One thing I have learned in a long life: that all our science, measured against reality, is primitive and childlike – and yet it is the most precious thing we have.

– Albert Einstein, 1879-1955, German-born American physicist.

University of Alberta

Probabilistic Methods for Discrete Labeling Problems in Digital Image Processing and Analysis

by

Rui Shen

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Computing Science

Fall 2012
Edmonton, Alberta

To my family and friends,
who have supported me through the toughest times in the past years of my life.

Abstract

Many problems in digital image processing and analysis can be interpreted as labeling problems, which aim to find the optimal mapping from a set of sites to a set of labels. A site represents a certain primitive, such as a pixel, while a label represents a certain quantity, such as disparity in stereo correspondence. Considering this labeling interpretation, instead of solving different problems individually, we propose a series of unified frameworks in the random walks (RW) context for labeling problems with regular sites and discrete labels. The first framework, which we term as the generalized random walks (GRW) framework, converts a discrete labeling problem into the calculation of steady-state probabilities of random walkers transiting on a graph. The performance of GRW is validated through experiments on stereo correspondence and multi-exposure fusion, which we formulate as labeling problems in the RW context. By incorporating hierarchical and multiscale schemes, the performance of GRW is further improved. This leads to two enhanced frameworks: the hierarchical random walks (HRW) framework, which reduces the computational cost of GRW but produces good approximations; and the multiscale random walks (MRW) framework, which produces robust solutions utilizing both inter- and intra-scale information. The performance of HRW is validated through perception-guided multi-exposure fusion, where we also introduce advanced perceptual metrics into multi-exposure fusion; the performance of MRW is verified through volumetric medical image fusion, where we also introduce a cross-scale fusion rule based on MRW. Furthermore, a multivariate Gaussian conditional random field model and its hierarchical version are proposed for more general multi-label problems, and their relationships with GRW and HRW are analyzed.

Acknowledgements

I owe much gratitude to a number of people, without whose help and encouragement, this dissertation would not have been possible. First of all, I would like to express my earnest gratitude to my supervisors, Dr. Anup Basu and Dr. Irene Cheng, for guiding and supporting me through my Ph.D. studies. I thank them for sharing with me their insightful perspectives, for giving me the freedom to explore different directions, and for helping me tap my potential.

I am indebted to Dr. Pierre Boulanger, my M.Sc. supervisor and Ph.D. supervisory committee member, and Dr. Nilanjan Ray, my Ph.D. thesis exam committee member, for sharing with me their knowledge and offering me helpful insights on my thesis research. Thanks also go to other faculty members in the Department of Computing Science, University of Alberta, including Dr. Xiaobo Li, Dr. Piotr Rudnicki, and Dr. Sherif Ghali. Special thanks to Dr. Guohui Lin for chairing my Ph.D. thesis exam. I would also like to thank Dr. Witold Pedrycz and Dr. Cha Zhang for their valuable comments and feedback as my Ph.D. thesis exam committee members.

I thank my labmates, especially Dr. Tao Wang, Nathaniel Rossol, and Dr. Hossein Azari, with whom I have had the pleasure to work with. I am also grateful to the support staff in the Department of Computing Science, University of Alberta, especially Edith Drummond, Fran Moore, and Sunrose Ko, for assisting me in many different ways. The financial support from this department, Alberta Innovates Technology Futures, and Killam Trusts is also greatly appreciated.

I thank my friends, especially Xingdong Yang, Guowei Wu, Shanny Lu, Roger Barbour, Jun Ma, Wei Ren, Gaopeng Gou, Dr. Baochun Bai, Xiaozhou Zhou, Dr. Le Luo, and Dr. Tiansheng Zhou, who have generously helped me through many difficult times and who have made my life in Edmonton much more fun and enjoyable. I cherish every precious moment that we have shared. Special thanks to Jie Pan, Dr. Qiang Guo, and Dr. Qin Wang. I owe sincere thankfulness to Jiaxi Wang, who always believes in me and supports me to achieve my goals.

Lastly, but most importantly, I express my deepest gratitude to my parents, Zhuyong Shen and Xiuhua Su, who always teach me to be an honorable, responsible, and forethoughtful man and help me stand strong in the face of adversity. Without their continuous support and faith in me, I would not have been able to succeed in my studies.

Table of Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

Human eyes and brains have been capturing and processing visual information for millennia, long before any special apparatus or algorithm was introduced. Yet during this time, people have never stopped the efforts to decipher human perception, to invent tools and methods to mimic, facilitate, or enhance human perception, to create informative and/or appealing visual data, and to develop techniques and devices to extract meaningful information from visual data. More specifically, various directions have been explored by researchers from different disciplines (including physiology, computing science, engineering, and physics) in order to solve different problems in visual information processing, from studying human perception [118, 158, 182] to building optical devices [2, 126], and from developing processing algorithms/software [3, 148] to constructing specialized hardware [84, 164]. Working in the diverse field of research into visual information processing, we focus on digital image processing and analysis. With the rapid advancement of digital image acquisition technology and computer hardware, digital image processing and analysis has gained significant popularity with a wide range of applications in medicine, astronomy, geography, *etc.*, since its debut in the 1960s [65]. Despite its popularity, different problems in digital image processing and analysis are usually treated separately with more emphasis on their distinct attributes than their common characteristics. While recognizing the former helps to construct problem-specific solution paradigms, respecting the latter helps to derive general frameworks, which can lead to a more systematic analysis of a given problem. A general framework also helps to better identify the unique attributes of a given problem and then more efficiently incorporate them into the design of a problem-specific algorithm, which is an instantiation of the general framework. Hence, in this thesis, we focus on designing a series of general frameworks for digital image processing and analysis exploiting a common characteristic and deriving specific algorithms for particular problems, which produce enhanced images or extract useful information from input images.

1.1 Motivation

Many seemingly different problems in digital image processing and analysis share a common characteristic, *i.e.,* they all need to assign a certain quantity to each given primitive (*e.g.,* a pixel). The following are several representative examples:

1. Foreground/background segmentation [68, 88]: Given an image, the goal is to distinguish the set of pixels that belong to the foreground from those belonging to the background. In other words, foreground/background segmentation aims to assign either foreground or background to each pixel of a given image.

2. Stereo correspondence [90, 177]: Given an image pair, the goal is to determine corresponding pixels between them, where only horizontal displacement is present. We can consider an alternative interpretation: Given an image pair where one image is taken as the reference image, the goal is to determine the displacement of each pixel of the reference image in the other image. Based on this interpretation, stereo correspondence amounts to assigning a displacement to each pixel of the reference image.

3. Optical flow [12, 104]: Given two consecutive frames, the goal is to estimate the motion for each pixel, *i.e.,* to assign a displacement vector for each pixel of the earlier frame. This problem is similar to stereo correspondence, except that both horizontal and vertical displacements are considered.

4. Image restoration [14, 55]: Given a degraded image, the goal is to recover the original pixel values (*i.e.,* intensities or colors), which means to assign a meaningful value to each corrupted pixel.

Considering this common characteristic, we can apply a general framework to such digital image processing and analysis problems. This unified modeling scheme allows us to tackle these problems at a higher level; and once a specific problem is appropriately formulated under this framework, an optimal solution can be quickly obtained. However, in the meanwhile, we have to acknowledge that these problems also possess distinct attributes. Therefore, after modeling the problems under this general framework, we need to utilize proper *distance metrics* to accommodate these attributes. Mathematically, a distance metric is a function that calculates the distance between two or more entities, such as the difference between the colors of two pixels. However, here, we consider distance metrics in a broader sense, in which perceptual metrics are also included. For instance, a distance metric can be a metric that evaluates how an image feature is perceived by a human observer (*e.g.,* the distance between a presented image feature and the ideal feature that can capture a high degree of attention). With these considerations, we attempt to develop a set of effective and efficient general frameworks and apply them to practical digital image processing and analysis problems in this thesis.

1.2 Problem Statement

The relationship between problems in digital image processing and analysis and labeling problems was closely studied in [110, 111]. A *labeling problem* can be defined as finding an optimal mapping f^* from $\mathcal{D} = \{d_1, d_2, \ldots, d_N\}$, a set of sites, to $\mathcal{L}$, a set of labels, among the set $\mathbf{F}$ of all feasible mappings, subject to some constraint(s):

$$f^* : \mathcal{D} \rightarrow \mathcal{L}. \tag{1.1}$$

Let $\mathcal{X}$ denote a set of random variables defined on $\mathcal{D}$, where each $x \in \mathcal{X}$ is associated with a site $d \in \mathcal{D}$ and takes a label $l \in \mathcal{L}$. Let x_i denote the variable associated with d_i. A mapping $f \in \mathbf{F}$, *i.e.,* a realization of $\mathcal{X} = \{x_1 = f_1, x_2 = f_2, \ldots, x_N = f_N\} = f$, is called a *(label) configuration*, *assignment* or *labeling*, where $f_i \in \mathcal{L}$ denotes the label assigned to the variable x_i. If $\mathcal{L}$ is discrete, *i.e.,* $\mathcal{L} = \{l_1, l_2, \ldots, l_K\}$, then the set $\mathcal{F}$ of all configurations can be defined as:

$$\mathcal{F} = \mathcal{L}^N. \tag{1.2}$$

Normally, $\mathcal{F} = \mathbf{F}$.

A site takes different forms in different problems: it can be a pixel, a region, a feature, etc. Similarly, a label also takes different meanings in different problems: it can be an intensity value, a disparity, an object class, etc. A set of sites can be spatially regular/homogeneous (*e.g.,* a lattice) or irregular/inhomogeneous. A set of labels can be either continuous (*e.g.,* a real interval) or discrete (*e.g.,* a finite set). Based on the spatial regularity of the sites and the continuity of the labels, labeling problems in digital image processing and analysis can be classified into four categories [110, 111]: regular sites with continuous labels (LP1); regular sites with discrete labels (LP2); irregular sites with discrete labels (LP3); and irregular sites with continuous labels (LP4). Restoration of images containing continuous pixel values is an LP1, where the sites are a lattice of pixels. Stereo correspondence with discrete disparities and image segmentation are LP2s. Feature-based object detection and recognition can be considered as LP3s, where the sites are extracted image features and the labels are discrete object classes. Region-based continuous optical flow can be considered as an LP4.

1.3 Scope and Significance

The focus of this thesis is on the design and application of probabilistic methods in discrete labeling problems in digital image processing and analysis, with a main focus on pixel labeling (LP2). The major contributions are:

1. Proposing a generalized random walks (GRW) framework for solving discrete labeling problems (Section 3.1); proposing the multivariate Gaussian conditional random field (MGCRF) and interpreting GRW from the perspective of MGCRF (Section 3.4).

2. Formulating stereo correspondence (Section 3.2) and multi-exposure fusion (Section 3.3) in the GRW framework, and proposing effective algorithms.

3. Proposing a hierarchical random walks framework reducing the temporal and spatial complexity of GRW (Section 4.1) and appling it to image fusion, where we also introduce perception-based metrics (Section 4.2); proposing the hierarchical MGCRF (HMGCRF)) and interpreting HRW from the perspective of HMGCRF (Section 4.3).

4. Proposing a multiscale random walks framework incorporating multiscale analysis into GRW (Section 5.1) and applying it to volumetric medical image fusion (Section 5.2).

1.4 Organization

The remainder of this thesis is organized as follows. **Chapter 2** discusses some popular graph-based probabilistic methods for discrete labeling problems in digital image processing and analysis. More specifically, we review the Markov random field model, belief propagation, graph cuts, and the random walks model.

The major contributions of this thesis are presented in three subsequent chapters. In **Chapter 3**, we propose a generalized random walks framework, which transforms a labeling problem into steady-state probability calculation in the random walks context. The application of GRW in two LP2 problems, stereo correspondence and multi-exposure fusion, is introduced. Using basic metrics and disparity refinement schemes, we show that our GRW-based stereo correspondence algorithm has achieved promising performance on standard test images. Using a GRW-based formulation of multi-exposure fusion, we show that our algorithm is able to achieve a good balance between detail reproduction and color fidelity preservation. We also propose a multivariate Gaussian conditional random field model and investigate the relationship between GRW and MGCRF.

In **Chapter 4**, we propose a hierarchical random walks framework, which reduces the computational complexity of GRW by computing the solution in a coarse-to-fine manner. The improved performance of HRW compared with GRW is validated through multi-exposure fusion. In addition, an enhanced multi-exposure fusion algorithm is proposed by introducing new perceptual factors. The possibility of applying this enhanced algorithm to other fusion problems is also investigated. In addition, we propose a hierarchical MGCRF model and analyze the relationship between HRW and HMGCRF.

In **Chapter 5**, we propose a multiscale random walks framework, which combines GRW with multiscale analysis. A cross-scale fusion rule is proposed under the MRW framework that guarantees intra- and inter-scale consistencies. This fusion rule is validated through volumetric medical image fusion. In addition, an efficient color fusion scheme is also proposed.

Finally, conclusions and possible future research directions are presented in **Chapter 6**.

Chapter 2

Background and Related Work

This section discusses some popular graph-based probabilistic methods for discrete labeling problems in digital image processing and analysis. Two probabilistic models are reviewed: the *Markov random field* (MRF) model and the *random walks* model. One widely-used inference algorithm that works on MRFs, called *belief propagation*, is discussed. One representative optimization algorithm that works for MRF posterior energy minimization, called *graph cuts* (GC), is also discussed. There also exist other optimization methods that can be applied to minimize an MRF posterior energy, such as simulated annealing [86] and relaxation labeling [112], but they are either inefficient or ineffective in finding the global minimum (or a high-quality approximation to the global minimum). Compared with these methods, GC has gained popularity in many digital image processing and analysis applications due to its effectiveness and efficiency. Therefore, we choose GC as a representative MRF posterior energy minimization method to be discussed in this chapter.

2.1 Markov and Conditional Random Fields

The main focus of this section is to provide a brief introduction to Markov random fields and maximum *a posteriori* labeling on MRFs, which are widely used in digital image processing and analysis problems. The descriptions in this section are mainly based on [111, 160].

2.1.1 Graphs and Graphical Models

To solve a labeling problem, it is convenient to represent the problem on a graph. A graph is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{X}$ is the set of all the nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of all the edges. An image is usually represented as a finite lattice as shown in Figure 2.1, because such representation facilitates both the encoding of the scene structure and the subsequent computation. In digital image processing and analysis problems, each scene node $x_i \in \mathcal{X}$ is normally associated with an observed quantity (*e.g.,* a pixel) $d_i \in \mathcal{D}$ at location i. An edge connecting two scene nodes x_i and x_j is denoted by $e_{ij} = (x_i, x_j) \in \mathcal{E}$. A weighted graph has a value assigned to each edge. Let w_{ij} denote the weight associated with e_{ij}. This graph can be either undirected where $w_{ij} = w_{ji}$ or directed. When

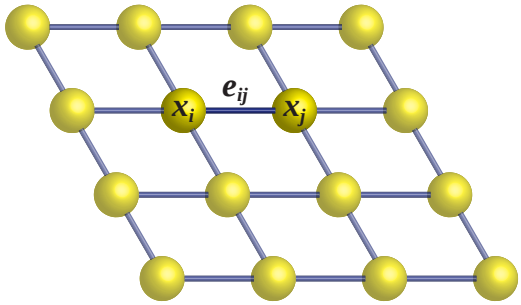


Figure 2.1: A finite lattice.

the probability theory is combined with a graph, where nodes represent random variables and edges represent dependency assumptions, this graph becomes a *(probabilistic) graphical model* [132]. A graphical model, like a graph, can be either undirected or directed. An MRF is an undirected graph that can take a cyclic form, in which each edge represents symmetric dependencies, as opposed to a Bayesian network, which is a directed acyclic graph, in which each edge represents the causal influence. While Bayesian networks are good at representing induced dependencies, MRFs are capable of representing cyclic dependencies. Because the associations of adjacent nodes are usually considered as correlational rather than causal [174], cyclic dependency representation is preferable in many digital image processing and analysis problems.

2.1.2 Markov Random Fields

MRFs were first applied to digital image processing and analysis problems by Geman and Geman [63] for image restoration. $\mathcal{X} = \{x_1, x_2, \ldots, x_N\}$, a set of random variables, is said to be an MRF with respect to a neighboring system $\mathcal{N}$ if and only if it satisfies the positivity property:

$$P(\mathcal{X} = f) > 0, \forall f \in \mathcal{F}, \tag{2.1}$$

and the Markov property:

$$P(x_i = f_i | \{x_j\}_{x_j \in \mathcal{X} \backslash x_i}) = P(x_i = f_i | \{x_j\}_{x_j \in \mathcal{N}_i}), \forall x_i \in \mathcal{X}, \tag{2.2}$$

where $P(\mathcal{X} = f) \triangleq P(x_1 = f_1, x_2 = f_2, \ldots, x_N = f_N)$ denotes the joint probability that each variable takes a certain value; $\mathcal{X} \backslash x_i \triangleq \mathcal{X} - \{x_i\}$; and $\mathcal{N}_i$ denotes the neighborhood around x_i. It is not necessary for a neighbor $x_j \in \mathcal{N}_i$ of x_i to be the immediate neighbor of x_i, but their distance should be less than a specified threshold δ:

$$\mathcal{N}_i = \{x_j | |x_i - x_j| \leq \delta, \forall x_j \in \mathcal{X} \backslash x_i\}. \tag{2.3}$$

For a two-dimensional (2D) image, the distance is usually computed as the Euclidean distance, and δ is usually taken as $\delta = 1$, *i.e.,* 4-neighbor connectivity is assumed. A neighboring system $\mathcal{N}$ is defined as the union of all $\mathcal{N}_i$'s:

$$\mathcal{N} = \{\mathcal{N}_i | \forall x_i \in \mathcal{X}\}. \tag{2.4}$$

When continuous labels are considered (*i.e.,* for an LP1 or LP4), the probabilities are replaced as probability densities.

2.1.3 Label Configurations on MRFs

According to the categorization of inference problems on graphical models (*e.g.,* MRFs) in [194], the configuration for a labeling problem can be a maximum *a posteriori* (MAP) configuration or a maximum posterior marginal (MPM) configuration[1].

2.1.3.1 Maximum *A Posteriori* Configuration

A labeling problem can be interpreted as a search for the most probable configuration f^* for all the unobserved variables $\mathcal{X}$ based on the observed data $\mathcal{D}$. Such a configuration f^* is called the MAP configuration and defined as:

$$f^* \triangleq \arg\max_{f \in \mathcal{F}} P(f|\mathcal{D}), \tag{2.5}$$

where $P(f|\mathcal{D}) \triangleq P(\mathcal{X} = f|\mathcal{D})$ denotes the probability that configuration f is assigned based on $\mathcal{D}$. It is sometimes difficult to directly calculate this posterior probability. Therefore, using the Bayes' theorem, $P(f|\mathcal{D})$ can be computed as:

$$P(f|\mathcal{D}) = \frac{P(\mathcal{D}|f)P(f)}{P(\mathcal{D})}, \tag{2.6}$$

where $P(\mathcal{D}|f) \triangleq (\mathcal{D}|\mathcal{X} = f)$ is the likelihood probability; $P(f) \triangleq P(\mathcal{X} = f)$ is the prior probability of $\mathcal{X} = f$; and $P(\mathcal{D})$ is the marginal probability of $\mathcal{D}$:

$$P(\mathcal{D}) = \sum_{f \in \mathcal{F}} P(\mathcal{D}|f)P(f). \tag{2.7}$$

Since $P(\mathcal{D})$ normally does not affect the results, it is usually treated as a constant. In general, finding the MAP configuration for an MRF is NP hard [169].

2.1.3.2 Maximum Posterior Marginal Configuration

A labeling problem can also be interpreted as a search for the configuration f^* that maximizes the posterior marginal probabilities of each unobserved variable $x \in \mathcal{X}$ given the observed data $\mathcal{D}$. Such a configuration f^* is called the MPM configuration and defined as:

$$f^* \triangleq \{f_i^* | f_i^* = \arg\max_{f_i \in \mathcal{L}} P(f_i|\mathcal{D}),\ i = 1 \dots N\}, \tag{2.8}$$

where $P(f_i|\mathcal{D}) \triangleq P(x_i = f_i|\mathcal{D})$.

[1] In [194], the MPM configuration is termed as maximum marginal configuration. Here, we add the word *posterior* as in [97, 111] to avoid the ambiguity with the normal use of *marginal* probability.

2.1.4 MAP-MRF Labeling

Many digital image processing and analysis problems [12, 55, 177] have been formulated as the MAP estimation on MRFs. Finding the MAP solution to a labeling problem on an MRF is equivalent to finding the minimum-energy solution of the following objective/energy function[2]:

$$E(f) = E_{data}(f) + E_{smooth}(f), \tag{2.9}$$

This energy function is viewed as the log likelihood of the posterior distribution of an MRF. The data energy $E_{data}(f)$ and the smoothness energy $E_{smooth}(f)$ are derived from the negative log likelihoods of the likelihood $P(\mathcal{D}|f)$ and the prior $P(f)$, respectively. In the MAP-MRF model, the prior distribution is formulated as an MRF, and the likelihood is due to the observation model [111].

Based on the equivalence of MRFs and Gibbs distributions [63], $P(f|\mathcal{D})$ can be expressed as follows:

$$P(f|\mathcal{D}) = \beta P(f, \mathcal{D}) = \frac{\beta}{Z} \exp(-E(f)/T), \tag{2.10}$$

where β and Z are normalizing constants, and T is a constant called *temperature*. Assuming all probabilities are above zero, the Hammersley-Clifford theorem [16, 76] guarantees that the probability distribution will factorize into a product of functions of the maximal cliques $\mathcal{C}$ of the graph. Therefore, for a pairwise MRF model (*e.g.,* Figure 2.1) where a maximal clique consists of two connected nodes[3], the joint probability $P(f, \mathcal{D})$ can be factorized as a function defined on single-site cliques and pair-site cliques:

$$P(f, \mathcal{D}) = \frac{1}{Z} \prod_{\{x_i, x_j\} \in \mathcal{C}} \Psi_{ij}(f_i, f_j) \prod_{x_i \in \mathcal{X}} \Psi_{ii}(f_i), \tag{2.11}$$

where $\Psi_{ij}(f_i, f_j) \triangleq \Psi_{ij}(x_i = f_i, x_j = f_j)$ and $\Psi_{ii}(f_i) \triangleq \Psi_{ii}(x_i = f_i, d_i)$ are called *potential functions*. By substituting Equation (2.10) into Equation (2.11) and assuming $T = 1$, $E(f)$ can be expressed as:

$$E(f) = \sum_{x_i \in \mathcal{X}} D_i(f_i) + \sum_{\{x_i, x_j\} \in \mathcal{C}} V_{ij}(f_i, f_j), \tag{2.12}$$

where $D_i(f_i) = -\log(\Psi_{ii}(f_i))$ and $V_{ij}(f_i, f_j) = -\log(\Psi_{ij}(f_i, f_j))$. $E_{data}(f)$ and $E_{smooth}(f)$ are then represented as follows:

$$E_{data}(f) = \sum_{x_i \in \mathcal{X}} D_i(f_i). \tag{2.13}$$

$$E_{smooth}(f) = \sum_{\{x_i, x_j\} \in \mathcal{C}} V_{ij}(f_i, f_j). \tag{2.14}$$

This formulation of energies indicates that the interaction between the scene nodes $\mathcal{X}$ and the data nodes $\mathcal{D}$ constitutes the data energy E_{data}, while the interaction among the scene nodes $\mathcal{X}$ constitutes the smoothness energy E_{smooth}. For pixel labeling problems, $E_{data}(f)$ is the sum of a

[2] Another form of the energy function used in practice is one that adds a weighting coefficient λ to the second term due to the individual modeling of the two energy terms. However, this variant does not make any fundamental difference to our discussion here.

[3] In this thesis, we do not distinguish between a scene node and a variable for conciseness, *i.e.,* the i-th scene node and the i-th variable are both denoted by x_i, unless otherwise mentioned.

set of per-pixel data costs, and $E_{smooth}(f)$ is the sum of spatially varying horizontal and vertical nearest neighbor smoothness costs (if 4-neighbor connectivity is assumed) [179]. In [206], $D_i(f_i)$ and $V_{ij}(f_i, f_j)$ are referred to as compatibility functions that model the dependency between the observed data $d \in \mathcal{D}$ and the scene nodes $x \in \mathcal{X}$, and the resemblance of a scene node x_i with its neighboring nodes x_j's, respectively. For different applications, the compatibility functions can take different forms.

2.1.4.1 MAP Estimation on Gaussian MRFs

A special case of the MRF is the Gaussian MRF (GMRF) [160]. Let $\mathbf{x}$ be the vector form of $\mathcal{X}$, *i.e.,* $\mathbf{x} = (x_1, x_2, \ldots, x_N)^T$, where $\cdot^T$ represents the transpose operator. A GMRF is an MRF under the assumption that the random vector $\mathbf{x}$ follows a multivariate Gaussian distribution with the probability density function (PDF) defined as[4]:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{N/2}|\mathbf{\Sigma}|^{1/2}} \exp(-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)), \tag{2.15}$$

where μ is the mean vector; $\mathbf{\Sigma}$ is the covariance matrix, which is symmetric positive definite; the determinant of $\mathbf{\Sigma}$ is denoted by $|\mathbf{\Sigma}|$; and the inverse of $\mathbf{\Sigma}$, denoted by $\mathbf{\Sigma}^{-1}$, is called the precision matrix. To simplify the notation, we use $\mathbf{Q} \triangleq \mathbf{\Sigma}^{-1}$ to denote the precision matrix. Normally, $\mathbf{Q}$ is a sparse matrix. Then, Equation (2.15) can be rewritten as:

$$p(\mathbf{x}) = (2\pi)^{-N/2}|\mathbf{Q}|^{1/2} \exp(-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{Q}(\mathbf{x}-\mu)). \tag{2.16}$$

This density function can be further reformatted in the canonical form:

$$p(\mathbf{x}) \propto \exp(-\frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{b}^T\mathbf{x}), \tag{2.17}$$

where $\mathbf{b} = \mathbf{Q}\mu$. This form facilitates computation by not requiring us to explicitly compute the mean. Like the MAP-MRF model, the GMRF model only assumes that the prior distribution follows Gaussian, and there is no restriction on the likelihood distribution. Therefore, the MAP configuration is:

$$\mathbf{f}^* = \arg\min_{\mathbf{f}} E(\mathbf{f}) = \arg\min_{\mathbf{f}} E_{data}(\mathbf{f}) + \frac{1}{2}\mathbf{f}^T\mathbf{Q}\mathbf{f} - \mathbf{b}^T\mathbf{f}, \tag{2.18}$$

where $\mathbf{f}$ is the vector form of f. To simplify computation in practical cases, it is normally assumed (either explicitly or implicitly) that $\mu = \mathbf{0}$ (*e.g.,* [23, 178]), which results in a quadratic smoothness energy. An alternative construction assuming piecewise smooth or piecewise constant [160], which also results in a quadratic smoothness energy, does not impose this zero-mean assumption, and we will discuss this later in Section 2.4.4. If the likelihood is also modeled as a Gaussian distribution as in [178], the MAP estimation can be efficiently obtained via quadratic energy minimization.

[4]For a GMRF, the variables are normally supposed to take continuous values, *i.e.,* the label set $\mathcal{L}$ is a real interval.

2.1.5 Conditional Random Fields

A conditional random field (CRF) [97, 99] models the posterior probability $P(f|\mathcal{D})$ directly as an MRF without explicit modeling of the prior and likelihood probabilities[5]. While an MRF is defined on the variable set $\mathcal{X}$, a CRF is defined on both $\mathcal{X}$ and the observed data $\mathcal{D}$ [97]. MRF is a generative model and independent of the observed data, while CRF is a discriminative model and dependent of the observed data. $(\mathcal{X}, \mathcal{D})$ is called a CRF when $\mathcal{X}$ conditioned on $\mathcal{D}$ obeys the positivity and Markov properties. Therefore, the posterior probability $P(f|\mathcal{D})$ takes the following form for a pairwise CRF:

$$P(f|\mathcal{D}) \propto \exp(-\sum_{x_i \in \mathcal{X}} D_i(f_i|\mathcal{D}) - \sum_{\{x_i,x_j\} \in \mathcal{C}} V_{ij}(f_i, f_j|\mathcal{D})). \tag{2.19}$$

There are two major differences between the CRF model and the MRF model: 1) in the CRF model, the unary potential, which corresponds to the data compatibility in MRF, is defined on all the observed data $\mathcal{D}$ and x_i, while it is only defined on d_i and x_i in the MRF model; 2) in a CRF, the pairwise potential, which corresponds to the neighborhood compatibility in MRF, is dependent of the observed data $\mathcal{D}$, while it is independent on $\mathcal{D}$ in an MRF. The CRF model is useful when explicit modeling of the likelihood is intractable and/or when interaction between the scene nodes is data-dependent. Like a GMRF, a CRF $(\mathcal{X}, \mathcal{D})$ is called a Gaussian CRF (GCRF) when $\mathcal{X}$ conditioned on $\mathcal{D}$ follows a Gaussian distribution [181].

2.2 Belief Propagation

Belief propagation (BP) solves inference problems on belief networks (*e.g.,* MRFs and Bayesian networks), which encode relevances as neighboring nodes in a graph. Labeling problems can be solved using BP through local message passing, which at equilibrium produces a globally optimal (or near optimal) solution. A higher-level labeling problem (such as texture classification [183]) may need a formulation on Bayesian networks, while a lower-level labeling problem (such as image segmentation [63]) may prefer a formulation on MRFs. In this section, we focus on BP algorithms for the MRF formulation of the labeling problems. Please refer to Appendix A for more details on BP algorithms for Bayesian networks.

Two belief propagation algorithms were originally proposed by Pearl [140] for Bayesian networks based on message passing between neighboring nodes. The first algorithm is called *belief update*, a.k.a. sum-product belief propagation (BP-S). When iterative message updating reaches equilibrium in BP-S, an MPM solution is found. The second algorithm, which is for producing the MAP configuration, is called *belief revision*, a.k.a. max-product belief propagation (BP-M). The correctness of the two algorithms is guaranteed for singly connected networks.

[5] In [97], a special case of a CRF, called *discriminative random field*, is proposed, but here we treat them under the same term.

Weiss [194] extended BP-S and BP-M to MRFs and proved the correctness of the MAP configuration using BP-M in networks with a single loop, although the numerical values of the beliefs were shown to be incorrect. Though BP-M produces good approximations at convergence, it converges less frequently than BP-S. However, the correctness of the MPM configuration using BP-S in networks with single or multiple loops is not guaranteed. Weiss and Freeman [196] further extended the BP-M algorithm to graphs of arbitrary topologies. Any Bayesian network can be converted into an MRF by adding a cluster node for every node that has multiple parents in the original graph. An MRF with loops can be unwrapped into a computation tree. It has been proved that for an arbitrary graphical model with arbitrary topology, when BP-M converges it assigns the MAP configuration to all nodes within a large region, *i.e.,* the single loops and trees neighborhood of those nodes. However, ties may arise, *i.e.,* $b_i(f_i) = b_i(f_j)$, which results in the MAP configuration not being unique. In this case, extra criteria are needed to determine the most appropriate configuration. These criteria can be applied after convergence; therefore, the message updating procedure can remain the same. The message updating procedures of BP-S and BP-M for pairwise MRFs are described in the following sections. Without loss of generality, it is assumed that each variable $x_i \in \mathcal{X}$ has a data node $d_i \in \mathcal{D}$ (which is observed) associated with it.

