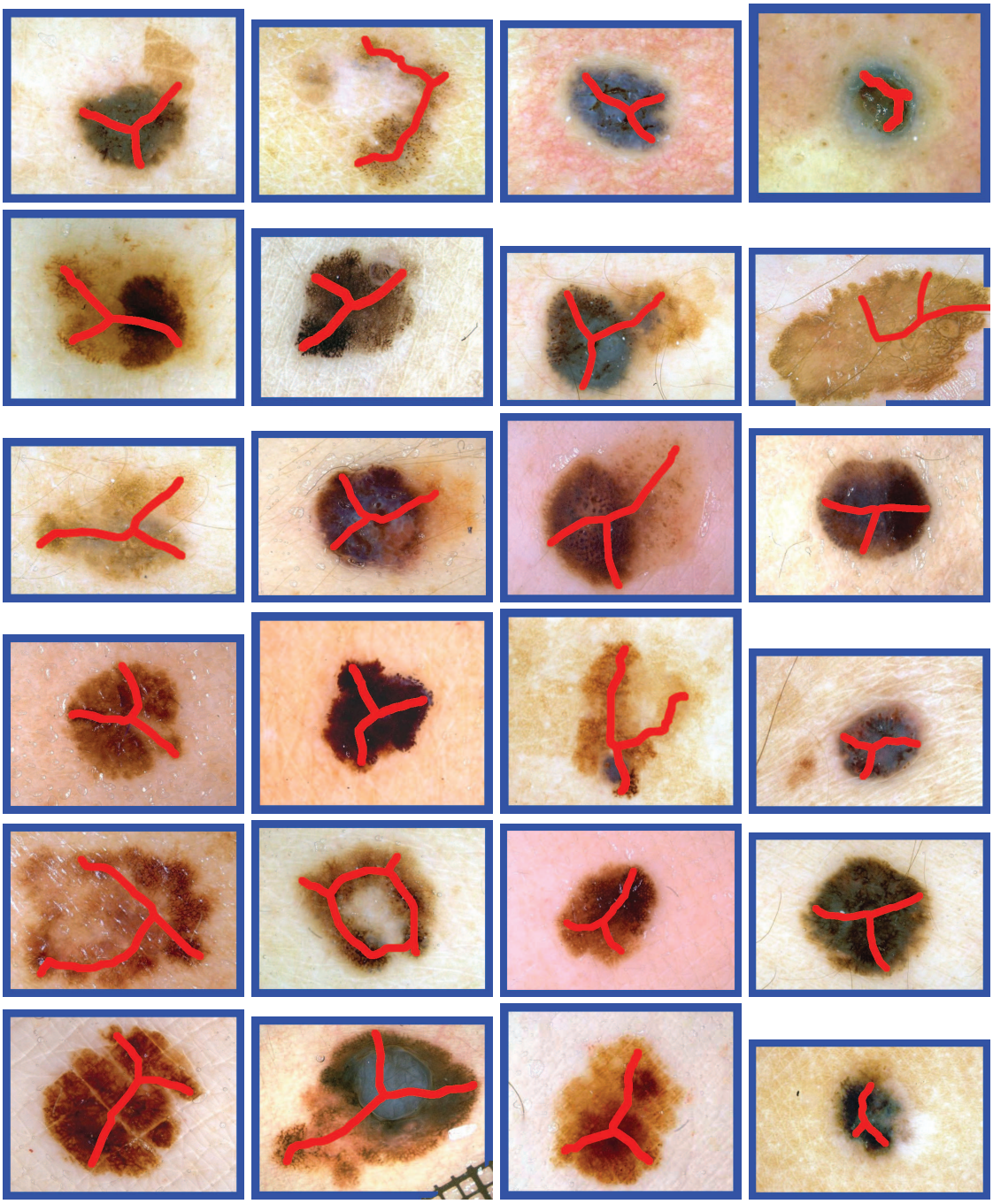


Figure 4.25: Results of the proposed Auto-Seeding approach on 30 dermoscopic images of melanomas and basal-cell carcinoma (BCC).

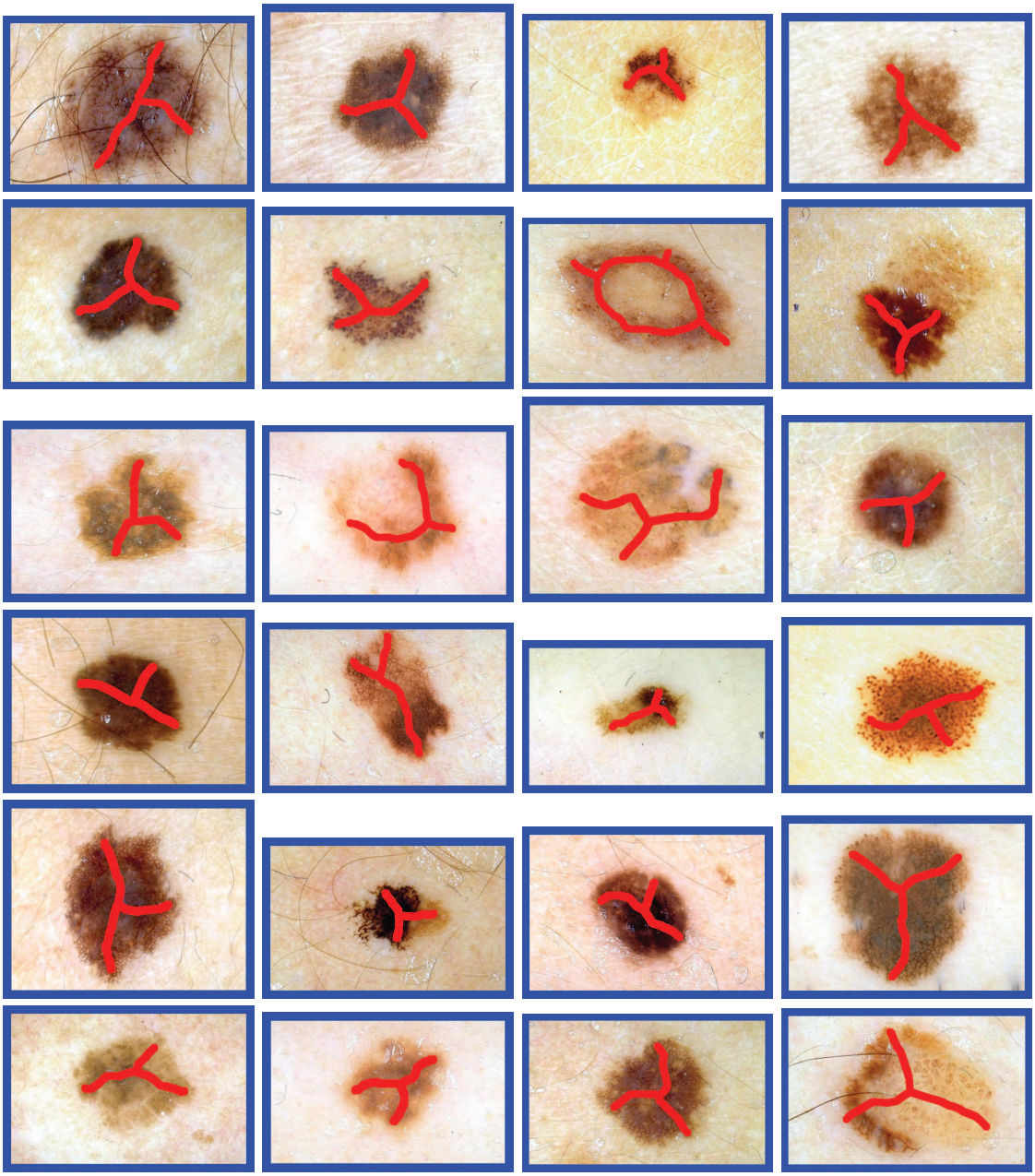


Figure 4.26: Results of the proposed Auto-Seeding approach on 30 dermoscopic images of dysplastic nevi.

4.5.3 Graph Cut based Segmentation with Skin Chromophore Features

4.5.3.1 Feature Configuration

Pigmented skin lesions can be distinct from the surrounding healthy skin tissues in terms of darker color, varied texture and other discriminative morphological features. However, from the histological point of view, concentration and distribution of two primitive skin chromophores, epidermal melanin and dermal hemoglobin, are primary and more reliable discriminators of PSLs since pigmented skin lesions are induced by absence or overmuch of these two chromophores. Based on the research in Section 4.3, we are able to quantify objectively the two chromophores with accuracy, particularly when the proposed model-fitting approach is involve. Therefore, we decide to introduce the chromophore feature into feature configuration under our proposed APS framework. Skin chromophore features, in terms of melanin/hemoglobin concentration distribution maps, are extracted using the proposed model-fitting approach and then combined with color feature to build a new feature vector. We expect that by adding chromophore feature to the feature configuration, SVM classifier can be well trained to improve the discriminability of our APS technique, resulting more accurate PSLs delineation.

4.5.3.2 Experiments and Discussions

In this section, we evaluate both qualitatively and quantitatively the performance of our proposed APS technique under different feature configurations, compared to state-of-the-art graph-cut based segmentation approaches on PSLs delineation. All competitor methods are initialized by our Auto-Seeding approach (cf. Figure 4.25 and Figure 4.26) except for GrabCut, which is initialized manually by one expert dermatologist. We experiment on 100 dermoscopic PSLs images including melanoma, dysplastic nevus, basal cell carcinoma and Spitz-Reed nevus, taken from a dermoscopy atlas [11]. These 24-bit RGB color images were acquired in three university hospitals (University of Graz, Austria, University of Naples, Italy and University of Florence, Italy), with a typical resolution of 768×512 pixels. For each image a manual delineation was performed by a expert dermatologist and the final segmentation results in terms of binary images were collected to build a reference dataset as ground truth.

General experimental settings are as follows:

- Feature: For the proposed APS technique, 3 different types of feature configurations are tested: (a) color feature only, i.e. 75-dimensional color-shape feature vector, (b) the combination of color and texture feature, i.e. 100-dimensional GLBP

texture-color-shape feature vector, (c) the combination of color and skin chromophore features, i.e. 100-dimensional EI/MI-color-shape feature vector. Other graph-cut based approaches are based on color feature only (3-dimensional color feature vector). Note that the neighborhood template (shape feature) is adopted by default in our proposed APS framework and the skin chromophore features are extracted by the proposed model fitting approach (introduced in Section 4.3.4) in terms of concentration distribution maps of epidermal melanin and dermal hemoglobin. For GLBP, parameter σ is fixed at 3 and the kernel size of the gaussian filter is fixed at 7×7.

- Classifier: Nonlinear SVM classifier is adopted in the proposed APS technique to generate more comprehensive posterior-probability distribution maps compared to the results of random forest, which is empirically found resulting *hard* binary classifications of skin and lesion pixels. Parameter of the SVM classifier is estimated using cross-validation method: kernel function is chosen as GRBF, with its parameter γ fixed at 0.5 and the soft-margin parameter C is 10.

- Energy function: Parameter λ is selected as 10 by optimizing the segmentation performance against ground truth on 30 dermoscopic PSLs images. Other settings of parameters particular for Boykov-Jolly approach, Lazy Snapping and GrabCut are detailed in Section 2.4.4.

- Graph: A 8-neighborhood adjacent directed graph is constructed with two terminal nodes s and t, respectively connected to the "lesion" seeds $\mathcal{O}$ and "skin" seeds $\mathcal{B}$. By satisfying the user-specified hard constraints, a minimum cut on a graph can be found with edge weights appropriately defined to generate an optimal segmentation. The assignment of edge weights for the proposed method are given in Table 3.3 while the assignments for other methods can be explicitly found from Table 3.2 to Table 2.4.

The conventional criterion for image segmentation evaluation, like the segmentation error rate (cf. Section 2.5.1, Equation (2.45)), is not sufficient for objective quantitative analysis in context of PSLs delineation. It merely gives quantitative evaluation on whether the delineated lesion area for each dermoscopy PSLs image is "acceptable" or "unacceptable" according to the manual segmentation by expert dermatologist (the ground truth). Therefore, we introduce here two additional criteria, *Precision* and *Recall*, and their derivation, *F-measure*. These criteria are commonly used in engineering and will be redefined in context of PSLs segmentation as follows:

- Precision

  This metric measures the rate of pixels classified as lesion by computer-based approach are identically classified as such by the expert dermatologist, i.e. how accurate the segmentation is:

$$Precision(SG, GT) = \frac{\#(SG \cap GT)}{\#(SG \cap GT) + \#(SG \cap \overline{GT})} = \frac{\#TP}{\#TP + \#FP}, \tag{4.44}$$

- Recall

  This metric measures the rate of pixels classified as lesion by the expert dermatologist are correctly identified as such by computer based approach. i.e. how well the lesion region is segmented:

$$Recall(SG, GT) = \frac{\#(SG \cap GT)}{\#(SG \cap GT) + \#(\overline{SG} \cap GT)} = \frac{\#TP}{\#TP + \#FN}, \tag{4.45}$$

  Note that in scope of classification, *recall* is also known as *sensitivity*.

- F-measure

  Since *precision* and *recall* are two opposing metrics and good segmentation requires high levels of both of them, we introduce here a criterion that combines precision and recall into the harmonic mean, the *F-measure*. Particularly, we consider the harmonic mean of evenly weighted precision and recall, named F_1 measure. In fact, the F_1 measure is equivalent to the Dice similarity coefficient (cf. Section 4.3.5, Equation (4.33)) which evaluates the similarity between graph-cut based segmentation and manual segmentation. We will use the term "Dice similarity coefficient (DSC)" instead of "F_1 measure" in the following quantitative evaluation task.

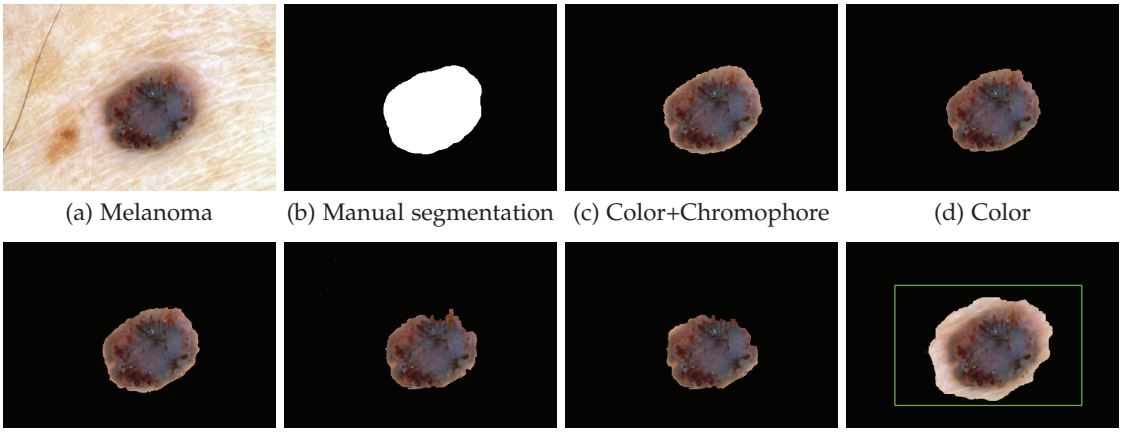
$$F_1(SG, GT) = \frac{2 \cdot Precison \cdot Recall}{Precision + Recall} = \frac{2 \cdot \#(SG \cap GT)}{\#(SG) + \#(GT)} = DSC(SG, GT). \tag{4.46}$$

  Note that the definitions of SG, GT as well as TP, FP and FN in context of PSLs delineation can be found explicitly in Section 4.3.5 and Section 4.5.2.2.

For the purpose of qualitative evaluation, 9 examples of challenging dermoscopy PSLs images are chosen for intuitive demonstration of segmentation results (cf. Figure 4.27 (a) to Figure 4.34 (a)). Besides ground truth (cf. Figure 4.27 (b) to Figure 4.34 (b)), three state-of-the-art graph-cut based segmentation methods and the proposed APS method are performed on each example. As can seen clearly, conventional color-feature-based segmentation approaches, like Boykov-Jolly's method and Lazy Sapping, trend to

over-segment the PSLs areas while GrabCut trends to under-segment the PSLs regions. None of these methods succeeded in segmenting adequate PSLs areas, only inner regions of PSLs or PSLs with part of skin tissues are extracted via an ambiguous border. For example, smooth transition between skin and lesion in Figure 4.27 (a) and Figure 4.29 (a) results gaps in the original PSLs edges and thus allows the extracted contours to leak through them, as shown in Figure 4.27 (f-h) and Figure 4.29. Also, artifacts like hairs in Figure 4.34 (a) and Figure 4.35 (a), specular reflection in Figure 4.28 (a) and irregular textures in Figure 4.31 (a) cause spurious PSLs "edges" that stopped the extracted contours from converging to the real PSLs boundaries, as shown in Figure 4.28 (f,g), Figure 4.31 (f-h), Figure 4.34 (f-h) and Figure 4.35 (f-h), correspondingly. Since border properties are important for the diagnosis of PSLs, an adequate PSLs delineation method should perform segmentation without losing much information on the peripheral parts of PSLs, for which the conventional color-feature-based segmentation approaches are inadequate. Nevertheless, our proposed APS technique in corporation with multiple features consistently outperforms the conventional color-feature-based segmentation approaches in delineating more accurate boundary close to the peripheral parts of "real" skin lesions. Significant improvement in PSLs delineation can be observed when the proposed APS technique is involved, especially on these challenging dermoscopy PSLs images mentioned above. Moreover, under the proposed APS framework, different combination of feature results in different accuracy level of delineation. In corporation with texture feature allows enhancement in discrimination between the lesion region and the skin region where they are similar in color but distinct in texture. For example, the proposed APS technique based on the combination of color and texture features (Color+LBP) yields better delineation results than that of color feature solely (color), as shown in Figure 4.30 (d,e) and Figure 4.31 (d,e). Otherwise, in cases of non-textured or evenly textured PSLs images, color feature based APS technique has comparable performance to that of combination of color and texture features, for instance, in Figure 4.33 (d,e), Figure 4.34 (d,e) and Figure 4.35 (d,e). In general, by combining color feature with chromophore feature quantified by the proposed model-fitting method (Color+Chromophore), our APS technique performs best in delineating out the entire lesion regions which is very close to those delineated manually by the expert dermatologist, except for the case shown in Figure 4.28 (c), where part of the actual lesion is segmented as healthy skin.

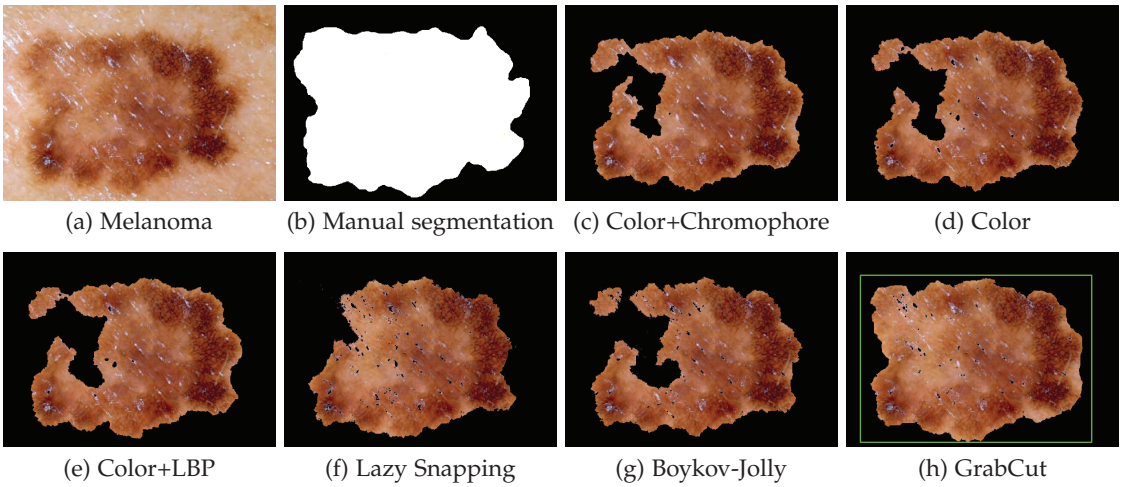
Qualitative evaluation addressed above can be verified objectively via quantitative evaluation concerning the segmentation performance on these 9 challenging examples. Table 4.5 shows the scores of four metrics, the Dice similarity coefficient (DSC), the segmentation error rate ϵ, precision and recall, which are calculated for each candidate



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

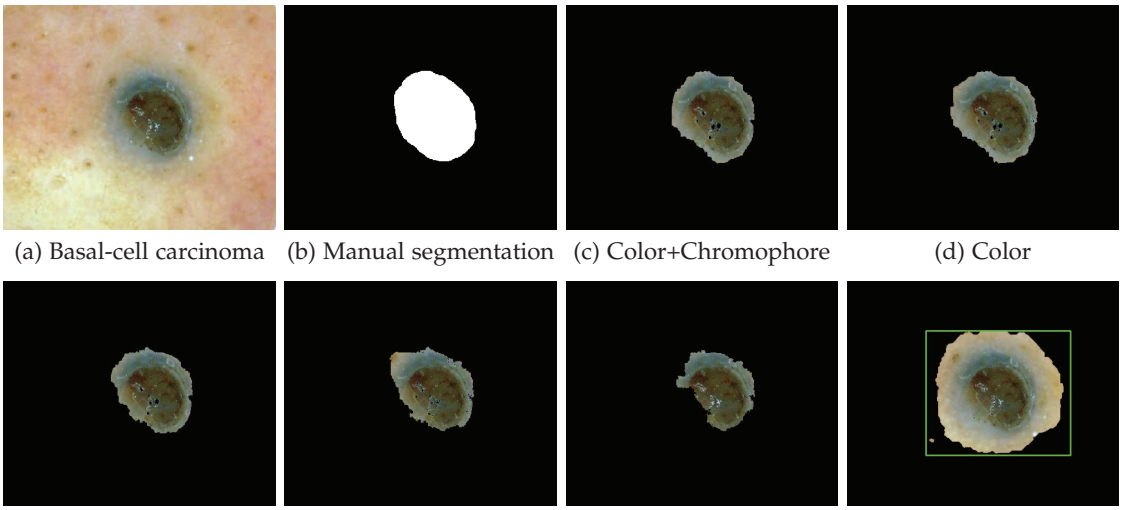
Figure 4.27: Segmentations on melanoma, Aal002 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

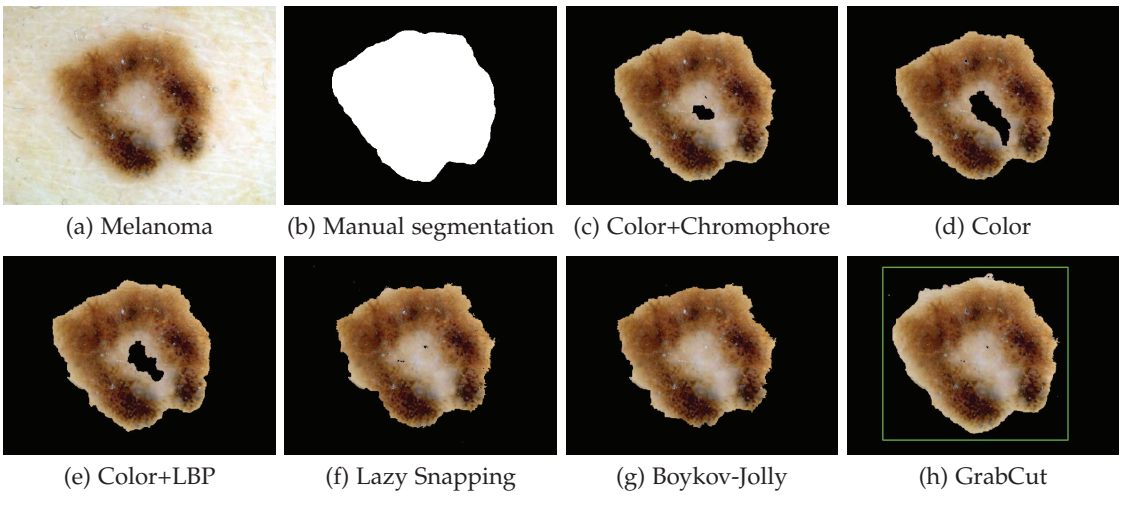
Figure 4.28: Segmentations on melanoma, Fbl028 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Basal-cell carcinoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

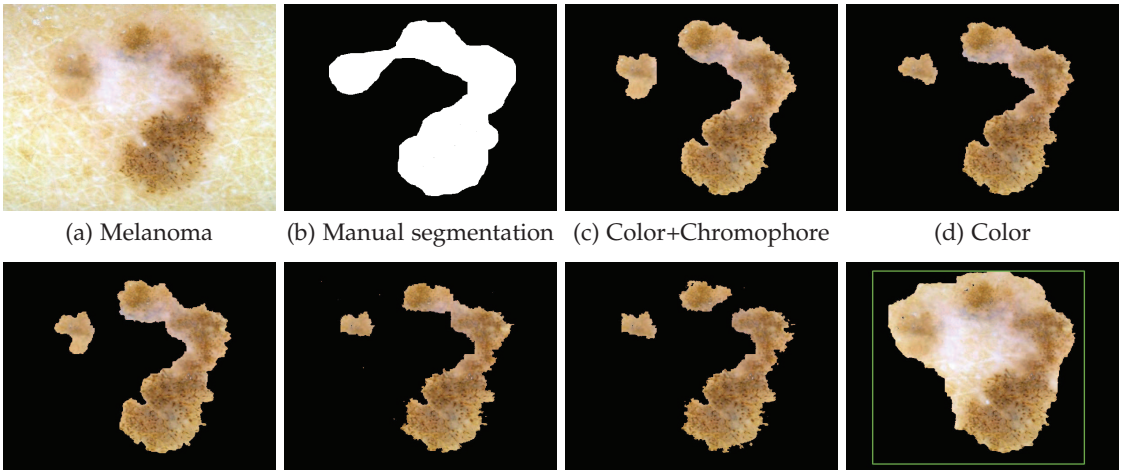
Figure 4.29: Segmentations on basal-cell carcinoma, Nelo39 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

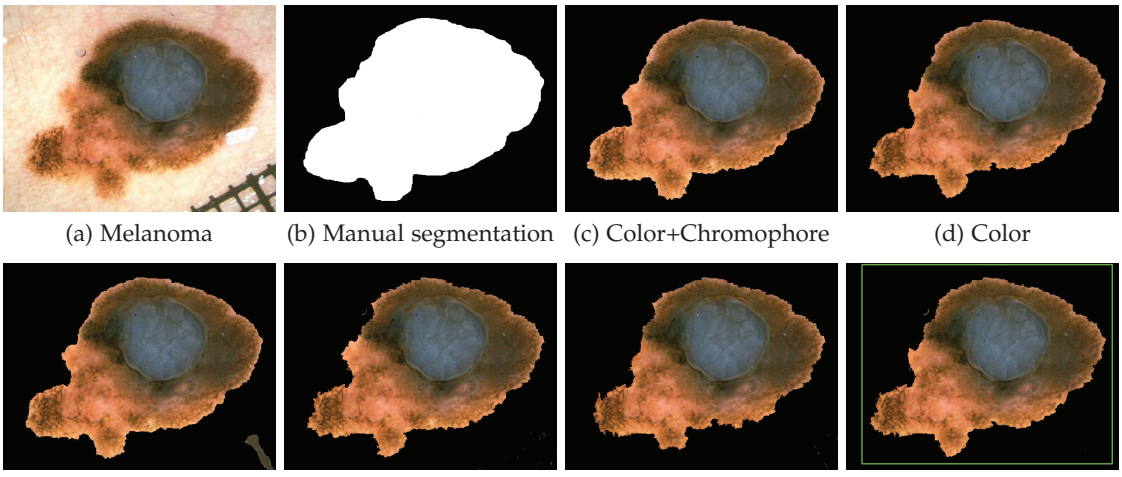
Figure 4.30: Segmentations on melanoma, Fclo40 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