2.2.1 Sum-Product Belief Propagation

The derivation of the relationship between beliefs and messages is not provided in [194, 196]. Here we first give a brief derivation based on the derivation for BP on Baysian networks [140] and then summarize the belief updating rules. In BP-S, a belief $b_i(f_i)$ represents the probability that $x_i = f_i$ given all the information $\mathcal{D}$:

$$b_i(f_i) \triangleq P(f_i|\mathcal{D}) \propto P(f_i, \mathcal{D}). \tag{2.20}$$

Using the Markov property and the Hammersley-Clifford theorem, $b_i(f_i)$ can be factorized into the product of a set of functions defined on neighboring nodes:

$$b_i(f_i) \propto P(f_i, \mathcal{D}_i) \propto \Psi_{ii}(f_i) \prod_{x_k \in \mathcal{N}_i} [\sum_{f_k \in \mathcal{L}} P(f_i, f_k, \mathcal{D}_{k\backslash i})], \tag{2.21}$$

where $\Psi_{ii}(f_i) \propto P(f_i, d_i)$; $\mathcal{D}_i$ denotes all the information received through node x_i; and $\mathcal{D}_{k\backslash i}$ denotes all the information received through node x_k except that from node x_i. A message $m_{ki}(f_i)$ represents the probability that $x_i = f_i$ given the information $\mathcal{D}_{k\backslash i}$:

$$m_{ki}(f_i) \triangleq P(f_i|\mathcal{D}_{k\backslash i}) \propto P(f_i, \mathcal{D}_{k\backslash i}) = \sum_{f_k \in \mathcal{L}} P(f_i, f_k, \mathcal{D}_{k\backslash i}). \tag{2.22}$$

Using the Markov property and the Hammersley-Clifford theorem, $m_{ki}(f_i)$ can be further factorized as:

$$\begin{aligned} m_{ki}(f_i) &\propto \sum_{f_k} [\Psi_{ki}(f_k, f_i) P(f_k, \mathcal{D}_{k\backslash i})] \\ &= \sum_{f_k} \{\Psi_{ki}(f_k, f_i) \Psi_{kk}(f_k) \prod_{x_j \in \mathcal{N}_k \backslash x_i} [\sum_{f_j \in \mathcal{L}} P(f_k, f_j, \mathcal{D}_{j\backslash k})]\}, \end{aligned} \tag{2.23}$$

where $\Psi_{ki}(f_k, f_i) \propto P(f_k, f_i)$. It is also possible to let $\Psi_{ii}(f_i)$ and $\Psi_{ki}(f_k, f_i)$ represent conditional probabilities (*e.g.,* [60, 206]) instead of the joint probabilities used here. This formulation normally makes an implicit assumption that $P(f_k|f_i) = P(f_i|f_k)$, $\forall\{x_i, x_k\} \in \mathcal{C}$ to simplify the computation. Using the formulation of $b_i(f_i)$ and $m_{ki}(f_i)$ from Equations (2.20) to (2.23), the posterior probabilities $P(f_i|\mathcal{D})$'s can be approximated through iterative local message passing. When this procedure reaches equilibrium (*i.e.,* $b_i(f_i)$'s make no more or negligibly little changes), $P(f_i|\mathcal{D})$'s can be considered as precise over a large neighborhood. Let $m_{ij}^t(f_j)$ denote the message sent from node x_i to node x_j at time t given $x_j = f_j$. The BP-S updating rules for pairwise MRFs are:

$$m_{ij}^t(f_j) \leftarrow \alpha \sum_{f_i \in \mathcal{L}} [\Psi_{ij}(f_i, f_j)\Psi_{ii}(f_i) \prod_{x_k \in \mathcal{N}_i \setminus x_j} m_{ki}^{t-1}(f_i)]. \tag{2.24}$$

$$b_i^t(f_i) \leftarrow \alpha \Psi_{ii}(f_i) \prod_{x_k \in \mathcal{N}_i} m_{ki}^{t-1}(f_i). \tag{2.25}$$

This procedure starts with all messages initialized as 1. α is a normalizing constant rendering $\sum_{f_i \in \mathcal{L}} b_i^t(f_i) = 1$. A belief vector $\mathbf{b}_i = (b_i(l_1), \ldots, b_i(l_K))^T$ is stored for each node x_i to represent the beliefs of that node taking different labels. Then, the MPM labeling problem expressed in Equation (2.8) is equivalent to finding the label with the highest belief for each variable:

$$f_i^* = \arg\max_{f_i \in \mathcal{L}} b_i(f_i). \tag{2.26}$$

2.2.2 Max-Product Belief Propagation

In BP-M, a belief $\hat{b}_i(f_i)$ represents the maximum value that the posterior probability $P(f|\mathcal{D})$ can take when $x_i = f_i$ is fixed. Unlike BP-S, in BP-M, the probabilities of the other unobserved nodes need to be maximized other than summed in order to get the MAP configuration. Therefore, the belief $\hat{b}_i(f_i)$ at node x_i is defined as:

$$\hat{b}_i(f_i) \triangleq \hat{P}(f_i|\mathcal{D}) \propto \max_{f \setminus f_i} P(f_i, f \setminus f_i, \mathcal{D}). \tag{2.27}$$

Then, the MAP labeling problem expressed in Equation (2.5) is equivalent to finding the label with the highest belief for each variable:

$$f_i^* = \arg\max_{f_i \in \mathcal{L}} \hat{b}_i(f_i). \tag{2.28}$$

Unlike BP-S, $\hat{b}_i(f_i)$ may not sum to unity over f_i in BP-M. A message $\hat{m}_{ki}(f_i)$ represents the maximum probability that $x_i = f_i$ given $\mathcal{D}_{k \setminus i}$:

$$\hat{m}_{ki}(f_i) \triangleq \hat{P}(f_i|\mathcal{D}_{k \setminus i}) \propto \max_{f_k \in \mathcal{L}} P(f_i, f_k, \mathcal{D}_{k \setminus i}). \tag{2.29}$$

With beliefs and messages defined in this way, the BP-M updating rules can be derived following the similar pattern used for BP-S:

$$\hat{m}_{ij}^t(f_j) \leftarrow \alpha \max_{f_i \in \mathcal{L}} [\Psi_{ij}(f_i, f_j)\Psi_{ii}(f_i) \prod_{x_k \in \mathcal{N}_i \setminus x_j} \hat{m}_{ki}^{t-1}(f_i)]. \tag{2.30}$$

$$\hat{b}_i^t(f_i) \leftarrow \alpha \Psi_{ii}(f_i) \prod_{x_k \in \mathcal{N}_i} \hat{m}_{ki}^{t-1}(f_i). \tag{2.31}$$

The initialization is the same as that of BP-S, and α is still a normalizing constant. The only difference is that the summation operator used in Equation (2.24) is replaced by the maximization operator.

In practice, the probabilities are often manipulated in the log domain; therefore if taking the negative log of the posterior probabilities, BP-M actually estimates the minimum-energy solution (MAP configuration) for Equation (2.12). BP-S does not directly search for a minimum-energy solution as BP-M does, but it computes the posterior marginal probability distribution for each node in the graph, which results in the MPM configuration.

2.2.3 Hierarchical Belief Propagation

To accelerate BP-M, an efficient hierarchical BP-M was proposed by Felzenszwalb and Huttenlocher [54, 55]. This algorithm works in the negative log domain of the probabilities, and the energy function used takes the same form as in Equation (2.12). In each iteration, the messages are updated as:

$$m_{ij}^t(f_j) \leftarrow \min_{f_i \in \mathcal{L}} [V_{ij}(f_i, f_j) + D_i(f_i) + \sum_{x_k \in \mathcal{N}_i \setminus x_j} m_{ki}^{t-1}(f_i)]. \tag{2.32}$$

All messages are initialized to 0. The updating of the beliefs in the iterative message-passing procedure is not necessary and the beliefs are only computed at the convergence of the algorithm after T iterations.

$$b_i(f_i) \leftarrow D_i(f_i) + \sum_{x_k \in \mathcal{N}_i} m_{ki}^T(f_i). \tag{2.33}$$

When the energy function in Equation (2.12) is defined following the Potts model [200], *i.e.,* the compatibility function $V_{ij}(f_i, f_j)$ has the following form:

$$V_{ij}(f_i, f_j) = \min(s|f_i - f_j|^q, \theta), \tag{2.34}$$

where $q \in \{1, 2\}$ and s is the rate of increase in the cost and θ is a positive thresholding constant, the computational efficiency of updating a single message between adjacent nodes is enhanced by reformatting the minimization problem as the computation of a lower envelope. The message $m_{ij}^t(f_j)$ can be computed as:

$$m_{ij}^t(f_j) = \min(\hat{h}(f_j), \min_{f_i} h(f_i) + \theta), \tag{2.35}$$

where $h(f_i) = D_i(f_i) + \sum_{x_k \in \mathcal{N}_i \setminus x_j} m_{ki}^{t-1}(f_i)$ and $\hat{h}(f_j)$ is computed on $h(f_i)$ following a two-pass process (one forward pass and one backward pass). This acceleration technique, which is also referred to as *distance transform*, reduces the complexity of message updating from $O(K^2)$ to $O(K)$. However, the limitation of the distance transform acceleration is that it is only applicable when the smoothness energy is of a particular form as given in Equation (2.34).

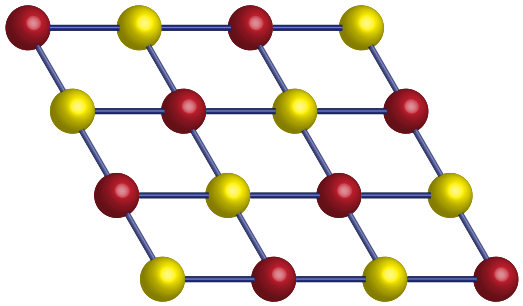


Figure 2.2: A lattice is a bipartite graph. Nodes with the same color belong to the same set.

For a grid (lattice) graph and any bipartite graph, it has been shown that the messages for each of the two sets of nodes can be updated alternately. Let $\mathcal{X}_1$ and $\mathcal{X}_2$ denote the two sets, with $\mathcal{X}_1 \cap \mathcal{X}_2 = \emptyset$ and $\mathcal{X}_1 \cup \mathcal{X}_2 = \mathcal{X}$. If $x_i \in \mathcal{X}_1$, then $\mathcal{N}_i \subseteq \mathcal{X}_2$. Following this rule, a lattice can be converted to a bipartite graph. As shown in Figure 2.2, nodes with the same color belong to the same set. At time t only messages sent to nodes in $\mathcal{X}_1$ from nodes in $\mathcal{X}_2$ are updated, while at time $t+1$ only messages from $\mathcal{X}_1$ to $\mathcal{X}_2$ are updated. This alternate message-updating scheme not only yields speedup but also reduces memory usage. In addition, the input images are organized in pyramidal structures, and the beliefs are propagated in a coarse-to-fine fashion. This structure accelerates the message passing over a long range. The message updating procedure runs for a constant number of iterations at each level, and the resulting messages are used to initialize the message updating procedure in the lower (next finer) level. With all the acceleration techniques applied, the overall computational complexity of BP-M is reduced to $O(NK)$ from $O(NK^2T)$.

2.3 Graph Cuts

The graph cuts (GC) method solves the min-cut/max-flow problem on a graph, a fundamental problem in combinatorial optimization [138]. More specifically, GC solves the problem of finding the maximum flow from a *source* to a *sink*, which is equivalent to finding the cut with the minimum capacity/cost. The structure of the graph used by GC for a two-label case is illustrated in Figure 2.3. Let $\mathcal{G}_{\alpha\beta} = (\mathcal{V}_{\alpha\beta}, \mathcal{E}_{\alpha\beta})$ denote this weighted graph. The structure of $\mathcal{G}_{\alpha\beta}$ is similar to the lattice graph shown in Figure 2.1 except that two terminals are added to $\mathcal{V}$ and their incident edges are added to $\mathcal{E}$. The terminals represent the set of labels $\mathcal{L}$, and the two terminals in this graph represent the two labels α and β, respectively. α can be considered as the source, and β as the sink. Each scene node x_i is connected to both the α and β terminals by edges t_i^α and t_i^β, respectively. Such edges are called *t-links* (terminal links). The edges e_{ij}'s connecting two scene nodes are called *n-links* (neighbor links). A *cut* $\mathbf{C}$ is defined as the set of edges without which all of the terminals are completely separated from each other. The cost of the cut $\mathbf{C}$, denoted by $|\mathbf{C}|$, equals the sum of its edge weights. If more than two terminals are present, the minimum cut problem is referred to as the *multiway cut* problem or *multiterminal cut* problem [42, 43]. A cut $\mathbf{C}$ is actually equivalent to a

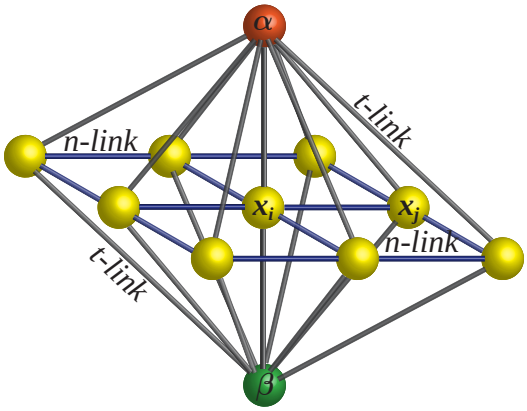


Figure 2.3: The structure of the graph used by GC. Yellow nodes represent unobserved variables; the orange and green nodes represent the labels α and β, respectively. Blue segments represent n-links and grey segments represent t-links.

labeling f^C with $|\mathbf{C}| = E(f^C) + \epsilon$, where ϵ is a constant [27]. Then, finding the minimum cut $\mathbf{C}^*$ is equivalent to finding the MAP configuration f^*.

Just like BP-M, GC can be considered as an MAP approximation algorithm for MRFs by minimizing the energy function in Equation (2.12). The weight of a t-link and the weight of an n-link corresponds to the data compatibility function $D_i(\cdot)$ and the neighborhood compatibility function $V_{ij}(\cdot,\cdot)$, respectively. As an MAP estimation technique, GC was first introduced into digital image processing and analysis by Greig *et al.* [74] for binary image restoration, in which only two labels are considered. In [74], the capacity of a cut was defined as a cost/energy function based on the Ising model, a special case of the Potts model. It was shown that given only two labels, the exact MAP configuration can be obtained using GC. For an energy function of a more general form as given in Equation (2.12), it was proved in [81] that the exact MAP configuration can be obtained via GC given that the neighborhood compatibility function $V(\cdot,\cdot)$, which defines the priors, is convex for a linearly ordered label set.

For an arbitrary finite set of labels $\mathcal{L}$, finding the minimum multiway cut is NP hard [42, 43]. However, Boykov *et al.* [26, 27] proposed two fast approximation algorithms. These algorithms iteratively execute large *moves* to obtain a strong local minimum at convergence when no further moves can produce a lower energy. A move refers to a change of the labels of a set of nodes. Two categories of the interaction penalty function (neighborhood compatibility function) V_{ij} are considered: metric and semimetric. V_{ij} is called a *metric* on $\mathcal{L}$ if for any $\alpha, \beta, \gamma \in \mathcal{L}$ it satisfies the following three conditions[6]:

$$V_{ij}(\alpha,\beta) = 0 \quad \Leftrightarrow \quad \alpha = \beta, \tag{2.36}$$

$$V_{ij}(\alpha,\beta) \quad = \quad V_{ij}(\alpha,\beta) \geq 0, \tag{2.37}$$

$$V_{ij}(\alpha,\beta) \quad \leq \quad V_{ij}(\alpha,\gamma) + V_{ij}(\gamma,\beta). \tag{2.38}$$

[6]The requirement of $V_{ij}(\alpha,\beta) = V_{ij}(\alpha,\beta)$ can be relaxed for directed graphs [27].

If V_{ij} only satisfies the first two conditions, it is called a *semimetric*. The two GC algorithms are known as the *swap* algorithm (GC-S), which is based on the α-β-*swap* moves, and the *expansion* algorithm (GC-E), which is based on the α-*expansion* moves. An α-β swap is a move in which nodes with label α or β can change their labels to the other one, and nodes with other labels remain the same. An α-expansion is a move in which any node can change its label to α and nodes already with label α remain the same. GC-S is applied to cases where V_{ij} is a semimetric, while GC-E is applied to cases where V_{ij} is a metric. These conditions are relaxed in [92] and the only requirement is that V_{ij} be submodular. Thus, the constraint for GC-S on binary labeling problems becomes:

$$V_{ij}(\alpha,\alpha)+V_{ij}(\beta,\beta)\leq V_{ij}(\alpha,\beta)+V_{ij}(\beta,\alpha),\forall\alpha,\beta\in\mathcal{L}. \tag{2.39}$$

And the constraint for multi-label problems and GC-E becomes:

$$V_{ij}(\alpha,\alpha)+V_{ij}(\beta,\gamma)\leq V_{ij}(\beta,\alpha)+V_{ij}(\alpha,\gamma),\forall\alpha,\beta,\gamma\in\mathcal{L}. \tag{2.40}$$

In a later review paper [89], Kolmogorov and Rother showed that GC may also produce satisfactory results when V_{ij} is modeled by non-submodular functions. Please note that the constraints discussed above are all on the neighborhood compatibility function $V_{ij}(\cdot,\cdot)$, and there is no constraint on the form of the data compatibility function $D_i(\cdot)$. In the following four sections, we briefly review the GC-S and GC-E algorithms.

2.3.1 Swap Algorithm

Let $\mathcal{P}_{\alpha\beta}$ denote a partition that consists of all the nodes with label α or β, *i.e.,* $\mathcal{P}_{\alpha\beta}=\{x_i|f_i=\alpha\}\cup\{x_i|f_i=\beta\}$. The α-β swap algorithm works on a graph constructed on $\mathcal{P}_{\alpha\beta}$ and on the two labels with a structure similar to the one shown in Figure 2.3. With an arbitrary initial configuration f^0, in each iteration t, GC-S finds the configuration $f^t=\arg\min_{f_{\alpha\beta}}E(f_{\alpha\beta})$ among all $f_{\alpha\beta}$'s that are within one α-β swap of f^{t-1}. If $E(f^T)=E(f^{T-1})$ after T iterations, the iteration terminates and the optimal configuration f^* is set to f^T. The structure of the graph changes dynamically after each iteration. The configuration f^t after each iteration is determined from the cut $\mathbf{C}^t$ as follows: if $t_i^\alpha\in\mathbf{C}^t$, then x_i is assigned label α; if $t_i^\beta\in\mathbf{C}^t$, then x_i is assigned label β; for any $x_i\notin\mathcal{P}_{\alpha\beta}^t$, its label remains unchanged.

2.3.2 Expansion Algorithm

The expansion algorithm works in a similar fashion to GC-S. It works on a graph constructed on $\mathcal{X}$, label α and label $\beta=\mathcal{L}-\{\alpha\}$. With an arbitrary initial configuration f^0, in each iteration t, the algorithm finds the configuration $f^t=\arg\min_{f_\alpha}E(f_\alpha)$ among all f_α's that are within one α expansion of f^{t-1}. If $E(f^T)=E(f^{T-1})$ after T iterations, the iteration terminates and the optimal configuration f^* is set to f^T. As in the case of GC-S, the structure of the graph changes dynamically after each iteration. The configuration f^t after each iteration is determined as follows: if $t_i^\alpha\in\mathbf{C}^t$, then x_i is assigned label α; for any other node, its label remains unchanged.

2.3.3 Max-Flow Computation

The max-flow algorithm proposed by Boykov and Kolmogorov [25] can be used to find the minimum cut in each iteration for GC-S and GC-E, and the minimum cut is actually a binary labeling. Two non-overlapping search trees S and T are built from the source s and the sink t, respectively. In S the direction of the flow is from a parent to its children, while in T the direction is reversed. The algorithm iteratively repeats three stages: growth stage, augmentation stage, and adoption stage. In the growth stage, S and T expand by acquiring neighboring free nodes, which are connected to the search trees via immediate edges. If an edge has residual capacity (*i.e.,* the edge is nonsaturated), the free node incident on that edge is added to the search tree as a child, whose parent is the other node incident on that edge. The growth stage terminates until S and T touch each other, in which case an $s \rightarrow t$ path is found. In the augmentation stage, the maximum flow allowed by the path $s \rightarrow t$ pushes through. The nodes incident on the saturated edges, including all the subtrees rooted in them, are removed from S and T. These removed nodes, not including their subtrees, are called orphans. This stage splits S and T into forests. At the adoption stage, a search in the neighborhood of each orphan is performed, in an effort to find it a new parent. A valid parent should belong to the same tree, which has its root at s or t, as the orphan, and its incident edge should have residual capacity. If this procedure fails, that orphan becomes a free node and all its children become orphans. The stage terminates when there is no orphan left. The whole algorithm terminates when both S and T cannot grow and they are separated by saturated edges. Then the minimum cut is obtained as:

$$\mathbf{C} = \begin{cases} \{(u,v) \in \mathcal{E} | u \in S, v \in T\}, & \text{no free nodes;} \\ \{(u,v) \in \mathcal{E} | u \in S, v \in \mathcal{V} - S\}, & \text{otherwise.} \end{cases} \tag{2.41}$$

When there are isolated free nodes, the costs of the cuts determined from $\{S, \mathcal{V} - S\}$ and $\{T, \mathcal{V} - T\}$ are equal. The worst-case complexity is $O(|\mathcal{E}|N^2|\mathbf{C}|)$, where $|\mathcal{E}|$ denotes the number of edges; N is the number of nodes; and $|\mathbf{C}|$ is the cost of the cut $\mathbf{C}$.

2.3.4 Multiway Cuts

For a multiway cut problem, both GC-S and GC-E compute the minimum cut by repeatedly solving the binary labeling problem [26, 27]. In each cycle of iterations, GC-S finds a minimum cut $\mathbf{C}_{\alpha\beta}$ for every distinct pair of labels (α, β). Therefore, $|\mathcal{L}|^2$ iterations are required in each cycle. A labeling f for the whole set of scene nodes is determined by combining all the minimum cuts. The algorithm terminates when the labeling obtained at the current iteration gives the same energy as in the previous iteration, *i.e.,* $E(f^T) = E(f^{T-1})$. The only difference in GC-E is that it finds a minimum cut $\mathbf{C}_\alpha$ for every label against all other labels in each cycle of iterations. Therefore, $|\mathcal{L}|$ iterations are required in each cycle.

2.4 Random Walks

A random walk (RW) on a graph is a finite Markov chain that is time-reversible [119]. Suppose a random walker is at node x_i of a graph $\mathcal{G}$ at time t, then its position on $\mathcal{G}$ at time $t+1$ depends solely on the probability distribution associated with x_i at time t. In addition, the overall probability of a random walker moving from x_i to x_j is the same as the probability of it moving from x_j to x_i. Early studies of RW mainly focused on such questions as whether a random walker returns to its origin with probability 1 and when the first return occurs (*e.g.,* [53]). RW was first introduced into digital image processing and analysis by Wechsler and Kidode [192] for texture discrimination. An image is divided into square patches, and the probability distribution of a random walker within a patch hitting its boundary is determined from simulation. These hitting distributions are used as features for the classification of different textures.

2.4.1 Random Walks for Labeling Problems: A Simple Example

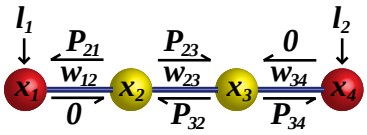


Figure 2.4: RW in a 1D case. Yellow nodes represent uninstantiated variables, while red nodes represent instantiated variables.

In this section, let us examine a simple example of applying RW to a binary labeling problem, as illustrated in Figure 2.4. In this one-dimensional (1D) case, we have four variables (represented as scene nodes), x_1 to x_4, and two labels, l_1 and l_2. Each variable is to take one and only one of the labels. The weight for an edge (blue segment) connecting two scene nodes x_i and x_j is denoted by w_{ij}. w_{ij} is usually determined based on the observed data associated with x_i and x_j. Let us assume x_1 and x_4 are pre-assigned the labels l_1 and l_2, respectively. Now, the problem is to determine the most probable configuration for x_2 and x_3 given $x_1 = l_1$, $x_4 = l_2$, and some observed data $\mathcal{D}$. The solution may be considered analogous to the MPM configuration of the labeling problem formulated in the MRF model; therefore, we denote this configuration as MPM*. This (discrete) labeling problem can be formulated in RW as: which labeled node is a random walker starting from an unlabeled node most likely to visit first?

To solve this problem, let us start by defining some concepts. The degree of a node x_i is defined as:

$$d_i \triangleq \sum_{x_k \in \mathcal{N}_i} w_{ik}. \tag{2.42}$$

The transition probability P_{ij} is defined as the probability that a random walker at node x_i moves to node x_j in a single move, which is determined by the weights of the edges incident on x_i:

$$P_{ij} = \frac{w_{ij}}{\sum_{x_k \in \mathcal{N}_i} w_{ik}} = \frac{w_{ij}}{d_i}, \tag{2.43}$$

where $\mathcal{N}_i$ denotes the immediate neighborhood of x_i. Since the transition probabilities of leaving node x_1 or x_4 are both 0, a random walker that arrives at either of them will conclude its walk. Therefore, nodes like x_1 and x_4 are called *boundary nodes*, *absorbing barriers*, or *traps*. Let $\mathbf{S}_{i,k}$ denote the set of all the possible paths that the random walker starting from node x_i can take to reach a node with label l_k before reaching a node with a different label. For the graph in Figure 2.4, $\mathbf{S}_{2,1} = \{\{x_2, \{x_3, x_2\}^n, x_1\} : n = 0, \ldots, +\infty\}$. Then, the steady-state probability of a random walker starting at x_2 and reaching the node x_1 with label l_1 (*i.e.,* the probability that the random walker has attempted any of the paths in $\mathbf{S}_{2,1}$), is $P_{2,1} \triangleq P(x_2 = l_1 | x_1 = l_1, x_4 = l_2, \mathcal{D}) = P_{21} \sum_{n=0}^{+\infty} (P_{23}P_{32})^n = P_{21}/(1 - P_{23}P_{32})$. Similarly, $P_{2,2}, P_{3,1}, P_{3,2}$ can be calculated. After that, the configuration with the highest probability is selected, *i.e.,* $f^* = \{f_i^* | f_i^* = \arg\max_{l_k : k=1,2} P_{i,k}, i = 2, 3\}$. For this simple graph, it is possible to calculate the probabilities this way. However, for a more complicated graph, such as a large finite lattice (which is usually used to represent an image), this brute-force approach will not be computationally tractable. In the following section, we will review an effective solution from the perspective of the Dirichlet problem.

2.4.2 Random Walks as Dirichlet Problem

It was established in [48] that the probability that a random walker starting from an unlabeled node first reaches a labeled node is equivalent to the solution of the Dirichlet problem with boundary conditions at the labeled nodes. Let $\mathcal{X}_L \subset \mathcal{X}$ denote the set of boundary nodes (instantiated variables) and $\mathcal{X}_U = \mathcal{X} - \mathcal{X}_L$ the set of interior nodes (uninstantiated variables). A function $g(\cdot)$ is called *harmonic* on $\mathcal{X}_U$ if for any node $x_i \in \mathcal{X}_U$ it satisfies:

$$g(x_i) = \frac{1}{d_i} \sum_{x_k \in \mathcal{N}_i} w_{ik} g(x_k). \tag{2.44}$$

Equation (2.44) means that the value of $g(\cdot)$ at any unlabeled node is the weighted average of $g(\cdot)$ at its neighboring nodes. The problem of finding a harmonic function given its boundary values (conditions) is called the *Dirichlet problem*. Given the same boundary values, the harmonic function defined on the same set of interior nodes is unique. This is called the *uniqueness principle*. A harmonic function always attains its maximum and minimum on the boundary. This is called the *maximum principle*. Using Equation (2.43) and the law of total probability, it is obvious that a probability mass function (PMF) $P_k(x_i) \triangleq P(x_i = l_k | \mathcal{D})$, which represents the steady-state probability, is a harmonic function. In addition, $\sum_{l_k \in \mathcal{L}} P_k(x_i) = 1$ for any $x_i \in \mathcal{X}$.

Based on this connection, Grady and Funka-Lea [71, 69] proposed an effective RW model for finding the MPM* configuration given labeled nodes and observed data, and they applied the model to image segmentation. Let the Dirichlet integral for a region $\mathcal{X}$ and a field g defined on $\mathcal{X}$ be formulated as:

$$D[g] = \frac{1}{2} \int_{\mathcal{X}} |\nabla g(x)|^2 dx, \tag{2.45}$$

where ∇ denotes the gradient operator. Following this definition, a harmonic function $g(\cdot)$ is a function that satisfies the Laplace equation:

$$\nabla^2 g(x) = 0. \tag{2.46}$$

A harmonic function satisfying its boundary conditions minimizes the Dirichlet integral in Equation (2.45). For an image represented as a graph $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, the harmonic function $g(\cdot)$ and the Dirichlet integral $D[\cdot]$ become discrete. Let each entry $\mathbf{L}_{ij}$ of the combinatorial Laplacian matrix $\mathbf{L}$ be defined as:

$$\mathbf{L}_{ij} = \begin{cases} d_i, & i = j; \\ -w_{ij}, & x_i \text{ and } x_j \text{ are adjacent nodes;} \\ 0, & \text{otherwise.} \end{cases} \tag{2.47}$$

Let $\mathbf{g} = (g(x_1), \ldots, g(x_N))^T$, the combinatorial Dirichlet integral can then be formulated as:

$$D[g] = \frac{1}{2}\mathbf{g}^T\mathbf{L}\mathbf{g} = \frac{1}{2}\sum_{e_{ij}\in\mathcal{E}} w_{ij}(g(x_i) - g(x_j))^2. \tag{2.48}$$

For a labeling problem, $P_k(\cdot)$ is substituted for $g(\cdot)$.

The nodes in the graph $\mathcal{G}$ can be divided into two disjoint sets: $\mathcal{X}_L$, which is composed of all the labeled nodes (boundary nodes), and $\mathcal{X}_U$, which is composed of all the unlabeled nodes (interior nodes). Then the Laplacian matrix $\mathbf{L}$ can be arranged in such a way that the submatrix at the upper left corner relies solely on the labeled nodes and the submatrix at the lower right corner relies solely on the unlabeled nodes, as given in the following equation:

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{LL} & \mathbf{L}_{LU} \\ \mathbf{L}_{UL} & \mathbf{L}_{UU} \end{pmatrix}. \tag{2.49}$$

Since $\mathbf{L}$ is symmetric, $\mathbf{L}_{LU}$ is just the transpose of $\mathbf{L}_{UL}$, *i.e.,* $\mathbf{L}_{LU} = {\mathbf{L}_{UL}}^T$. $\mathbf{L}_{LL}$ represents the interaction within $\mathcal{X}_L$, $\mathbf{L}_{UU}$ the interaction within $\mathcal{X}_U$, and $\mathbf{L}_{LU}$ the interaction between $\mathcal{X}_L$ and $\mathcal{X}_U$. Then the Dirichlet integral in Equation (2.48) can be rewritten with respect to l_k as:

$$D[P_k] = \begin{pmatrix} \mathbf{P}_{L,k} \\ \mathbf{P}_{U,k} \end{pmatrix}^T \begin{pmatrix} \mathbf{L}_{LL} & {\mathbf{L}_{UL}}^T \\ \mathbf{L}_{UL} & \mathbf{L}_{UU} \end{pmatrix} \begin{pmatrix} \mathbf{P}_{L,k} \\ \mathbf{P}_{U,k} \end{pmatrix}, \tag{2.50}$$

where $\mathbf{P}_{L,k}$ and $\mathbf{P}_{U,k}$ are column vectors representing $P_{L,k}(\cdot)$ and $P_{U,k}(\cdot)$, the PMFs defined on $\mathcal{X}_L$ and $\mathcal{X}_U$, respectively. For a given label l_k, the value of $\mathbf{P}_{L,k}$ is known, and $P_{L,k}(\cdot)$ is a binary function:

$$P_{L,k}(x) = \begin{cases} 1, & x = l_k; \\ 0, & x \neq l_k. \end{cases} \tag{2.51}$$

Therefore, $\mathbf{P}_{U,k}$ can be calculated by first differentiating $D[P_k]$ with respect to $\mathbf{P}_{U,k}$ and then finding the critical points:

$$\mathbf{L}_{UU}\mathbf{P}_{U,k} = -\mathbf{L}_{UL}\mathbf{P}_{L,k}. \tag{2.52}$$

The above formulation requires some of the variables to be instantiated. This requirement is relaxed in [68] by incorporating label priors and approaching the problem from a Bayesian perspective,

which results in a similar linear system to Equation (2.52). (Please refer to Appendix B for more details.)[7]

To solve this large sparse linear system in Equation (2.52), many efficient methods are available: direct solvers, such as CHOLMOD[8] [38], which is based on supernodal Cholesky factorization [44]; and iterative solvers, such as conjugate gradient [161], algebraic multigrid [50, 204], and Gaussian belief propagation [18, 195]. Among these algorithms, the supernodal Cholesky factorization and the algebraic multigrid can achieve near-optimal performance, *i.e.,* $O(|\mathcal{X}_U|)$, where $|\cdot|$ denotes cardinality.

An algorithm closely related to RW is diffusion maps [51], but it focuses on computing the t-step transition probabilities instead of the steady-state probabilities. Another type of methods, called spectral clustering, may also involve the construction of the Laplacian matrix of a graph. An RW interpretation of one representative spectral clustering algorithm, the normalized cuts (NCut) algorithm [168], was given in [128]. Four major differences between this RW interpretation and the RW model discussed in this thesis are: 1) in the RW interpretation, the goal of NCut is to minimize the sum of the clusters' escaping probabilities (the probabilities that random walkers leave a cluster), but the goal of the RW model is to compute the steady-state probabilities that random walkers starting from interior nodes reach boundary nodes/absorbing barriers (nodes with escaping probabilities equivalent to zero); 2) the minimization of the objective function in NCut is NP hard, but the exact minimum solution of the objective function in the RW model can be computed in linear time; 3) the graph in the RW interpretation contains no absorbing barriers, but the graph in the RW model contains them, which form the boundary conditions; 4) in the RW interpretation, the eigenvectors of a transformed Laplacian matrix are computed and their elements are grouped into clusters, which correspond to clusters of the nodes in the graph, based on a criterion of piecewise constancy, but in the RW model, the minimum solution (*i.e.,* the steady-state probabilities) to an energy function defined using a Laplacian matrix is computed.

2.4.3 Connection with Electrical Networks

The relationship between the RW model discussed in the previous section and electrical networks was closely studied in [48][9]. Given a graph $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, the underlying electrical network is the network obtained by replacing the edges with electrical resistors [48]. Let r_{ij} denote the resistance associated with edge $e_{ij} \in \mathcal{E}$. Then the conductance of e_{ij} is $w_{ij} = 1/r_{ij}$.

According to Ohm's law, the current/flow i_{ij} from node x_i to node x_j through a resistor e_{ij} is determined by the voltages/potentials at x_i and x_j and the resistance r_{ij}:

$$i_{ij} = \frac{u_i - u_j}{r_{ij}} = w_{ij}(u_i - u_j), \tag{2.53}$$

[7] Please note that the derivation in [68] is different from the Bayesian inference on MRFs.
[8] CHOLMOD has been included in all recent versions of Matlab, beginning with 7.2.
[9] This connection was also briefly discussed in [69].

where u_i denotes the voltage at x_i. If connecting a battery voltage U to node $x_a \in \mathcal{X}$ and connecting node $x_b \in \mathcal{X}$ to ground, *i.e.,* $u_a = U, u_b = 0$, Kirchhoff's current law requires that the sum of currents flowing into any other $x_i \in \mathcal{X}$ is 0:

$$\sum_{x_j \in \mathcal{N}_i} i_{ij} = \sum_{x_j \in \mathcal{N}_i} w_{ij}(u_i - u_j) = 0. \tag{2.54}$$

Equation (2.54) is equivalent to the following equation:

$$u_i = \frac{1}{d_i} \sum_{x_j \in \mathcal{N}_i} w_{ij} u_j. \tag{2.55}$$

Just like the PMF $P_k(\cdot)$ in Section 2.4.2, the voltage u_i is harmonic at any node except x_a and x_b.

For a resistor e_{ij}, the *energy dissipation* is calculated as:

$$E_{ij} = i_{ij}^2 r_{ij} = w_{ij}(u_i - u_j)^2. \tag{2.56}$$

Therefore, for an electrical network $\mathcal{G}$, the total energy dissipation is:

$$E = \frac{1}{2} \sum_{e_{ij} \in \mathcal{E}} E_{ij} = \frac{1}{2} \sum_{e_{ij} \in \mathcal{E}} w_{ij}(u_i - u_j)^2. \tag{2.57}$$

The factor $1/2$ is used to remove the energy caused by duplicated counting of edges (*i.e.,* $e_{ij} = e_{ji}$). The minimization of Equation (2.57) is achieved when u_i satisfying Equation (2.55) for each node x_i is found. When $U = 1$, the obtained u_i's are equivalent to $\mathbf{P}_{U,k}$ in Section 2.4.2 with $\mathcal{X}_L = \{x_a, x_b\}$.

2.4.4 Connection with Gaussian Random Fields

The RW model can be interpreted from the perspective of the Gaussian random fields (GaRFs) [160]. Given a set of continuous variables $\mathcal{X}$, let $\mathbf{x} = (x_1, \ldots, x_N)^T$ be the vector form of $\mathcal{X}$ and let $x_i \in \mathcal{X}$ represent a certain quantity. For the ease of exposition, let us call this quantity a *potential* and $\mathcal{X}$ a *potential field*. $\mathcal{X}$ and its internal interactions are represented as a graph $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, where only interacting variable pairs have edges connecting them. With the assumption that $\mathcal{X}$ is piecewise smooth/constant, the local increment follows a Gaussian distribution with zero mean:

$$\Delta_{ij} \triangleq x_i - x_j \sim N(0, \frac{1}{\kappa w_{ij}}), \forall e_{ij} \in \mathcal{E}, \tag{2.58}$$

where w_{ij} is a positive and symmetric weight ($w_{ij} = w_{ji}$) assigned to the edge e_{ij} connecting x_i and x_j; and κ is a constant. Now the joint probability density expressed in Equation (2.16) becomes:

$$\begin{aligned} p(\mathbf{x}) &\propto \kappa^{(N-1)/2} \exp[-\frac{\kappa}{2}(\frac{1}{2} \sum_{e_{ij} \in \mathcal{E}} w_{ij}(\Delta_{ij})^2)] \\ &= \kappa^{(N-1)/2} \exp[-\frac{\kappa}{2}(\frac{1}{2} \sum_{e_{ij} \in \mathcal{E}} w_{ij}(x_i - x_j)^2)] \\ &= \kappa^{(N-1)/2} \exp(-\frac{1}{2}\mathbf{x}^T \mathbf{Q} \mathbf{x}), \end{aligned} \tag{2.59}$$

where each entry $\mathbf{Q}_{ij}$ of the precision matrix $\mathbf{Q}$ is:

$$\mathbf{Q}_{ij} = \kappa \begin{cases} d_i, & i = j; \\ -w_{ij}, & x_i \text{ and } x_j \text{ are adjacent nodes;} \\ 0, & \text{otherwise} \end{cases} \tag{2.60}$$

where $d_i = \sum_{x_j \in \mathcal{N}_i} w_{ij}$ is the degree of node x_i. From this precision matrix, it can be derived that a variable x_i has the following Gaussian distribution:

$$x_i \sim N(\frac{\sum_{x_j \in \mathcal{N}_i} w_{ij} x_j}{d_i}, \frac{1}{\kappa d_i}). \tag{2.61}$$

Note the resemblance of the precision matrix to the Laplacian matrix $\mathbf{L}$ defined in Equation (2.47). When $\kappa = 1$, $\mathbf{Q}$ is equivalent to $\mathbf{L}$. If some of the variables are instantiated/observed, we can estimate the posterior distribution given these observations. Let $\mathbf{x}_L$ and $\mathbf{x}_U$ denote the observed and unobserved variables, respectively. Then, the MAP estimation/configuration $\mathbf{f}_U^*$ is obtained via quadratic energy minimization:

$$\begin{aligned} \mathbf{f}_U^* &= \arg\min_{\mathbf{f}_U} E_{data}(\mathbf{f}_U) + E_{smooth}(\mathbf{f}_U) \\ &= \arg\min_{\mathbf{f}_U} {\mathbf{f}_U}^T \mathbf{Q}_{UL} \mathbf{f}_L + \frac{1}{2} {\mathbf{f}_U}^T \mathbf{Q}_{UU} \mathbf{f}_U, \end{aligned} \tag{2.62}$$

where $\mathbf{Q}_{UL}$ and $\mathbf{Q}_{UU}$ are equivalent to their corresponding submatrices in $\mathbf{L}$ in Equation (2.49) except the parameter κ. Minimizing the above energy is equivalent to solving the following linear system:

$$\mathbf{Q}_{UU} \mathbf{f}_U^* = -\mathbf{Q}_{UL} \mathbf{f}_L. \tag{2.63}$$

If the edge weights w_{ij}'s are data-independent, then $\mathcal{X}$ is a GMRF[10]; otherwise, $(\mathcal{X}, \mathcal{D})$ is a GCRF. In [181], a GCRF was used for grayscale image restoration, where a globally constant mean for all variables was assumed. This is a much more strict constraint than the one mentioned here (Equation (2.58)), which only imposes local constancy/smoothness. In [212], a GaRF, which is actually a GCRF with $\kappa = 1$, was employed for semi-supervised learning. Though not explicitly mentioned, local smoothness was assumed. The implicit local smoothness constraint, together with $\kappa = 1$, makes the labeling framework in [212] equivalent to the RW model discussed in Section 2.4.2, except that alternative further processing schemes on the obtained steady-state probabilities were proposed in [212].

Despite the resemblance of the equations in this section to those in Section 2.4.2, there are major differences between this GaRF interpretation and the RW model. First, the MAP-GaRF labeling problem is a continuous labeling problem, while the labeling problem that RW deals with is discrete in nature. Second, in this GaRF interpretation, the variables are not steady-state probabilities (though they can be considered as so if the instantiated variables are binary-valued, *i.e.,* either 0 or 1), but some locally smooth potentials. Third, in the MAP-GaRF labeling problem, the estimated potential field is the configuration, while the estimated probabilities in RW are further processed to get the configuration.

[10] Strictly speaking, it should be called an *intrinsic* GMRF, because the precision matrix $\mathbf{Q}$ is not of full rank [160].

2.5 Discussion

The methods discussed in this chapter were initially proposed for different purposes: MRF (or CRF) for modeling inference problems, BP for solving inference problems formulated in belief networks (*e.g.,* MRFs), GC for max-flow/min-cut optimization, and RW for modeling stochastic behaviors. However, they can be and have been widely applied to solving labeling problems. When solving (discrete) labeling problems, the problem formulation (*e.g.,* MRF, RW) and the solution derivation (*e.g.,* GC, BP, linear system solvers) are both important[11].

For the MAP-MRF model, BP-M and GC solve the formulated energy minimization problem. GC estimates the MAP solution by solving the min-cut/max-flow problem. No numerical values that indicate partial belongingness of each site to different labels are explicitly computed[12]. BP-M estimates the MAP solution through local message passing. When applied to MRFs of loops, BP-M provides good approximation at convergence, although it converges less often than BP-S and produces the incorrect numerical values of the beliefs [194]. However, BP-M and GC can only efficiently minimize Potts model-like energy functions [179]. As pointed out in [27], an energy minimization algorithm may fail to solve a problem due to either a poorly defined energy function or a poor approximation of the global minimum. These two causes are difficult to distinguish, except in some special cases where the true global minima can be computed efficiently. For binary image restoration, the exact MAP configuration was computed using GC in [74]. However, due to the simple construction of the energy function following the Ising model, the quality of the restored images was poorer than that of those restored using the iterated conditional modes [17], a local greedy energy minimization method. The performance of BP-M, GC and some other energy minimization algorithms were compared in [179]. When these algorithms were evaluated using the Middlebury stereo benchmarks [162], it was revealed that BP-M and GC produce energies lower than the ground-truth energies. This suggests the limitation of the Potts model-like energy function (Equation (2.34)) in addressing the stereo correspondence problem.

Like BP-M, BP-S can also work on MRFs, but it does so in order to approximate the MPM configuration. Although BP-M has been largely applied to stereo disparity estimation, it is shown in [180] that BP-S outperforms BP-M at sub-pixel accuracy. However, the distance transform acceleration, which provides the most speedup in hierarchical BP-M [54, 55], is not applicable to BP-S; and the correctness of BP-S cannot be guaranteed for loopy networks [194].

RW converts a discrete labeling problem into a continuous quadratic energy minimization problem. Compared to the MRF modeling scheme for discrete labeling problems, the main advantage

[11] When solving a particular labeling problem, the distance metric definition also plays an important role. However, since our discussion in this chapter focuses on general frameworks, we will postpone the distance metric factor to the next two chapters where we discuss the multi-exposure fusion problem.

[12] A continuous max-flow algorithm for foreground/background segmentation was proposed in [10], where the potentials at each site can be considered as partial belongingness; however, this algorithm is for sites in a continuous space instead of the discrete space discussed in this thesis, and it only applies to binary labeling problems instead of the general multi-label problems discussed in this thesis.

of RW is the transformation of discrete problems to the continuous convex domain, where they can be solved efficiently and accurately. As analyzed in [69], a foreground/background segmentation algorithm using the RW energy with MPM* configuration performs better on weak edges than using the Potts model energy with MAP configuration[13]. In this thesis, we mainly focus on developing, applying, and validating a set of RW-based frameworks for discrete labeling problems.

[13]Some attempts [41, 172, 171] have been made to unify the Potts model energy used in [24], where GC was applied to obtain the exact minimum energy solution, and the RW energy used in [69], where a direct linear system solver was applied to obtain the exact minimum energy solution, for the background/foreground image segmentation problem. However, energy form-dependent solvers are still required.

Chapter 3

The Generalized Random Walks Framework and Its Applications

3.1 Generalized Random Walks

In order to provide a general framework for labeling problems in digital image processing and analysis, we propose the generalized random walks (GRW) framework based on the RW model [68, 69] (see Section 2.4.2) and the relationship between RW and electrical networks [48] (see Section 2.4.3).

3.1.1 Image Representation

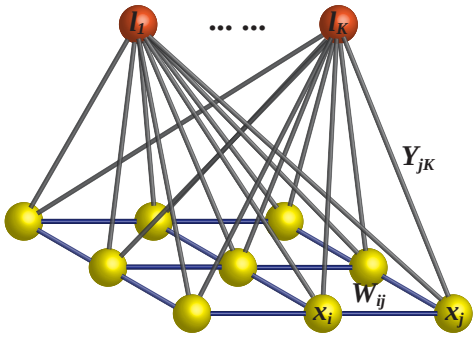


Figure 3.1: The graph used in GRW. The yellow nodes are scene nodes and the orange nodes are label nodes.

As shown in Figure 3.1, the variable set $\mathcal{X}$ and the label set $\mathcal{L}$ are represented in a weighted undirected graph similar to [68]. Each variable is associated with a pixel location in our case, and each label is associated with a certain quantity. The graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is constructed as $\mathcal{V} = \mathcal{L} \cup \mathcal{X}$ and $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ including edges both within $\mathcal{X}$ and between $\mathcal{X}$ and $\mathcal{L}$. The yellow nodes are scene nodes and the orange nodes are label nodes. For a scene node $x \in \mathcal{X}$, edges $\mathcal{E}_{\mathcal{X}}$ are drawn between it and each of its immediate neighbors in $\mathcal{X}$ (4-connectivity is assumed in our case). In addition, edges $\mathcal{E}_{\mathcal{L}}$ are drawn between a scene node and each label node. $W_{ij} \triangleq W(x_i, x_j)$ is a function, $W : \mathcal{E}_{\mathcal{X}} \rightarrow \mathbb{R}_{\geq 0}$, which models the compatibility/similarity between nodes x_i and x_j;

$Y_{ik} \triangleq Y(x_i, l_k)$ is a function, $Y : \mathcal{E}_{\mathcal{L}} \to \mathbb{R}_{\geq 0}$, which models the compatibility between x_i and l_k[1].

3.1.2 Dirichlet Problem

Let $\mathcal{V}$ be arranged in a way that the first K nodes are label nodes, *i.e.,* $\{v_1, \ldots, v_K\} = \mathcal{L}$, and the rest N nodes are scene nodes, *i.e.,* $\{v_{K+1}, \ldots, v_{K+N}\} = \mathcal{X}$. With two positive coefficients γ_1 and γ_2 introduced to balance the weights between $Y(\cdot, \cdot)$ and $W(\cdot, \cdot)$[2], we can define a node compatibility function $C(\cdot, \cdot)$ on $\mathcal{E}$ with the following form:

$$C_{ij} \triangleq C(v_i, v_j) = \begin{cases} \gamma_1 Y_{i-K,j}, & (v_i, v_j) \in \mathcal{E}_{\mathcal{L}} \wedge v_j \in \mathcal{L}; \\ \gamma_2 W_{i-K,j-K}, & (v_i, v_j) \in \mathcal{E}_{\mathcal{X}}. \end{cases} \tag{3.1}$$

Because the graph is undirected, we have $C_{ij} = C_{ji}$. Let $U(v_i)$ denote the potential associated with v_i. Based on the description in Section 2.4.3, the total energy of the system given in Figure 3.1 is:

$$E = \frac{1}{2} \sum_{(v_i, v_j) \in \mathcal{E}} C_{ij} (U(v_i) - U(v_j))^2. \tag{3.2}$$

Our goal is to find a function $U(\cdot)$ defined on $\mathcal{X}$ that minimizes this quadratic energy with boundary values $U(\cdot)$ defined on $\mathcal{L}$. If $U(\cdot)$ satisfies $\nabla^2 U = 0$, then it is called harmonic, and the harmonic function is guaranteed to minimize such quadratic energy E [69]. The problem of finding this harmonic function is called the Dirichlet problem.

The harmonic function $U(\cdot)$ can be computed efficiently using matrix operations. As in [69], a Laplacian matrix $\mathbf{L}$ can be constructed following Equation (3.3); however, in contrast with the procedure in [69], $\mathbf{L}$ here contains both the label nodes and the scene nodes and becomes a $(K + N) \times (K + N)$ matrix:

$$\mathbf{L}_{ij} = \begin{cases} D_i, & i = j; \\ -C_{ij}, & (v_i, v_j) \in \mathcal{E}; \\ 0, & \text{otherwise} \end{cases} \tag{3.3}$$

where $D_i = \sum_{v_j \in \mathcal{N}_i} C_{ij}$ is the degree of the node v_i defined on its immediate neighborhood $\mathcal{N}_i$. Without loss of generality, we assume $D_i > 0, \forall v_i$. Then, Equation (3.2) can be rewritten in matrix form as:

$$\begin{aligned} E &= \begin{pmatrix} \mathbf{U}_{\mathcal{L}} \\ \mathbf{U}_{\mathcal{X}} \end{pmatrix}^T \mathbf{L} \begin{pmatrix} \mathbf{U}_{\mathcal{L}} \\ \mathbf{U}_{\mathcal{X}} \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{U}_{\mathcal{L}} \\ \mathbf{U}_{\mathcal{X}} \end{pmatrix}^T \begin{pmatrix} \mathbf{L}_{\mathcal{L}\mathcal{L}} & \mathbf{L}_{\mathcal{L}\mathcal{X}} \\ \mathbf{L}_{\mathcal{X}\mathcal{L}} & \mathbf{L}_{\mathcal{X}\mathcal{X}} \end{pmatrix} \begin{pmatrix} \mathbf{U}_{\mathcal{L}} \\ \mathbf{U}_{\mathcal{X}} \end{pmatrix} \end{aligned} \tag{3.4}$$

where $\mathbf{U}_{\mathcal{L}} = (U(l_1), \ldots, U(l_K))^T$ and $\mathbf{U}_{\mathcal{X}} = (U(x_1), \ldots, U(x_N))^T$; $\mathbf{L}_{\mathcal{L}\mathcal{L}}$ is the upper left $K \times K$ submatrix of $\mathbf{L}$ that encodes the interactions within $\mathcal{L}$; $\mathbf{L}_{\mathcal{X}\mathcal{X}}$ is the lower right $N \times N$ submatrix that encodes the interactions within $\mathcal{X}$; and $\mathbf{L}_{\mathcal{L}\mathcal{X}} = {\mathbf{L}_{\mathcal{X}\mathcal{L}}}^T$ is the upper right $K \times N$ submatrix that

[1] Here, we have borrowed the term *compatibility function* from the MRF/CRF model; however, one should note that the compatibility functions in the MRF/CRF model contain the variables, but here they do not.

[2] In Section 3.4.2, when we introduce the multivariate Gaussian conditional random fields, we will reinterpret the meanings of these parameters.

encodes the interactions between $\mathcal{L}$ and $\mathcal{X}$. Hence, the minimum energy solution can be obtained by setting $\nabla E = 0$ with respect to $\mathbf{u}_{\mathcal{X}}$, *i.e.,* solving the following equation:

$$\mathbf{L}_{\mathcal{X}\mathcal{X}}\mathbf{U}_{\mathcal{X}} = -\mathbf{L}_{\mathcal{X}\mathcal{L}}\mathbf{U}_{\mathcal{L}}. \tag{3.5}$$

In some cases, part of $\mathcal{X}$ may be already labeled. These pre-labeled nodes can also be represented naturally in the current framework without altering the structure of the graph. Suppose x_i is one of the pre-labeled nodes and is assigned label l_k. Then, we can simply assign a sufficiently large value to Y_{ik} and solve the same Equation (3.5) for the unlabeled scene nodes.

3.1.3 Probability Calculation

Let $P_k(v_i) \triangleq P(v_i = l_k|\mathcal{D})$ denote the probability that a node v_i is assigned the kth label l_k given all the observed data $\mathcal{D}$. The probability $P_{i,k} \triangleq P_k(x_i)$ can be considered as the probability that a random walker starting at a scene node $x_i \in \mathcal{X}$ first reaches the label node $l_k \in \mathcal{L}$ on the graph $\mathcal{G}$. Let $P^t(l_k|x_i, \mathcal{D})$ denote the t-step transition probability determined based on observed data $\mathcal{D}$ that a random walker starting from location x_i reaches the label node l_k by the t-th step. The goal is to calculate the steady-state probability $P_{i,k} = \lim_{t\to\infty} P^t(l_k|x_i, \mathcal{D})$ for all $x_i \in \mathcal{X}$ and all $l_k \in \mathcal{L}$. Thus, $P_{i,k}$ can be computed for each pair of (x_i, l_k) by solving K Dirichlet problems in K iterations. Note that the probabilities here are used in the context of RW, which is different from the log-probabilities used in MAP-MRF energy minimization [111].

$\gamma_1 Y_{ik}/D_i$ can be considered as the initial probability that the scene node x_i is assigned label l_k given data $\mathcal{D}_i$ associated with x_i and data $\mathcal{D}^k$ associated with l_k, *i.e.,* the probability that a random walker transits from x_i to l_k in a single move:

$$\frac{\gamma_1 Y_{ik}}{D_i} = P(x_i = l_k|\mathcal{D}_i, \mathcal{D}^k) = P^1(l_k|x_i, \mathcal{D}_i, \mathcal{D}^k). \tag{3.6}$$

$\gamma_2 W_{ij}/D_i$ can be considered as the probability that the scene nodes x_i and x_j are assigned the same label given $\mathcal{D}_i$ and $\mathcal{D}_j$, *i.e.,* the probability that a random walker transits from x_i to x_j in a single move:

$$\frac{\gamma_2 W_{ij}}{D_i} = P(x_i = x_j|\mathcal{D}_i, \mathcal{D}_j) = P^1(x_j|x_i, \mathcal{D}_i, \mathcal{D}_j). \tag{3.7}$$

Y_{ik}'s and W_{ij}'s are initialized at the beginning of the algorithm and remain the same in each iteration.

Let $U_k(v_i)$ be the potential associated with node v_i in the kth iteration, which we define to be proportional to $P_k(v_i)$:

$$U_k(v_i) = \gamma_3 P_k(v_i), \tag{3.8}$$

where γ_3 is a positive constant. Since $P_k : \mathcal{V} \to [0, 1]$, $U_k : \mathcal{V} \to [0, \gamma_3]$. $U_k(\cdot)$ is a binary function on $\mathcal{L}$: $U_k(l) = \gamma_3$, when $l = l_k$; otherwise, $U_k(l) = 0$. For any $x_i \in \mathcal{X}$, $\sum_{k=1}^{K} U_k(x_i) = \gamma_3$. Once Y_{ik}'s and W_{ij}'s are defined, the probabilities $P_{i,k}$'s can then be determined from Equations (3.5) and (3.8). The whole algorithm is summarized in Algorithm 3.1.

Algorithm 3.1 Basic steps of the generalized random walks framework.

1: Construct problem-dependent functions $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$
2: Construct function $C(\cdot,\cdot)$ from Equation (3.1)
3: Construct $\mathbf{L}$ from Equation (3.3)
4: **for** $k = 1$ to K **do**
5: Calculate $U_k(\cdot)$ ($P_k(\cdot)$) for all $x_i \in \mathcal{X}$ by solving Equation (3.5)
6: **end for**
7: Find the appropriate configuration from $P_k(x_i)$'s using problem-dependent criteria

The RW model [69] requires some variables to be instantiated, *i.e.,* some scene nodes to be pre-labeled. This requirement is relaxed in GRW. The RW model with label priors (RWLP) [68] is proposed for image segmentation and derives a linear system similar to that of Equation (3.5) from a Bayesian point of view. When setting $\gamma_2 = \gamma_3$ in GRW, we can get a linear system in the same format as derived in [68][3]. The label priors in [68], which correspond to our compatibility function $Y(\cdot,\cdot)$, need to be defined following a probability expression. Although it is mentioned in [68] that the solution of RWLP is equivalent to that of the original RW [69] on an augmented graph with the label nodes as extra pre-labeled nodes, the requirement on the format of the label priors limits the choice of $Y(\cdot,\cdot)$. This requirement is relaxed in GRW, where we formulate the problem from the original RW point of view [48], where probabilities are considered to be transition probabilities of a random walker moving between nodes. The edge weighting function in [68], which corresponds to our compatibility function $W(\cdot,\cdot)$, serves as a regularization term. In GRW, $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$ are not probability quantities; instead, they represent compatibility/similarity and are used to define transition probabilities. In GRW, we have relaxed the extra requirements in [68, 69] and provided a more flexible framework, where the compatibility functions (and potential function) may be defined in any form according to the need of a particular problem. For example, for the multi-exposure fusion problem, the form of the compatibility functions is presented in Section 3.3.5. The proposed GRW can be applied to many different problems that can be formulated as estimating the probability of a site (*e.g.,* a pixel) being assigned a label given known information.

3.2 Application in Stereo Correspondence

3.2.1 Introduction

Stereo vision is the focus of very active research within the field of computer vision. One crucial and traditional task in stereo vision is stereo correspondence/stereo matching, the goal of which is to determine the disparities (differences of pixel locations) of corresponding pixels in a pair of stereo images. The disparity of a pixel is inversely proportional to the depth of the 3D point that is projected onto that pixel's location. Therefore, with the disparities calculated for the image pair, the 3D scene can be reconstructed digitally [100] or a virtual viewpoint can be generated [213]. Although several

[3]Some equations in [68] are inaccurate. Please refer to Appendix B or Grady's personal webpage (http://cns.bu.edu/~lgrady/publications.html) for the corrected version.

other techniques were proposed for recovering the depth information from a single lens utilizing artifacts that are intentionally introduced by the capture devices, such as color shifting [13] and focus/defocus [78], one major limitation of these techniques is the requirement of large apertures. For cameras with small apertures, *e.g.,* a commodity webcam, disparity is more reliable for extracting the depth information. Many algorithms have been proposed to solve this stereo correspondence problem. According to the comprehensive survey by Scharstein and Szeliski [162], the existing algorithms can be classified into two categories: local (window-based) algorithms that compute disparities within a finite window, and global algorithms that find the best disparity configuration by minimizing a global energy function. Compared with local methods, global methods normally produce more accurate results by formulating the problem in terms of finding the MAP configuration of a labeling problem. The MAP estimation is performed in the negative log domain; therefore it becomes a discrete energy minimization problem. Previously, researchers have solved this energy minimization problem using graph cuts (GC) [27] or belief propagation (BP) [177]. Here, we formulate and solve the stereo correspondence problem in the GRW framework. Experiments show that the proposed method produces more accurate results than many GC-based and BP-based MAP-MRF methods.

3.2.2 Preliminaries

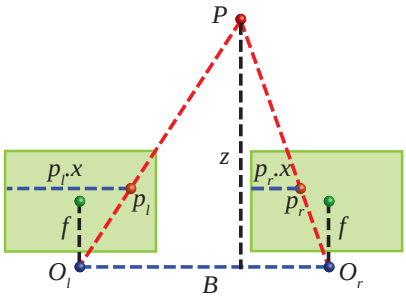


Figure 3.2: The parallel camera configuration.

Given a pair of stereo images $\mathcal{D} = \{\mathbf{I}_l, \mathbf{I}_r\}$, where $\mathbf{I}_l$ is the left image (also serving as the reference image in the discussion here) and $\mathbf{I}_r$ is the right image, the goal of stereo correspondence is to find the corresponding pixel in $\mathbf{I}_r$ for each pixel in $\mathbf{I}_l$. The difference between the coordinates of the corresponding pixels is called *disparity*. It is assumed that the images are captured under the parallel stereo configuration, as shown in Figure 3.2. Images captured under general stereo configurations can be converted to such parallel pairs using rectification techniques [61, 77]. The pinhole camera model is used. O_l and O_r are the camera centers (*i.e.,* optical centers) for the left and right cameras, respectively. p_l and p_r are the images of the 3D point P in $\mathbf{I}_l$ and $\mathbf{I}_r$, respectively. The focal lengths of the left and right cameras are assumed to be equal. The images are vertically aligned and only horizontal disparities are considered. The disparity d between p_l and p_r is defined

as the difference between their x-coordinates, *i.e.,* $d = p_l.x - p_r.x$. Let B denote the length of the baseline, f the focal length and z the depth of P (perpendicular distance from P to the baseline). The disparity d and the depth z are related as follows using stereo triangulation:

$$z = \frac{fB}{d}. \tag{3.9}$$

3.2.3 Problem Formulation

Suppose there are K different disparities from 0 to $K-1$. We can then define the set of labels $\mathcal{L}$ with $l_k = k - 1$. Let us define a set of variables $\mathcal{X}$, where a variable $x_i \in \mathcal{X}$ is associated with a pixel $p_i \in \mathbf{I}_l$ and takes a label from $\mathcal{L}$. Let f_i denote the label that x_i takes. Then, we can formulate the stereo correspondence problem as finding the most probable label for every $x_i \in \mathcal{X}$ given $\mathcal{D}$:

$$f_i^* = \arg\max_{f_i \in \mathcal{L}} P(x_i = f_i | \mathcal{D}). \tag{3.10}$$

Then, the configuration for $\mathcal{X}$ is $f^* = \{f_1^*, \ldots, f_N^*\}$, where $N = |\mathcal{X}|$.

3.2.4 Direct Application of GRW in Stereo Correspondence

GRW can be directly applied to the stereo correspondence problem with properly defined node compatibility functions. If we take the left image as the reference image, then the data compatibility function $Y(\cdot,\cdot)$ and the neighborhood compatibility function $W(\cdot,\cdot)$ can be defined as follows. Let $p_{i,l}$ denote the i-th pixel in $\mathbf{I}_l$ and assume $p_{j,r}.x + l_k = p_{i,l}.x$, then we can define Y_{ik} based on a similarity measure between the two pixels:

$$Y_{ik} = \exp(-\sigma \min(S_{ik}, \kappa)), \tag{3.11}$$

where $S_{ik} \triangleq S(p_{i,l}, l_k)$ is the similarity measure, and a lower S_{ik} denotes a higher similarity; κ sets an upper bound to the dissimilarity between corresponding areas and we take it as $\kappa = 0.05$; and σ is a weighting factor, which we take as $\sigma = 100$. A straightforward definition of S_{ik} is the root mean square deviation (RMSD) between corresponding local windows:

$$S_{ik} = \sqrt{\frac{\sum_{p_{i,l} \in \Omega_i} \|p_{i,l} - p_{j,r}\|^2}{|\Omega_i|}}, \tag{3.12}$$

where $p_{j,r}.x = p_{i,l}.x - l_k$ and Ω_i denotes an area around $p_{i,l}$, which we take as a 3×3 window. Please note that more advanced similarity measures, such as segment-based measures [87] and mutual information-based measures [80], could be employed to enhance the performance. Our focus here is to demonstrate the effectiveness of GRW in stereo correspondence even with a basic similarity measure. Assuming $p_{i,l}$ and $p_{j,l}$ are adjacent pixels, we can then define W_{ij} as:

$$W_{ij} = \exp(-\beta \min(\|p_{i,l} - p_{j,l}\|, \tau)), \tag{3.13}$$

where we take $\tau = 0.05$ and $\beta = 200$. Here, Y_{ik}'s and W_{ij}'s are calculated in the YCbCr color space[4], which provides better color discrimination than the RGB color space in our calculation. The

[4]The Y value is in the range [16/255, 235/255], and the Cb and Cr values are in the range [16/255, 240/255].

Algorithm 3.2 Basic steps of the proposed stereo correspondence algorithm.

1: Construct function $Y(\cdot,\cdot)$ from Equation (3.11)
2: Construct function $W(\cdot,\cdot)$ from Equation (3.13)
3: Apply Algorithm 3.1
4: Compute the disparity map from Equation (3.10)

probabilities $P_{i,k}$'s are then calculated using Equations (3.1) to (3.5). Here, we fix $\gamma_3/\gamma_2 = 1$ and only use the ratio $\gamma = \gamma_1/\gamma_2$ to determine the relative weight between $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$. We take $\gamma = 1e{-}4$. This basic stereo correspondence algorithm using GRW is summarized in Algorithm 3.2.

3.2.5 GRW-Based Stereo Correspondence with Disparity Refinement

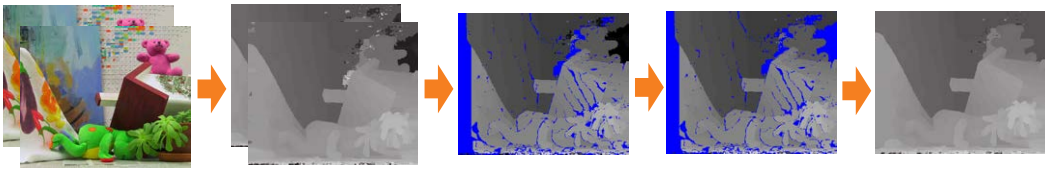


Figure 3.3: Processing procedure of the proposed algorithm.

The stereo correspondence algorithm proposed in the previous section can be further enhanced by adding refinement steps, as illustrated in Figure 3.3. More specifically, Algorithm 3.2 is first applied to both the left and right images to generate a pair of initial disparity maps. These maps then go through a left-right checking step and a textureless region handling step to determine a set of reliable matching pixels in the reference image (left image). Unreliable matching pixels are indicated in blue. With these reliable matching pixels as pre-labeled nodes, the disparities of the unreliable matching pixels are calculated using Algorithm 3.2, which results in the final disparity map.

3.2.5.1 Left-Right Checking

A left-right checking technique similar to the one used in [103] is applied to extract a set $\mathcal{X}_M$ of pixels whose reliable matching are found. The remaining pixels are categorized into the unreliable set $\mathcal{X}_U$. Since the left image $\mathbf{I}_l$ is used as the reference image, this reliable set is only determined on $\mathbf{I}_l$. In order to perform this left-right checking, an initial disparity map for the right image $\mathbf{I}_r$ is also required, and it is obtained following the same procedure as in Algorithm 3.2. For two matching pixels $(p_l, f_l) \in \mathbf{I}_l$ and $(p_r, -f_r) \in \mathbf{I}_r$ in the two images (*i.e.*, $p_r.x = p_l.x - f_l$ and $p_r.y = p_l.y$), the reliability of (p_l, f_l) is determined by:

$$(p_l, f_l) \in \begin{cases} \mathcal{X}_M, & f_l \leqslant f_r; \\ \mathcal{X}_U, & \text{otherwise.} \end{cases} \tag{3.14}$$

Algorithm 3.3 Basic steps of the proposed stereo correspondence algorithm with disparity refinement.