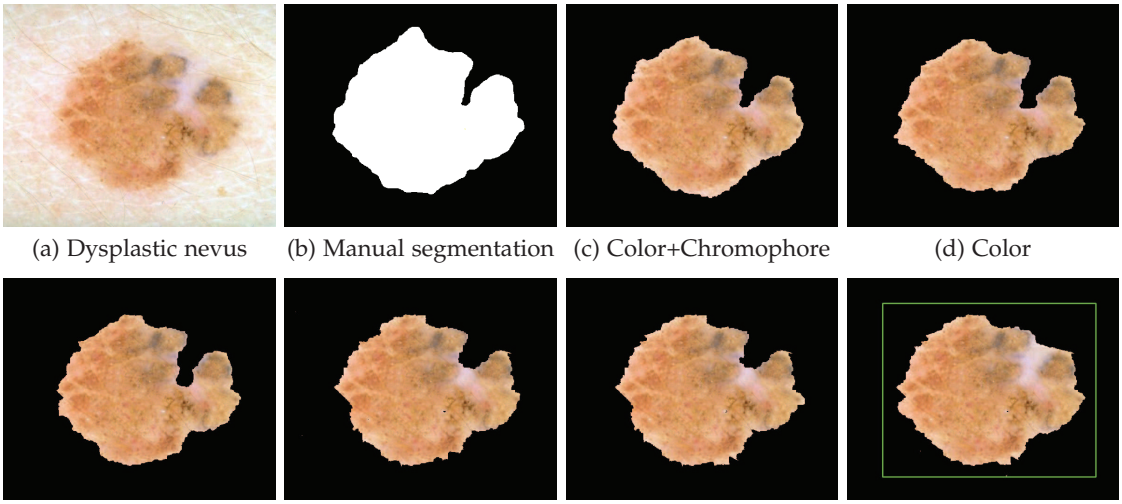
Figure 4.31: Segmentations on melanoma, Nalo20 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

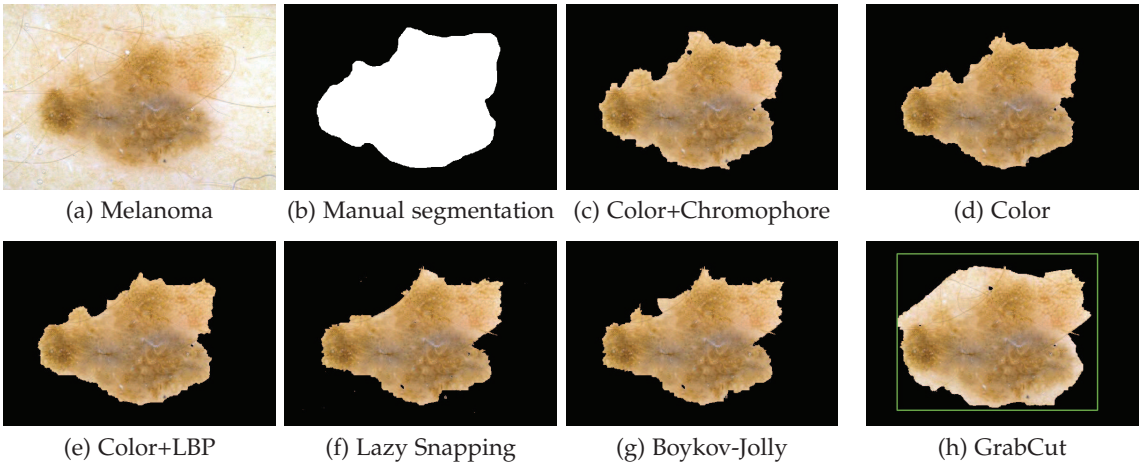
Figure 4.32: Segmentations on melanoma, Nglo15 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Dysplastic nevus (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

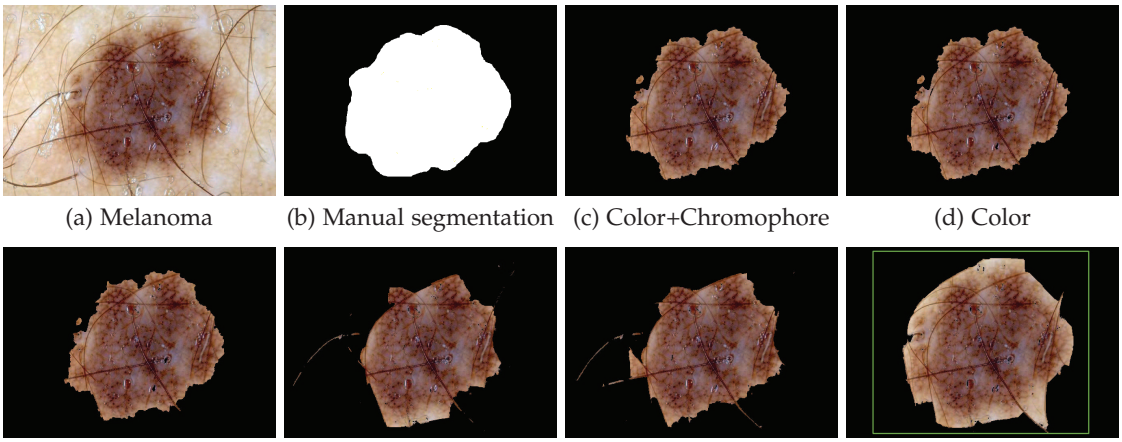
Figure 4.33: Segmentations on dysplastic nevus, Nml030 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

Figure 4.34: Segmentations on melanoma, Nml077 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).



(a) Melanoma (b) Manual segmentation (c) Color+Chromophore (d) Color

(e) Color+LBP (f) Lazy Snapping (g) Boykov-Jolly (h) GrabCut

Figure 4.35: Segmentations on dysplastic nevus, Fclo32 (a) obtained by our proposed APS technique with different feature configurations (c-e) and other color-feature-based graph-cut segmentation approaches (f-h).

approach on a particular example. As can be seen in Table 4.5, the proposed APS technique with different combinations of features achieves higher DSC score and lower segmentation error rate on 8 examples (except for the melanoma image 'Fblo28', on which GrabCut achieves highest DSC score and lowest error rate), compared to the three color-feature-based segmentation approaches. In particular, the proposed method in corporation with skin chromophore feature has the highest DSC score and lowest error rate (highlighted in pink color), which indicates that fundamental properties of PSLs like chromophore concentrations can be a more reliable and discriminative feature for improvement of the delineation performance on challenging PSLs images under the APS framework. Note that the lesion image (cf. Figure 4.28 (a)) to fail using our method is associated with strong specular artifacts dominant on the left part and only GrabCut can obtain comparative result to the manual segmentation due to its iterative energy minimization scheme which is converged to a good optimum. In terms of precision and recall, our proposed method achieves better with both high value of precision and recall, especially when chromophore feature is combine to the feature configuration (highlighted in blue color). This indicates that the proposed APS technique obtains results close to those of dermatologist. On the other hand, Boykov-Jolly's approach and Lazy Snapping show high precision but low recall, because these methods tend to extract insufficient inner regions of the PSLs. GrabCut, on the contrary, shows low precision but high recall, which indicates that it tends to overestimate the lesion area and misclassifies the surrounding healthy skin part as lesion. In both cases, these conventional graph-cut based approaches are not ideal as a preprocessing step in an automatic melanoma

diagnosis system, since their inadequate segmentations may result in loss of information on the peripheral parts of PSLs, which is very important for the diagnosis of melanoma.

Table 4.5: Quantitative evaluation on 9 dermoscopic PSLs images (presented from Figure 4.27 to Figure 4.34), measured by four metrics, the Dice similarity coefficient (DSC), the segmentation error rate (ϵ), Precision and Recall. Value in bold corresponds to the best performance according to a certain criterion. Cell in color indicates the best performance subject to a certain pair of criteria. 'Chromo': hemoglobin/melanin indices for chromophore information, 'LS': Lazy Snapping, 'BKJ': Boykov-Jolly proposed graph-cut algorithm.

Input	Criterion \ Approach	Color	Color + LBP	Color + Chromo	LS	BKJ	GrabCut
Aal002	DSC (%)	93.84	95.62	**96.94**	89.44	88.14	80.84
	ϵ (%)	11.62	8.46	**6.07**	19.11	21.26	47.41
	Precision (%)	**99.97**	99.17	97.73	99.92	99.70	67.84
	Recall (%)	88.41	92.31	96.16	80.95	78.97	**100**
Fbl028	DSC (%)	84.40	85.23	86.63	85.17	82.00	**91.97**
	ϵ (%)	27.05	25.77	23.67	25.89	30.55	**15.06**
	Precision (%)	99.67	99.80	99.59	99.71	**99.82**	99.51
	Recall (%)	73.19	74.38	76.63	74.32	69.58	**86.24**
Nel039	DSC (%)	93.79	92.59	**94.60**	89.78	82.30	61.95
	ϵ (%)	12.57	14.00	**10.73**	19.27	30.20	122.73
	Precision (%)	92.69	92.59	95.27	95.56	**99.43**	44.89
	Recall (%)	94.91	86.20	93.94	84.66	70.21	**99.90**
Fcl040	DSC (%)	93.82	94.67	**96.40**	94.56	94.17	95.84
	ϵ (%)	11.75	10.26	**7.13**	10.41	11.17	8.48
	Precision (%)	98.96	98.54	97.55	**99.01**	98.52	93.99
	Recall (%)	89.19	91.09	95.27	90.50	90.18	**97.77**
Nal020	DSC (%)	86.81	88.05	**90.28**	80.15	79.43	74.37
	ϵ (%)	23.54	21.71	**18.47**	33.14	34.16	65.62
	Precision (%)	98.71	97.95	95.26	**99.95**	99.83	61.02
	Recall (%)	77.47	79.96	85.80	66.90	65.96	**95.18**
Ngl015	DSC (%)	95.22	96.40	**97.30**	95.59	93.67	94.98
	ϵ (%)	9.13	7.05	**5.28**	8.45	11.92	9.57
	Precision (%)	**99.94**	98.51	99.69	99.83	99.88	**99.94**
	Recall (%)	90.93	94.38	**95.11**	91.70	88.19	90.48
Nml030	DSC (%)	95.57	94.01	**96.20**	93.91	94.42	95.10
	ϵ (%)	8.52	11.31	**7.51**	11.56	10.69	9.61
	Precision (%)	99.52	**99.99**	97.35	99.22	98.87	97.09
	Recall (%)	91.92	88.70	**95.09**	89.14	90.35	93.19
Nml077	DSC (%)	91.67	91.89	**94.40**	90.23	91.40	90.52
	ϵ (%)	15.40	15.04	**10.57**	17.95	15.97	20.58
	Precision (%)	**99.89**	99.73	98.73	98.92	99.05	83.91
	Recall (%)	84.69	85.19	90.44	82.95	84.85	**98.27**
Fcl032	DSC (%)	94.77	93.53	**94.95**	85.50	88.72	91.00
	ϵ (%)	9.99	12.18	**9.66**	26.22	20.53	19.63
	Precision (%)	99.49	**99.78**	99.53	95.62	98.48	84.03
	Recall (%)	90.47	88.02	90.77	77.32	80.72	**99.23**

Table 4.6 summarizes the overall performance for each method by calculating the mean value and standard deviation of each metric on 100 dermoscopy PSLs images. As can be seen in Table 4.6, GrabCut consistently achieves worse precision of 84.37% but excellent recall of 95.53% while Boykov-Jolly's approach consistently achieves excellent

precision of 99.07% but worse recall of 80.66%. In fact, a precision of 100% can be easily achieved as long as the delineated region is completely included in the actual lesion area whereas a recall of 100% can be achieved if the delineated region totally includes the actual lesion area. However, high scores of both precision and recall require more accurate segmentation, for example, the proposed method scores high precision above 98.36% along with high recall above 84.91%. For Lazy Snapping, it achieves both intermediate values of precision and recall (98.76% and 80.89%, repectively). When examining the more comprehensive metrics such as the Dice similarity coefficient (DSC) and segmentation error rate ϵ, it is apparent that our propose APS technique outperforms the alternative graph-cut based approaches. Our proposed method achieves considerably higher score of DSC above 91.19% and lower scores of ϵ under 16.05% while the other conventional segmentation approaches show poorer performance with DSC score lower than 88.73% and ϵ higher than 20.13%.

Table 4.6: Quantitative evaluation of segmentation methods on 100 dermoscopic PSLs images compared with manual segmentation by a expert dermatologist, measured by four metrics, the Dice similarity coefficient (DSC), the segmentation error rate (ϵ), Precision and Recall. For each metric, the corresponding average (Mean) value and standard deviation (SD) are specifically presented. Value in bold corresponds to the best performance according to a certain criterion. 'Chromo': hemoglobin/melanin indices for chromophore information, 'LS': Lazy Snapping, 'BKJ': Boykov-Jolly proposed graph-cut algorithm.

Criterion / Approach	DSC		Error ϵ		Precision		Recall	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Color	91.19	4.24	16.05	6.59	98.98	1.99	84.91	7.59
Color+LBP	91.41	3.89	15.70	6.47	98.74	2.02	85.69	7.68
Color+Chromo	**93.85**	**3.08**	**12.26**	**5.95**	98.36	1.68	88.27	7.56
LS	88.73	4.68	20.15	7.61	98.76	1.53	80.89	7.67
BKJ	88.70	5.00	20.13	7.90	**99.07**	**0.81**	80.66	8.14
GrabCut	87.90	9.90	29.97	32.39	84.37	16.45	**95.53**	**4.08**

The above experimental results based on average scores of metrics are effective for the overall accuracy assessment, however, they do not contain information on the dispersion of the segmentation scores. Therefore, the standard deviation (SD) of each metric is adopted as a criterion for the robustness assessment. Not surprisingly, the conventional methods show higher variability compared to the proposed method. Overall, the proposed APS technique based on the combination of color and chromophore features is proven to be the most reliable and stable PSLs delineation approach since it achieves best scores of both DSC and ϵ while maintains lowest degree of dispersion of the two metrics.

4.5.3.3 Conclusion

In this section, we have presented a fully automatic segmentation approach especially suitable for delineation of dermoscopy PSLs images. It is well developed under the Graph-Cut/MAP-MRF binary segmentation framework consisting of two innovative processing stages, automatic seeding and color-texture-chromophore feature configuration. Automatic seeding stage guarantees the precision and diversity of observed sample selection to build a reliable training set, while adding chromophore features of PSLs to the multi-feature configuration stage enhances the discriminability of SVM classifier to predict objective soft classification. Finally, graph-cut based segmentation promises a globally optimal skin lesion delineation with t-link edge weights appropriately defined by posterior classification probabilities.

Qualitative and quantitative evaluations were performed on 100 varied types of dermoscopic PSLs images of different difficulty levels, in proportion to the similarity between foreground lesion and background skin, as well as the degree of artifacts like hairs, reflections or bubbles. For each dermoscopic image, a corresponding ground-truth binary mask is provided via manual delineation. Scores of four statistical metrics calculated based on this reference dataset indicate promising segmentation results obtained by our proposed APS technique, compared to the alternative graph-cut based segmentation methods. Improvements on lesion delineation accuracy are particularly remarkable for challenging PSLs images. In addition, the proposed APS framework is essentially classification based thus additional features other than color information can be naturally fed into the algorithm to further enhance the performance. Both mean values and standard deviations of Dice similarity coefficient and segmentation error rate suggest that our proposed APS technique based on the combination of color and chromophore features is accurate and robust as expected and it can be effective and useful for pigmented skin lesion delineation in scope of computer-aided skin cancer diagnosis system.

Future work will focus on the simplification of automatic seeding procedure while maintain its precision. Also, multi-spectral PSLs images which are able to reveal in-depth chromophore features will be involved in our proposed APS framework for more precise classification of foreground lesion and background healthy skin.

Chapter 5

Conclusion

The main motivation of this thesis is to propose a novel fully automatic technique for pigmented skin lesion delineation in computer-aided skin cancer diagnosis system. In order to attain our goal, we have developed a tool based on graph cuts segmentation and binary clustering/classification, which have been successfully applied to segmentation of challenging synthetic textured image, natural color images and dermoscopic PSLs images. Furthermore, our work on automatic lesion localization and feature selection on multi-spectral PSLs images opens new perspectives which could lead to valuable improvements of existing image segmentation techniques. In the following, contributions made in this thesis are reviewed by chapter.

5.1 Contributions

Interactive Graph Cuts Based Image Segmentation

We have compared objectively state-of-the-art interactive graph-cut based image segmentations, Boykov-Jolly's approach, *Lazy Snapping* and *GrabCut* via qualitative segmentation accuracy assessment on 30 natural color images and highlighted the strengths and the weakness of each method. We have also addressed the optimization of two fundamental parameters λ and σ in the framework of graph cut segmentation in order to yield most preferable and robust segmentation results for later comparison of some state-of-the-art techniques as well as the proposed approaches in this work.

Combining Classification Techniques and Graph Cut Segmentation Framework

We have proposed to define the likelihood energy term of the binary segmentation energy function using the posterior classification of a classifier, Support Vector Machine or Random Forest. We have shown that the combination of shape feature, LBP texture

feature, RGB color feature and other potential pixel-wise features can be to used to build a powerful multi-feature vector. Finally, we have shown by qualitative and quantitative evaluations on 100 natural color images that the proposed color-texture-shape feature based graph cuts segmentation approach outperforms the conventional color-feature based segmentation methods.

Application to Skin Chromophore Extraction

We have developed a novel skin chromophore extraction method named Surface Fitting and Flattening (SF^2) approach which fits a quadratic surface in logarithmic normalized RGB space and flatten it onto a 2-dimensional color plane. As an extension of Tsumura's approach, we have introduced the Pearson correlation coefficient (PCC) to distinguish the estimated chromophore content distributions. For skin chromophore quantification on RGB color and mutispectral pigmented lesion images, we have proposed to apply Non-negative Matrix Factorization (NMF) to obtain concentration distributions of skin chromophores. In addition, we have developed a Model-Fitting approach using tabulated extinction coefficients of skin chromophores to objectively and precisely quantify skin chromophores.

Application to Melanoma Detection

Our main interest was to develop a robust and accurate segmentation tool for automatic dermoscopic PSL image delineation. First, we have developed an automatic approach for selection of seed regions (Auto-Seeding) based on binary clustering and mathematical morphological processing. Then, we have proposed to add chromophore features of pigmented lesions to the feature configuration to improve the algorithm's segmentation accuracy. Distributions of skin chromophore concentrations quantified by the proposed Model-Fitting approach are selected as chromophore feature. Finally, qualitative and quantitative evaluations on 100 dermoscopic PSLs images indicate that the developed Automatic PSLs Segmentation (APS) method incorporation with skin chromophore features outperforms state-of-the-art interactive graph-cut based methods, even they are equivalently initialized manually or by the proposed Auto-Seeding approach.

5.2 Perspectives

This chapter presents several perspectives concerning the improvement of pigmented skin lesion delineation under our proposed APS framework. At the automatic seed-region selection stage, background seeds can be generated using the same scheme of the lesion seeding (except for the skeletonization part, Parker's method will be preferred),

instead of simple border-adding. Also, inspired by the initialization of GrabCut, an automatic approximated lesion localization in terms of bounding rectangle is possible to simplify the auto-seeding approach. At the feature configuration and classification stage, since multi-spectral imaging allows more accurate acquisition of chromophore properties of deeper skin layer which is unseen by dermoscopy, we could select multi-spectral PSLs images at certain wave bands (ranges) using existing feature selection algorithms to directly complement the existing feature configuration. Likewise, the resulting chromophore distribution maps can be combined with other low level visual features to build a new multi-feature vector.

Automatic Lesion Localization

According to the approximate lesion localization method proposed by Celebi et al. [30], we could determine the lesion region of a dermoscopic PSLs image approximately via an accurate bounding rectangle which encloses the probable lesion area. This kind of initialization might allow: (a) improvement of lesion delineation accuracy since the subsequent segmentation procedure is focused on the region guaranteed in containing the lesion. In addition, from the classification perspective, it increases the foreground/background seeds number to provide better training set compared to the stroke-like small region of seeds. (b) simplification and speed-up of lesion delineation since skeletonization process in order to explicitly extract the stroke-like seed regions is omitted and the segmentation algorithm could simply focus on a smaller region rather than two whole dermoscopic image.

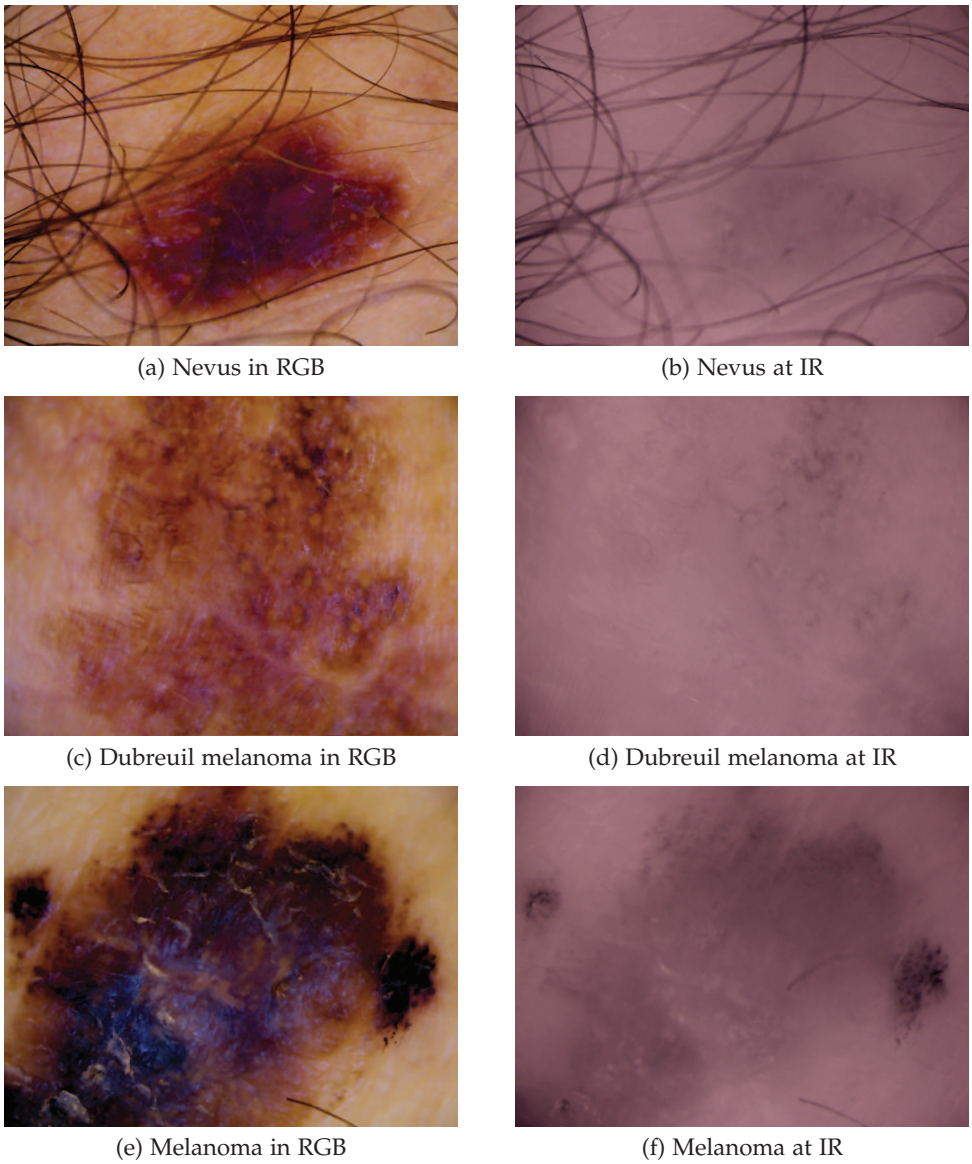
With this kind of initialization approach, an iterative optimization scheme of segmentation should be adopted and after each iteration, new updated training samples will be fed into a classifier to better estimate segmentations until converged (i.e. optimal segmentation is found).

Feature Selection

Multi-spectral imaging relies on the principle that light of different wavelengths, for example, of the visible and infrared spectrum, penetrates the skin in different depths. When coupled with computer-based analysis, some features of deeper skin layer which are not visible in macroscopic and dermoscopic analysis can be visualized.

Infrared light image, for instance, is specifically complemental to the computer-aided melanoma diagnosis. Indeed, the IR light at 850 nm reveals the melanin present in the deeper layers of the skin. We see in particular in Figure 5.1 (b,d) that for nevus or superficial spreading melanoma (e.g. Dubreuil melanoma), distribution of melanin concentration appears uniform and slightly visible in infrared light, whereas for deep

extensive melanoma in Figure 5.1 (f), the distribution of pigmentation appears more clear and especially non-uniform, which is a possible sign of malignancy.



(a) Nevus in RGB (b) Nevus at IR

(c) Dubreuil melanoma in RGB (d) Dubreuil melanoma at IR

(e) Melanoma in RGB (f) Melanoma at IR

Figure 5.1: RGB and spectral images of PSLs at wavelength of 850 nm.

The involvement of multi-spectral images increases the dimensionality of feature space. Therefore, feature selection which aims to reduce the dimensionality of the feature space by eliminating redundant, irrelevant or noisy features could be a preferred pre-processing step before training a classifier. From the classification perspective, five main benefits associated with feature selection can be concluded: (a) reduced feature extraction time and storage requirements, (b) reduced classifier complexity, (c) increased prediction accuracy, (d) reduced training and testing times, (e) enhanced data interpretability and (f) enhanced generalization by reducing overfitting.

Appendix A

Résumé en Français

Sommaire

Cette annexe est un résumé long en français de ce manuscrit de thèse. Après une introduction du contexte et des objectifs de ce travail qui concerne la détection automatique du contour des lésions pigmentaires sur des images couleurs ou multispectrales de la peau, nous présentons un rapide état de l'art des algorithmes de segmentation interactive d'images. Ils sont formulées comme la minimisation d'énergie par coupes de graphes en terme de Maxima A-Posteriori d'un champ de Markov (cadre MAP-MRF). Dans ce cadre, nous proposons une fonction d'énergie basée sur la classification et la combinaison de caractéristiques multiples. Nous présentons ensuite l'extraction des chromophores de la peau et son application à la détection du mélanome via un schéma global de segmentation automatique des lésions cutanées pigmentées. Enfin, les contributions des cette thèse sont résumées.

A.1 Introduction

A.1.1 Motivations

Le mélanome est une tumeur maligne des mélanocytes. C'est le cancer de la peau le plus mortel qui entraîne un nombre considérable de décès en particulier de la population mondiale à peau claire [52]. Puisque le traitement du mélanome métastatique est difficile [64], le diagnostic précoce et l'excision rapide sont essentiels pour améliorer le taux de survie des patients. Il existe plusieurs systèmes de diagnostic de mélanome, par exemple, la règle «ABCDE» [77] et la liste de «7-points» [10]. L'interprétation d'images dermoscopiques est couteuse en temps, subjective, et manque de précision et la reproductibilité. Elle s'appuie enfin sur l'expérience clinique du dermatologue. Par conséquent, il est utile de développer des systèmes de diagnostic assisté par ordinateur.

Une approche standard de l'analyse d'image dermoscopique automatique comprend généralement quatre étapes : (i) l'acquisition d'images ; (ii) la segmentation d'images ; (iii) l'extraction et la sélection de caractéristiques et (iv) la classification de la lésion. L'étape de segmentation qui délimite les contours des lésions cutanées pigmentés est très importante puisque les contours fournissent des informations importantes pour un diagnostic [32]. Par exemple, la dimension des lésions, la forme, l'irrégularité du contour sont des caractéristiques importantes calculées sur le contour de la lésion délimitée. Cependant, la mise en œuvre de la délimitation précise reste une tâche difficile pour plusieurs raisons : (i) un faible contraste entre la lésion et la peau environnante ; (ii) pigmentations variées à l'intérieur de la lésion ; (iii) le contour de la lésion flou et irrégulièr et (iv) la présence d'objets comme des cheveux, des bulles et des réflexion spéculaires. Cette thèse propose une nouvelle approche de la segmentation automatique des lésions cutanées pigmentées, basées sur des images dermoscopique et spectroscopiques dans le cadre MAP-MRF. Le logiciel développé a été testé sur différentes images de lésions dermoscopique et des évaluations qualitatives et quantitatives indiquent l'efficacité et la fiabilité de la méthode proposée.

A.1.2 Le cadre général : coupes de graphes

Nos outils de segmentation d'images sont basées sur la coupe de graphe. Dans la première partie, nous décrivons les problèmes d'optimisation dans le contexte de la vision par ordinateur et clarifions la relation entre coupe de graphe minimum et estimation du maximum *a posteriori* d'un champ de Markov en segmentation d'images. Du point de vue pratique, nous comparons objectivement des méthodes plus récentes de segmen-

tation interactive d'images par coupe de graphe via l'évaluation de la précision de la segmentation qualitative sur les images couleurs naturelles et mettre en évidence les forces et les faiblesses de chaque méthode. En outre, nous nous adressons à l'optimisation de deux paramètres fondamentaux dans le cadre de la segmentation par coupe de graphe. Les points sur lesquels le mise a focalise sont les suivants :

- la gestion de contraintes faibles
- l'amélioration de terme d'attache aux données
- la détermination automatique de graines

A.1.3 La combinaison des techniques de classification et segmentation par coupe de graphe

La deuxième partie de cette thèse est consacrée à des méthodes innovantes en combinant segmentation par coupe de graphe avec des techniques de classification. Nous proposons dans cette section de la formulation du terme d'attache aux données de la fonction d'énergie de segmentation binaire défini par la probabilité à postérieur d'un classificateur (*Machines à support de vecteurs* ou *Forêts aléatoires*). Nous montrons comment combiner les caractéristiques de couleur, de texture (LBP) dans un voisinage local (caractéristique de forme) pour améliorer la discrimination entre l'objet et le fond.