1: Apply Algorithm 3.2 to the left image
2: Apply Algorithm 3.2 to the right image
3: Apply left-right checking using Equation (3.15)
4: Apply textureless region handling as in Section 3.2.5.2
5: Apply Algorithm 3.2 to the left image with reliable matching pixels as pre-labeled nodes
6: Compute the disparity map from Equation (3.10)

3.2.5.2 Textureless Region Handling

Textureless regions tend to cause inaccurate disparity estimation. Due to the high degree of similarity of a pixel to its neighboring pixels, a small disparity is very likely to be assigned while the true disparity may be much larger. Pixels in these regions are considered to be unreliable matching pixels. A textureless region is considered to be a region within $\mathcal{X}_M$, where each pixel (p_i, f_i) has a local variation no larger than a threshold θ_1 and is assigned a disparity no larger than θ_2:

$$(p_i, f_i) \in \begin{cases} \mathcal{X}_U, & (\frac{1}{|\Omega_i|}\sum_{p_j \in \Omega_i} \|p_j - \bar{p}_j\| \leqslant \theta_1) \wedge (f_i \leqslant \theta_2); \\ \mathcal{X}_M, & \text{otherwise} \end{cases} \tag{3.15}$$

where Ω_i denotes a region centered at p_i and $\bar{p}_j$ the average pixel in Ω_j. A larger θ_1 or θ_2 will result in more pixels being classified as unreliable ones. In our current implementation, we take Ω_i as a 1×3 horizontal window, $\theta_1 = 0.05$, and $\theta_2 = 0.15 l_K$. This enhanced stereo correspondence algorithm is summarized in Algorithm 3.3.

3.2.6 Experimental Results

The proposed algorithms were evaluated using the standard benchmarks from Middlebury[5]. A set of constant parameters was used for all images. The accuracy of the algorithm was compared with several BP-based and GC-based algorithms in Table 3.1. Each decimal entry from the second column to the last column represents the percentage of bad pixels (absolute disparity error larger than 1.0) for different regions. The abbreviation *nocc* stands for non-occluded regions; *all* stands for the whole region of the image; and *disc* stands for regions near disparity discontinuities, which include the boundaries of occluded regions. The last column shows the average percentage of bad pixels over all the previous twelve columns, and the algorithms compared in the table are sorted by the values in this column. A border of 10 pixels for Venus and 18 pixels for Tsukuba is ignored when computing statistics. The disparity maps are shown against the ground truths in Figure 3.4, together with the reference images and bad pixel maps. Bad pixels in non-occluded regions are marked in black, while those in occluded regions are marked in gray. The reference images (from top to bottom) are from the Tsukuba, Venus, Teddy and Cones stereo image pairs, respectively.

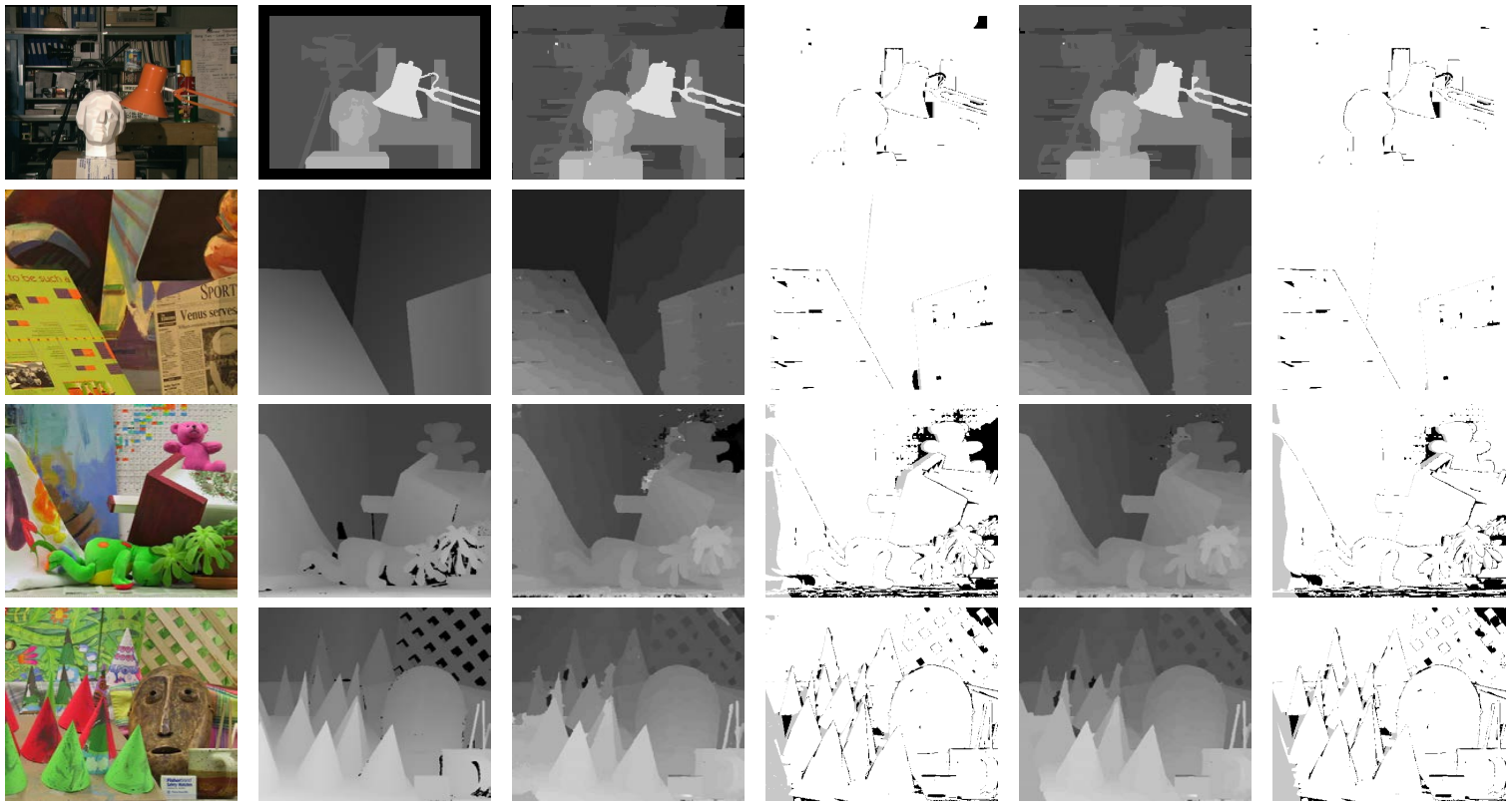
[5] http://vision.middlebury.edu/stereo/.

Table 3.1: The Middlebury stereo evaluation results of the proposed algorithms. The unit is percentage (%) of bad pixels.

Algorithm	Tsukuba			Venus			Teddy			Cones			Average percentage of bad pixels
	nocc	*all*	*disc*	*nocc*	*all*	*disc*	*nocc*	*all*	*disc*	*nocc*	*all*	*disc*	
Algorithm 3.3	**1.67**	**2.12**	**8.36**	**0.80**	**0.93**	**2.65**	**7.15**	**12.3**	**16.6**	**4.86**	**10.3**	**10.9**	**6.55**
EnhancedBP [100]	0.94	1.74	5.05	0.35	0.86	4.34	8.11	13.3	18.5	5.09	11.1	11.0	6.69
BP+DirectedDiff [15]	2.90	4.47	15.1	0.65	1.20	4.52	5.07	14.7	15.7	2.94	12.6	7.50	7.29
Algorithm 3.2	**1.98**	**2.54**	**7.83**	**1.04**	**1.41**	**3.95**	**10.1**	**15.0**	**18.8**	**4.85**	**10.7**	**10.8**	**7.41**
BPcompressed [207]	2.68	3.63	9.59	1.33	1.89	9.09	8.36	13.9	16.4	3.71	9.85	9.92	7.53
GC+occ [90]	1.19	2.01	6.24	1.64	2.19	6.75	11.2	17.4	19.8	5.36	12.4	13.0	8.26
MultiCamGC [91]	1.27	1.99	6.48	2.79	3.13	3.60	12.0	17.6	22.0	4.89	11.8	12.1	8.31
GC [27]	1.94	4.12	9.39	1.79	3.44	8.75	16.5	25.0	24.9	7.70	18.2	15.3	11.4

Table 3.2: More evaluation results of the proposed algorithms using the Middlebury datasets.

Image pair	Size (unit: pixel)	Disparity range (unit: pixel)	Algorithm 3.2				Algorithm 3.3			
			Time (unit: sec)	RMSE (unit: pixel)			Time (unit: sec)	RMSE (unit: pixel)		
				nocc	*all*	*disc*		*nocc*	*all*	*disc*
Tsukuba	$384 \times 288 \times 2$	$[0, 15]$	1.35	0.4670	0.4722	0.4493	2.53	0.4668	0.4709	0.4538
Venus	$434 \times 383 \times 2$	$[0, 19]$	2.22	0.3895	0.4080	0.5292	4.07	0.3762	0.3845	0.4740
Teddy	$450 \times 375 \times 2$	$[0, 59]$	3.96	1.1082	1.2517	1.4302	6.78	0.8710	1.0529	1.3005
Cones	$450 \times 375 \times 2$	$[0, 59]$	3.85	0.8093	1.2046	1.1585	6.99	0.8108	1.1852	1.1624

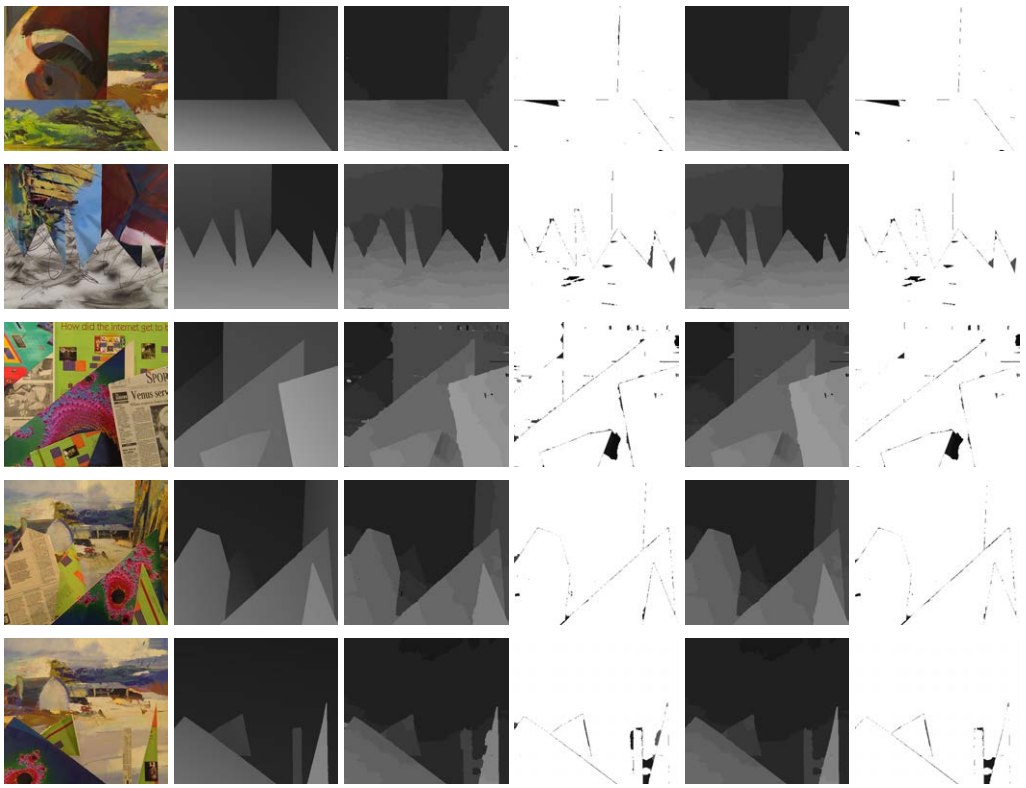


(a) (b) (c) (d) (e) (f)

Figure 3.4: Dense disparity maps on the Middlebury stereo datasets generated by our algorithms. (a) Reference images. (b) Ground truths. (c) Our results (Algorithm 3.2). (d) Bad pixels (Algorithm 3.2). (e) Our results (Algorithm 3.3). (f) Bad pixels (Algorithm 3.3).

Table 3.3: Evaluation results of the proposed algorithms using other datasets from Middlebury.

Image Pair	Size (unit: pixel)	Algorithm 3.2			Algorithm 3.3		
		Time (unit: sec)	Bad Pixels (unit: %)	RMSE (unit: pixel)	Time (unit: sec)	Bad Pixels (unit: %)	RMSE (unit: pixel)
Bull	$433 \times 381 \times 2$	2.12	0.82	0.3411	4.09	0.78	0.3403
Sawtooth	$434 \times 380 \times 2$	2.15	1.58	0.3906	4.17	1.38	0.3793
Poster	$435 \times 383 \times 2$	2.16	3.10	0.4603	4.14	2.48	0.4320
Barn 1	$432 \times 381 \times 2$	2.21	0.74	0.3287	4.04	0.69	0.3263
Barn 2	$430 \times 381 \times 2$	2.24	1.75	0.4267	4.24	1.25	0.4040



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

Figure 3.5: Dense disparity maps for other stereo datasets from Middlebury. (a) Reference images. (b) Ground truths. (c) Our results (Algorithm 3.2). (d) Bad pixels (Algorithm 3.2). (e) Our results (Algorithm 3.3). (f) Bad pixels (Algorithm 3.3).

The proposed algorithms are currently implemented in Matlab. The computational time ranges from 1.35 (Tsukuba) to 3.96 seconds (Teddy) by Algorithm 3.2 and from 2.51 (Tsukuba) to 6.99 seconds (Cones) by Algorithm 3.3 on a 2.53-GHz dual-core laptop with 4-GB memory, as shown in Table 3.2[6]. The size of each dataset and the root mean squared error (RMSE) between each disparity map and its corresponding ground truth are also given in Table 3.2.

More disparity maps are shown in Figure 3.5 using other datasets from Middlebury. Bad pixels

[6]Unless otherwise noted, all of the proposed algorithms in this thesis were implemented in Matlab and the processing times were recorded on a 2.53-GHz dual-core laptop with 4-GB memory.

for all regions are indicated in black. The disparity ranges of all the image pairs are between $[0, 19]$. The size, computational time, percentage of bad pixels and RMSE for all regions of each data set are given in Table 3.3 with the same order as in Figure 3.5.

3.3 Application in Multi-Exposure Fusion

3.3.1 Introduction

A natural scene often has a high dynamic range (HDR) that exceeds the capture range of common digital cameras. Therefore, a single digital photo is often insufficient to provide all the details in a scene due to under- or over-exposed regions. On the other hand, given an HDR image, current displays are capable of handling only a very limited dynamic range. Researchers have explored various directions to resolve the contradiction between the HDR nature of real-world scenes and the low dynamic range (LDR) limitation of current image acquisition and display technologies. Although cameras with spatially varying pixel exposures [133], cameras that automatically adjust exposure for different parts of a scene [126, 185], and displays that directly display HDR images [163] have been developed by previous researchers, their technologies are only at a prototyping stage and are unavailable to ordinary users.

Instead of employing specialized image sensors, an HDR image can be reconstructed digitally using HDR reconstruction (HDR-R) techniques from a set of images of the same scene taken by a conventional LDR camera [46] or a panoramic camera [4] under different exposure settings. While the rich data contained in these reconstructed HDR images facilitate many tasks (*e.g.,* scientific visualization) [157], special treatment is required to adapt the HDR images for viewing on consumer displays, which have limited dynamic ranges. Therefore, tone mapping (TM) operators [157] have been developed to compress the dynamic range of an HDR image.[7] This two-phase workflow, HDR-R+TM (HDR-R and TM), has several advantages: no specialized hardware is required; various operations, such as virtual exposure, can be performed on the HDR images; and user interactions are allowed in the TM phase to generate a tone-mapped image with desired appearance. However, this workflow is usually not as efficient as image fusion (IF, *e.g.,* [67, 129]) in presenting the captured scene for viewing on LDR displays, because IF directly generates a composite LDR image from a source sequence without involving HDR-R, and minimal user intervention is required, as shown in Figure 3.6. While the objective of TM is to preserve information from a single image during tone manipulation, IF aims to combine information from multiple images. Since it skips the HDR-R step, IF does not need the calibration of the camera response function, which is required in HDR-R if the camera response function is not linear. IF is a preferred method of quickly generating a well-exposed image from an input set of multi-exposure images, especially when the number of input images is small and speed is crucial.

[7] TM was initially proposed in graphics for synthetic HDR images generated using rendering methods such as ray tracing [186].

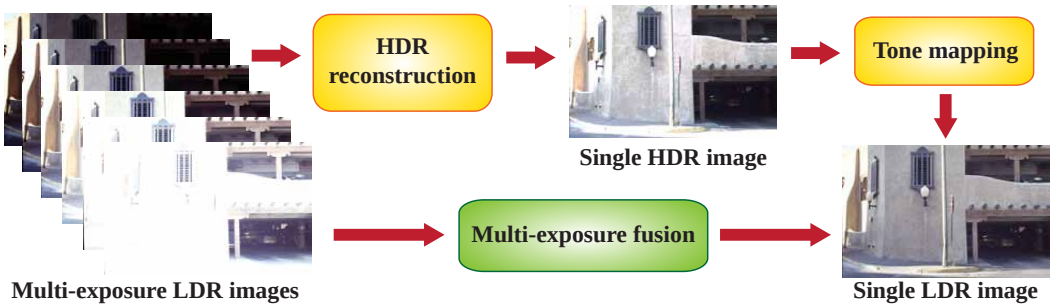


Figure 3.6: Comparison between multi-exposure fusion and the HDR reconstruction and tone mapping workflow. (The Garage image sequence courtesy of Shree Nayar.)

Since its introduction in the 1980s, IF has been employed in various applications, such as multi-sensor fusion [96, 211] (combining information from multi-modality sensors), multi-focus fusion [108, 210] (extending depth-of-field from multi-focus images), and multi-exposure fusion (MEF) [21, 129] (merging details of the same scene revealed under different exposures). Although some general fusion approaches [147, 144] have been proposed by previous researchers, they are not optimized for individual applications and have only been applied to grayscale images. Here, we focus exclusively on MEF, proposing a novel fusion algorithm that is both efficient and effective. Unlike previous MEF methods [67, 129], our algorithm is based on a probabilistic model and global optimization taking neighborhood information into account. GRW is employed to calculate an optimal set of probabilities subject to two quality measures: local variation and color consistency. The local variation measure preserves details; the color consistency measure, which is not considered by previous methods, includes both consistency in a neighborhood and consistency with the natural scene. The proposed fusion algorithm produces a final fused LDR image with fine details and an optimal balance between the two quality measures.

3.3.2 Related Work

3.3.2.1 Multi-Exposure Fusion

Image fusion methods combine information in different images into a single composite image. The history of IF research dates back to 1984 when Burt [31] proposed Laplacian pyramid-based fusion for binocular monochrome images. In 1993, Burt and Kolczynski [33] applied this method to the fusion of multi-exposure monochrome images. For color images, a number of MEF methods have been proposed over the last decade, and some representative methods are reviewed in this section. Please refer to [19, 173] for discussions on IF methods in different applications. For multi-exposure images, IF methods directly work on the input LDR images and focus on enhancing dynamic range while preserving details. Goshtasby [67] partitioned the input images into uniform blocks and tried to maximize the information in each block based on an entropy measure. However, the approach may generate artifacts on object boundaries if the block is not sufficiently small. Instead of divid-

ing images into blocks, Cheng and Basu [39] combined images in a column-by-column fashion. This algorithm maximizes the contrast within a column by selecting pixels from different images. However, since no neighborhood information is considered, it cannot preserve color consistency, and artifacts may occur on object boundaries. Cho and Hong [40] focused on substituting the well-exposed regions in one image for the under- or over-exposed regions in another image, which are determined by a saturation mask. Region boundaries are blended by minimizing an energy function defined in the gradient domain. Although this algorithm works better on object boundaries, it is only applicable to two images. Our algorithm works at the pixel level and applies a global optimization taking neighborhood information into account, which allows it to avoid boundary artifacts.

Image fusion can also be interpreted as an analogy to alpha blending. Raman and Chaudhuri [153] generated the fused image by solving an unconstrained optimization problem. The weight function for each pixel was modeled based on local variance in such a way that the fused image tends to have uniform illumination or contrast. Raman and Chaudhuri [154] generated mattes for each pixel in an image using the difference between the original pixel value and the pixel value after bilateral filtering. This measure strengthens weak edges, but may not be sufficient to enhance the overall contrast. Our algorithm defines two quality measures and finds the optimal balance between them, *i.e.,* enhancing local contrast while imposing color consistency.

Multi-resolution analysis based fusion has demonstrated good performance in enhancing main image features. Bogoni and Hansen [21] proposed a pyramid-based pattern-selective fusion technique. Laplacian and Gaussian pyramids are constructed for the luminance and chrominance components, respectively. However, the color scheme of the fused image may be very close to that of the average image because pixels with saturation closest to the average saturation are selected for blending. Mertens *et al.* [129] constructed Laplacian pyramids for the input images and Gaussian pyramids for the weight maps. A weight for a pixel is determined by three quality measures: local variation, saturation, and well-exposedness. The Laplacian and Gaussian pyramids are blended at each level, and are then collapsed to form the final image. However, with only local calculation involved and no measure to preserve color consistency, the fusion results may be unnatural. Our algorithm also uses local variation as one quality measure, but another quality measure that we consider is color consistency, which is not employed by [129]. Moreover, our algorithm does not use pyramid decomposition but solves a linear system defined at the pixel level.

3.3.2.2 HDR Reconstruction and Tone Mapping

For multi-exposure images, unlike MEF, TM does not directly take an LDR sequence as input, but works on the reconstructed HDR image. Although related, TM and MEF are fundamentally different and were initially proposed for different purposes. As a branch of IF, MEF focuses on combining and enhancing information from multiple sources (*i.e.,* multi-exposure images), while TM aims to preserve information from a single source (*i.e.,* an HDR image) during its dynamic

range reduction process. Although the HDR-R+TM workflow is usually used in different scenarios than IF methods, we still give a brief discussion of those HDR-R and TM methods, which share some similar features with our MEF algorithm, because the original input and the final output of this workflow is the same as those of MEF. Please refer to [34, 157] for more detailed discussion of TM. Given an input LDR sequence and exposure times associated with each image in the sequence, HDR-R techniques [46] first recover the camera response function, which is a mapping from the exposure at each pixel location to the pixel's digital LDR value, and then use the camera response function to reconstruct an HDR image via a weighting function. Debevec and Malik [46] recovered the camera response function by minimizing a quadratic objective function defined on exposures, pixels' LDR values, and exposure times. Then, a hat-shaped weighting function is used to reconstruct the HDR image. Granados *et al.* [73] assumed that the camera response function is linear and focused on the development of an optimal weighting function using a compound-Gaussian noise model. Our MEF algorithm also solves a quadratic objective function, but the function is defined on local features and the solution leads to probability maps.

Given an HDR image, tone mapping methods [34] aim to reduce its dynamic range while preserving details. The history of TM research in computer graphics can be traced back to 1993, when Tumblin and Rushmeier [186] introduced a brightness sensation model-based operator for synthetic monochrome HDR images.[8] Depending on whether the mapping scheme between HDR and LDR is spatially invariant or not, TM methods can be classified into global or local operators [102]. Global TM methods [101, 49], which apply spatially invariant mapping of luminance values, usually have high speed, while local TM methods [93, 156, 113, 167], which apply spatially varying mapping, usually produce images with better quality, especially when strong local contrast is present [156, 167].

Reinhard *et al.* [156] proposed a global operator using log average luminance and a local operator that implements a printing technique called dodging-and-burning. Automatic parameter estimation for these two operators was introduced in [155]. Li *et al.* [113] decomposed the luminance channel of the input HDR image into multiscale subbands and applied local gain control to the subbands. Shan *et al.* [167] defined a linear system for each overlapping window in the HDR image, and the solution of each linear system are two coefficients that map luminance values from HDR to LDR. Although our MEF algorithm also defines a linear system, we define the linear system on pixels from all of the original LDR images, and the solution is a set of probabilities that determine the contributions from each pixel of each original LDR image to its corresponding pixel in the fused image. Krawczyk *et al.* [93] segmented an HDR image into frameworks with consistent luminance and computed the belongingness of each pixel to each framework using the framework's centroid, which resulted in a set of probability maps. Our MEF algorithm also generates probability maps, but directly from the original LDR sequence with no HDR or segmentation involved. One typical

[8] An earlier TM method, which handles a relatively limited dynamic range, was introduced in the lighting community in 1984 [157].

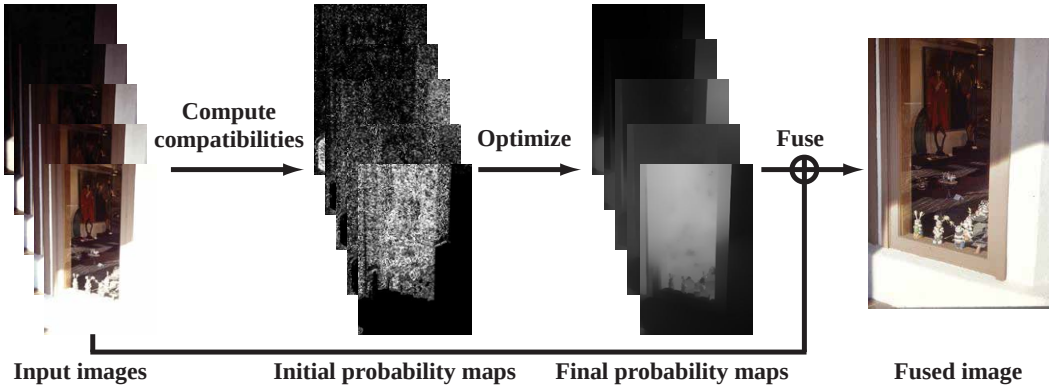


Figure 3.7: Processing procedure of the proposed fusion algorithm. (The Window image sequence courtesy of Shree Nayar.)

problem with some local TM methods is that halo artifacts are introduced as a result of contrast reversals near high contrast edges [34, 102]. Our MEF algorithm balances between contrast and consistency in a neighborhood, which can prevent contrast reversals. Another problem with some TM methods is that color artifacts such as oversaturation may be introduced into the results, because operations are usually performed in the luminance channel without involving chrominance [34]. This problem does not apply to our MEF algorithm, because every color channel is treated equally.

3.3.3 Algorithm Overview

Unlike most previous MEF methods, we consider image fusion as a probabilistic composition process, as illustrated in Figure 3.7[9]. The initial probability that a pixel in the fused image belongs to each input image is estimated based on local features. Taking neighborhood information into account, the final probabilities are obtained by global optimization using the generalized random walks. These probability maps serve as weights in the linear fusion process to produce the fused image. In a probability map, the brighter a pixel is, the higher the probability. These processes are explained in detail below.

3.3.4 Problem Formulation

The fusion of a set of multi-exposure images can be formulated as a probabilistic composition process. Let $\mathcal{D} = \{\mathbf{I}_1, \dots \mathbf{I}_K\}$ denote the set of input images and $\mathcal{L} = \{l_1, \dots, l_K\}$ the set of labels, where a label $l_k \in \mathcal{L}$ is associated with the kth input image $\mathbf{I}_k \in \mathcal{D}$. $\mathbf{I}_k$'s are assumed to be already registered and to be the same size with N pixels each. Normally, $K \ll N$. Let us define a set $\mathcal{X}$ of variables such that an $x_i \in \mathcal{X}$ is associated with the ith pixel $\bar{p}_i$ in the fused image $\bar{\mathbf{I}}$ and takes a value from $\mathcal{L}$. Then, each pixel in the fused image $\bar{\mathbf{I}}$ can be represented as:

$$\bar{p}_i = \sum_{k=1}^{K} P_{i,k} p_{i,k}, \tag{3.16}$$

[9]The initial probability maps are normalized at each pixel.

where $p_{i,k}$ denotes the ith pixel in $\mathbf{I}_k$ and $P_{i,k} \triangleq P(x_i = l_k|\mathcal{D})$ the probability of pixel $\bar{p}_i$ being assigned label l_k given $\mathcal{D}$ with $\sum_k P_{i,k} = 1$. This probabilistic formulation converts the fusion problem to the calculation of marginal probabilities given the input images subject to some quality measures and helps to achieve an optimal balance between different quality measures. If every pixel is given equal probability, *i.e.,* $P_{i,k} = \frac{1}{K}, \forall i, k$, $\bar{\mathbf{I}}$ is simply the average of $\mathbf{I}_k$'s. Although it is also possible to combine only the pixels with the highest probabilities instead of using Equation (3.16), artifacts may appear at locations with significant brightness changes between different input images.

If $P_{i,k}$'s are simply viewed and calculated as local weights, after which some relatively simple smoothing filters, such as Gaussian filter and bilateral filter, are applied, two results may occur. Artifacts (such as halos) may appear at object boundaries, or it may be difficult to determine the termination criteria of the filtering. Therefore, the results are usually unsatisfactory [129]. An alternative is to use multi-resolution fusion techniques [21, 129], where the weights are blended at each level to produce more satisfactory results. However, the weights at each level are still determined locally, which may not be optimal in a large neighborhood. In contrast to multi-resolution techniques, we propose an efficient single-resolution fusion technique that involves formulating $P_{i,k}$'s in Equation (3.16) as probabilities in GRW. A set of optimal $P_{i,k}$'s that balances the influence of different quality measures is computed based on global optimization in GRW. Experiments (Section 5.2.6) show that the results of the proposed probabilistic fusion technique are comparable to the results of multi-resolution techniques and the results of the HDR-R+TM workflow.

3.3.5 Compatibility Functions

The compatibility functions $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$ are defined to represent the two quality measures, *i.e.,* local variation and color consistency, respectively. Since image contrast is usually related to variations in image luminance [34], in order to preserve local contrasts, the probability calculation should be biased towards pixels from the images that provide more local variations in luminance. Let $G_{i,k}$ denote the second-order partial derivative computed in the luminance channel at the ith pixel in image $\mathbf{I}_k$. The higher the magnitude of $G_{i,k}$ (denoted by $\|G_{i,k}\|$) is, the more variations occur near the pixel $p_{i,k}$, which may indicate more local contrast. In [129], a Laplacian filter is used to calculate the variations. Here both a Laplacian filter and a central difference in a 3×3 neighborhood were tested. With all other settings the same, central difference produces slightly better visual quality in the fused images. Therefore, central difference is used to approximate the second-order derivative. However, if the frequency (*i.e.,* number of occurrences) of a value $\|G_{i,k}\|$ in $\mathbf{I}_k$ is very low, the associated pixels may constitute noise, or may belong to unimportant features. Therefore, unlike previous methods [129, 153], our variation measure is normalized by the frequencies of each $\|G_{i,k}\|$. In addition, $G_{i,k}$'s are modified using a sigmoid-shaped function to suppress the difference in high contrast regions. Hence, taking into account both the magnitude and the frequency of $G_{i,k}$, the

Algorithm 3.4 Basic steps of the proposed fusion algorithm.

1: Construct function $Y(\cdot,\cdot)$ from Equation (3.17)
2: Construct function $W(\cdot,\cdot)$ from Equation (3.18)
3: Apply Algorithm 3.1
4: Compute the fused image from Equation (3.16)

compatibility between a pixel and a label is computed as:

$$Y_{ik} = \theta_{ik}[\mathrm{erf}(\frac{\|G_{i,k}\|}{\sigma_y})]^K, \tag{3.17}$$

where θ_{ik} represents the frequency of the value $\|G_{i,k}\|$ in $\mathbf{I}_k$; $\mathrm{erf}(\cdot)$ is the Guass error function, which is monotonically increasing and sigmoid shaped; the exponent K is equal to the number of input images and controls the shape of $\mathrm{erf}(\cdot)$ by placing less emphasis on the difference in high contrast regions as the number of input images increases; and σ_y is a weighting coefficient, which we take as the standard deviation of all $G_{i,k}$'s.