A.1.4 Application à la détection de mélanome

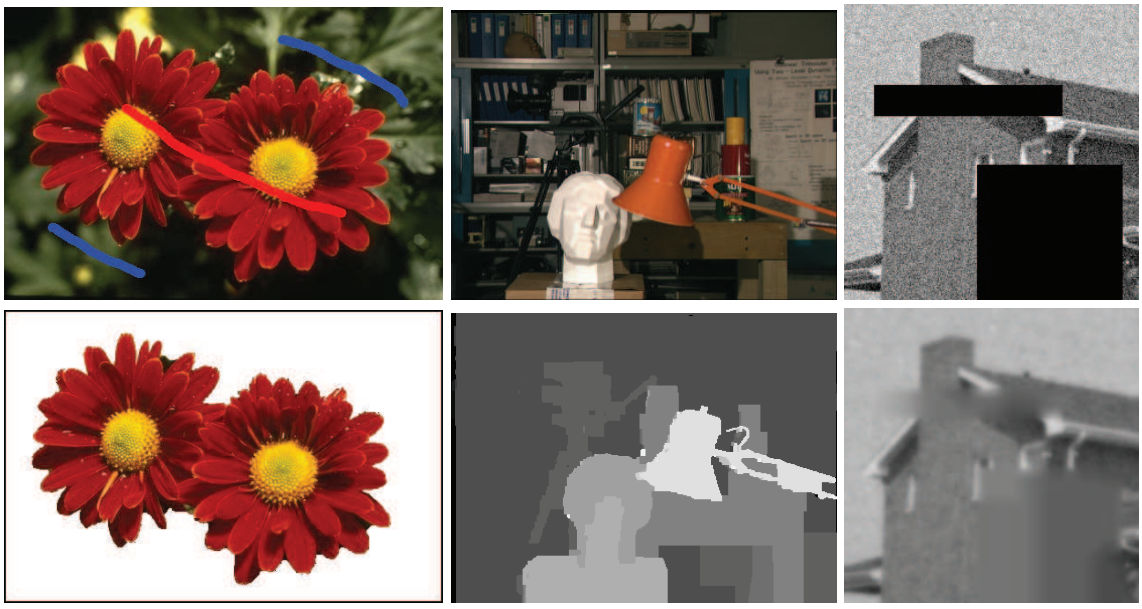
Nous détaillons dans cela section de l'extraction des chromophores de la peau et la détection du mélanome. La premiére partie consiste à estimer les distributions des chromophores de la peau. L'application vise à segmenter précisément les lésions cutanées pigmentées sur des images dermoscopiques. Nous développons une approche determinant automatiquement les graines donnant un ensemble d'apprentissage suffisant pour le classificateur et proposons d'ajouter les chromophores à la configuration des caractéristiques du classificateur. Dans les études qualitatives et quantitatives, les résultats de segmentation sont comparés aux délimitations manuelles et l'approche proposée surpasse les méthodes classiques.

A.2 Le cadre général : coupes de graphes

A.2.1 Problèmes d'étiquetage que la minimisation de l'énergie

Les problèmes existantes de vision par ordinateur de bas niveau, telles que la segmentation d'image, l'appariement stéréo, la restauration de l'image (cf. Figure A.1), peuvent être formulés comme des problèmes d'étiquetage d'images. Ces problèmes visent à affecter une étiquette de l_p d'un ensemble d'étiquettes $\mathcal{L}$ à chaque pixel p de l'image observée $\mathcal{P}$. Un exemple d'étiquetage d'image binaire est illustré dans la figure A.2. Un tel problème d'étiquetage est naturellement représenté en termes de minimisation de l'énergie, où la fonction d'énergie E a deux termes : un terme appelé «l'énergie d'attache aux données», qui pénalise les solutions qui sont incompatibles avec les données observées, tandis que l'autre terme appelé «l'énergie de régularisation» assure la cohérence spatiale :

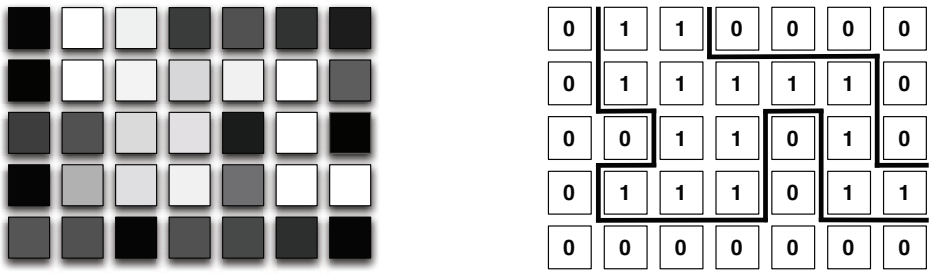
$$E = E_{data} + \lambda \cdot E_{smooth}. \tag{A.1}$$



(a) Segmentation d'images binaires (b) Appariement stéréo (c) Restauration de l'image

FIGURE A.1 – Application des problèmes d'étiquetage d'images binaires. (a) Segmentation d'images binaires sur «Flower» (Ref. 124084) par coupe de graphe [22] ; (b) Appariement stéréo sur «Tsukuba» par l'algorithme *α-expansion* [59] ; (c) Restauration de l'image «House» par l'algorithme *α-β-swap* [26].

où la constante λ contrôle l'importance relative des données et de l'énergie de régulari-



(a) Une image en niveaux de gris (b) Un étiquetage

FIGURE A.2 – Un exemple d'étiquetage d'image binaire.

sation. En particulier, l'énergie d'attache aux données est de la forme :

$$E_{data} = \sum_{p \in \mathcal{P}} D_p(l_p). \tag{A.2}$$

Compte tenu du système de voisinage $\mathcal{N}$, l'énergie de régularisation a la forme suivante :

$$E_{smooth} = \sum_{\{p,q\} \in \mathcal{N}} V_{\{p,q\}}(l_p, l_q), \tag{A.3}$$

En combinant l'équation (A.1), l'équation (A.2) et l'équation (A.3) nous obtenons la forme générale de la fonction de l'énergie :

$$E(\mathcal{L}) = \sum_{p \in \mathcal{P}} D_p(l_p) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} V_{\{p,q\}}(l_p, l_q). \tag{A.4}$$

Une des raisons pour lesquelles cette forme générale est si populaire, c'est qu'il peut être justifié en termes d'estimation du maximum *a posteriori* (MAP) d'un champ de Markov (MRF) [16, 44].

A.2.2 Notions de base de coupes de graphes

Dans cette section, nous passons en revue brièvement quelques définitions et notations du graphe et les théorèmes fondamentaux ainsi que des algorithmes de base dans le contexte de l'optimisation combinatoire en vision par ordinateur.

Définition A.2.1. (Graphe). Un graphe orienté $\mathcal{G}$ est une paire ordonnée $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, où $\mathcal{V}$ est l'ensemble des nœuds d'un graphe (sommets) et $\mathcal{E} \subseteq \{(p,q) | p, q \in \mathcal{V}, p \neq q\}$ est l'ensemble des arêtes du graphe reliant ces nœuds. Une arête (p,q) de $\mathcal{E}$ est notée $e_{p,q}$. Pour chaque arête $e_{p,q}$ de $\mathcal{G}$, nous attribuons une valeur de $w_{p,q} \in \mathbb{R}^+$, qui est la capacité de l'arête $e_{p,q}$.

Définition A.2.2. (Coupe). Une coupe est un sous-ensemble des arêtes $C \subset \mathcal{E}$ telles que deux nœuds supplémentaires sont ajoutés au graphe induit $\mathcal{G}(C) = \langle \mathcal{V}, \mathcal{E} \backslash C \rangle$. Le coût (ou la capacité) de la coupe est définie comme la somme des capacités des arêtes qu'il sectionne :

$$|C| = \sum_{e_{p,q} \in C} w_{p,q} \tag{A.5}$$

Définition A.2.3. (*s*/*t* Coupe). Une *s*/*t* coupe est un cas particulier de coupe de graphe, où l'ensemble des nœuds $\mathcal{V} = \{s, t\} \cup \mathcal{P}$ contient deux nœuds spéciaux de terminaux, la source s et le puits t, et un ensemble de nœuds non terminaux $\mathcal{P}$. Par conséquent, une *s*/*t* coupe est définie comme une partition des nœuds dans le graphe en deux sous-ensembles disjoints $\mathcal{S}$ et $\mathcal{T}$, c.-à-d. $\mathcal{S} \cup \mathcal{T} = \mathcal{V}$ and $\mathcal{S} \cap \mathcal{T} = \varnothing$, où $s \in \mathcal{S}$ and $t \in \mathcal{T}$.

En optimisation, trouver la *s*/*t* coupe avec le plus petit coût indique donc l'endroit où le réseau est le plus limité, ce qui revient à établir le «flot» avec la valeur maximale qui peut passer de la source s au puits t, selon le **théorème flot-max/coupe-min** (cf. Théorème A.2.1) [2, 41]. En d'autres mots, le *problème de coupe minimale* et le *problème de flot maximum* sont équivalentes.

Définition A.2.4. (Flot). Etant donné un graphe orienté $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ avec deux nœuds terminaux $\{s, t\} \in \mathcal{V}$, un *flot* dans le graph $\mathcal{G}$ est définie comme une fonction $f : \mathcal{E} \to \mathbb{R}_0^+$ attribuant à chaque arête $(p, q) \in \mathcal{E}$ une valeur non négative $f_{p,q}$. Ce flot doit vérifier deux contraintes :

$$0 \leq f_{p,q} \leq w_{p,q} \quad \forall (p, q) \in \mathcal{E}, \quad (\textbf{La contrainte de capacité}) \tag{A.6}$$

$$\sum_{(p,q) \in \mathcal{E}} f_{p,q} = \sum_{(q,p) \in \mathcal{E}} f_{q,p} \quad \forall p \in \mathcal{V} \setminus \{s, t\}. \quad (\textbf{La conservation du flot}) \tag{A.7}$$

La valeur du flot représente la quantité de flot allant de la source s au puits t. Elle se définit par

$$|f| = \sum_{(s,p) \in \mathcal{E}} f_{s,p} - \sum_{(p,s) \in \mathcal{E}} f_{p,s} = \sum_{(p,t) \in \mathcal{E}} f_{p,t} - \sum_{(t,p) \in \mathcal{E}} f_{t,p}. \tag{A.8}$$

Théorème A.2.1. (Théorème Flot-Max/Coupe-Min). *Pour tout graphe orienté $\mathcal{G}$, tout couple $\{s, t\}$ de sommets, et pour tout vecteur de capacités positives, la valeur maximale du flot de s à t est égale à la capacité minimale d'une coupe séparant s de t.*

Selon l'inférence bayésienne, la fonction d'énergie associée au problème d'étiquetage binaire (cf. Equation (A.4)) est traitée comme une configuration optimale d'un champ de Markov (MRF) qui peut être résolu via estimation du maximum *a posteriori* (MAP).

En trouvant une coupe minimale d'un graphe, il peut être exactement minimisé par la suite (cf. Corollaire A.2.1).

Corollaire A.2.1. (Minimisation de l'Energie via Coupe Minimale de Graphe dans le Cadre MAP-MRF). *Prenons l'étiquetage binaire* $\mathcal{L} = \left\{ l_p | l_p \in \{0,1\} \right\}_{p=1}^{|\mathcal{V}|}$ *sur une observation donnée* $\mathcal{X} = \{x_p | p \in \mathcal{V}\}$*, l'énergie de ce problème d'étiquetage*

$$E(\mathcal{L}) = \sum_{p \in \mathcal{V}} -\ln(\Pr(x_p|l_p)) + \sum_{p \in \mathcal{V}} \sum_{q \in \mathcal{N}_p} U_{p,q} \cdot \delta_{l_p \neq l_q} \tag{A.9}$$

peut être minimisé exactement via le calcul de la coupe minimale d'un graphe avec ses capacités d'arêtes attribués de façon appropriée, si et seulement si la condition $U_{p,q} \in \mathbb{R}^+$ *est satisfaite.*

A.2.3 Coupes de graphes dans la segmentation d'images

Segmentation d'image par coupe de graphe a été introduit par Boykov et Jolly [23] en 2001. Contrairement aux autres méthodes de segmentation d'images, par exemple, le Modéle de contour actif [55, 28] et les approches fondées sur les surfaces de niveau, le coupe de graphe garantit un optimum global, pour la segmentation d'image binaire. La figure A.3 illustre le processus de segmentation par coupe de graphe proposé par Boykov et Jolly. L'utilisateur marque certains pixels comme «objet» (le trait rouge) ou «fond» (le trait bleu), «contraintes dures» pour la segmentation. Chaque nœud dans le graphe représente un pixel de l'image d'entrée, tandis que deux nœuds supplémentaires, la source et le puits, représentent l'objet et le fond, respectivement. La capacité de l'arête entre deux nœuds voisins est donnée par le «terme de régularisation» de l'énergie (cf. Equation (A.3)), tandis que la capacité de l'arête entre la source et le nœud ou nœud et le puits, est définie par le «terme d'attache aux données» de l'énergie (cf. Equation (A.2)). Pour le flot maximum, les arêtes sont saturées. Par conséquent, nous pouvons obtenir la segmentation globalement optimale.

La fonction de l'énergie adoptée par Boykov et Jolly est défini sur l'équation (A.4) comme suit :

$$E(\mathcal{L}) = -\sum_{p \in \mathcal{V}} \log h(x_p; l_p) + \lambda \cdot \sum_{\{p,q\} \in \mathcal{N}} \exp\left(-\frac{(x_p - x_q)^2}{2\sigma^2} \right) \cdot \frac{1}{dist(p,q)} \cdot \delta_{l_p \neq l_q} \tag{A.10}$$

L'énergie d'attache aux données est définie par l'estimation de la distribution d'intensité de pixel utilisant un histogramme, alors que l'énergie de régularisation par pénalise les discontinuités. Deux extensions de l'approche de segmentation par coupe de graphe de Boykov-Jolly, *Lazy Snapping* et *GrabCut*, ont été proposés par Li et al. [65] et Rother et

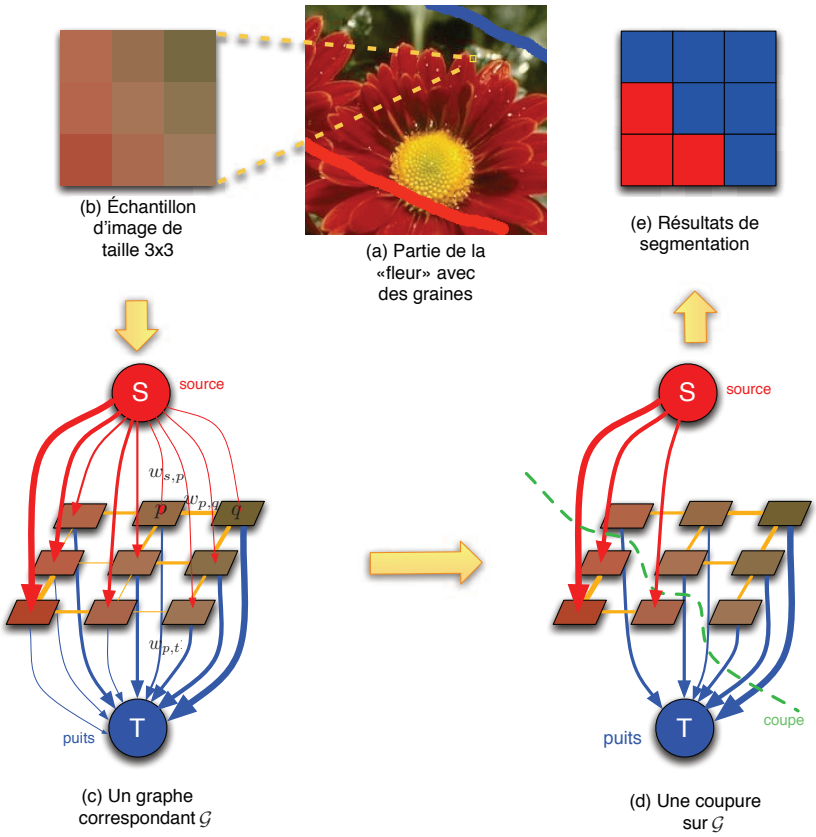


FIGURE A.3 – Un cadre de la segmentation basée sur le coupe de graphe proposé par Boykov et Jolly [22].

al. [78] en 2004. Contrairement à la méthode Boykov-Jolly, Lazy Snapping adopte un algorithme de partitionnement des données nommé «*k*-Means» pour définir l'énergie de données, tandis que GrabCut utilise un modèle de mélange gaussien pour estimer la distribution des couleurs pour la définition de l'énergie des données.

A.2.4 Détermination des paramètres de segmentation par coupe de graphe

Le mauvais choix des paramètres λ et σ peut entraîner segmentation insatisfaisanté. λ peut être estimée par apprentissage. σ est étroitement liée au niveau de variation d'intensité de pixels d'une image, il est raisonnable de donner à σ de la moyenne des différences d'intensité absolue entre deux pixels voisins :

$$\sigma = \sqrt{\left\langle (I_p - I_q)^2 \right\rangle} \tag{A.11}$$

A.3 La combinaison des techniques de classification et segmentation par coupe de graphe

Les classiques méthodes utilisent uniquement les caractéristique de couleur pour définir l'énergie de données. Par exemple, dans la figure A.4, l'objet de la «croix» et le fond de «l'église» sont très semblables en couleur. Ici, la caractéristique de la couleur ne suffit pas à les distinguer. La caractéristique de la texture peut être un meilleur choix dans ce cas.

Par conséquent, nous avons proposé dans cette thèse d'utiliser non seulement la couleur, mais aussi la texture et la forme pour construire un vecteur de caractéristique pour améliorer la perfomance de segmentation. Ce vecteur de caractéristiques, combinaison de couleur, de texture dans un voisinage local (caractéristique de forme) est utilisé pour classificateur (Machines à support de vecteurs ou Forêts aléatoires). La sortie probabiliste de ce classificateur est utilisée pour définir l'énergie des données.

A.3.1 Local Binary Pattern (LBP) pour reconnaître des textures

Pour la description de la texture, nous avons proposé d'utiliser **Local Binary Pattern (LBP)**, qui tout d'abord introduit par Ojala et al. [71] en 1996.

La figure A.5 montre le schéma de processus de codage LBP. En outre, le lissage gaussien peut être adoptée pour améliorer la capacité discriminative pour la texture à grande échelle (léopard) d'un opérateur LBP. Tout au long de cette thèse, la combinaison de filtrage gaussien et l'opérateur LBP est notée par GLBP.

(a) Initialisation par «graines» (b) Boykov-Jolly ; $\epsilon = 75.81\%$ (c) Lazy Snapping ; $\epsilon = 18.43\%$

(d) IInitialisation par rectangle (e) GrabCut ; $\epsilon = 57.82\%$ (f)

FIGURE A.4 – Un exemple négatif pour les techniques de segmentation par coupe de graphe à base de couleurs.

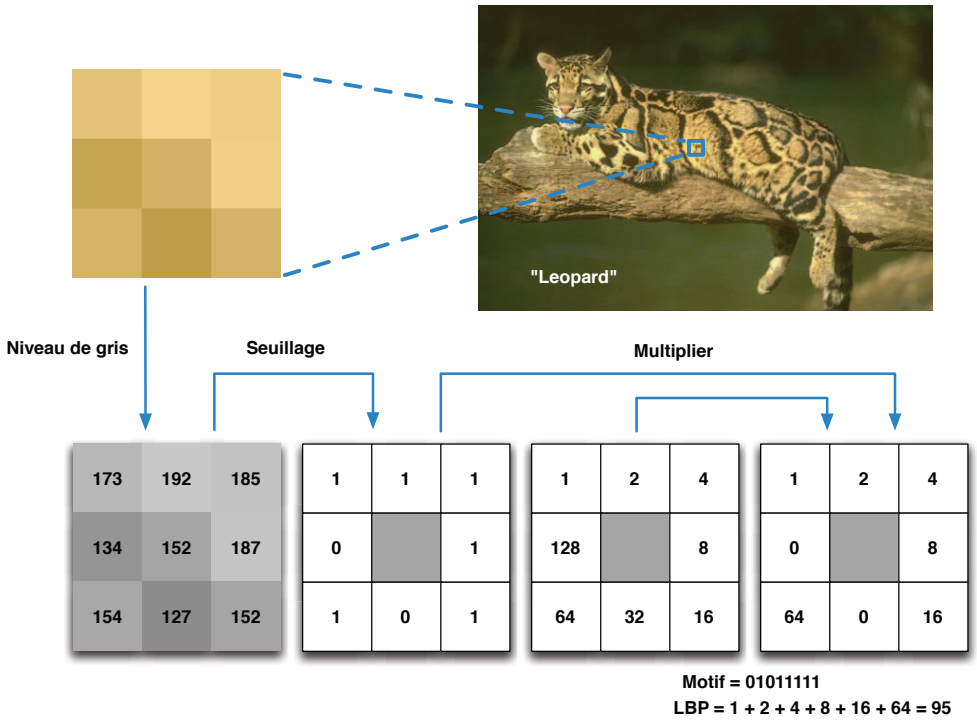


FIGURE A.5 – Calcul du code LBP original.

A.3.2 Un voisinage local pour reconnaître des formes

Comme la caractéristique de la texture est une propriété basée sur l'aspect local, nous avons introduit un voisinage local (une fenêtre glissante, cf. Figure A.6).

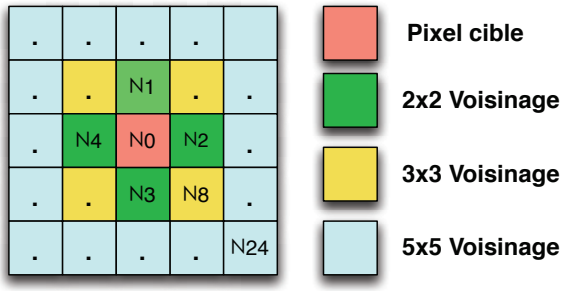


FIGURE A.6 – Un voisinage local.

Au cours du balayage de cette fenêtre, le code LBP de chaque pixel dans le voisinage, ainsi que le pixel central est utilisé pour construire un vecteur de caractéristique.

Le voisinage local montre aussi une excellente généralisation à d'autres propriétés locales des pixels, comme des informations de couleur. Le nombre de voisins (taille de la fenêtre) est fixée à 5×5 comme un compromis de faible complexité de calcul dans l'espace de caractéristique et bonne discrimination spatiale. Ainsi, la configuration de la caractéristique de texture en utilisant le voisinage local est un vecteur de dimensions

100 (cf. Equation (A.12)).

$$\mathbf{F} = [R_0, G_0, B_0, T_0 \cdots, R_n, G_n, B_n, T_n]^\top \tag{A.12}$$

où $T_i,\ i = 0, \cdots, n$ représente le code LBP.

A.4 Application à la détection de mélanome

Dans le contexte de la détection du mélanome, nous avons utilisé non seulement les caractéristiques de la couleur, la texture et de la forme, mais aussi des caractéristiques de chromophores. Afin d'extraire les chromophores, nous introduisons ici deux groupes de méthodes d'extraction : i) les approches fondées traitement de l'image et ii) les approches basées sur des modèles physiques.

A.4.1 Structure de la peau et la propriété optique

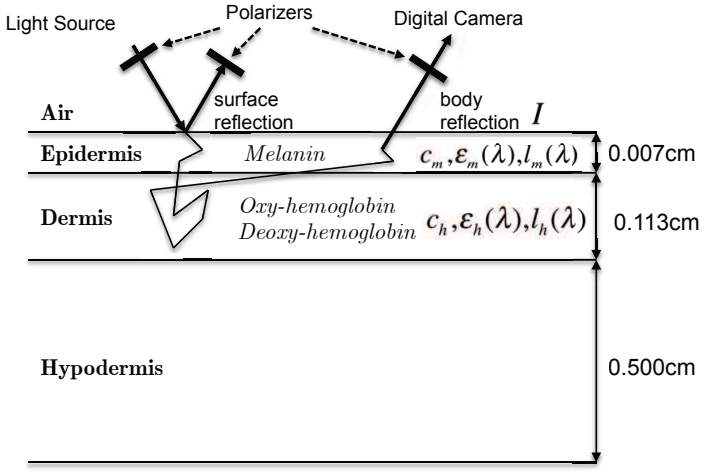


FIGURE A.7 – Schéma de la voie optique dans un modèle de peau à 3 couches (dispersion est omis).

Comme on peut le voir sur la figure A.7, une lumière qui pénètre dans la peau humaine est principalement absorbée par trois chromophores prédominants, la mélanine, oxyhémoglobine et désoxyhémoglobine. Ce sont les trois principaux déterminants de la couleur de peau. Sur la base de la loi de Beer-Lambert, l'absorbance de ce modèle de peau à une longueur d'onde λ peuvent être exprimée pour chaque pixel dans l'image

de la peau comme ceci :

$$\begin{aligned} A(\lambda) &= \log(1/R(\lambda)) \\ &= \epsilon_{\mathrm{HbO_2}}(\lambda) l_{\mathrm{HbO_2}}(\lambda) c_{\mathrm{HbO_2}} + \epsilon_{\mathrm{Hb}}(\lambda) l_{\mathrm{Hb}}(\lambda) c_{\mathrm{Hb}} + \epsilon_{\mathrm{Mel}}(\lambda) l_{\mathrm{Mel}}(\lambda) c_{\mathrm{Mel}} \end{aligned} \quad \text{(A.13)}$$

où l est la profondeur de pénétration de la lumière, c désigne la concentration du chromophore, et ϵ est le coefficient d'extinction.

A.4.2 Extraction de chromophore : approches fondées traitement de l'image

Parmi les approaches est basées sur le traitement d'image dans différents espaces colorimétriques, qui comprend : i) la décomposition du chromophore de la peau dans l'espace «TSL» proposé par Kim et Kim [56] et ii) séparation des distributions spatiales de la mélanine et l'hémoglobine dans l'espace «RVB» proposé par Tsumura et al. [91]. Un schéma de la méthode de Tsumura est représenté dans la figure A.8. La répartition de la couleur de la peau a d'abord été modélisée dans le domaine de densité optique. Ensuite, l'Analyse en composantes principales (PCA) a été adoptée pour extraire un plan bidimensionnel. Enfin, l'Analyse en composantes indépendantes (ICA) a été utilisé pour l'estimation de la mélanine et l'hémoglobine.

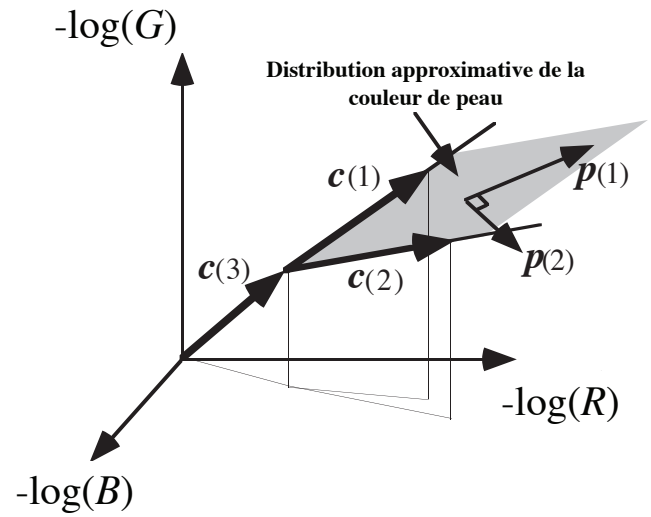
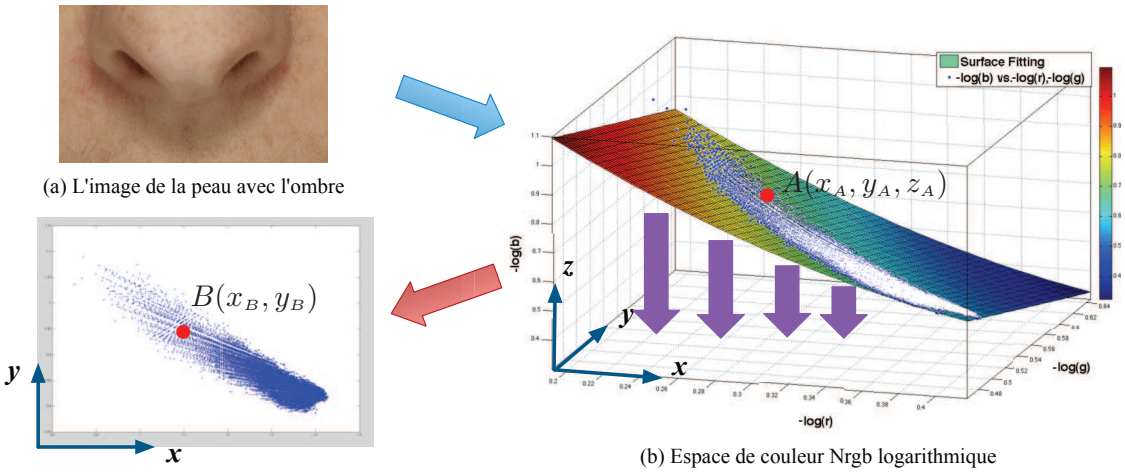


FIGURE A.8 – Un schéma de la méthode d'extraction de chromophore de la peau proposé par Tsumura et al. [91]

Cependant, la méthode de Tsumura est sensible à la variation de l'éclairage, et n'est valable que pour des petites régions de l'échantillon de peau. Pour pallier la faiblesse de la méthode de Tsumura, nous proprosed une nouvelle méthode, nommée «Surface Fitting and Flattening» (SF^2).