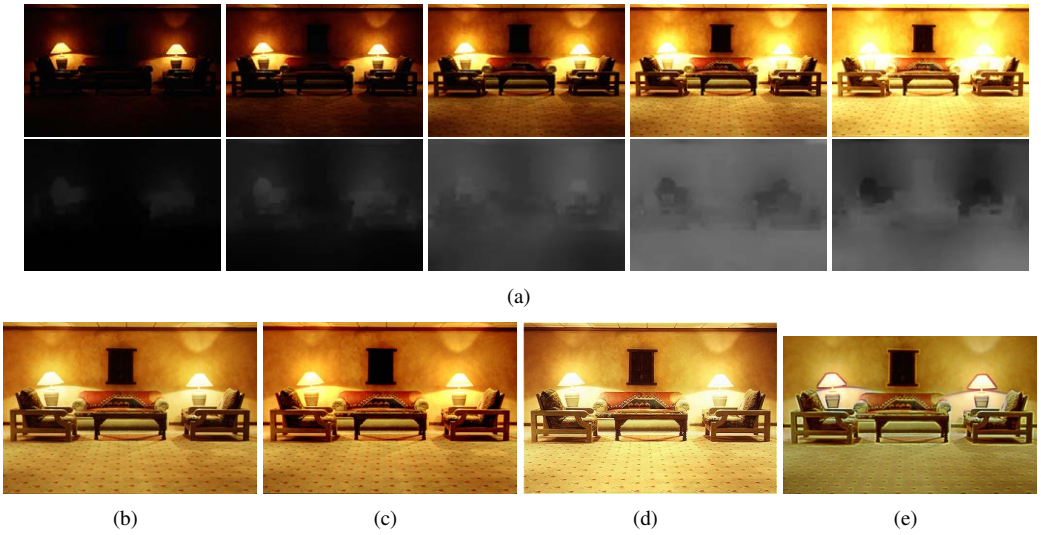
The other quality measure used in our algorithm is color consistency, which is not considered in previous methods [67, 129]. Although Bogoni and Hansen [21] also suggested a color consistency criterion by assuming that the hue component for all the input images is constant, this assumption is not true if the images are not taken with close exposure times. In addition, they did not consider consistency in a neighborhood. Our color consistency measure imposes not only color consistency in a neighborhood but also consistency with the natural scene. If two adjacent pixels in most images have similar colors, then it is very likely that they have similar colors in the fused image. Also, if the colors at the same pixel location in different images under appropriate exposures (not under- or over-exposed) are similar, they indicate the true color of the scene, and the fused color should not deviate from these colors. Therefore, the following equation is used to evaluate the similarity/compatibility between adjacent pixels in the input image set using all three channels in the RGB color space:

$$W_{ij} = \prod_k \exp(-\frac{\|p_{i,k} - p_{j,k}\|}{\sigma_w}) \approx \exp(-\frac{\|\overline{p}_i - \overline{p}_j\|}{\overline{\sigma}_w}), \tag{3.18}$$

where $p_{i,k}$ and $p_{j,k}$ are adjacent pixels in image $\mathbf{I}_k$; $\exp(\cdot)$ is the exponential function; $\|\cdot\|$ denotes Euclidean distance; $\overline{p}_i = \frac{1}{K}\sum_k p_{i,k}$ denotes the average pixel; and σ_w and $\overline{\sigma}_w = \sigma_w/K$ are free parameters. Although the two quality measures are defined locally, a global optimization in GRW is carried out to produce a fused image that maximizes contrast and details, as well as imposing color consistency. Once $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$ are defined using Equations (3.17) and (3.18), the probabilities $P_{i,k}$'s are calculated using Equations (3.1) to (3.5). Here, we fix $\gamma_3/\gamma_1 = 1$ and only use the ratio $\gamma = \gamma_2/\gamma_1$ to determine the relative weight between $Y(\cdot,\cdot)$ and $W(\cdot,\cdot)$. The basic steps of our algorithm are summarized in Algorithm 3.1.

3.3.6 Acceleration

To accelerate the computation as well as reduce memory usage, the final probability maps are computed at a lower resolution of the Laplacian matrix and then interpolated back to the original resolu-



(a)

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

Figure 3.8: Comparison of our method with EF, EntropyF, and VC using the Chairs image sequence. The result of EF is comparable to ours. The result of EntropyF suffers some over-exposure. The result of VC shows serious halo artifacts on object boundaries. (a) Source sequence (top) and final probability maps (bottom). (b) Proposed. (c) EF. (d) EntropyF. (e) VC. (Source sequence courtesy of Shree Nayar.)

tion before being used in Equation (3.16). The local variation $G_{i,k}$ of each pixel is calculated at the original resolution. Then, the images are divided into uniform blocks of size $\eta \times \eta$. The average of $G_{i,k}$'s in a block is used to calculate the compatibility Y_{ik} between that block and the label l_k. The compatibility between two adjacent blocks is computed as W_{ij} of the average pixel in one block and the average pixel in the other block.

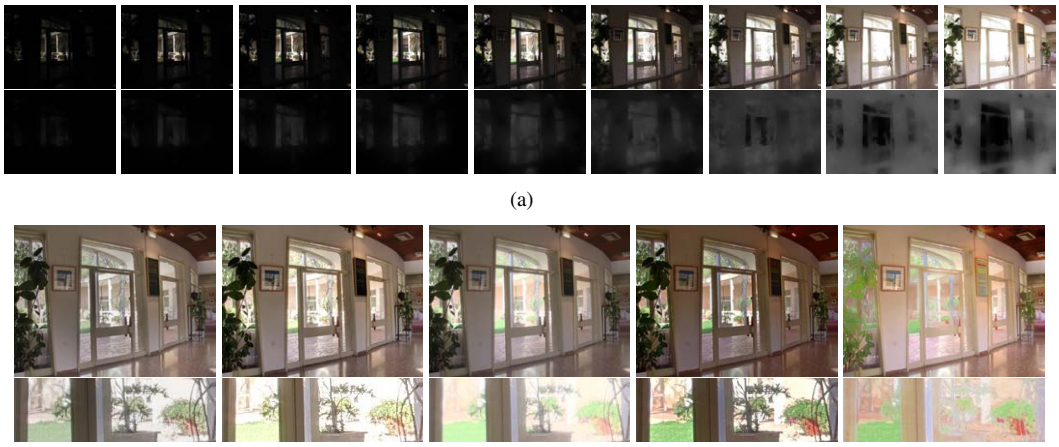
3.3.7 Experimental Results and Discussion

3.3.7.1 Comparison with Other Methods

Only three free parameters are used in our algorithm, and we take $\overline{\sigma}_w = 0.1, \gamma = 1.0, \eta = 4$ in all experiments unless otherwise stated. Figure 4.15 shows the comparison of our MEF algorithm with three other MEF methods on the Chairs image sequence. The input image sequence and final probability maps are given in Figure 4.15(a). Brighter pixels in a probability map stand for higher probabilities. The result of entropy fusion (EntropyF) [67] was taken from its project webpage[10]. The result of variational composition (VC) [153] was taken from its paper. The results of exposure fusion (EF) [129] in all experiments were generated by the Matlab implementation provided by its authors. Its default parameter setting was used in all experiments. The result of EF is comparable to ours. The result of EntropyF suffers some over-exposure. The result of VC shows serious halo artifacts on object boundaries.

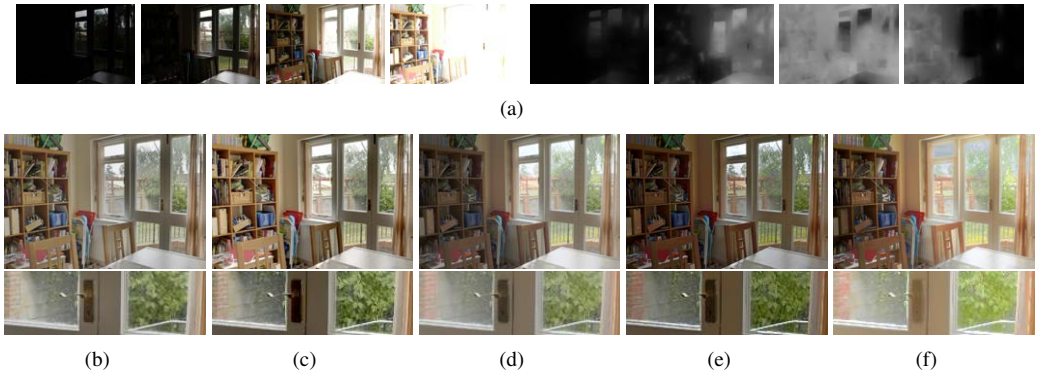
Figures 3.9 and 3.10 give comparison of our MEF algorithm with EF and three TM methods on two image sequences. In all experiments, the intermediate HDR images for photographic tone

[10] http://www.cs.wright.edu/~agoshtas/hdr.html.



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

Figure 3.9: Comparison of our algorithm with EF, PTR, SC and LW using the Belgium House image sequence. The intermediate HDR image for PTR, SC and LW is generated by HDRShop. The results of EF and PTR suffer over-exposure in the window regions. The results of SC and LW show some color distortion (looking pink). (a) Source image sequence (top) and final probability maps (bottom). (b) Proposed. (c) EF. (d) PTR. (e) SC. (f) LW. (Source sequence courtesy of Dani Lischinski.)



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

Figure 3.10: Comparison of our algorithm with EF, PTR, SC and LW using the House image sequence. The intermediate HDR image for PTR, SC and LW is generated by HDRShop. EF introduces color artifacts that assign different colors to the two chairs of the same type. The result of PTR is slightly dark for the indoor scene. The result of SC looks somewhat pink and dark through the entire image. The result of LW shows some color distortion. Our result reveals more detail than EF, especially in the door lock region. (a) Source image sequence (left four) and final probability maps (right four). (b) Proposed. (c) EF. (d) PTR. (e) SC. (f) LW. (Source sequence courtesy of Tom Mertens.)

reproduction (PTR) [156], subband compression (SC) [113] and linear windowed (LW) [167] were generated from the corresponding LDR sequences using HDRShop[11], which employed the HDR-R algorithm in [46]. The results of PTR were generated by an HDRShop plugin[12]. The results of SC and LW were generated by the Matlab implementations provided by their respective authors. There are no constant sets of parameters in their original papers, because TM methods usually depend on

[11] http://www.hdrshop.com/.

[12] http://www.gregdowning.com/HDRI/tonemap/Reinhard/.

user-controlled parameters to generate desired tone-mapped images. However, in order to give a relatively fair image quality comparison with our and other MEF methods where constant sets of parameters were used throughout the experiments, we used the default parameter settings in their programs in all experiments.

Figure 3.9 shows the experimental results on the Belgium House sequence. The last row gives a closeup view of the window regions. Although the result of SC preserves as much detail as ours, it looks a little pink due to color distortion. The results of EF and PTR suffer over-exposure in all the window regions. The result of LW shows some color artifacts, *e.g.,* color reversal of the blackboard. Color artifacts in the results of TM methods are usually caused by operations carried out solely in the luminance channel without involving chrominance [34]. Our method treats each color channel equally and imposes color consistency, which helps to avoid color artifacts. Note that adjusting the parameters in SC and LW may reduce color distortion and generate more pleasing images, which is considered as a common user interaction in TM methods. However, we use constant sets of parameters for both TM and MEF methods in our experiments in order to give a relatively fair comparison of these two types of methods.

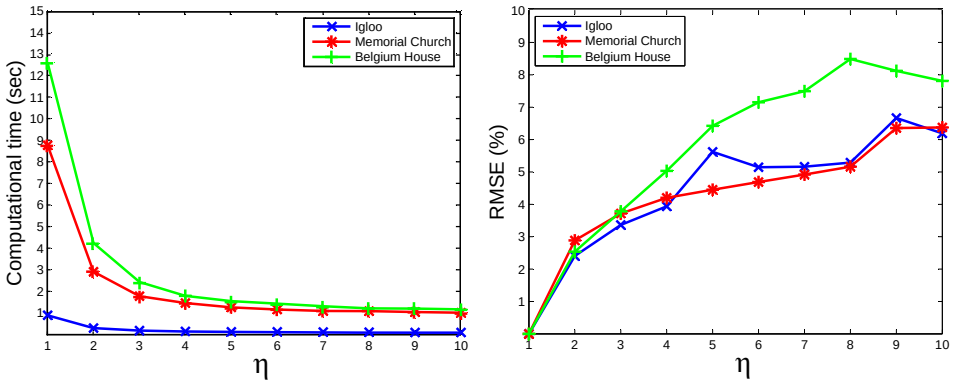
Figure 3.10 shows the results on the House sequence. The last row gives a closeup view of the door lock region. The result of PTR is slightly dark for the indoor scene. The result of SC reveals many details of the outdoor scene but looks somewhat pink and dark through the entire image. The result of LW also reveals many details but shows some color distortion of the outdoor scene. Although both our method and EF use local variations to indicate local contrasts, we use a nonlinear function to modify the contrast indicators. This nonlinear mapping aims to reflect the nonlinear human perception of contrast [197]. In addition, the well-exposedness measure in EF is biased towards pixels with a specific luminance. Therefore, our result reveals more detail than EF, especially in the door lock region. The two chairs in the original sequence have the same color, but for EF the color difference between them is quite obvious. Our method keeps the colors consistent while EF fails, because our method imposes consistency in large neighborhoods and consistency with the natural scene via the color consistency measure.

3.3.7.2 Computational Complexity

In the initialization step, the input image sequence is converted to gray scale for calculating Y_{ik}'s, and an average image is also computed for calculating W_{ij}'s. The complexity of this step is $O(NK)$, where N is the number of pixels in an input image and K is the number of images in the input sequence. Since the number of operations is proportional to the total number of input pixels, the complexity for computing the compatibilities is also $O(NK)$. We employ CHOLMOD [38], which is based on supernodal Cholesky factorization, to solve the linear system in Equation (3.5). The complexity of this direct solver is proportional to the number of the nonzero entries in $\mathbf{L}_{\mathcal{X}}$, and in our case it is $O(\frac{N}{\eta^2})$. Since there are a total of K linear systems to be solved, the complexity of this

Table 3.4: Computational times of the proposed algorithm and EF on the test image sequences. Times are recorded in seconds.

Input sequence	Size	Proposed	EF
Window	$226 \times 341 \times 5$	**0.17**	0.68
Chairs	$343 \times 231 \times 5$	**0.17**	0.68
Garage	$348 \times 222 \times 6$	**0.20**	0.80
Igloo	$236 \times 341 \times 6$	**0.20**	0.85
House	$752 \times 500 \times 4$	**0.72**	2.64
Memorial Church	$512 \times 768 \times 16$	**2.18**	9.82
Belgium House	$1025 \times 769 \times 9$	**2.74**	11.68



(a) (b)

Figure 3.11: Analysis of acceleration with different block width η. Error is defined as the relative difference from the results generated with $\eta = 1$. Significant efficiency improvement is only observed when $\eta \leqslant 5$ and significant error increase only occurs when $\eta \leqslant 5$. (a) Effectiveness of acceleration. (b) Error introduced.

step is $O(\frac{NK}{\eta^2})$. With only linear operations involved, the complexity of the fusion step is $O(NK)$. Therefore, the total complexity of our algorithm is $O(NK)$.

The computational times for seven image sequences are reported in Table 3.4, along with the comparison with EF [129]. Both our and EF's Matlab implementations were executed on the same computer. Times for reading the input sequences and writing the output images were excluded. Our algorithm took only 25% of the total time of EF on the average.

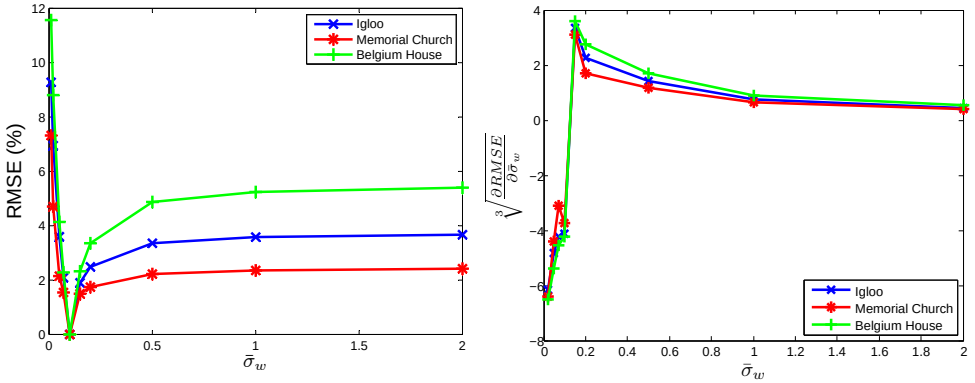
3.3.7.3 Analysis of Free Parameters

The effectiveness of acceleration is illustrated in Figure 3.11(a). All of the image sequences in Table 3.4 were used in this and the following analyses. For illustration, we plotted three representative image sequences at different scales (*i.e.,* Igloo, Memorial Church, and Belgium House) in the graph. We fixed $\overline{\sigma}_w = 0.1, \gamma = 1.0$ in this analysis. The horizontal axis represents the block width η and the vertical axis represents the computational time in seconds. Time used in initialization is excluded because the acceleration does not affect this step. Computational time decreases as η increases. However, significant efficiency improvement is only observed when $\eta \leqslant 5$.

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

Figure 3.12: Errors introduced by acceleration on the Belgium House image sequence. The result generated with $\eta = 1$ is used as the reference image. The error increases when η increases, as shown in the color-coded error maps (warmer colors indicate larger errors). However, the total error is still below 9% even when $\eta = 10$. (a) $\eta = 1$. (b) $\eta = 2$. (c) $\eta = 4$. (d) $\eta = 5$. (e) $\eta = 6$. (f) $\eta = 10$.

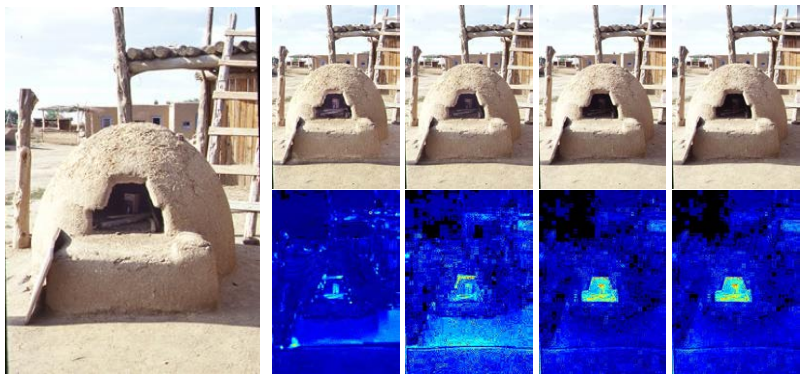
Errors introduced by acceleration are shown against block width η in Figure 3.11(b). The error in a pixel p_i^* is calculated using Euclidean distance between a pixel in a resulting image with $\eta > 1$ and the corresponding pixel in the reference image with $\eta = 1$, *i.e.,* $E(p_i^*) = \|p_i^* - p_i^{ref}\|$. The pixel values are normalized between $[0, 1]$. The total error in a resulting image is measured using the root mean square error (RMSE), *i.e.,* $E(\mathbf{I}^*) = \sqrt{\frac{1}{N}\sum_i E(p_i^*)^2}$. The error increases as the block size η increases. However, a significant error increase only occurs when $\eta \leqslant 5$. Even when $\eta = 10$, the total error is still below 9%. Some results on the Belgium House image sequence with different η's and their corresponding color-coded error maps are shown in Figure 3.12. In an error map, warmer colors indicate larger errors. The error increases as η increases. In order to balance the speed and error, we tested different values and found that using a block width $\eta \in [2, 5]$ generates reasonably good results.



(a) (b)

Figure 3.13: Sensitivity analysis of the free parameter $\overline{\sigma}_w$. The error increases dramatically as $\overline{\sigma}_w$ becomes too small. It increases slowly when $\overline{\sigma}_w \geqslant 0.4$ and converges after $\overline{\sigma}_w = 1.0$. (a) RMSE at different $\overline{\sigma}_w$ values. (b) Changes of RMSE relative to $\overline{\sigma}_w$.

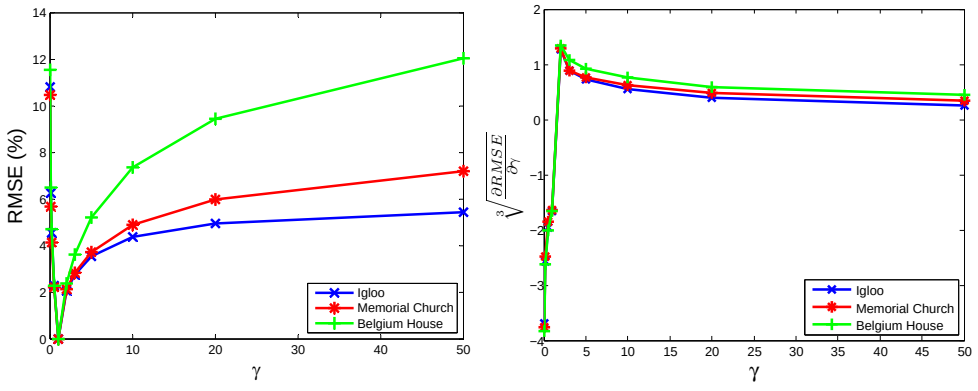
In the analysis of $\overline{\sigma}_w$, we fixed $\gamma = 1.0, \eta = 4$. The error is defined similarly as in the analysis of η, and the results with $\overline{\sigma}_w = 0.1$ are used as reference images. The analysis on three representative image sequences is plotted in Figure 3.13. Some results on the Igloo sequence with different



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

Figure 3.14: Sensitivity analysis of the free parameter $\overline{\sigma}_w$ on the Igloo image sequence. The result generated with $\overline{\sigma}_w = 0.1$ is used as the reference image. When $\overline{\sigma}_w$ decreases, the image gets brighter, and therefore more information is revealed in the under-exposed regions. However, if $\overline{\sigma}_w$ becomes too small, adjacent pixels with large difference may be ignored, which produces artifacts at object boundaries. (a) $\overline{\sigma}_w = 0.1$. (b) $\overline{\sigma}_w = 0.01$. (c) $\overline{\sigma}_w = 0.05$. (d) $\overline{\sigma}_w = 0.5$. (e) $\overline{\sigma}_w = 2.0$. (Source sequence courtesy of Shree Nayar.)

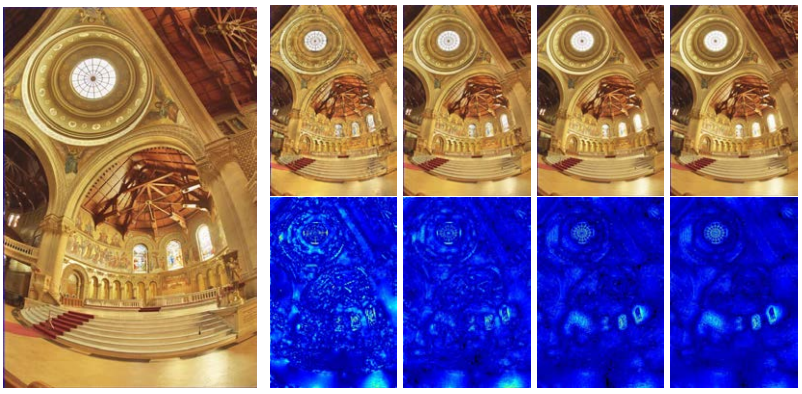
$\overline{\sigma}_w$'s and their corresponding color-coded error maps are shown in Figure 3.14. The error increases dramatically as $\overline{\sigma}_w$ becomes too small. It increases slowly when $\overline{\sigma}_w \geqslant 0.4$ and converges after $\overline{\sigma}_w = 1.0$. When $\overline{\sigma}_w$ decreases, the image gets brighter, and more information is therefore revealed in the under-exposed regions. However, if $\overline{\sigma}_w$ becomes too small, adjacent pixels with large difference may be ignored, which leads to artifacts at object boundaries. Therefore, we suggest using $\sigma_w \in [0.05, 0.4]$.



(a) (b)

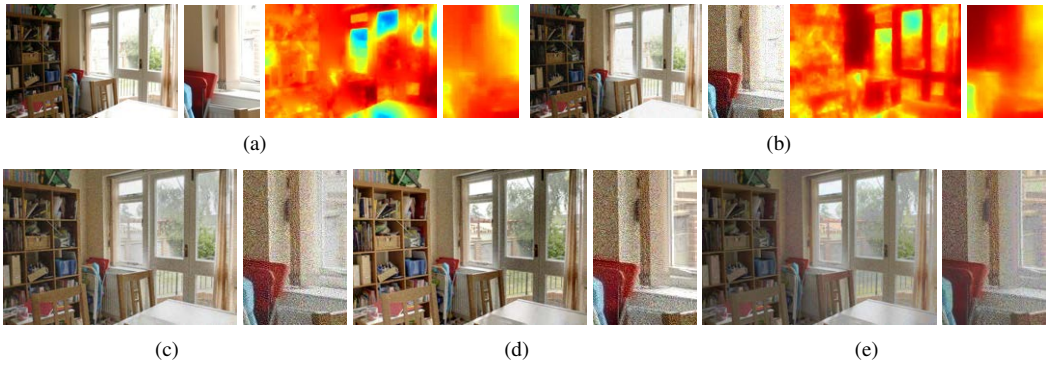
Figure 3.15: Sensitivity analysis of the free parameter γ. The error increases dramatically as γ becomes too small, but increases slowly when $\gamma \geqslant 5.0$. (a) RMSE at different γ values. (b) Changes of RMSE relative to γ.

In the analysis of γ, we fix $\overline{\sigma}_w = 0.1, \eta = 4$. The error is defined similarly as in the analysis of η and the results with $\gamma = 1.0$ are used as reference images. The analysis on three representative image sequences is plotted in Figure 3.15. Some results on the Memorial Church sequence with different



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

Figure 3.16: Sensitivity analysis of the free parameter γ on the Memorial Church image sequence. The result generated with $\gamma = 1.0$ is used as the reference image. When γ decreases, more detail is revealed in the over-exposed regions. However, if γ is too small, artifacts may occur at object boundaries. (a) $\gamma = 1.0$. (b) $\gamma = 0.01$. (c) $\gamma = 0.2$. (d) $\gamma = 5.0$. (e) $\gamma = 10.0$. (Source sequence courtesy of Paul Debevec.)



(a) (b)

(c) (d) (e)

Figure 3.17: Analysis of our algorithm's sensitivity to Gaussian white noise and comparison with other algorithms using the House image sequence. A closeup view of regions near the window is placed beside each image to give a clearer view of the effect of noise. As the Gaussian noise variance increases, errors introduced by the noise become more obvious. Our algorithm, EF, and PTR are all affected by the noise in the input image. (a) Original image (left) and its corresponding color-coded probability map (right) calculated using the proposed method (warmer colors represent higher probabilities). (b) Gaussian noise-corrupted image with 0 mean and 0.01 variance (left) and its corresponding color-coded probability map (right) calculated using the proposed method. (c) Fused image generated by the proposed method. (d) Fused image generated by EF. (e) Tone-mapped image generated by PTR.

γ's and their corresponding color-coded error maps are shown in Figure 3.16. The error increases dramatically as γ becomes too small, but increases slowly when $\gamma \geqslant 5.0$. When γ decreases, more detail is revealed in the over-exposed regions. However, if γ is too small, as in the analysis of $\overline{\sigma}_w$, artifacts may occur at object boundaries. Therefore, we suggest using $\gamma \in [0.2, 5]$. More objective and subjective evaluations can be found in Section 4.2.8.

(a) (b) (c)

Figure 3.18: Fusion result using our algorithm after adding a denoising step. The probability map calculated for this denoised image and the fused image are close to the ones computed from the clean input images. However, the object boundaries are a little blurred during the denoising process, which also affects the fusion result. (a) Denoised image. (b) Color-coded probability map. (c) Fused image.

3.3.7.4 Effect of Noise

One limitation of our algorithm is that it is sensitive to noise in the input image sequence. One example is given in Figure 3.17. White Gaussian noise with 0 mean and variance from 0.001 to 0.01 with increments of 0.001 is added to one of the four input images (pixel values are scaled to the range $[0, 1]$). For the sake of brevity, only the corrupted image with variance 0.01 is shown in Figure 3.17, along with the original image. In our algorithm, the initial probabilities are determined by local variation, which is sensitive to white Gaussian noise. Since a global optimization scheme is used afterwards, the error caused by the noise tends to propagate to a larger neighborhood. The color-coded probability maps are given in Figure 3.17(a)(b), where warmer colors represent higher probabilities. Compared to the probability map generated for the original image, higher probabilities are assigned to pixels in the corrupted image, especially in textureless regions such as the walls. Even if we use the correctly computed probability maps, pixels from the noisy images still contribute to the fused image when Equation (3.16) is used. Therefore, the noisy images have a significant influence on the fused images, as shown in Figure 3.17(c). A closeup view of regions near the window is placed beside each image to give a clearer view of the effect of noise. As the Gaussian noise variance increases, errors introduced by the noise become more obvious. However, the results by EF are also affected by the noise, as shown in Figure 3.17(d). In addition, the noise in an input image also affects the HDR reconstruction process, which results in a noisy HDR image. The tone mapping process fails to correct the noise in this HDR image and the noise remains in the tone-mapped image. PTR [156], SC [113], and LW [167] generate similarly affected results. Therefore, only the result of PTR is given in Figure 3.17(e).

One solution to this problem is to add a pre-processing step to reduce the noise in the input images. One example is given in Figure 3.18. The input sequence is the House image sequence, where one image is corrupted with white Gaussian noise with 0 mean and 0.01 variance, as shown in Figure 3.17(b). Figure 3.18(a) gives the denoised image using the method proposed by Portilla *et al.* [150]. The noisy image is first decomposed into subbands using the steerable pyramid transform [170], and then Bayesian least square estimation and Gaussian scale mixture model are

employed to denoise each subband. The probability map calculated for this denoised image and the fused image are given in Figure 3.18(b) and Figure 3.18(c), respectively. They are close to the ones computed from the clean input images (see Figure 3.17(a) and Figure 3.10(b)). However, the object boundaries are a little blurred during the denoising process, which also affects the fusion result.

3.4 Multivariate Gaussian Conditional Random Fields

In this section, we introduce the (pairwise) multivariate Gaussian conditional random field (MGCRF) model based on the multivariate GMRF [160] and analyze its relationship with GRW. MGCRF is a more general model for multi-label problems than GRW, although more sophisticated (*i.e.,* multi-dimensional) distance metrics may be required and higher computational cost is expected.

3.4.1 Multivariate Gaussian Markov Random Fields

A multivariate GMRF (MGMRF) is a multivariate extension of a GMRF. Let $\tilde{\mathcal{X}} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ denote a set of variables, where $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,K})^T$ is a random vector. Let $\tilde{\mathbf{x}}$ be the vector form of $\tilde{\mathcal{X}}$, *i.e.,* $\tilde{\mathbf{x}} = ({\mathbf{x}_1}^T, {\mathbf{x}_2}^T, \ldots, {\mathbf{x}_N}^T)^T$. Then, according to [160], $\tilde{\mathcal{X}}$ is called an MGMRF with respect to a graph $\mathcal{G} = (\tilde{\mathcal{X}}, \mathcal{E})$ with mean $\tilde{\mu}$ and precision matrix $\tilde{\mathbf{Q}}$, if and only if its density has the form:[13]

$$\begin{aligned} p(\tilde{\mathbf{x}}) &= (2\pi)^{-\mathrm{rk}(\tilde{\mathbf{Q}})/2} |\tilde{\mathbf{Q}}|^{1/2} \exp(-\frac{1}{2}(\tilde{\mathbf{x}} - \tilde{\mu})^T \tilde{\mathbf{Q}} (\tilde{\mathbf{x}} - \tilde{\mu})) \\ &\propto \exp[-\frac{1}{2}(\sum_{i=1}^{N} (\mathbf{x}_i - \tilde{\mu}_i)^T \tilde{\mathbf{Q}}_{ii} (\mathbf{x}_i - \tilde{\mu}_i) + \frac{1}{2} \sum_{e_{ij} \in \mathcal{E}} (\mathbf{x}_i - \tilde{\mu}_i)^T \tilde{\mathbf{Q}}_{ij} (\mathbf{x}_j - \tilde{\mu}_j))]. \end{aligned} \tag{3.19}$$

and

$$\tilde{\mathbf{Q}}_{ij} \neq \mathbf{0} \Leftrightarrow e_{ij} \in \mathcal{E}, \forall i \neq j, \tag{3.20}$$

where $\mathrm{rk}(\cdot)$ denotes the rank of a matrix, and each $\tilde{\mathbf{Q}}_{ij}$ is a $K \times K$ submatrix of $\tilde{\mathbf{Q}}$. When $K = 1$, an MGMRF degenerates to a GMRF.