(a) L'image de la peau avec l'ombre

(b) Espace de couleur Nrgb logarithmique

(c) Aplatie plan à deux dimensions

FIGURE A.9 – Un schéma de la méthode proposée «SF^2».

Comme on peut le voir sur la figure A.9, la distribution de couleur de peau dans l'espace de couleur RVB normalisée (Nrgb) se trouve à approximativement sur une surface quadratique en trois dimensions. Nous proposons régression par un modèle de polynôme du second ordre et dépliement de cette surface pour obtenir un plan de couleur à deux dimensions. La distribution de la couleur de la peau peut être séparé par la suite en deux composantes, de l'hémoglobine et de la mélanine, par l'ICA.

A.4.3 Extraction de chromophore : approches basées sur des modèles physiques

Nous pouvons réécrire l'équation (A.13) sous une forme matricielle (cf. Equation (A.15)) :

$$\begin{bmatrix} \log(1/\mathbf{R}(\lambda_1)) \\ \vdots \\ \log(1/\mathbf{R}(\lambda_m)) \end{bmatrix} = \begin{bmatrix} \epsilon_{\mathrm{HbO_2}}(\lambda_1) & \epsilon_{\mathrm{Hb}}(\lambda_1) & \epsilon_{\mathrm{Mel}}(\lambda_1) \\ \vdots & \vdots & \vdots \\ \epsilon_{\mathrm{HbO_2}}(\lambda_m) & \epsilon_{\mathrm{Hb}}(\lambda_m) & \epsilon_{\mathrm{Mel}}(\lambda_m) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{\mathrm{HbO_2}} \\ \mathbf{c}_{\mathrm{Hb}} \\ \mathbf{c}_{\mathrm{Mel}} \end{bmatrix} \tag{A.14}$$

$$\mathbf{X} = \mathbf{AS} \tag{A.15}$$

où $\mathbf{X}$ est la matrice des données observées, $\mathbf{A}$ est une matrice de mélange et $\mathbf{S}$ est une matrice de données de source.

Lorsque la matrice de mélange $\mathbf{A}$ est inconnue, les méthodes basées sur la sépara-

tion aveugle de source (BSS), par exemple, la factorisation de la matrice non-négative (NMF), peuvent être utilisés pour résoudre la matrice de données de source **S**, dans le cas contraire, si la matrice de mélange **A** est donnée par les coefficients d'extinction tableaux de trois chromophores, nous avons proposé une méthode «Modèle-Fitting» (MF). Sur images couleur, nous pouvons résoudre le système d'équations linéaires (cf. Equation (A.14)) :

$$\mathbf{S} = \mathbf{A}_{\text{tabulate}}^{-1}\mathbf{X} \tag{A.16}$$

Si la méthode proposée est appliquée à des images multispectrales, nous pouvons résoudre ce système surdéterminé (cf. Equation (A.14)) utilisant une estimation des moindres carrés avec une seule contrainte donnée dans l'équation (A.17) :

$$\begin{aligned} &\arg\min_{\mathbf{A}_{\text{tabulated}}} \|\mathbf{X} - \mathbf{A}_{\text{tabulated}}\mathbf{S}\|^2 \\ &\textit{Subject to} : \mathbf{S} \geq 0 \end{aligned} \tag{A.17}$$

A.4.4 La Segmentation Automatique de lésion cutanée Pigmentée (APS) sur les images dermoscopiques

Puisque nous sommes maintenant en mesure d'estimer les concentrations des chromophores de la peau, il est utile d'ajouter les informations de chromophore de la peau á la configuration des caractéristiques multiples (cf. Figure A.10 de la partie «la configuration des caractéristiques»). Des expériences sur des images de lésions dermoscopique vérifient que la combinaison d'informations des chromophores de la peau avec les caractéristiques de la couleur, la texture et la forme peut conduire segmenter la lésion plus precis et robuste (cf. Table A.1).

Table A.1 – L'évaluation quantitative des différentes méthodes de segmentation sur 100 images dermoscopique. 'DSC' : Dice similarity coefficient, 'LS' : Lazy Snapping, 'BKJ' : Boykov-Jolly.

Criterion / Approach	DSC		Error ϵ		Precision		Recall	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Couleur	91.19	4.24	16.05	6.59	98.98	1.99	84.91	7.59
Couleur+LBP	91.41	3.89	15.70	6.47	98.74	2.02	85.69	7.68
Couleur+Chromophore	**93.85**	**3.08**	**12.26**	**5.95**	98.36	1.68	88.27	7.56
LS	88.73	4.68	20.15	7.61	98.76	1.53	80.89	7.67
BKJ	88.70	5.00	20.13	7.90	**99.07**	**0.81**	80.66	8.14
GrabCut	87.90	9.90	29.97	32.39	84.37	16.45	**95.53**	**4.08**

En outre, une approche a été proposé pour générer automatiquement des «graines» (cf. Figure A.10 de la partie «Auto-Seeding»). Tout d'abord, l'algorithme de *k*-Means++

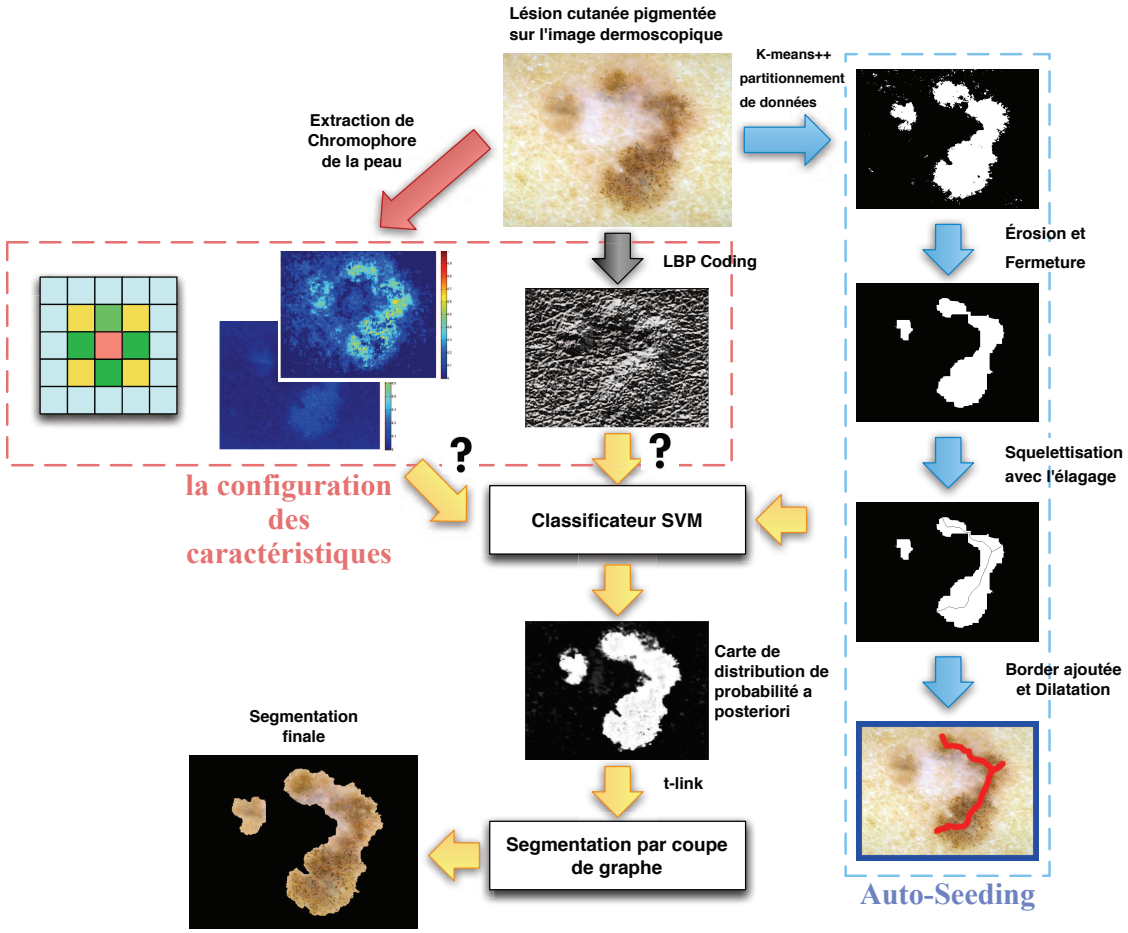


FIGURE A.10 – Un schéma de notre approche de **S**egmentation **A**utomatique de lésion cutanée **P**igmentée (APS), qui est particulièrement adapté pour des images de lésions dermoscopiques.

[12] a été adopté afin d'obtenir une segmentation grossière de lésion de la peau sur l'image dermoscopique. Ensuite, les opérations morphologiques tels que l'érosion et la fermeture ont été utilisés pour combler les trous et lisser la segmentation binaire obtenu. Après cela, l'image binaire est squelettée via un algorithme de squelettisation avec élagage [14] pour former les «graines» de l'objet. Les «graines» d'objets obtenues sont placés dans la région de la lésion, malgré de la présence d'artefacts tels que les poils ou la réflexion spéculaire. Pour obtenir les «graines» de fond, le bord de l'image suive la dilatation ont été utilisé.

A.5 Conclusion

Cette thèse utilise la notion de coupe de graphe pour segmenter automatiquement des images de mélanomes. Les contributions concernent l'utilisation d'un classifieur performant comme terme d'attache aux donneés, la gestion de contraintes faibles ainsi que la détermination automatique des graines pour notre application. Les résultats de cette methode sont precis et robuste.

Bibliography

[1] T. Ahonen, A. Hadid, and M. Pietikäinen. Face recognition with local binary patterns. In *Computer Vision-ECCV 2004*, volume 3021, pages 469–481. Springer, 2004. 46, 60

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. Network flows: theory, algorithms, and applications. 1993. 12, 155

[3] A. Aizerman, E. M. Braverman, and L. I. Rozoner. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964. 52

[4] M. Alam and D. Ratner. Cutaneous squamous-cell carcinoma. *New England Journal of Medicine*, 344(13):975–983, 2001. 76

[5] D. Aloise, A. Deshpande, P. Hansen, and P. Popat. Np-hardness of euclidean sum-of-squares clustering. *Machine Learning*, 75(2):245–248, 2009. 121

[6] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9(7):1545–1588, 1997. 57

[7] R. R. Anderson and J. A. Parrish. The optics of human skin. *J Investig Dermatol*, 77(1):13–19, 1981. 78, 80

[8] E. Angelopoulou. The reflectance spectrum of human skin. *Technical Reports (CIS)*, pages 583–584, 1999. 79

[9] F. J. Anthony, Jennifer T. J., D. S. Catherine, and J. C. Timothy. Early detection and treatment of skin cancer. *American Family Physician*, 62(2):357–368, 2000. 76

[10] G. Argenziano, G. Fabbrocini, P. Carli, V. De Giorgi, E. Sammarco, and M. Delfino. Epiluminescence microscopy for the diagnosis of doubtful melanocytic skin lesions: comparison of the abcd rule of dermatoscopy and a new 7-point checklist based on pattern analysis. *Archives of Dermatology*, 134(12):1563, 1998. 1, 76, 151

[11] G. Argenziano, H. P. Soyer, V. De Giorgi, D. Piccolo, P. Carli, and M. Delfino. Dermoscopy: a tutorial. *Milan, Italy: EDRA, Medical Publishing & New Media*, 2002. 132

[12] D. Arthur and S. Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete Algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007. 33, 123, 166

[13] P. A. Ascierto, G. Palmieri, E. Celentano, R. Parasole, C. Caracò, A. Daponte, M. G. Chiofalo, M. T. Melucci, N. Mozzillo, and R. A. Satriano. Sensitivity and specificity of epiluminescence microscopy: evaluation on a sample of 2731 excised cutaneous pigmented lesions. *British Journal of Dermatology*, 142(5):893–898, 2000. 115

[14] X. Bai, L. J. Latecki, and W.-Y. Liu. Skeleton pruning by contour partitioning with discrete curve evolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):449–462, 2007. 118, 123, 127, 166

[15] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 192–236, 1974. 20

[16] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 259–302, 1986. 6, 10, 154

[17] M. Binder, H. Kittler, S. Dreiseitl, H. Ganster, K. Wolff, and H. Pehamberger. Computer-aided epiluminescence microscopy of pigmented skin lesions: the value of clinical data for the classification process. *Melanoma research*, 10(6):556–561, 2000. 1

[18] M. Binder, M. Schwarz, A. Winkler, A. Steiner, A. Kaider, K. Wolff, and H. Pehamberger. Epiluminescence microscopy: a useful tool for the diagnosis of pigmented skin lesions for formally trained dermatologists. *Archives of Dermatology*, 131(3):286, 1995. 77

[19] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 489–495. IEEE, 1999. 24

[20] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. Interactive image segmentation using an adaptive gmmrf model. In *Computer Vision-ECCV 2004*, pages 428–441. Springer, 2004. 38, 39, 41, 65, 68, 70, 74

[21] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational Learning Theory*, pages 144–152. ACM, 1992. 55

[22] Y. Boykov and G. Funka-Lea. Graph cuts and efficient nd image segmentation. *International Journal of Computer Vision*, 70(2):109–131, 2006. xi, xii, 9, 11, 153, 157

[23] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Proceedings of the Eighth IEEE International Conference on Computer Vision*, volume 1, pages 105–112. IEEE, 2001. xi, xv, 24, 25, 27, 34, 35, 38, 45, 61, 116, 156

[24] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1124–1137, 2004. xi, 13, 16, 17, 27, 29