3.4.2 Multivariate Gaussian Conditional Random Fields and GRW

Like an MGMRF, $(\tilde{\mathcal{X}}, \mathcal{D})$ is considered as an MGCRF with respect to a graph $\mathcal{G} = (\tilde{\mathcal{X}}, \mathcal{E})$, if the conditional density $p(\tilde{\mathbf{x}}|\mathcal{D})$ satisfies Equations (3.19) and (3.20). Now, let us consider a graph of the form in Figure 3.1. In addition to the variable set $\tilde{\mathcal{X}}$, there is a boundary set $\tilde{\mathcal{B}} = \{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_K\}$, where $\mathbf{b}_k = (b_{k,1}, b_{k,2}, \ldots, b_{k,K})^T$ is a known vector. The boundary set $\tilde{\mathcal{B}}$ defines the K-dimensional (K-D) bounding box for the potentials. Let $\tilde{\mathbf{b}}$ be the vector form of $\tilde{\mathcal{B}}$, *i.e.,* $\tilde{\mathbf{b}} = ({\mathbf{b}_1}^T, {\mathbf{b}_2}^T, \ldots, {\mathbf{b}_K}^T)^T$. The graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is constructed as $\mathcal{V} = \tilde{\mathcal{X}} \cup \tilde{\mathcal{B}}$ and $\mathcal{E} = \mathcal{E}_{\tilde{\mathcal{X}}} \cup \mathcal{E}_{\tilde{\mathcal{B}}}$, where $\mathcal{E}_{\tilde{\mathcal{X}}}$ is the set of edges within the variable set and $\mathcal{E}_{\tilde{\mathcal{B}}}$ is the set of edges between the variable and the boundary sets. Let us

[13] In [160], it also requires $\tilde{\mathbf{Q}}$ to be positive definite, but here we do not impose this requirement to distinguish between an MGMRF and an intrinsic MGMRF, whose $\tilde{\mathbf{Q}}$ is positive semi-definite.

denote the maximum and minimum allowable potentials in the k-th dimension as $U_{\max,k}$ and $U_{\min,k}$, respectively. Then, $\mathbf{b}_k = (U_{\min,1}, \ldots, U_{\max,k}, \ldots, U_{\min,K})^T$. We assume that the K-D potential field $\tilde{\mathcal{X}}$ is piecewise smooth and the local difference follows a multivariate Gaussian distribution with zero mean and precision matrix $\tilde{\mathbf{S}}_{ij}$:

$$\Delta_{ij}^{\tilde{\mathcal{X}}} \triangleq \mathbf{x}_i - \mathbf{x}_j \sim N(\mathbf{0}, \tilde{\mathbf{S}}_{ij}^{-1}), \forall e_{ij} \in \mathcal{E}_{\tilde{\mathcal{X}}}. \tag{3.21}$$

We also assume the difference between a variable $\mathbf{x}_i$ and a boundary potential $\mathbf{b}_k$ follows a multivariate Gaussian distribution with zero mean and precision matrix $\tilde{\mathbf{T}}_{ik}$:

$$\Delta_{ik}^{\tilde{\mathcal{B}}} \triangleq \mathbf{x}_i - \mathbf{b}_k \sim N(\mathbf{0}, \tilde{\mathbf{T}}_{ik}^{-1}), \forall e_{i,k} \in \mathcal{E}_{\tilde{\mathcal{B}}}. \tag{3.22}$$

Given the above assumptions, the conditional density $p(\tilde{\mathbf{x}}|\tilde{\mathcal{B}}, \mathcal{D})$ follows a multivariate Gaussian distribution[14]:

$$\begin{aligned} p(\tilde{\mathbf{x}}|\tilde{\mathcal{B}}, \mathcal{D}) &\propto \exp(-\frac{1}{2}(\sum_{k=1}^{K}\sum_{i=1}^{N} {\Delta_{ik}^{\tilde{\mathcal{B}}}}^T \tilde{\mathbf{T}}_{ik} \Delta_{ik}^{\tilde{\mathcal{B}}} + \frac{1}{2}\sum_{e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}} {\Delta_{ij}^{\tilde{\mathcal{X}}}}^T \tilde{\mathbf{S}}_{ij} \Delta_{ij}^{\tilde{\mathcal{X}}})) \\ &\propto \exp(-\frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{Z}}\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T\tilde{\mathbf{R}}\tilde{\mathbf{b}}), \end{aligned} \tag{3.23}$$

where $\tilde{\mathbf{Z}}$ is an $NK \times NK$ matrix with each $K \times K$ element defined as:

$$\tilde{\mathbf{Z}}_{ij} = \begin{cases} \sum_{k=1}^{K} \tilde{\mathbf{T}}_{ik} + \sum_{\mathbf{x}_m \in \mathcal{N}_i} \tilde{\mathbf{S}}_{im}, & i = j; \\ -\tilde{\mathbf{S}}_{ij}, & e_{ij} \in \mathcal{E}_{\tilde{\mathcal{X}}}; \\ 0, & \text{otherwise.} \end{cases} \tag{3.24}$$

and $\tilde{\mathbf{R}}$ is an $NK \times KK$ matrix with each $K \times K$ element defined as:

$$\tilde{\mathbf{R}}_{ik} = -\tilde{\mathbf{T}}_{ik}. \tag{3.25}$$

Then, the MAP configuration $\tilde{\mathbf{f}}^*$ is:

$$\begin{aligned} \tilde{\mathbf{f}}^* &= \arg\max_{\tilde{\mathbf{f}}} \exp(-\frac{1}{2}\tilde{\mathbf{f}}^T\tilde{\mathbf{Z}}\tilde{\mathbf{f}} - \tilde{\mathbf{f}}^T\tilde{\mathbf{R}}\tilde{\mathbf{b}}) \\ &= \arg\min_{\tilde{\mathbf{f}}} \frac{1}{2}\tilde{\mathbf{f}}^T\tilde{\mathbf{Z}}\tilde{\mathbf{f}} + \tilde{\mathbf{f}}^T\tilde{\mathbf{R}}\tilde{\mathbf{b}}. \end{aligned} \tag{3.26}$$

This quadratic energy minimization problem is equivalent to solving the following linear system:

$$\tilde{\mathbf{Z}}\tilde{\mathbf{f}}^* = -\tilde{\mathbf{R}}\tilde{\mathbf{b}}. \tag{3.27}$$

Now, if we assume that $\tilde{\mathbf{S}}_{ij}$ and $\tilde{\mathbf{T}}_{ik}$ are diagonal matrices, which indicates that each dimension is independent, then estimating the K-D potential field $\tilde{\mathcal{X}}$ can be done in each dimension separately. The (m, n)-th entry in $\tilde{\mathbf{S}}_{ij}$ is defined as:

$$\tilde{\mathbf{S}}_{ij,mn} = \begin{cases} W_{ij,m}, & m = n; \\ 0, & \text{otherwise.} \end{cases} \tag{3.28}$$

[14]Please refer to Appendix C for the detailed derivation of the following equation.

The (m, n)-th entry in $\tilde{\mathbf{T}}_{ik}$ is defined as:

$$\tilde{\mathbf{T}}_{ik,mn} = \begin{cases} Y_{ik,m}, & m = n; \\ 0, & \text{otherwise.} \end{cases} \tag{3.29}$$

In this case, Equation (3.27) can be rewritten as:

$$\mathbf{Z}^k \mathbf{x}^k = -\mathbf{R}^k \mathbf{b}^k, \forall k, \tag{3.30}$$

where $\mathbf{Z}^k$ is an $N \times N$ matrix; $\mathbf{R}^k$ is an $N \times K$ matrix; and $\mathbf{x}^k = (x_{1,k}, \ldots, x_{N,k})^T$ and $\mathbf{b}^k = (b_{1,k}, \ldots, b_{k,k}, \ldots, b_{K,k})^T = (U_{\min,k}, \ldots, U_{\max,k}, \ldots, U_{\min,k})^T$ are column vectors. The (i, j)-th entry in $\mathbf{Z}^k$ equals the $(Ki - K + k, Kj - K + k)$-th entry in $\tilde{\mathbf{Z}}$, *i.e.,*

$$\mathbf{Z}^k_{ij} = \begin{cases} \sum_{m=1}^{K} Y_{im,k} + \sum_{\mathbf{x}_m \in \mathcal{N}_i} W_{im,k}, & i = j; \\ -W_{ij,k}, & e_{ij} \in \mathcal{E}_{\tilde{\mathcal{X}}}; \\ 0, & \text{otherwise.} \end{cases} \tag{3.31}$$

The (i, j)-th entry in $\mathbf{R}^k$ equals the $(Ki - K + k, Kj - K + k)$-th entry in $\tilde{\mathbf{R}}$, *i.e.,*

$$\mathbf{R}^k_{ij} = -Y_{ij,k}. \tag{3.32}$$

Now, let us consider a more restrictive case, where $\tilde{\mathbf{T}}_{ik}$ and $\tilde{\mathbf{S}}_{ij}$ are identity matrices subject to individual scaling factors, $\gamma_1 Y_{ik}$ and $\gamma_2 W_{ij}$, *i.e.,*

$$\tilde{\mathbf{T}}_{ik} = \gamma_1 Y_{ik} \mathbf{I}, \ \tilde{\mathbf{S}}_{ij} = \gamma_2 W_{ij} \mathbf{I}, \tag{3.33}$$

where $\mathbf{I}$ represents a $K \times K$ identity matrix; γ_1 and γ_2 are data-independent scaling factors; and Y_{ik} and W_{ij} are data-dependent scaling factors. Then, Equation (3.30) becomes:

$$\mathbf{ZX} = -\mathbf{RB}. \tag{3.34}$$

Here, $\mathbf{Z}$ is an $N \times N$ symmetric positive definite matrix defined as:

$$\mathbf{Z}_{ij} = \begin{cases} \gamma_1 \sum_{k=1}^{K} Y_{ik} + \gamma_2 \sum_{\mathbf{x}_m \in \mathcal{N}_i} W_{im}, & i = j; \\ -\gamma_2 W_{ij}, & e_{ij} \in \mathcal{E}_{\tilde{\mathcal{X}}}; \\ 0, & \text{otherwise.} \end{cases} \tag{3.35}$$

$\mathbf{R}$ is an $N \times K$ matrix defined as:

$$\mathbf{R}_{ik} = -Y_{ik}. \tag{3.36}$$

$\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^T$ is an $N \times K$ matrix, and $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_K)^T$ is a $K \times K$ symmetric matrix:

$$\mathbf{B}_{ij} = \begin{cases} U_{\max,i}, & i = j; \\ U_{\min,j}, & i < j; \\ U_{\min,i}, & i > j. \end{cases} \tag{3.37}$$

When $U_{\max,k} = \gamma_3, \forall k$ and $U_{\min,k} = 0, \forall k$, it is easy to see that an MGCRF under the constraint in Equation (3.33) is equivalent to the GRW model (Section 3.1) in terms of steady-state probability calculation.

3.5 Summary

In this chapter, we proposed a generalized random walks framework, which converts a labeling problem into steady-state probability calculation in the random walks context. Two applications of GRW were introduced: stereo correspondence and multi-exposure fusion. With basic similarity measure and disparity refinement scheme, GRW is able to produce reasonable results on standard stereo test images. Experiments also demonstrated the effectiveness of GRW in stereo correspondence compared with other global stereo correspondence methods (*i.e.,* BP-based and GC-based). A new formulation of multi-exposure fusion in the random walks context as a probabilistic composition process was also proposed. Two quality measures were considered: local variation and color consistency. Unlike previous fusion methods, our algorithm achieves an optimal balance between the two measures via GRW. Experimental results demonstrated that our probabilistic fusion produces good results, in which contrast is enhanced and details are preserved with high computational efficiency. Compared to other fusion methods and tone mapping methods, our algorithm produces images with comparable or even better qualities. In addition, we introduced the multivariate Gaussian conditional random field model and analyzed its relationship with GRW.

Chapter 4

The Hierarchical Random Walks Framework and Its Application

4.1 Hierarchical Random Walks

In GRW, the probability maps are calculated by directly solving the linear systems. However, a direct solver is computationally expensive, which restricts GRW from being applied to images in high resolution or data of higher dimensions. Therefore, in this section, we propose a hierarchical random walks (HRW) model, which provides a good approximation to the solution obtained from a direct solver in GRW but with improved efficiency and memory usage. Hierarchical methods work on a pyramidal representation of the data and aim to solve the problem in a coarse-to-fine fashion, which normally has lower temporal and spatial complexity. Because of their computational efficiency, hierarchical methods have gained popularity in many applications [55, 70]. Although hierarchical RW has been examined in other studies such as [45], their focus was on the behavior of a random walker traveling in high dimensions, *e.g.,* moving between several two-dimensional (2D) planes. The HRW introduced here differs in that the hierarchy is used to facilitate the calculation of the probabilities at the finest level and that a random walker does not transit between levels. The basic idea of HRW is to use the solution obtained on a coarser-level graph to guide the probability calculation on a finer-level graph.

The basic procedure of HRW is illustrated in Figure 4.1. For the sake of brevity, only a 1D 2-level case is depicted. Let $\mathcal{X}^n$ denote the variable set at level n, where $n \in [0, N_h - 1]$, $\mathcal{X}^0 = \mathcal{X}$, and $\mathcal{X}^n = [\mathcal{X}^{n-1}]^{\downarrow 2}$. A weighted undirected graph is built for each level from finest to coarsest to represent the relationship within and between the variable set $\mathcal{X}^n$ (scene nodes) and the label set $\mathcal{L}$ (label nodes). In Figure 4.1, yellow and orange nodes denote scene nodes at level 0 and level 1, respectively; and red nodes denote label nodes. $x_i^n \in \mathcal{X}^n$ denotes the i-th scene node at level n. Y_{ik}^n denotes the data compatibility that measures the similarity between x_i^n at level n and l_k. W_{ij}^n denotes the neighborhood compatibility between adjacent scene nodes x_i^n and x_j^n at level n. A random walker starting from any scene node transits freely within the graph until it reaches a label

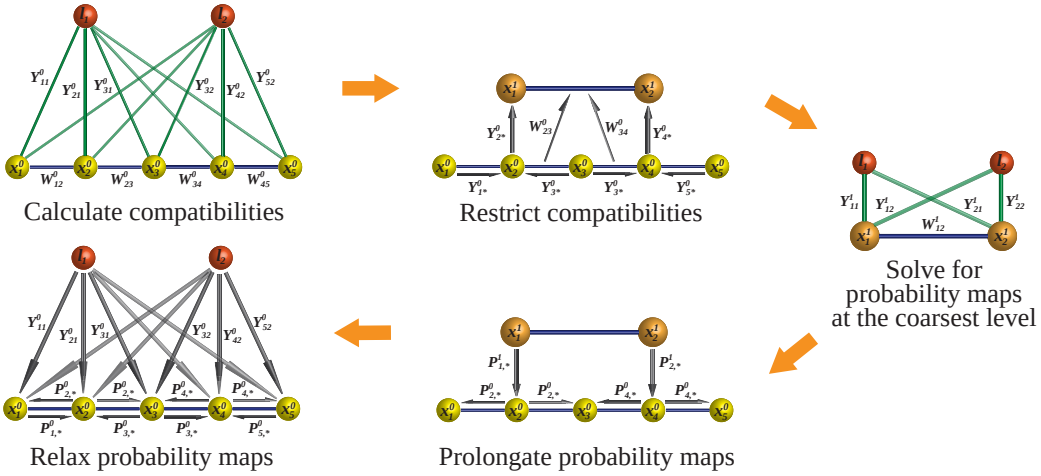


Figure 4.1: The basic procedure of HRW in a 1D 2-level case.

node, which concludes its walk. Let $P^t(l_k|x_i^n, \mathcal{D})$ denote the t-step transition probability determined based on observed data $\mathcal{D}$ that a random walker starting from location x_i^n reaches the label node l_k by the t-th step. The goal is to calculate the steady-state probability $P_{i,k}^0 \triangleq \lim_{t\to\infty} P^t(l_k|x_i^0, \mathcal{D})$ for all $x_i^0 \in \mathcal{X}^0$ at level 0 and all $l_k \in \mathcal{L}$.

The data and neighborhood compatibilities, which are problem dependent, are defined at the finest level. Then, they are progressively restricted to a coarser/higher level until the coarsest/highest level is reached, where the linear systems are constructed following GRW[1]. These linear systems have only a fraction of the amount of unknown variables at the finest level; therefore, the solution can be computed exactly and efficiently using a direct solver. Finally, the solution from a coarser level is prolongated/interpolated to a finer/lower level and then relaxed at the finer level until the finest/lowest level is reached. Because a coarser-level solution serves as a good initialization for a finer level, it requires less time to converge. These operations for the 2D case are described below.

4.1.1 Restriction

The restriction operators are defined in a way that preserves boundaries between nearby scene nodes in order to avoid over-smoothed solutions. The restriction operator for data compatibilities is defined similarly to the restriction operator for variables in [72]. Suppose x_i^n at level n is the projection of a higher-level node x_s^{n+1}, and $\mathcal{N}_i^n$ is the first-order neighborhood of x_i^n. Then, Y_{sk}^{n+1} is determined as a weighted average of the data compatibilities of $\mathcal{N}_i^n$ and x_i^n, where the weights are defined on the neighborhood compatibilities W_{ij}^n's between x_i^n and $\forall x_j^n \in \mathcal{N}_i^n$. This operator in the 2D case is illustrated in Figure 4.2. Four-connectivity is assumed. For any level $n \in [0, N_h - 2]$, we group the nodes according to their x- and y-coordinates. Let $\mathcal{X}_O^n$, $\mathcal{X}_E^n$, and $\mathcal{X}_B^n$ denote the sets containing nodes with only odd coordinates (yellow nodes), only even coordinates (orange

[1]Here, we borrow the terms *restriction* and *prolongation* from multigrid.

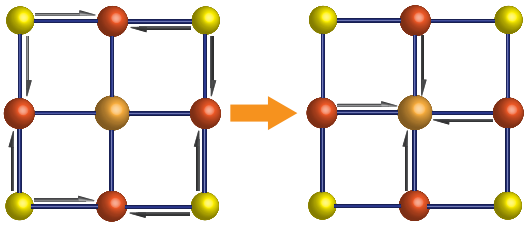


Figure 4.2: Restriction operator for data compatibilities.

nodes), and one odd and one even coordinate (red nodes), respectively. First, Y_{ik}^n's of all scene nodes in $\mathcal{X}_B^n$ are updated with contributions from their respective neighboring scene nodes that are in $\mathcal{X}_O^n$. Then, Y_{ik}^n's of all scene nodes in $\mathcal{X}_E^n$ are updated with contributions from their respective neighboring scene nodes, all of which are in $\mathcal{X}_B^n$. Following our hierarchy construction scheme, we have $\mathcal{X}^{n+1} = [\mathcal{X}^n]^{\downarrow 2} = \mathcal{X}_E^n$. Let x_i^n denote the projection of x_s^{n+1} at level n, and $\mathcal{N}_i^n$ the first-order neighborhood of x_i^n. Mathematically, the restriction operator for data compatibilities can be defined as in Equations (4.1) and (4.2), where the data compatibility of a scene node at a higher level is determined as a weighted average of the data compatibilities of its corresponding nodes at a lower level.

$$Y_{ik}^n = Y_{ik}^n + \frac{\sum_{x_j^n \in \mathcal{N}_i^n \cap \mathcal{X}_O^n} W_{ij}^n Y_{jk}^n}{\sum_{x_j^n \in \mathcal{N}_i^n \cap \mathcal{X}_O^n} W_{ij}^n}, \ \forall x_i^n \in \mathcal{X}_B^n \tag{4.1}$$

$$Y_{sk}^{n+1} = Y_{ik}^n + \frac{\sum_{x_j^n \in \mathcal{N}_i^n} W_{ij}^n Y_{jk}^n}{\sum_{x_j^n \in \mathcal{N}_i^n} W_{ij}^n}, \ \forall x_i^n \in \mathcal{X}_E^n \tag{4.2}$$

Let x_s^{n+1} and x_t^{n+1} denote two adjacent scene nodes at level $n+1$, and x_i^n and x_j^n their projections at level n. Then, the restriction operator for neighborhood compatibilities is defined as in Equation (4.3), where W_{st}^{n+1} is calculated as the minimum of the neighborhood compatibilities at a lower level that are defined in the common neighborhood $\mathcal{N}_i^n \cap \mathcal{N}_j^n$ of x_i^n and x_j^n.

$$W_{st}^{n+1} = \min\{W_{ih}^n, W_{jh}^n | x_h^n \in \mathcal{N}_i^n \cap \mathcal{N}_j^n\}. \tag{4.3}$$

4.1.2 Coarsest-Level Solution

The coarsest-level solution can be computed the same way as in GRW (Section 3.1) by minimizing a quadratic energy function, which is equivalent to solving the following equation for $\mathbf{P}_k^{N_h-1}$ for each label l_k:

$$\mathbf{L}_{\mathcal{X}}^{N_h-1} \mathbf{P}_k^{N_h-1} = \mathbf{Y}_k^{N_h-1}, \tag{4.4}$$

where $\mathbf{P}_k^{N_h-1} = (P_{1,k}^{N_h-1}, \ldots, P_{Q,k}^{N_h-1})^T$ and $\mathbf{Y}_k^{N_h-1} = (Y_{1k}^{N_h-1}, \ldots, Y_{Qk}^{N_h-1})^T$. Here, $\cdot^T$ denotes the transpose operator and Q is the number of coefficients at level $n = N_h - 1$. $\mathbf{L}_{\mathcal{X}}^{N_h-1}$ is a $Q \times Q$ matrix with the (i,j)-th entry defined as:

$$\mathbf{L}_{ij}^n = \begin{cases} \gamma \sum_{x_h^n \in \mathcal{N}_i^n} W_{ih}^n + \sum_k Y_{ik}^n, & i = j; \\ -\gamma W_{ij}^n, & x_j^n \in \mathcal{N}_i^n; \\ 0, & \text{otherwise.} \end{cases} \tag{4.5}$$

where γ is a weighting coefficient balancing the contributions from label nodes and neighboring scene nodes. A larger γ places more emphasis on neighborhood consistency/smoothness. In GRW, there are three weighting coefficients: $\gamma_1, \gamma_2, \gamma_3$. Here, we take $\gamma = \gamma_2/\gamma_1$, and γ_3 is not needed because it appears on both sides of Equation (4.4)[2].

4.1.3 Prolongation

The prolongation operator is defined as the inverse operation of restriction except that it is applied to the probability maps instead of the data compatibilities. First, the coarser-level probability maps $\mathbf{P}_k^{n+1}$'s are injected onto $\mathcal{X}_E^n$ at the finer level as in Equation (4.6). x_i^n denotes the projection of x_s^{n+1} at level n. Then, as in Equation (4.7), $P_{i,k}^n$'s for every scene node in $\mathcal{X}_B^n$ are determined based on contributions from their respective neighboring scene nodes that are in $\mathcal{X}_E^n$. Finally, $P_{i,k}^n$'s for every scene node in $\mathcal{X}_O^n$ are determined based on contributions from their respective neighboring scene nodes, all of which are in $\mathcal{X}_B^n$, as given in Equation (4.8).

$$P_{i,k}^n = P_{s,k}^{n+1}, \ \forall x_i^n \in \mathcal{X}_E^n \tag{4.6}$$

$$P_{i,k}^n = \frac{\sum_{x_j^n \in \mathcal{N}_i^n \cap \mathcal{X}_E^n} W_{ij}^n P_{j,k}^n}{\sum_{x_j^n \in \mathcal{N}_i^n \cap \mathcal{X}_E^n} W_{ij}^n}, \ \forall x_i^n \in \mathcal{X}_B^n \tag{4.7}$$

$$P_{i,k}^n = \frac{\sum_{x_j^n \in \mathcal{N}_i^n} W_{ij}^n P_{j,k}^n}{\sum_{x_j^n \in \mathcal{N}_i^n} W_{ij}^n}, \ \forall x_i^n \in \mathcal{X}_O^n \tag{4.8}$$

4.1.4 Relaxation

We use the red-black Gauss-Seidel relaxation [29] to update the prolongated probability maps for better approximations. $\mathcal{X}^n$ at level n forms a bipartite graph, in which every edge connects two nodes from two different sets. One set is $\mathcal{X}_B^n$, and the other is $\mathcal{X}_E^n \cup \mathcal{X}_O^n$. In this relaxation step, Equation (4.9) is alternately applied to the nodes in the two sets for a pre-defined number of iterations. In our implementation, we set the number of iterations as $n+1$ for level n.

$$P_{i,k}^n = \frac{1}{D_i^n}(Y_{ik}^n + \gamma \sum_{x_j^n \in \mathcal{N}_i^n} W_{ij}^n P_{j,k}^n), \tag{4.9}$$

where $D_i^n = \gamma \sum_{x_h^n \in \mathcal{N}_i^n} W_{ih}^n + \sum_k Y_{ik}^n$ is the degree of node x_i^n.

The basic procedure of HRW is summarized in Algorithm 4.1. The normalization in the last step is applied because the quick approximation by HRW may cause $\sum_k P_{i,k}^0 \neq 1$. $P_{i,k}^0$'s are normalized for every pixel location as $P_{i,k}^0 = P_{i,k}^0/(\sum_k P_{i,k}^0)$.

Compared with GRW, HRW requires less time and memory to generate the probability maps but with good accuracy. In Table 4.1, their performances are compared on the multi-exposure fusion problem using four representative source sequences of increasing sizes. The time and memory usage recorded in this table is only for the final probability calculation step, where GRW or HRW is

[2]The parameter γ can also be associated with Y_{ik}^n in the equation, which assumes $\gamma = \gamma_1/\gamma_2$. However, this makes no fundamental difference to our current setting.

Algorithm 4.1 Basic steps of the hierarchical random walks framework.

1: Define problem-specific data and neighborhood compatibilities at original resolution (level 0)
2: **for** $n = 1$ to $N_h - 1$ **do**
3: Restrict data compatibilities from level $n - 1$ to n using Equations (4.1) and (4.2)
4: Restrict neighborhood compatibilities from level $n - 1$ to n using Equation (4.3)
5: **end for**
6: Obtain solution at level $N_h - 1$ using GRW (Algorithm 3.1)
7: **for** $n = N_h - 2$ to 0 **do**
8: Prolongate the probability maps from level $n + 1$ to n using Equations (4.6) to (4.8)
9: Relax for $n + 1$ iterations using Equation (4.9)
10: **end for**
11: Normalize the probability maps at level 0

Table 4.1: Comparison between HRW and GRW

Source sequence	Size	Time (sec)		Memory (MB)		RMSE (%)
		HRW	GRW	HRW	GRW	
House	$752 \times 500 \times 4$	**1.176**	2.998	**43**	245	1.297
Chateau	$1500 \times 644 \times 5$	**3.995**	8.418	**152**	955	1.108
Belgium House	$1025 \times 769 \times 9$	**5.362**	7.449	**148**	773	0.800
Lamp	$1600 \times 1200 \times 15$	**23.08**	130.7	**757**	2386	0.904

applied, and all other settings remain the same. Both HRW and GRW were applied to the original resolution of the images. Five levels were used in HRW, *i.e.,* $N_h = 5$. On average, HRW takes about 44.081% of the time and 21.085% of the memory needed by GRW. The root mean squared error (RMSE) between the probability maps generated by HRW and those by GRW for each sequence is only around 1%, which indicates good approximations. The results on the Belgium House sequence are given in Figure 4.3 for visual comparison. Figure 4.3(a) shows two of the nine source images. The color-coded probability maps corresponding to the source image on the left are given on the left in each row with the fused images on the right. HRW produces probability maps close to those generated by GRW. Although there are subtle differences in the probability maps, there is no noticeable difference between the fusion results.

A multigrid algorithm was proposed in [72] also for solving the linear system that arises in the RW context. Later, a maximally connected neighbor coarsening operator was proposed in [70] for more stable performance. The key idea behind multigrid is to recursively restrict the residual of the current solution and the coefficient matrix to a coarser level and then use the correction obtained at the coarser level to update the current solution. In contrast, the proposed HRW only restricts the compatibilities to a coarser level and then prolongates the coarser-level solution to a finer level. This is a more efficient scheme; and as shown in Table 4.1 and Figure 4.3, HRW provides a good approximation to the exact solution obtained by a direct solver. In addition, while a general multigrid algorithm [29] is capable of solving different types of linear systems, HRW is more flexible in terms of solving a labeling problem. With different types of energy functions defined at the coarsest level, our hierarchical scheme may be applied to other formulations of labeling problems as well, *e.g.,*

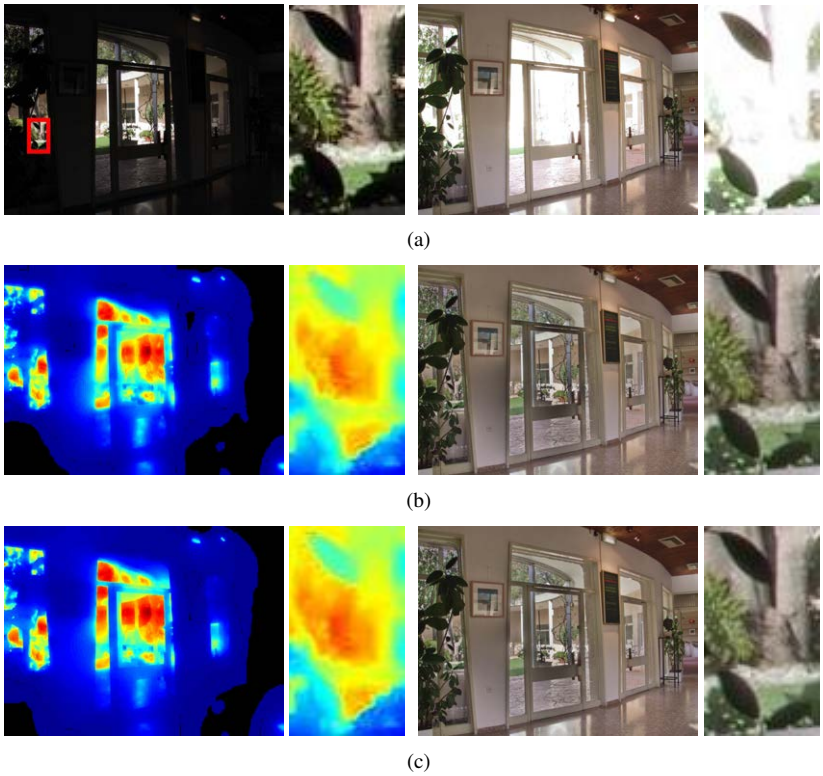


Figure 4.3: Comparison of GRW and HRW on the Belgium House sequence. The color-coded final probability maps corresponding to the source image on the left are given on the left in each row with the fused images on the right. HRW produces probability maps close to those generated by GRW, and there is no noticeable difference between the fusion results generated by these two algorithms. (a) Low-exposure (left) and high-exposure (right) source images. (b) Probability map (left) and fusion result (right) generated by GRW. (c) Probability map (left) and fusion result (right) generated by HRW. (Source sequence courtesy of Dani Lischinski.)

MRF modeling.

A multiscale decomposition (MSD) based solver was proposed in [115] for solving a similar linear system for image tone adjustment. The major differences between our HRW and this solver are: 1) no MSD is performed on the input images in our method, but instead a coarser Laplacian matrix is obtained by our restriction operator; 2) at each level other than the coarsest level in our hierarchy, the initial solution is obtained by our prolongation operator that respects node boundaries, which gives better initial approximation than upsampling in [115]; 3) at each level, relaxation is applied to refine the prolongated solution, which requires much less computation than the residual minimization performed in [115].

4.2 Application of HRW in Perception-Guided Image Fusion

4.2.1 Introduction

Human perceptual factors have attracted a growing amount of attention within the research on visual communication techniques [135, 114]. The rationale behind this trend is to appeal the human observer with high visual quality images, videos, graphics and animations. Therefore, it is important for researchers to take the human visual system [158] into consideration when designing image processing algorithms and evaluation metrics. For example, a unified color space providing maximum perceptual uniformity was developed for perception-based image processing [116]; visual stimulating factors for capturing human attention were employed for lossy compression of motion capture data [57]; human color perception and orientation sensitivity were utilized for color and texture feature extraction to achieve perceptually uniform image segmentation [37]; a quantitative metric based on geometry and texture resolutions was developed to approximate the perceptual quality of rendered images of three-dimensional (3D) objects [137]. Contrast and color are generally recognized to be important parameters [30, 176] in image quality. Motivated by these research findings, we study the visual impact of perceived local contrast and color saturation on fused images.

Multi-exposure fusion (MEF) is necessary because a conventional digital camera often produces images with insufficient details of a natural scene due to the incompatibility of the camera's limited/low dynamic range (LDR) and the high dynamic range (HDR) of the scene. As shown in Figure 4.4(a), none of the source images, which are captured under different exposures, is able to present all the details in the scene. The insets illustrate the loss of local details caused by under- or over-exposure, although the individual images combined contain complementary high quality details of the scene, as in Figure 4.4(b). This composition of local details can be achieved using MEF techniques (*e.g.,* [130]), which select different elements from separate source images to generate one single image based on certain quality computation. Differing from an alternative approach, which needs to adapt reconstructed HDR images for viewing on consumer displays by applying tone mapping (TM) methods to compress the high dynamic range [157], the MEF approach bypasses the HDR generation process.

Although some interactive fusion tools (*e.g.,* [3]) have been proposed, their focus is on interactive image editing tasks such as selective composition and relighting, and no specialized algorithm for MEF is incorporated. In this section, we propose a novel MEF method based on perceptual quality measures that exploit both contrast and color information. A transducer function and a psychometric function are applied to model the nonlinear human perception of local contrast. Such a perceptual modeling scheme allows us to effectively identify properly exposed pixels and thus generate fusion results of high visual quality. The contributions from the pixels of different source images to the fused image are then efficiently evaluated based on these quality measures using HRW. A fusion weighting function, which can be interactively controlled by the user, is introduced to generate

(a) (b)

Figure 4.4: Details from multi-exposure images in the Lamp image sequence are combined into a single image. Because of the high dynamic range of the scene, any single captured image suffers either over-exposure or under-exposure in some regions (*e.g.,* in the insets) and fails to present all of the details. Using our image fusion technique, details from different regions in each source image are transferred into one well-exposed image. (a) Four of the fifteen source images taken under different exposures. (b) Fused image. (Source sequence courtesy of Martin Čadík.)

different appearances of the fused image based on these evaluated pixel contributions.

4.2.2 Related Work

4.2.2.1 GRW-Based Multi-Exposure Fusion

A probabilistic fusion method was proposed in Section 3.3, in which the fusion problem was formulated as a labeling problem. A local variation measure, which was used to indicate local contrast, and a color consistency measure were considered. Here, we introduce two perceptual quality measures (*i.e.,* perceived local contrast and color saturation) instead of the local variation measure to give a more accurate evaluation of the pixel contributions, which results in higher-quality fused images. The perceived contrast measure was not considered in previous MEF methods. In addition, we propose a more flexible fusion model, where the fusion weights are defined as a function of the probabilities of the pixels in the fused image coming from each source image, and a practical form of this fusion weighting function is introduced.

4.2.2.2 Perception-Based Tone Mapping

Human perception has been exploited in some TM methods. Ferradans *et al.* [56] proposed a two-stage TM method. In the first stage, a global operator that models perceived brightness is applied. In the second stage, a local operator is applied to enhance local contrast and approximate color constancy. Kuang *et al.* [94] proposed iCAM06, an image appearance model, for HDR image rendering, which models perceived brightness and applies chromatic adaptation. Mantiuk *et al.* [127] proposed a global operator that minimizes the distortion of perceived contrast, which is modeled by a transducer function, between the original HDR image and the tone-mapped image under a specific viewing condition. We also model human perception, but we do not model perceived brightness; instead we consider perceived local contrasts and the probability of detecting such contrasts. We do not use the perceived contrasts to minimize visual distortions, but to maximize the reproduction of

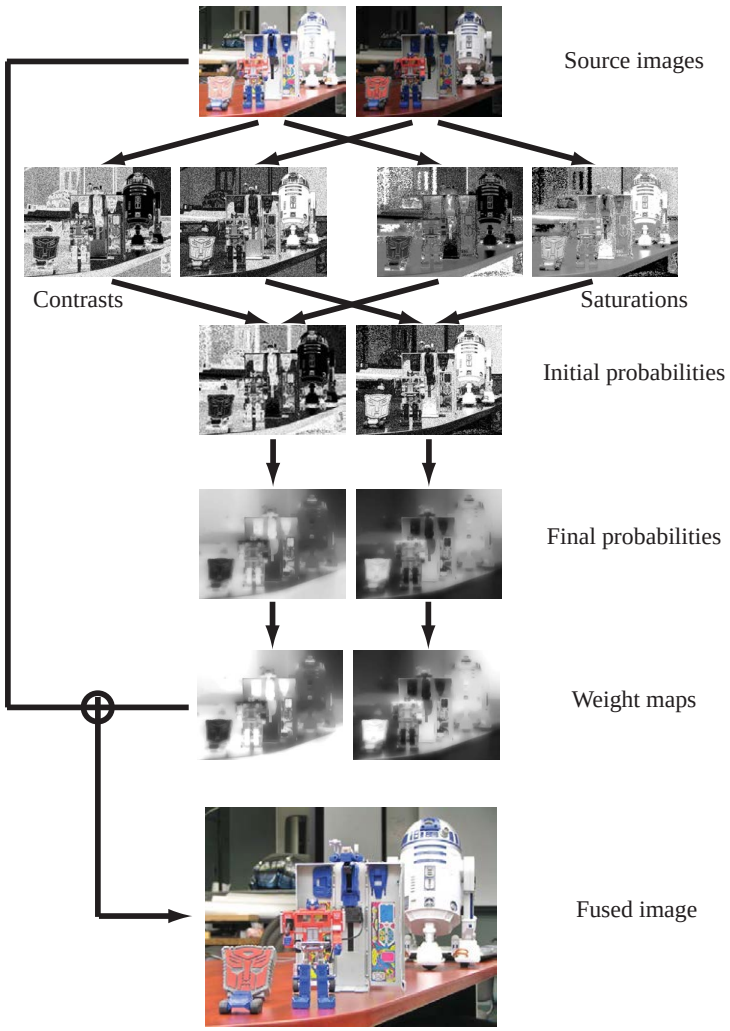


Figure 4.5: Processing procedure of the proposed fusion algorithm.

local contrasts and details. Moreover, we apply a color saturation measure to reproduce the colors of properly exposed objects. In addition, we impose color consistency over a neighborhood, rather than imposing color constancy over an entire color channel, which may reduce the colorfulness of the original scene.

4.2.3 Algorithm Overview

Figure 4.5 illustrates the basic steps in our fusion algorithm. Given a source image sequence, the initial probability that each pixel location in the fused image is assigned a label is determined from two locally-defined quality measures, *i.e.,* perceived local contrast (Section 4.2.4) and color saturation (Section 4.2.5). Contrasts are calculated for each pixel in the luminance channel. To be in accordance with the nonlinear human perception of contrast [62], these contrasts are then modified using a transducer function followed by a psychometric function. Saturations are calculated for each pixel

in the LHS (luminance, hue, and saturation) color space [176]. Contrasts and saturations are combined to determine the initial probabilities without considering neighborhood information. Then, the hierarchical random walks framework (Sections 4.1 and 4.2.7) is applied to efficiently calculate the final probabilities based on these two quality measures while taking neighborhood consistency into account. These probabilities are then used in a weighting function to generate the weight maps. Finally, the fused image is produced by combining the source images based on the weight maps.

4.2.4 Perceived Local Contrast

Local contrast represents the perception of local luminance variations with respect to the surrounding luminance [198]. Different measures of local contrast exist in the literature [198]. Simple definitions like Weber contrast and Michelson contrast normally assume small targets on a large uniform background [141, 198]. In order to deal with complex images of natural scenes in multi-exposure fusion, we modified the local band-limited contrast proposed by Peli [141], which defines local contrast as a function of the spatial frequency band. Although there are other sophisticated definitions of contrast, *e.g.,* isotropic local contrast [198], they require a large computational cost. Peli's contrast gives a good balance between quality and complexity.

For a color image, we convert it to the LHS color space and perform a contrast calculation on the luminance components.[3] Luminance values are normalized between $[0, 1]$ in our calculations. Peli's contrast $C_{i,k}$ at the i-th pixel location in $\mathbf{I}_k$ is defined in the image's luminance channel $\mathbf{L}_k$ as:

$$C_{i,k} = \frac{[\psi * \mathbf{L}_k]_i}{[\phi * \mathbf{L}_k]_i}, \tag{4.10}$$

where the operator $*$ denotes convolution; the subscript $[\cdot]_i$ denotes the i-th pixel/coefficient; ψ is a band-pass filter and ϕ is the corresponding low-pass filter.

Although Peli's original definition works well for complex images by taking into account the dependence of human contrast sensitivity on spatial frequency [141], it is biased towards pixels with low-luminance neighborhoods. If we directly use Equation (4.10) to represent contrasts in multi-exposure images, under-exposed regions, which are normally noisy, may produce stronger responses than well-exposed regions. This will make under-exposed regions contribute more to the fused image and reduce the overall brightness. As evaluated in [34], brightness has a strong impact on the overall image quality. Therefore, if the local background luminance at a pixel is below a threshold θ, we weight its contrast by the background luminance to suppress noise in the under-exposed regions. When combined with the Gaussian pyramid representation of a luminance image, we can construct a contrast pyramid. Let $\tilde{C}^n_{i,k}$ denote the weighted contrast at the i-th pixel location at level n of the Gaussian pyramid of $\mathbf{L}_k$, where $n \in [0, N_c - 1]$. Then, $\tilde{C}^n_{i,k}$ can be calculated as:

$$\tilde{C}^n_{i,k} = \begin{cases} G^n_{i,k} - [\phi * \mathbf{G}^n_k]_i, & [\phi * \mathbf{G}^n_k]_i < \theta; \\ (G^n_{i,k} - [\phi * \mathbf{G}^n_k]_i)/[\phi * \mathbf{G}^n_k]_i, & \text{otherwise.} \end{cases} \tag{4.11}$$

[3] Different luminance conversion methods (*e.g.,* taking the L^* component in the CIELAB space) slightly affect the fusion results.

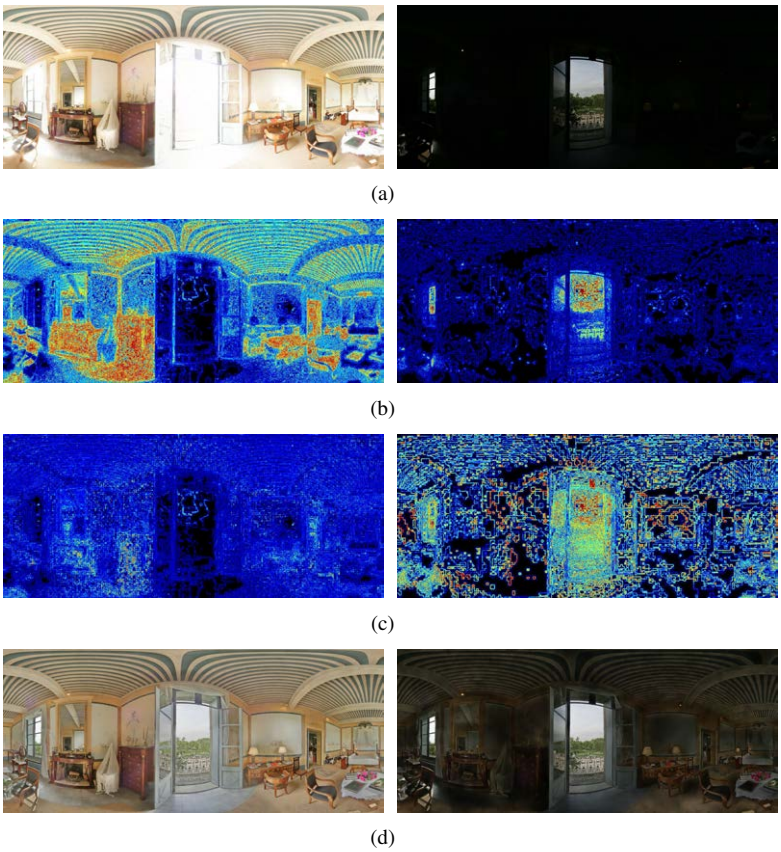


Figure 4.6: Comparison of our weighted and Peli's contrasts for multi-exposure fusion on the Chateau sequence. In a color-coded contrast magnitude image, warmer color indicates higher magnitude. Our weighted contrast places more emphasis on the high-exposure image, where more indoor details are revealed. In contrast, Peli's contrast produces noisy responses in the under-exposed regions of the low-exposure image, which reduces the brightness of the fused image. (a) High-exposure (left) and low-exposure (right) source images. (b) Magnitude of our weighted contrast. (c) Magnitude of Peli's contrast. (d) Results using our weighted (left) and Peli's (right) contrasts. (Source sequence courtesy of HDRsoft.com.)

where $\mathbf{G}_k^n$ denotes the n-th level of the Gaussian pyramid representation of $\mathbf{L}_k$ and $G_{i,k}^n$ the i-th coefficient in $\mathbf{G}_k^n$. $\mathbf{G}_k^0 = \mathbf{L}_k$ and $\mathbf{G}_k^{n+1} = [\phi * \mathbf{G}_k^n]^{\downarrow 2}$, where $[\cdot]^{\downarrow 2}$ denotes downsampling by a factor of 2 in each dimension. In our implementation, we take ϕ as a 5×5 Gaussian filter with variance 1. Figure 4.6 gives a comparison between our weighted and Peli's contrasts. In this comparison, the same transducer and psychometric functions are applied to the contrasts; and the color saturation measure is not used. Two of the five source images are shown in Figure 4.6(a). Their corresponding color-coded contrast magnitude images generated by the two contrast measures are given in Figures 4.6(b)(c), where warmer color indicates higher magnitude. Our weighted contrast places more emphasis on the high-exposure image, where more indoor details are revealed. In contrast, Peli's contrast produces noisy responses in the under-exposed regions of the low-exposure image, which reduces the brightness of the fused image, as shown in Figure 4.6(d).

In order to impose inter-level consistency, we pass information from a lower level to a higher

level by taking the maximum coefficient among corresponding coefficients in adjacent pyramid levels as the contrast magnitude $\hat{C}^n_{i,k}$ at the higher level:

$$\hat{C}^n_{i,k} = \max\{|\tilde{C}^n_{i,k}|, |[\tilde{\mathbf{C}}^{n-1}_k]^{\downarrow 2}_i|\}, \tag{4.12}$$

where $|\cdot|$ denotes the absolute value operator or magnitude. The polarity of contrasts has little effect on object detection in the human visual system, as long as the magnitudes are the same [193]. Therefore, we only consider contrast magnitude in our method. A contrast pyramid representation for image fusion was also proposed by Toet [184], which is known as the ratio of low-pass (RoLP) pyramid. However, no special treatment is applied in the RoLP pyramid to suppress noise in under-exposed/low-luminance regions; and no information is passed between levels to ensure inter-level consistency.

4.2.4.1 Transducer and Psychometric Functions

The nonlinearity of the human perception of contrast has been studied by many researchers [58, 62, 151, 197]. According to [58], contrast perception can be considered as a two-stage procedure. In the first stage, referred to as the mechanism response stage, the stimulus contrast is mapped to the internal/physiological response of the sensory system via a transducer function μ. In the second stage, referred to as the decision stage, the probability of correctly discriminating a stimulus with a certain contrast from the standard stimulus with a fixed contrast C_s is expressed by a psychometric function Ψ. When $C_s = 0$, the discrimination task degenerates to the detection task. A formal relationship between the psychometric function Ψ and the transducer function μ was developed in [62], where Ψ is determined by μ and the distribution of internal responses. It is assumed in [62] that the stimulus contrast is between $[0, 1]$; thus, without loss of generality, $\hat{C}^n_{i,k}$'s from Equation (4.12) are normalized between $[0, 1]$ before being used in μ.

Because information is passed from a lower level to a higher/coarser level in a contrast pyramid as introduced in Section 4.2.4, the highest level (level $N_c - 1$) captures all of the important features present in the lower levels. Hence, the transducer function and the psychometric function are only applied to level $N_c - 1$ of the contrast pyramids. This also reduces the computational cost. We adopted the transducer function proposed by Foley and Schwarz [58]:

$$\mu(\hat{C}^{N_c-1}_{i,k}) = \frac{(\hat{C}^{N_c-1}_{i,k} S_E)^p}{(\hat{C}^{N_c-1}_{i,k} S_I)^q + Z}, \tag{4.13}$$

where $S_E = 100$ is a constant; S_I, p, q, Z are four free parameters, which we set to the mean values reported from the experiments in [58], *i.e.,* $S_I = 75.70, p = 4.03, q = 3.59, Z = 24.87$. Two other forms of μ were also tested. A three-parameter transducer function defined on log contrast was proposed in [62], but it does not produce satisfactory results for multi-exposure fusion, where over- and under-exposures appear in the fusion results. Wilson's [197] transducer function for threshold and suprathreshold vision produces fusion results with a quality very close to those generated using Equation (4.13).

Because our goal is to preserve and enhance local contrasts/details in the fusion process, stimuli at different locations in an image are equally important, and it is very likely that making one or several stimuli highly discriminative from others (*i.e.,* performing contrast discrimination) will cause loss of details in some regions. Therefore, we adopted the psychometric function proposed in [62] for contrast detection instead of contrast discrimination:

$$\Psi(\hat{C}_{i,k}^{N_c-1}) = 1/(1+\exp(-\mu(\hat{C}_{i,k}^{N_c-1})/b)), \tag{4.14}$$

where b is a scale parameter, which we take as $b = \sqrt{6}/\pi$ as used in [62]; and $\Psi(\hat{C}_{i,k}^{N_c-1}) \in [0.5, 1]$. When used in HRW, $\Psi(\hat{C}_{i,k}^{N_c-1})$ is normalized to $[0, 1]$.

4.2.5 Color Saturation

Since the perceived local contrast measure only works in the luminance channel, using it alone may not produce satisfactory results for color images in some cases where high local contrasts are achieved at the cost of low colorfulness/saturation. Objects captured at proper exposures normally exhibit more saturated colors. For instance, as shown in Figure 4.4, the red book in one source image presents more saturated red than in the others. Therefore, we employ color saturation as another quality measure.

We incorporated the formula in the LHS color space [176] to measure the saturation S of a given pixel:

$$S = 1 - 3 \cdot \frac{\min(R, G, B)}{R + G + B}, \tag{4.15}$$

where $S \in [0, 1]$ and R, G, B denote the red, green, and blue components in the RGB color space, respectively. Other saturation measures also exist in the literature. Levkowitz and Herman [105] showed that the LHS space, as well as other related models like HSV, is one special case of the generalized LHS model. Lübbe [121] measures saturation as the relationship of chroma to the complete color sensation in the CIELAB space. Mertens *et al.* [129] measure saturation as the standard deviation between the R, G, and B components. All of these definitions produce similar results under our fusion scheme. However, Lübbe's measure requires about 4 times more computation than the others due to its relatively more complex color space conversion.

As in the case of creating the contrast pyramid introduced in Section 4.2.4, we can construct a saturation pyramid for each source image by first building a Gaussian pyramid and then calculating the saturation components at every level using Equation (4.15). To be consistent with the contrast pyramid, this saturation pyramid also has N_c levels. We observe that saturation calculation performed only at the highest pyramid level without information passing between levels is sufficient to produce satisfactory fusion results with no noticeable difference. This is because: 1) Gaussian smoothing has little influence on the objects/regions' color information when the filter's size and variance parameters are small (we use the same Gaussian filter parameters as for local contrast, *i.e.,* size 5×5 and variance 1); 2) Gaussian smoothing mainly affects object/region boundaries, where

(a) GRW without the two proposed perceptual measures

(b) With perceived local contrast

(c) With both perceptual measures

Figure 4.7: Incorporation of the proposed quality measures in MEF models using the GRW model as an example. With the perceived contrast measure, more local details are preserved, as shown in the insets. With the saturation measure, the fused image exhibits more vivid colors, *e.g.,* for the trees and grass. (House sequence in top row courtesy of Tom Mertens, Mask sequence in bottom row courtesy of HDRsoft.com.)

relatively large color and luminance changes occur after filtering, and these changes are captured by the perceived local contrast measure.

4.2.6 Perceptual Impact of the Proposed Quality Measures

In order to illustrate the quality contribution of the proposed local contrast and color saturation measures, we incorporate these two measures into the GRW model. The results from two test sequences are provided in Figure 4.7. Instead of employing local variations in a non-linear function to indicate contrasts, we believe modeling the probability of the human visual system to perceive a given contrast will generate better perceptual quality, because this new modeling scheme offers a more accurate estimation of the amount of visual stimuli delivered in each image region, which leads to better detail preservation, as shown in the insets in Figure 4.7(b). With the color saturation measure, Figure 4.7(c) (*e.g.,* for the trees and grass of the House scene and for the sky of the Mask

scene) shows that the fused image can exhibit more vivid colors.

4.2.7 Using Quality Measures in HRW for Fusion

The contrast pyramid (with transducer and psychometric functions applied) and the saturation pyramid are combined at each level to form a single pyramid $\mathcal{H}$ representing the quality measures. As discussed in Sections 4.2.4 and 4.2.5, the information at the highest level (level $N_c - 1$) is sufficient to generate satisfactory fusion results. Hence, in our implementation, only the highest levels in the two pyramids are actually combined, which also reduces computational cost. The coefficients at level $N_c - 1$ of $\mathcal{H}$ are taken as the data compatibilities at the finest level in HRW, *i.e.,*

$$Y_{ik}^{N_c-1} = \Psi(\hat{C}_{i,k}^{N_c-1}) \cdot S_{i,k}^{N_c-1}. \tag{4.16}$$

The initial probability that a scene node x_i^n takes a label l_k is determined as $P_{i,k} = Y_{ik}^n / (\sum_k Y_{ik}^n)$. These probabilities are then updated in HRW exploiting neighborhood information.

As in Section 3.3, we also employ the neighborhood consistency measure to impose consistency both in a neighborhood and with the natural scene. The similarity between adjacent scene nodes at level 0 of $\mathcal{H}$ is evaluated based on pixel values in the source images:

$$W_{ij}^0 = \prod_{k=1}^{K} W_{ij,k}^0 = \prod_{k=1}^{K} \exp(-\frac{\|p_{i,k} - p_{j,k}\|}{\sigma}), \tag{4.17}$$

where $\exp(\cdot)$ is the exponential function; $\|\cdot\|$ denotes Euclidean distance; and σ is a free parameter. For color images, the RGB color space is used for computing neighborhood similarity scores. These scores are then restricted to level $N_c - 1$ using Equation (4.3) to be used as neighborhood compatibilities at the finest level in HRW. After the probability maps are obtained using HRW (Section 4.1) for level $N_c - 1$ of $\mathcal{H}$, they are interpolated to the original resolution (level 0) using the prolongation operator in HRW. Because of this quick approximation, $P_{i,k}^0$'s may not sum up to 1 for a given x_i. Hence, $P_{i,k}^0$'s are normalized for every pixel location, *i.e.,* $P_{i,k}^0 = P_{i,k}^0 / (\sum_k P_{i,k}^0), \forall i$.

4.2.7.1 Fusion Weighting Function

In order to utilize the probabilities to guide the image composition process, we define a weighting function $\Phi(\cdot)$, which increases monotonically on the probabilities. The derived weights are then used to combine the source images as follows:

$$\bar{p}_i = \sum_{k=1}^{K} \Phi(P_{i,k}^0) p_{i,k}, \tag{4.18}$$

where $\bar{p}_i$ and $p_{i,k}$ denotes the i-th pixel in the fused image $\bar{\mathbf{I}}$ and the k-th source image $\mathbf{I}_k$, respectively.

One straightforward fusion weighting function is to directly take the probabilities as weights, *i.e.,*

$$\Phi(P_{i,k}^0) = P_{i,k}^0. \tag{4.19}$$

(a) (b) (c)

Figure 4.8: Effects of different parameter settings of the fusion weighting function. (a) $\alpha_1 = 1, \alpha_2 = 0$. (b) $\alpha_1 = 1.3, \alpha_2 = 0.01$. (c) $\alpha_1 = 1.5, \alpha_2 = 0$.

Algorithm 4.2 Basic steps of the perception-guided multi-exposure fusion algorithm.

1: Construct the contrast and saturation pyramids following the procedures in Sections 4.2.4 and 4.2.5
2: Combine the contrast and saturation measures at level $N_c - 1$ to form the data compatibilities in HRW using Equation (4.16)
3: Compute the neighborhood similarity scores at level 0 using Equation (4.17)
4: **for** $n = 1$ to $N_c - 1$ **do**
5: Restrict the neighborhood similarity scores from level $n - 1$ to n using Equation (4.3)
6: **end for**
7: Take the neighborhood similarity scores at level $N_c - 1$ as the neighborhood compatibilities in HRW
8: Obtain the probability maps for level $N_c - 1$ using Algorithm 4.1
9: **for** $n = N_c - 2$ to 0 **do**
10: Prolongate the probability maps from level $n + 1$ to n using Equations (4.6) to (4.8)
11: **end for**
12: Generate weight maps using Equation (4.19) or (4.20)
13: Generate the fused image using Equation (4.18)

This weighting function ensures that all fused pixel values are within the valid range of $[0, 1]$. It also preserves local details well. However, the fused image may not have enhanced global contrast/dynamic range. To achieve this, we take $\Phi(\cdot)$ as a linear function of $P^0_{i,k}$:

$$\Phi(P^0_{i,k}) = \alpha_1 P^0_{i,k} - \alpha_2, \tag{4.20}$$

where α_1 and α_2 are two free parameters. When $\alpha_1 = 1, \alpha_2 = 0$, Equation (4.20) becomes equivalent to Equation (4.19). α_1 is used to stretch or compress the weights, which results in higher or lower global contrast in the fused image. Because this weighting function does not guarantee a valid range of pixels, when the fused pixel values fall outside $[0, 1]$, they are truncated. α_2, together with the truncation operator, is used to select the range of the weights, which results in a darker or brighter fused image. We take truncation instead of normalization in our implementation because it generally provides higher global contrast. The values of these two parameters can be pre-defined, evaluated based on the source images, or interactively controlled by the user to adjust the appearance of the fused image. An example is given in Figure 4.8. The whole fusion algorithm is summarized in Table 4.2.

4.2.8 Experimental Results

Seven free parameters, *i.e.,* $\theta, N_c, N_h, \sigma, \gamma, \alpha_1, \alpha_2$, are used in our algorithm. We take the luminance threshold for computing the weighted contrast in Equation (4.11) as $\theta = 0.3$. The other six parameters are computed as follows based on image information. The number of levels for the quality measure pyramids (*i.e.,* N_c) and the number of levels for the hierarchy in HRW (*i.e.,* N_h) are both determined based on the size of the source images, which are assumed to have the same size. Let the image size be $M \times N$. Then, we compute $N_c = \min(2, \lceil \log_2(\min(M, N)/\kappa) \rceil + 1)$ and $N_h = \lceil \log_2(\min(M, N)/\kappa) \rceil - N_c + 2$. Here, $\lceil \cdot \rceil$ is the ceiling function; and κ is maximum number of nodes allowed along the shorter dimension of the coarsest-level graph, which we take as $\kappa = 32$. For a source sequence $\mathcal{D}$ with K images, the parameter σ in Equation (4.17) is taken as $\sigma = 0.1K$. The purpose here is to place less emphasis on the differences between adjacent scene nodes, which helps to include contributions from larger neighborhoods, when the number of source images is large. γ in Equation (4.5) is computed as $\gamma = 0.2K\sqrt{MN}/\kappa$. This is to impose consistency over larger neighborhoods, when the number and/or size of source images is large. When Equation (4.20) is used as the fusion weighting function, we compute $\alpha_1 = 0.6 + \exp(-\bar{L})$ and $\alpha_2 = 0.02\exp(-\bar{L})$, where $\bar{L} \in [0, 1]$ is the average luminance of $\mathcal{D}$. If $\bar{L}$ is low, it may indicate that the amount of under-exposed regions in $\mathcal{D}$ is large. Hence, a larger α_1 is required to increase the global contrast; and a larger α_2 is required to select the middle portion of the stretched dynamic range avoiding over- or under-exposure. The above parameter setting was used in all experiments unless otherwise mentioned.

4.2.8.1 Comparison with Other Methods

The fusion results of our perception-guided algorithm were compared with two other multi-exposure fusion methods, *i.e.,* our probabilistic fusion (PF) (Section 3.3) and exposure fusion (EF) [130], which have previously demonstrated better performance than many other methods. The default parameter settings in PF and EF were used. In addition, two TM methods were also considered: the photographic tone reproduction (PTR) method [156][4], which is one of the best-rated TM methods in evaluations conducted by different researchers [34, 102], and iCAM06 [94], which demonstrated even better performance than PTR for some scenes in different evaluations [94, 102]. The parameters in PTR were estimated using the method in [155]. For iCAM06, the default parameter setting was used. The HDR images for TM methods were generated using HDR reconstruction (HDR-R) [46]. The default or automatically generated parameter settings were used for all methods in our analysis, because we aimed to achieve a relatively fair comparison with minimal personal manipulation/bias from our side.

Figure 4.9 depicts the comparison on the National Cathedral sequence. The bottom row shows closeup views of the upper window. The images in Figures 4.9(a)(b) are both generated by our

[4]Here, we refer to the local TM operator in [156].

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

Figure 4.9: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the National Cathedral sequence. The HDR input for PTR and iCAM06 are generated using HDR-R. PGF and PGF-A generate the best results, where details in all regions are preserved with high local contrasts and saturated colors. iCAM06 preserves as many details as ours in the window regions but with less vivid colors. (a) PGF. (b) PGF-A. (c) PF. (d) EF. (e) PTR. (f) iCAM06. (Source sequence courtesy of Max Lyons.)

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

Figure 4.10: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Memorial Church sequence. The HDR input for PTR and iCAM06 are generated using HDR-R. PGF and PGF-A preserve more details than the others, especially in the window regions. PGF-A produces higher global contrast than PGF, but may cause loss of subtle details, *e.g.,* in the upper portion of the left window in the closeup view. (a) PGF. (b) PGF-A. (c) PF. (d) EF. (e) PTR. (f) iCAM06. (Source sequence courtesy of Paul Debevec.)

method, but with different fusion weighting functions. The image in Figure 4.9(a) is produced using Equation (4.19), which is denoted as PGF; and the image in Figure 4.9(b) uses Equation (4.20), which is denoted as PGF-A. Our methods (PGF and PGF-A) and iCAM06 produce better local details in the window regions than the other three methods. However, the colors are not reproduced well by iCAM06. Our methods produce the best images, where details are preserved with high local contrasts and vivid colors.

Figure 4.10 gives the comparison on the Memorial Church sequence. The bottom row gives closeup views of regions near the three windows. Our PGF-A produces higher global contrast than our PGF, but some subtle details may be lost. For example, compared with Figure 4.10(a), the upper portion of the left window in the closeup view in Figure 4.10(b) shows slight over-exposure. However, both images contain more details than the results generated by other methods.

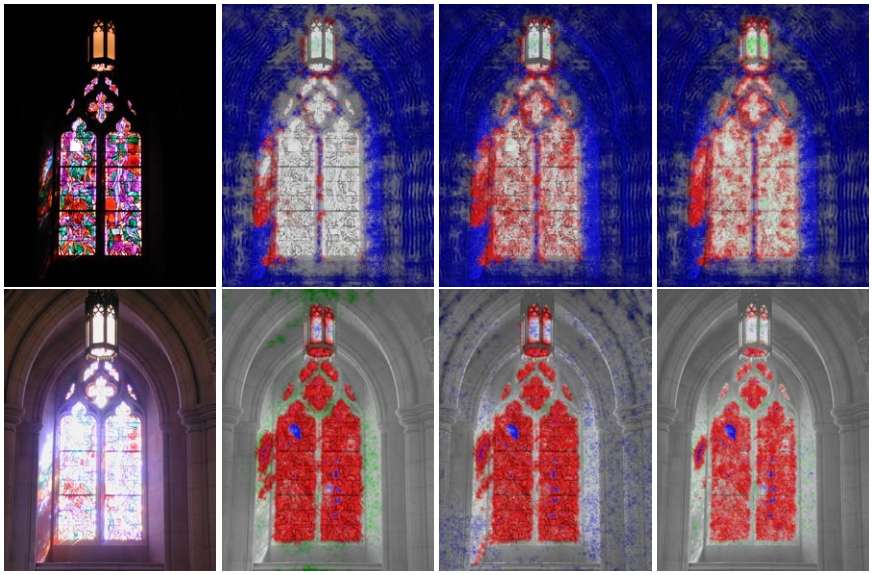
4.2.8.2 Objective Evaluation

Table 4.2: Evaluation using the $Q^{AB/F}$ metric.

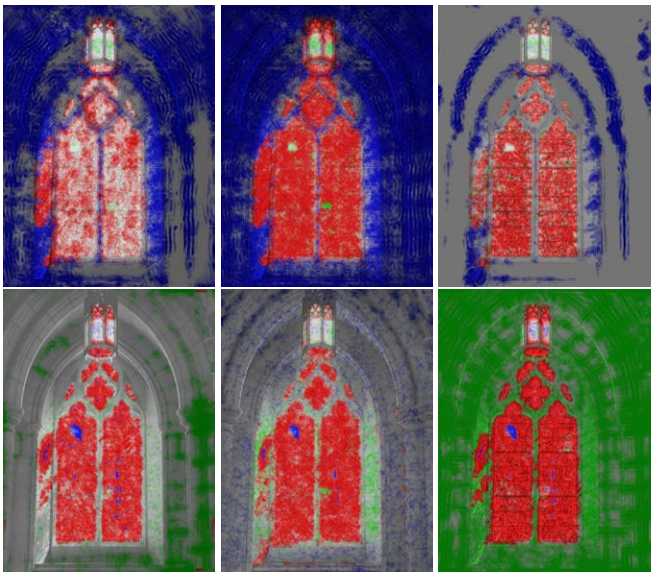
Source sequence	PGF	PGF-A	PF	EF	PTR	iCAM06
House	0.715	**0.739**	0.715	0.732	0.515	0.578
Chateau	**0.774**	0.772	0.766	0.769	0.419	0.418
Memorial Church	**0.665**	0.658	0.661	0.662	0.628	0.495
Belgium House	0.609	0.617	**0.621**	0.609	0.568	0.560
National Cathedral	0.697	**0.703**	0.578	0.670	0.286	0.281
Lamp	0.564	**0.583**	0.552	0.571	0.365	0.478

Two objective evaluation metrics were employed to assess the fusion quality of our methods (PGF and PGF-A). Xydeas and Petrović [202] proposed the $Q^{AB/F}$ metric to measure edge information correctly transferred from a set of input monochrome images to a fused image. Reproduction of both edge strength and orientation is considered. This metric assumes that stronger edges attract more visual attention, and therefore it also associates each edge with an importance coefficient defined by the edge strength. This metric gives a performance score between $[0, 1]$ for each test image, where a higher score means better performance. This metric is widely used in the image fusion literature [117, 109]. Because this metric works on monochrome images, we first converted both source images and resulting images into gray-scale before performing the evaluation. The reported metric parameter values from [202] were used in our experiment. Six standard test sequences were used in this evaluation, and the performance scores are reported in Table 4.2. Although PTR and iCAM06 are not MEF methods, they are included in the table for reference purposes only. All of the compared MEF methods successfully transferred most of the edge information into the results. On average, PGF-A had the best performance, followed by PGF, EF, and PF.