[25] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–655. IEEE, 1998. 24

[26] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001. 9, 24, 153

[27] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 57

[28] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997. 24, 156

[29] M. E. Celebi, S. Hwang, H. Iyatomi, and G. Schaefer. Robust border detection in dermoscopy images using threshold fusion. In *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP)*, pages 2541–2544. IEEE, 2010. 116

[30] M. E. Celebi, H. Iyatomi, G. Schaefer, and W. V. Stoecker. Approximate lesion localization in dermoscopy images. *Skin Research and Technology*, 15(3):314–322, 2009. 147

[31] M. E. Celebi, H. A. Kingravi, H. Iyatomi, Y. Alp Aslandogan, W. V. Stoecker, R. H. Moss, J. M. Malters, J. M. Grichnik, A. A. Marghoob, and H. S. Rabinovitz. Border detection in dermoscopy images using statistical region merging. *Skin Research and Technology*, 14(3):347–353, 2008. 116

[32] M. E. Celebi, G. Schaefer, H. Iyatomi, and W. V. Stoecker. Lesion border detection in dermoscopy images. *Computerized medical imaging and graphics: the official journal of the Computerized Medical Imaging Society*, 33(2):148, 2009. 1, 151

[33] A. Chardon, I. Cretois, and C. Hourseau. Skin colour typology and suntanning pathways. *International Journal of Cosmetic Science*, 13(4):191–208, 1991. 81, 83

[34] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995. 52, 54

[35] Martin D., Fowlkes C., Tal D., and Malik J. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the 8th International Conference on Computer Vision*, volume 2, pages 416–423, July 2001. 31, 34, 35, 36, 37, 38, 39, 65, 68, 69, 70, 74

[36] G. Dantzig and D. R. Fulkerson. On the max flow min cut theorem of networks. *Linear inequalities and related systems*, 38:225–231, 2003. 12

[37] E. A. Dinic. Algorithm for Solution of a Problem of Maximum Flow in a Network with Power Estimation. *Soviet Math Doklady*, 11:1277–1280, 1970. 13, 16

[38] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19(2):248–264, 1972. 13

[39] B. Erkol, R. H. Moss, Joe S. R., W. V. Stoecker, and E. Hvatum. Automatic lesion boundary detection in dermoscopy images using gradient vector flow snakes. *Skin Research and Technology*, 11(1):17–26, 2005. 116

[40] V. Estivill-Castro. Why so many clustering algorithms: a position paper. *ACM SIGKDD Explorations Newsletter*, 4(1):65–75, 2002. 121

[41] D. R. Ford and D. R. Fulkerson. *Flows in networks*. Princeton university press, 2010. xi, 11, 12, 13, 14, 15, 155

[42] D. Freedman and P. Drineas. Energy minimization via graph cuts: Settling what is possible. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 939–946. IEEE, 2005. 19, 20

[43] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997. 42

[44] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984. 6, 10, 11, 154

[45] A. V. Goldberg and R. E. Tarjan. A new approach to the maximum-flow problem. *Journal of the ACM (JACM)*, 35(4):921–940, 1988. 13

[46] D. M. Greig, B. T. Porteous, and A. H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 271–279, 1989. 6, 9, 11

[47] M. Hiraoka, M. Firbank, M. Essenpreis, M. Cope, S. R. Arridge, P. Van Der Zee, and D. T. Delpy. A monte carlo investigation of optical pathlength in inhomogeneous tissue and its application to near-infrared spectroscopy. *Physics in Medicine and Biology*, 38(12):1859, 1993. 85

[48] C. M. Holt, A. Stewart, M. Clint, and R. H. Perrott. An improved parallel thinning algorithm. *Communications of the ACM*, 30(2):156–160, 1987. 127

[49] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7):1483–1492, 1997. 101

[50] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *ECCV'98*, pages 232–248. Springer, 1998. 24

[51] H. Iyatomi, H. Oka, M. Saito, A. Miyake, M. Kimoto, J. Yamagami, S. Kobayashi, A. Tanikawa, M. Hagiwara, and K. Ogawa. Quantitative assessment of tumour extraction from dermoscopy images and evaluation of computer-based extraction methods for an automatic melanoma diagnostic system. *Melanoma Research*, 16(2):183–190, 2006. 116

[52] A. Jemal, T. Murray, E. Ward, A. Samuels, R. C. Tiwari, A. Ghafoor, E. J. Feuer, and M. J. Thun. Cancer statistics, 2005. *CA: A Cancer Journal for Clinicians*, 55(1):10–30, 2005. 1, 151

[53] T. Joachims. *Text categorization with support vector machines: Learning with many relevant features*. Springer, 1998. 52

[54] B. Jung, B. Choi, A. J. Durkin, K. M. Kelly, and J. S. Nelson. Characterization of port wine stain skin erythema and melanin content using cross-polarized diffuse reflectance imaging. *Lasers in Surgery and Medicine*, 34(2):174–181, 2004. 81, 83

[55] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988. 24, 156

[56] D. H. Kim and M.-J. Kim. Skin color analysis in hsv color space and rendering with fine scale skin structure. In *Advances in Computer Graphics*, pages 254–264. Springer, 2006. xii, 81, 91, 92, 162

[57] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, pages 1033–1040. IEEE, 2003. 24

[58] V. Kolmogorov, Y. Boykov, and C. Rother. Applications of parametric maxflow in computer vision. In *IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE, 2007. 41

[59] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV 2002*, pages 82–96. Springer, 2002. 9, 153

[60] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004. 19, 20, 38

[61] H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proceedings of the second Berkeley symposium on Mathematical Statistics and Probability*, volume 5. California, 1951. 53

[62] L. J. Latecki and R. Lakämper. Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73(3):441–454, 1999. 118

[63] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999. 102

[64] M. B. Lens and M. Dawes. Global perspectives of contemporary epidemiological trends of cutaneous malignant melanoma. *British Journal of Dermatology*, 150(2):179–185, 2004. 1, 151

[65] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum. Lazy snapping. *ACM Transactions on Graphics (ToG)*, 23(3):303–308, 2004. xi, xv, 27, 28, 29, 34, 35, 44, 45, 116, 156

[66] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. 121

[67] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on Mathematical Statistics and Probability*, volume 1, page 14. California, USA, 1967. 28, 121

[68] R. Melli, C. Grana, and R. Cucchiara. Comparison of color clustering algorithms for segmentation of dermatological images. In *Medical Imaging*, pages 61443S–61443S–9. International Society for Optics and Photonics, 2006. 116

[69] S. W. Menzies. A method for the diagnosis of primary cutaneous melanoma using surface microscopy. *Dermatologic clinics*, 19(2):299–305, 2001. 116

[70] T. Ojala and M. Pietikäinen. Unsupervised texture segmentation using feature distributions. *Pattern Recognition*, 32(3):477–486, 1999. 46, 60

[71] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996. 46, 158

[72] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. 46, 48, 60

[73] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 130–136. IEEE, 1997. 52

[74] J. R. Parker. *Algorithms for image processing and computer vision*. Wiley, 2010. 127

[75] B. Peng and O. Veksler. Parameter selection for graph cut based image segmentation. In *British machine vision conference*, 2008. 40, 41

[76] J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74, 1999. 56

[77] D. S. Rigel, R. J. Friedman, A. W. Kopf, and D. Polsky. Abcde–an evolving concept in the early detection of melanoma. *Archives of dermatology*, 141(8):1032, 2005. 1, 76, 151

[78] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (ToG)*, volume 23, pages 309–314. ACM, 2004. xi, xv, 29, 31, 36, 37, 40, 44, 116, 158

[79] P. Rubegni, A. Ferrari, G. Cevenini, D. Piccolo, M. Burroni, R. Perotti, K. Peris, P. Taddeucci, M. Biagioli, and G. Dell'Eva. Differentiation between pigmented spitz naevus and melanoma by digital dermoscopy and stepwise logistic discriminant analysis. *Melanoma Research*, 11(1):37–44, 2001. 116

[80] P. Schmid. Segmentation of digitized dermatoscopic images by two-dimensional color clustering. *IEEE Transactions on Medical Imaging*, 18(2):164–171, 1999. 116

[81] B. Schölkopf, P. Simard, V. Vapnik, and A. J. Smola. Improving the accuracy and speed of support vector machines. In *Advances in Neural Information Processing Systems: Proceedings of the 1996 Conference on Neural Information, Denver, USA*, volume 9, page 375. The MIT Press, 1997. 52

[82] P. Scott. Tabulated molar extinction coefficient for hemoglobin in water. `http://omlc.ogi.edu/spectra/hemoglobin/summary.html`. 104, 111

[83] M. D. Shriver and E. J. Parra. Comparison of narrow-band reflectance spectroscopy and tristimulus colorimetry for measurements of skin and hair color in persons of different biological ancestry. *American Journal of Physical Anthropology*, 112(1):17, 2000. 84

[84] M. Silveira, J. C. Nascimento, J. S. Marques, A. R. S. Marçal, T. Mendonça, S. Yamauchi, J. Maeda, and J. Rozeira. Comparison of segmentation methods for melanoma diagnosis in dermoscopy images. *IEEE Journal of the Selected Topics in Signal Processing*, 3(1):35–45, 2009. 116

[85] G. N. Stamatas, B. Z. Zmudzka, and N. Kollias. Non-invasive measurements of skin pigmentation in situ. *Pigment Cell Research*, 17(6):618–626, 2004. 84, 101, 108

[86] A. Steiner, M. Binder, M. Schemper, K. Wolff, and H. Pehamberger. Statistical evaluation of epiluminescence microscopy criteria for melanocytic pigmented skin lesions. *Journal of the American Academy of Dermatology*, 29(4):581–588, 1993. 1

[87] F. W. M. Stentiford and R. G. Mortimer. Some new heuristics for thinning binary handprinted characters for OCR. *IEEE Transactions on Systems, Man and Cybernetics*, (1):81–84, 1983. 127

[88] J. Steven. Extinction coefficient of melanin. `http://omlc.ogi.edu/spectra/melanin/eumelanin.html`. 104, 111

[89] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1068–1080, 2008. 8

[90] V. Takala, T. Ahonen, and M. Pietikäinen. Block-based methods for image retrieval using local binary patterns. In *Image Analysis*, pages 882–891. Springer, 2005. 46, 60

[91] N. Tsumura, H. Haneishi, and Y. Miyake. Independent-component analysis of skin color image. *JOSA A*, 16(9):2169–2176, 1999. xii, xiii, 82, 84, 86, 162

[92] N. Tsumura, N. Ojima, K. Sato, M. Shiraishi, H. Shimizu, H. Nabeshima, S. Akazaki, K. Hori, and Y. Miyake. Image-based skin color and texture analysis/synthesis by extracting hemoglobin and melanin information in the skin. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 770–779. ACM, 2003. 86

[93] V. N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5):988–999, 1999. 56

[94] O. Veksler. *Efficient graph-based energy minimization methods in computer vision*. PhD thesis, Cornell University, 1999. 8

[95] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 37–44. IEEE, 2006. 57

[96] L. Xu, M. Jackowski, A. Goshtasby, D. Roseman, S. Bines, C. Yu, A. Dhawan, and A. Huntley. Segmentation of skin cancer images. *Image and Vision Computing*, 17(1):65–74, 1999. 116

[97] T. Yamamoto, H. Takiwaki, and H. Ohshima. Derivation and clinical application of special imaging by means of digital cameras and image j freeware for quantification of erythema and pigmentation. *Skin Research and Technology*, 14(1):26–34, 2008. 100, 108

[98] A. R. Young. Chromophores in human skin. *Physics in Medicine and Biology*, 42(5):789, 1997. 78

[99] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–239, 1984. 127

[100] H. Zhou, M. Chen, L. Zou, R. Gass, L. Ferris, L. Drogowski, and J. M. Rehg. Spatially constrained segmentation of dermoscopy images. In *Proceedings of the 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 800–803. IEEE, 2008. 116

[101] M. Zortea, S. O. Skrøvseth, T. R. Schopf, H. M. Kirchesch, and F. Godtliebsen. Automatic segmentation of dermoscopic images by iterative classification. *Journal of Biomedical Imaging*, 2011:3, 2011. 117

Title: Segmentation of Color and Multspectral Skin Images

Abstract: Accurate border delineation of pigmented skin lesion (PSL) images is a vital first step in computer-aided diagnosis (CAD) of melanoma. This thesis presents a novel approach of automatic PSL border detection on color and multispectral skin images. We first introduce the concept of energy minimization by graph cuts in terms of maximum *a posteriori* estimation of a Markov random field (MAP-MRF framework). After a brief state of the art in interactive graph-cut based segmentation methods, we study the influence of parameters of the segmentation algorithm on color images. Under this framework, we propose an energy function based on efficient classifiers (support vector machines and random forests) and a feature vector calculated on a local neighborhood. For the segmentation of melanoma, we estimate the concentration maps of skin chromophores, discriminating indices of melanomas from color and multispectral images, and integrate these features in a vector. Finally, we detail a global framework of automatic segmentation of melanoma, which comprises two main stages: automatic selection of "seeds" useful for graph cuts and the selection of discriminating features. This tool is compared favorably to classic graph-cut based segmentation methods in terms of accuracy and robustness.

Keywords: Image segmentation, Graphs, Melanoma, Lesion delineation, Classification.

Titre : Segmentation de l'Images Couleurs et Multispectrales de la Peau

Résumé : La délimitation précise du contour des lésions pigmentées sur des images est une première étape importante pour le diagnostic assisté par ordinateur du mélanome. Cette thèse présente une nouvelle approche de la détection automatique du contour des lésions pigmentaires sur des images couleurs ou multispectrales de la peau. Nous présentons d'abord la notion de minimisation d'énergie par coupes de graphes en terme de Maxima A-Posteriori d'un champ de Markov. Après un rapide état de l'art, nous étudions l'influence des paramètres de l'algorithme sur les contours d'images couleurs. Dans ce cadre, nous proposons une fonction d'énergie basée sur des classifieurs performants (Machines à support de vecteurs et Forêts aléatoires) et sur un vecteur de caractéristiques calculé sur un voisinage local. Pour la segmentation de mélanomes, nous estimons une carte de concentration des chromophores de la peau, indices discriminants du mélanomes, à partir d'images couleurs ou multispectrales, et intégrons ces caractéristiques au vecteur. Enfin, nous détaillons le schéma global de la segmentation automatique de mélanomes, comportant une étape de sélection automatique des «graines» utiles à la coupure de graphes ainsi que la sélection des caractéristiques discriminantes. Cet outil est comparé favorablement aux méthodes classiques à base de coupure de graphes en terme de précision et de robustesse.

Mots clés : Segmentation d'images, Graphes, Mélanome, Délimitation des lésions, Classification.