The other objective metric employed is the dynamic range independent visible difference predictor (DRIVDP) [11]. This metric evaluates visual local contrast distortions between a reference image and a test image under a specific viewing condition. This metric is widely used in the TM literature [56, 123]. Here, we use it to assess the visual distortions between a test image and each source image. Three distortions are considered in DRIVDP: loss of visible contrast, amplification of invisible contrast, and reversal of visible contrast. We assume that the images were viewed on a typical LCD with a max luminance equivalent to $100cd/m^2$, a gamma value of 2.2, and a visual resolution of 30 pixels per degree at a viewing distance of 0.5 meter, and that the peak contrast sensitivity of the viewer is 0.25%. The evaluation result on the National Cathedral sequence is given in Figure 4.11 (please refer to Figure 4.9 for visual comparison). The two source images with good exposures respectively for the windows and the walls are given in Figure 4.11(a). The distortion maps for each method are given in Figure 4.11(b)-(g). In a distortion map, green, blue, red, and gray pixels indicate contrast loss, amplification, reversal, and no distortion, respectively. For the window region (top row in Figure 4.11), PGF shows very little distortion; PGF-A, PF, and EF show various



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



(e) (f) (g)

Figure 4.11: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the National Cathedral sequence using DRIVDP. The two source images offers good clarity for the windows and the wall, respectively. In a distortion map, green, blue, red, and gray pixels indicate contrast loss, amplification, reversal, and no distortion, respectively. PGF performs best in preventing contrast distortions, followed by PGF-A, PF, EF, PTR, and iCAM06. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

degrees of contrast reversal; PTR and iCAM06 show strong contrast reversal and some contrast loss. For the wall region (bottom row in Figure 4.11), PF shows very little distortion; PGF-A shows some contrast amplification; PGF shows a slight loss of contrast; EF shows strong contrast loss for the lower right region of the wall; PTR shows some contrast loss and amplification; and iCAM06 shows strong contrast loss. Please note that although contrast amplification may be viewed as distortion in

some applications, it is normally considered as one of the objectives and usually preferred in image fusion [173].

4.2.8.3 Subjective Evaluation

Subjective evaluations, which can cover more aspects than objective evaluations, are widely conducted in both the image fusion and TM communities to compare and validate algorithms [34, 142]. The objective evaluation in the previous section considers only characteristics in the luminance channel, but does not consider other important aspects of image quality, *e.g.,* colors. Thus, we also conducted a subjective test to evaluate the performance of PGF, PGF-A, PF, EF, PTR, and iCAM06 on six different scenes. Thirteen subjects (8 males and 5 females) aged between 25 and 35 participated in this evaluation. All of the participating subjects had normal or corrected-to-normal vision and were non-experts in the field of MEF or TM. The test was performed under normal lighting conditions. For each scene, the results of different methods were anonymized and placed side by side in different orders, along with the source sequence. No other reference image, either manually or automatically fused or tone-mapped, was provided to guide/influence a subject's judgement. Subjects were asked to rank the results on a scale of 5 (best) to 0 (worst) in four categories: global contrast, details, colors, and overall appearance. These four criteria were also considered in [34]. The *global contrast* criterion measures the global luminance variations, where an image with a high global contrast tends to span a large dynamic range. The *details* and the *colors* criteria measure the local details and colors reproduced and/or enhanced from the source images, respectively. The *overall appearance* criterion measures the overall impression of a fused or tone-mapped image.

The average ranking scores of different algorithms under the four criteria are reported in Figure 4.12. Our PGF-A (green lines) performed consistently well compared to the other methods. It had the best performance under all of the four criteria on 5 out of 6 scenes, and performed similarly to our PGF (red lines) on the National Cathedral sequence. PGF and EF (cyan lines) performed similarly on average, followed by PF (blue lines), iCAM06 (black lines), and PTR (magenta lines). Although some images may suffer from loss of subtle details due to the enhanced contrast (*e.g.,* compare the insets in Figure 4.10(a)(b)), high contrast and good color scheme may compensate for this in the visual impression in some cases, which results in higher ranking scores (see the plots on MC in Figure 4.12). For some scenes, one may notice that higher overall appearance ratings seem to be related to images with higher brightness or global contrast. However, these two factors do not necessarily dominate the overall user preference. For the Cathedral scene (Figure 4.9), the results generated by PGF-A, PF, EF, and PTR are all brighter than the result by PGF. However, PGF preserved the details and colors better for that scene, and it therefore received a higher rating under the overall quality criterion. For the Memorial Church scene (Figure 4.10), both PF and EF had higher global contrast ratings than PGF, but the three methods received very close overall appearance ratings. In summary, there is no single factor that exclusively influenced users' preferences, but rather

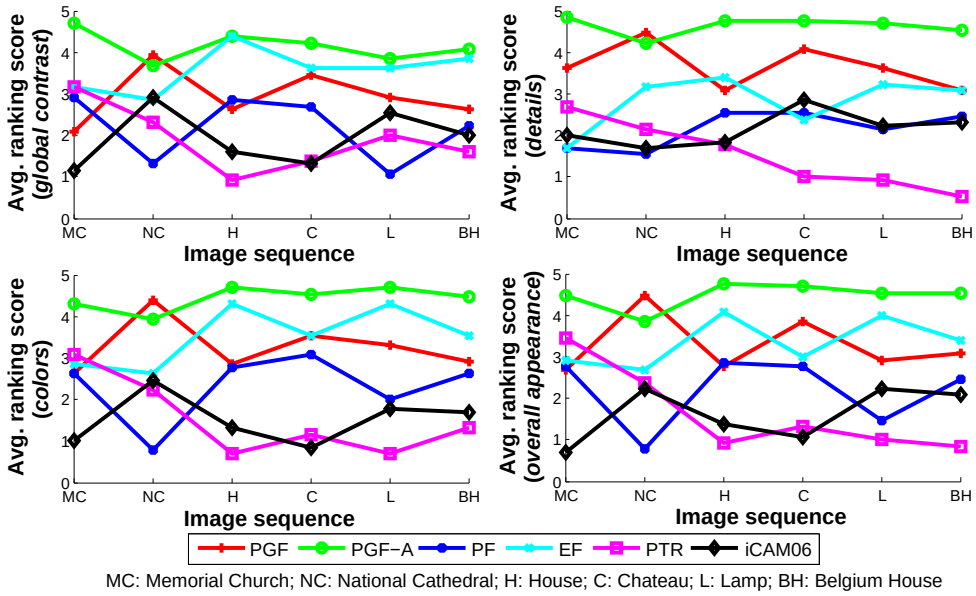


Figure 4.12: Average ranking scores of different algorithms in the subjective evaluation. Our PGF-A has the best performance under all four criteria for 5 out of 6 scenes, and shows similar performance to our PGF on the other scene. PGF and EF have similar performance on the average, followed by PF, iCAM06, and PTR.

it is the combination of different factors (*e.g.,* global contrast and details) that determines a user's choice.

Please note that the above experiments were not meant to formally evaluate TM operators. The main purpose was to demonstrate that the proposed fusion method is capable of producing high-quality fused images directly from the source sequences with qualities comparable to those of tone-mapped images and images generated by other fusion methods, or even better qualities in some cases. It is also worth mentioning that while our concern here is the production of a well-exposed image from a set of multi-exposure images, the HDR images input into TM do not necessarily come from a multi-exposure sequence. Please refer to [34, 95] for more formal evaluations of TM operators. Because user interaction is normally exploited in TM, it is possible that PTR and iCAM06 may produce comparable results to ours from the reconstructed HDR images by manually fine-tuning their parameters for individual scenes. In contrast, our method is capable of automatically producing high-quality results directly from a source sequence without any user intervention. In addition, if desired, a user can also adjust the parameters of our method to generate even more appealing results based on his or her preferences for individual scenes, such as in Figure 4.8. More evaluation results can be found in Appendix D.

4.2.8.4 Computational Complexity

In the quality measure calculation step, only linear operations are involved. Therefore, the complexity of this step is $O(KMN)$, where K denotes the number of images in the source sequence and

each source image has $M \times N$ pixels. In HRW, the compatibility calculation, restriction, prolongation, and relaxation are all linear operations. The coarsest-level solution is obtained as in GRW, the complexity of which is $O(KMN/(4^{N_c+N_h-2}))$. The operations in the final composition step are linear. Therefore, the total complexity of our algorithm is $O(KMN)$, which means the computational time is proportional to the number of pixels in the whole source sequence. The computational times for the six source sequences used in previous experiments range from 1.504 seconds (House sequence) to 28.373 seconds (Lamp sequence). Times for reading source sequences and writing fused images are excluded.

4.2.9 Applications in Other Fusion Problems

Our fusion method can be applied to other fusion problems as well. This section gives two examples: multi-focus fusion and ambient/flash fusion. Note that the two examples serve only to show some possible future applications of our method. In order to fully adapt our MEF method to other fusion problems, more specialized quality measures may need to be developed.

4.2.9.1 Multi-Focus Fusion

A 3D scene may contain a large depth of field (DOF), and different focal lengths of a camera are required to clearly capture objects at various distances. The goal of multi-focus fusion [3, 210] is to generate an all-in-focus image with extended DOF by combining in-focus objects from different images. Figure 4.13 gives an example of applying our method to multi-focus fusion. The same procedure as in Table 4.2 was used. The parameter setting was the same as in MEF except that γ was computed as $0.01K\sqrt{MN}/\kappa$. We also explored the possibility of applying two other MEF methods, *i.e.,* PF and EF, to this multi-focus fusion problem. The default parameter settings in PF and EF were used. In the left image of Figure 4.13(a), the focus is set on the frog, and the beaver and the R2-D2 at the back are blurred, while the situation is reversed in the right image. The in-focus regions in individual images are combined into one all-in-focus image. Our perception-guided methods (Figures 4.13(b)(e)) generated clearer images than our previous PF (Figure 4.13(c)) and EF (Figure 4.13(d)), *e.g.,* the boundary of the beaver's teeth is sharper in our results.

4.2.9.2 Ambient/Flash Fusion

Without the use of a flash, it is difficult to clearly capture a scene under poor illumination. However, a flash usually creates undesirable artifacts, such as highlights on reflective surfaces. Ambient/flash fusion aims to remove artifacts in individual images by integrating information from an image pair, where one image is captured under ambient lighting condition and the other is captured with a flash. An example of removing highlights in a flash image is given in Figure 4.14. The ambient image captures the pattern on the book cover, but the interior of Optimus Prime's trailer is almost invisible. In the flash image, the roller inside the trailer is clearly visible, but there are highlights on the book

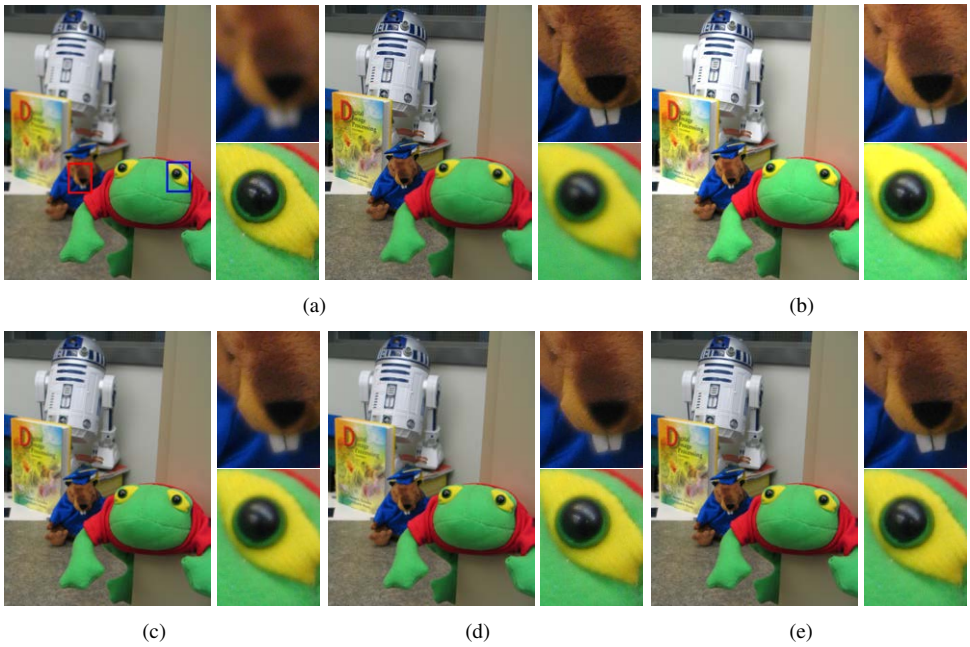


Figure 4.13: Multi-focus fusion using PGF and PGF-A on the Frog sequence, in comparison with PF and EF. The in-focus regions in individual source images are combined into one all-in-focus image. PGF and PGF-A generate clearer images than PF and EF, *e.g.,* the boundary of the beaver's teeth is sharper in results of PGF and PGF-A. (a) Source images with focuses on the frog (left) and on the beaver (right). (b) PGF-A. (c) PF. (d) EF. (e) PGF.

cover. Compared with PF and EF, PGF and PGF-A generate better fused images (Figures 4.14(b)(e)) with improved appearance, where the highlights in the flash image are removed and details from both images are combined. The same procedure in Table 4.2 was used except that the contrast measure was not applied. This is because the clear boundaries around the highlights created by the flash produce very high local contrasts, which will cause unnatural illumination changes near those boundaries in the fused image when the contrast measure is used. The parameter setting is the same as in MEF except that σ equals $0.5K$ and $\gamma = 1.0K\sqrt{MN}/\kappa$. Because our algorithm currently estimates the fusion weights/probabilities in a way that maximizes local details, it is not suitable for the removal of more complex artifacts, such as removing reflections in a flash image or noise in an ambient image, which requires more specialized algorithms (*e.g.,* [5, 145]).

4.2.10 Limitation

Although our method produces high-quality results, it requires the source images to be registered and to contain minimal sensor noise, which are common requirements for multi-exposure fusion methods. These requirements may restrict our method from being applied in some situations (*e.g.,* camera not held in a static position) without pre-processing the captured images. Therefore, incorporating special treatment for imperfectly registered or noisy source images during the fusion process would be a useful extension. Shrinkage of wavelet coefficients was used in [120] for noise

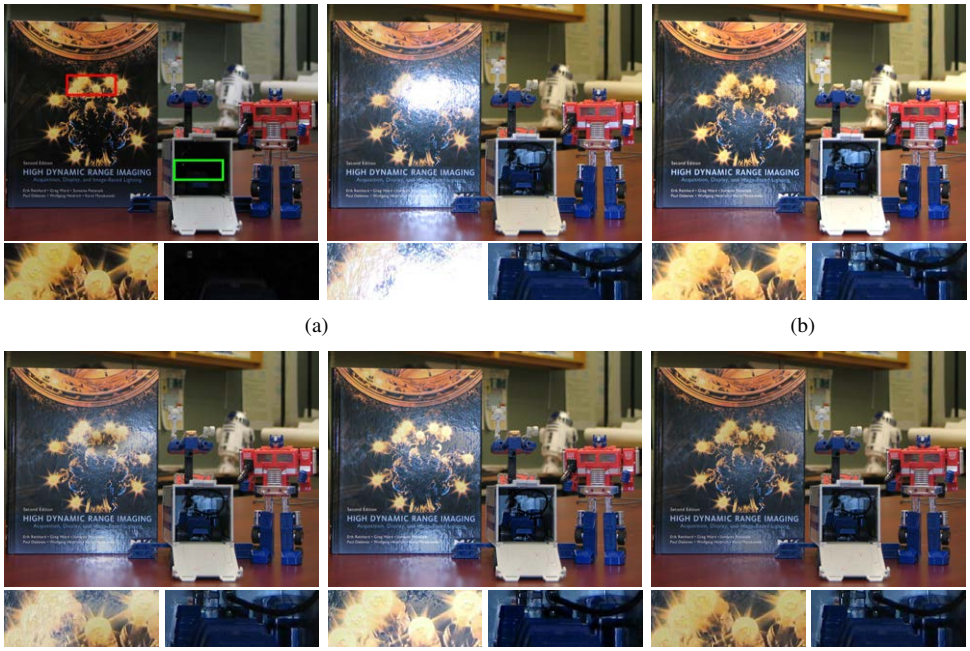


Figure 4.14: Ambient/flash fusion using PGF and PGF-A on the Book sequence, in comparison with PF and EF. PGF and PGF-A generate better fused images with improved appearance, in which the highlights in the flash image are removed and details from both images are integrated. (a) Source image pair of an ambient image (left) and a flash image (right). (b) PGF-A. (c) PF. (d) EF. (e) PGF.

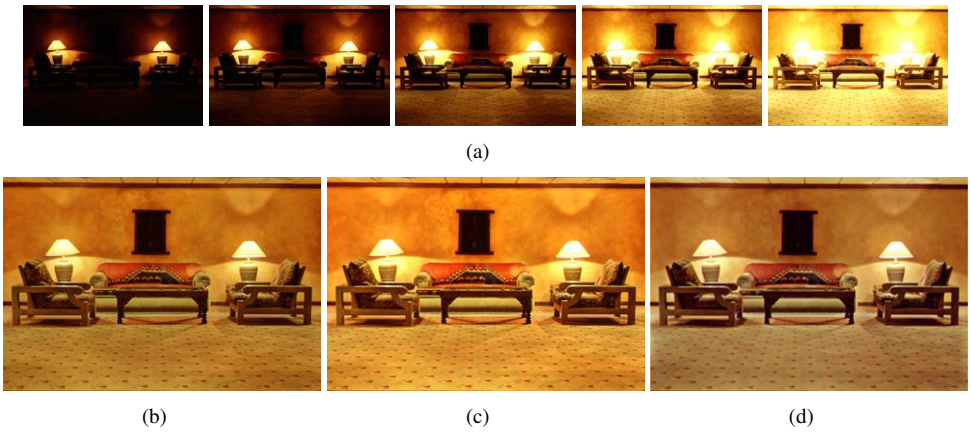


Figure 4.15: Comparison of our methods (PGF and PGF-A) with GDC on the Chairs sequence. The result of GDC is taken from its project website. (a) Source sequence. (b) PGF. (c) PGF-A. (d) GDC. (Source sequence courtesy of Shree Nayar.)

reduction during the fusion of multimodal images. This shrinkage function may be incorporated as a quality measure in our method in order to handle noisy source images.

Another useful extension to our fusion scheme is to incorporate special treatment for dynamic scenes, in which there are moving objects. A gradient-directed composition (GDC) method is proposed in [208], where gradient orientation changes are used to identify dynamic objects in a multi-

exposure sequence. Gradient magnitudes are used to measure the contributions from each source pixel. Bilateral filtering is applied to get smooth weight maps and an MSD-based fusion scheme is employed. Although this method is good at removing ghosting artifacts, our methods produce better results for static scenes. An example is given in Figure 4.15. The contrasts and colors of the well-exposed regions in the source sequence do not transfer well into the result of GDC, while our results feature better contrasts and colors. In addition, GDC requires that for a given region, the static background scene should not be occluded in most source images. When a multi-exposure sequence is taken within a short time period, it is very likely that no noticeable movement of objects is observed, and it is also possible that a portion of the background scene is constantly occluded by a moving foreground object. In the former case, the source sequence can be treated as a static scene; in the latter case, techniques such as image inpainting [85] may need to be incorporated in order to infer the background. Since the change of gradient orientations was used in [208] for deghosting, it may be also helpful in the treatment of slightly misaligned source images.

4.3 Hierarchical Multivariate Gaussian Conditional Random Fields

Similar to HRW, we can also construct a hierarchical model of the multivariate Gaussian conditional random fields. In this section, we introduce the hierarchical MGCRF model and analyze its relationship with HRW. As in Section 3.4.2, we impose the piecewise-smooth assumption (Equations (3.21) and (3.22)). We restrict our discussion here to lattice graphs with 4-connectivity assumed. The basic idea is to efficiently calculate the MAP estimate of the K-D potential field $\tilde{\mathcal{X}}$ given the boundary set $\tilde{\mathcal{B}}$ and the observed data $\mathcal{D}$ in a coarse-to-fine fashion. Given an MGCRF $(\tilde{\mathcal{X}}, \mathcal{D})$ at the original resolution (*i.e.,* level 0) with $\tilde{\mathcal{X}}^0 = \tilde{\mathcal{X}}$, we iteratively construct a coarser representation of it until a pre-defined level (say, level $N-1$) is reached. At level $N-1$, $\tilde{\mathcal{X}}^{N-1}$ can be accurately and quickly estimated by solving a linear system as in Section 3.4.2.

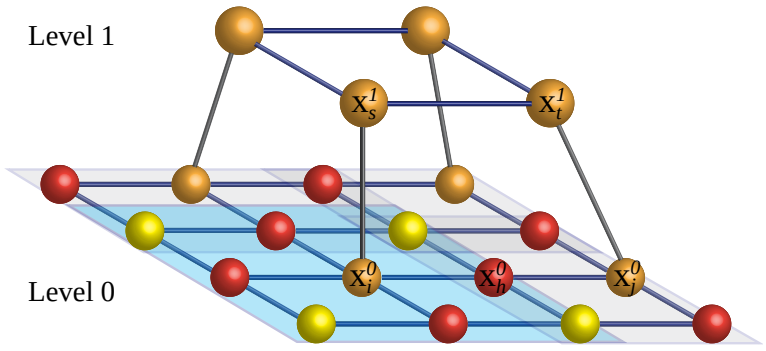


Figure 4.16: A two-level hierarchical multivariate Gaussian conditional random field.

A two-level HMGCRF is shown in Figure 4.16. The nodes at a finer level are divided into three sets: $\tilde{\mathcal{X}}_O^n$ (nodes with only odd coordinates, *i.e.,* yellow nodes in the figure), $\tilde{\mathcal{X}}_E^n$ (nodes with only

even coordinates, *i.e.,* orange nodes in the figure), and $\tilde{\mathcal{X}}_B^n$ (nodes with one even and one odd coordinates, *i.e.,* red nodes in the figure). The orange nodes at a finer level are the projections of the nodes at a coarser level. The precision matrices associated with each of the nodes at a coarser level are determined by the precision matrices associated with their projections and the projections' neighboring nodes at a finer level. In the following, we introduce the coarser-level MGCRF construction scheme and the finer-level potential field approximation scheme.

4.3.1 Construction of Coarser-Level MGCRF

The construction of a coarser-level MGCRF is essentially deriving the coarser-level precision matrices based on the precision matrices at the current level. Let $\mathbf{x}_i^n$ at level n denote the projection of $\mathbf{x}_s^{n+1}$ at level $n+1$. The precision matrix $\tilde{\mathbf{T}}_{sk}^{n+1}$, which defines the probability density of $\mathbf{x}_s^{n+1}-\mathbf{b}_k$, is derived as follows:

$$\tilde{\mathbf{T}}_{ik}^n = \tilde{\mathbf{T}}_{ik}^n + (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n \cap \tilde{\mathcal{X}}_O^n} \tilde{\mathbf{S}}_{ij}^n)^{-1} (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n \cap \tilde{\mathcal{X}}_O^n} \tilde{\mathbf{S}}_{ij}^n \tilde{\mathbf{T}}_{jk}^n),\ \forall \mathbf{x}_i^n \in \tilde{\mathcal{X}}_B^n \tag{4.21}$$

$$\tilde{\mathbf{T}}_{sk}^{n+1} = \tilde{\mathbf{T}}_{ik}^n + (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n} \tilde{\mathbf{S}}_{ij}^n)^{-1} (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n} \tilde{\mathbf{S}}_{ij}^n \tilde{\mathbf{T}}_{jk}^n),\ \forall \mathbf{x}_i^n \in \tilde{\mathcal{X}}_E^n \tag{4.22}$$

Let $\mathbf{x}_i^n$ and $\mathbf{x}_j^n$ at level n denote the projections of two adjacent nodes $\mathbf{x}_s^{n+1}$ and $\mathbf{x}_t^{n+1}$ at level $n+1$. The precision matrix $\tilde{\mathbf{S}}_{st}^{n+1}$, which defines the probability density of $\mathbf{x}_s^{n+1}-\mathbf{x}_t^{n+1}$, is derived as:

$$\tilde{\mathbf{S}}_{st}^{n+1} = \arg\min_{\tilde{\mathbf{S}}_{hm}^n} |\tilde{\mathbf{S}}_{hm}^n|, \forall \mathbf{x}_h^n \in \mathcal{N}_i^n \cap \mathcal{N}_j^n, \forall \mathbf{x}_m^n \in \{\mathbf{x}_i^n, \mathbf{x}_j^n\}, \tag{4.23}$$

where $|\cdot|$ denotes determinant.

4.3.2 Approximation of Finer-Level Potential Field

Once the estimate of the potential field $\tilde{\mathcal{X}}^{n+1}$ at a coarser level is obtained, it is used to guide the approximation at a finer level. First, the coarser-level potentials are injected to the finer level as follows. These potentials spread from their projections $\tilde{\mathcal{X}}_E^n$ to their first-order neighborhood $\tilde{\mathcal{X}}_B^n$ and then to their second-order neighborhood $\tilde{\mathcal{X}}_O^n$.

$$\mathbf{x}_i^n = \mathbf{x}_s^{n+1},\ \forall \mathbf{x}_i^n \in \tilde{\mathcal{X}}_E^n \tag{4.24}$$

$$\mathbf{x}_i^n = (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n \cap \tilde{\mathcal{X}}_E^n} \tilde{\mathbf{S}}_{ij}^n)^{-1} (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n \cap \tilde{\mathcal{X}}_E^n} \tilde{\mathbf{S}}_{ij}^n \mathbf{x}_j^n),\ \forall \mathbf{x}_i^n \in \tilde{\mathcal{X}}_B^n \tag{4.25}$$

$$\mathbf{x}_i^n = (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n} \tilde{\mathbf{S}}_{ij}^n)^{-1} (\sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n} \tilde{\mathbf{S}}_{ij}^n \mathbf{x}_j^n),\ \forall \mathbf{x}_i^n \in \tilde{\mathcal{X}}_O^n \tag{4.26}$$

Second, these injected potentials are updated iteratively using the following relaxation scheme:

$$\mathbf{x}_i^n = (\sum_{k=1}^K \tilde{\mathbf{T}}_{ik}^n + \sum_{\mathbf{x}_j \in \mathcal{N}_i} \tilde{\mathbf{S}}_{ij}^n)^{-1} (\sum_{k=1}^K \tilde{\mathbf{T}}_{ik}^n \mathbf{b}_k + \sum_{\mathbf{x}_j^n \in \mathcal{N}_i^n} \tilde{\mathbf{S}}_{ij}^n \mathbf{x}_j^n). \tag{4.27}$$

Because this is a quick approximation, when the original resolution is reached (*i.e.,* $n = 0$), each K-D potential $\mathbf{x}_i^0$ is normalized. The normalization takes three steps. First, element-wise normalization of $\mathbf{x}_i^0$ is performed, and we denote the vector after this normalization as $\mathbf{y}_i$ with each element calculated as:

$$y_{i,k} = \frac{x_{i,k}^0 - U_{\min,k}}{U_{\max,k} - U_{\min,k}}. \tag{4.28}$$

Second, $\mathbf{y}_i$ is normalized to render $\sum_k y_{i,k} = 1$. Finally, element-wise scaling of $\mathbf{y}_i$ is performed to obtain the normalized $\mathbf{x}_i^0$:

$$x_{i,k}^0 = U_{\min,k} + y_{i,k}(U_{\max,k} - U_{\min,k}). \tag{4.29}$$

4.3.3 Relationship with HRW

When the same assumption as in Equation (3.33) is made (*i.e.,* identity matrices subject to individual scaling factors) and $U_{\max,k} = \gamma_3, U_{\min,k} = 0, \forall k$, the HMGCRF model degenerates to the HRW model in terms of steady-state probability calculation. In the multi-exposure fusion application, this leads to Equation (4.19), and setting $U_{\max,k} = \alpha_1, U_{\min,k} = -\alpha_2, \forall k$ leads to Equation (4.20).

4.4 Summary

In this chapter, a hierarchical random walks framework was proposed to reduce the computational cost incurred in GRW. This improved performance was achieved by calculating the probabilities in a coarse-to-fine manner with properly passed information between levels. A novel fusion algorithm based on perceptual quality measures, *i.e.,* perceived local contrast and color saturation, was also proposed, where HRW was applied for efficient calculation of fusion weights. A transducer function and a psychometric function were applied to model the nonlinear human perception of local contrast, which helped us achieve maximum local detail preservation. Experiments demonstrated the superior performance of our algorithm in multi-exposure fusion, compared with other state-of-the-art fusion methods and some state-of-the-art tone mapping methods coupled with HDR reconstruction. Two possible applications of our algorithm, *i.e.,* multi-focus fusion and ambient/flash fusion, were also demonstrated. In addition, we introduced the hierarchical multivariate Gaussian conditional random field model and analyzed its relationship with HRW.

Chapter 5

The Multiscale Random Walks Framework and Its Application

5.1 Multiscale Random Walks

In this section, we propose a multiscale random walks (MRW) framework based on GRW and multiscale decomposition (MSD) schemes. Unlike HRW, MRW does not aim to provide an approximation of GRW but rather to produce robust results utilizing both inter- and intra-scale information.

5.1.1 Multiscale Decomposition

The pyramid transform (PT) and the wavelet transform (WT) are the two categories of MSD schemes that are most commonly employed in digital image processing and analysis tasks. Among different PT schemes, Laplacian pyramid transform (LPT) [32, 31] is one of the most frequently used. A Laplacian pyramid (LP) is constructed based on its corresponding Gaussian pyramid by subtracting two adjacent levels. Thus, a detail level in the LP encodes the local variations at that scale. The ratio of low-pass pyramid (RoLP) [184] is also constructed based on the Gaussian pyramid, but by taking the ratio of two adjacent levels. When an RoLP is used in subsequent calculations, the local contrast (*i.e.,* local luminance variation with respect to the background [198]) at each coefficient location in the RoLP is derived from the ratio. The gradient pyramid (either explicitly [33] or implicitly [144] constructed) is another type of PT, which is built by applying gradient filters of different orientations to each level of a Gaussian pyramid.

A standard WT technique is the discrete WT (DWT) [125], which decomposes a signal into an MSR using scaling (low-pass filtering) and wavelet (high-pass filtering) functions. One drawback of DWT is shift variance, *i.e.,* the wavelet coefficients are very sensitive to translations of the original signal. The redundant DWT (RDWT) [59] is an over-complete WT that solves the shift variance problem by dropping the subsampling operation in DWT. Selesnick *et al.* [165] proposed the dual-tree complex WT (DT-CWT), which also provides shift invariance but via complex-valued operations. In addition, DT-CWT provides increased directional selectivity over DWT and RDWT,

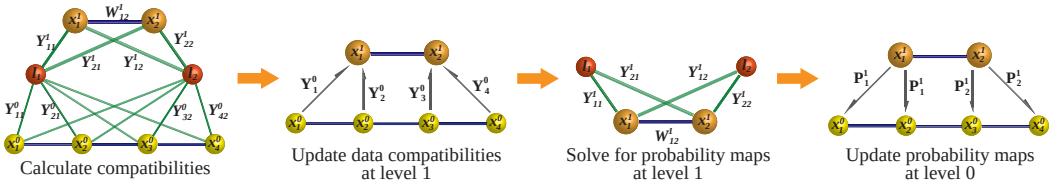


Figure 5.1: The basic procedure of MRW in a 1D 2-level case.

and a more compact representation than RDWT.

5.1.2 The MRW Framework

Given an image, we can build a multiscale representation (MSR) of it by applying an MSD scheme. Based on this MSR, a weighted graph can be constructed. The basic procedure of MRW is illustrated in Figure 5.1. For the sake of brevity, only a 1D 2-level case is depicted. For each level n ($n \in [0, N-1]$), we can define a variable set $\mathcal{X}^n$, where $x_i^n \in \mathcal{X}^n$ corresponds to the i-th coefficients in all the subbands at that level. At any level n, the scene nodes x_i^n's (yellow and orange nodes) are connected with their immediate neighbors $\mathcal{N}_i^n$. Let $\mathcal{X} = \{\mathcal{X}^n | n \in [0, N-1]\}$ denote the set of all variables (scene nodes). Each scene node x_i^n is also connected with all the label nodes $\mathcal{L}$ (red nodes). $W_{ij}^{N-1} \triangleq W(x_i^{N-1}, x_j^{N-1})$ is a function defined on the approximation level that models the neighborhood compatibility/similarity between adjacent scene nodes x_i^{N-1} and x_j^{N-1}. $Y_{ik}^n \triangleq Y(x_i^n, l_k)$ is a function defined on every decomposition level that models the data compatibility between each scene node x_i^n and each label node l_k. Information, $\mathbf{P}_i^n$'s and $\mathbf{Y}_i^n$'s, is transferred between corresponding nodes at adjacent levels, where $\mathbf{P}_i^n = \{P_{i,1}^n, \ldots, P_{i,K}^n\}$ and $\mathbf{Y}_i^n = \{Y_{i1}^n, \ldots, Y_{iK}^n\}$. This information-passing scheme is problem dependent. As in HRW, we let $P^t(l_k | x_i^n, \mathcal{D})$ denote the t-step transition probability determined based on observed data $\mathcal{D}$ that a random walker starting from location x_i^n reaches the label node l_k by the t-th step. The goal here is to calculate the steady-state probability $P_{i,k}^n \triangleq \lim_{t\to\infty} P^t(l_k | x_i^n, \mathcal{D})$ for all $x_i^n \in \mathcal{X}$ and all $l_k \in \mathcal{L}$.

In order to pass information upwards in the MSR, Y_{ik}^n needs to be updated as follows:

$$\tilde{Y}_{ik}^n = \begin{cases} Y_{ik}^n, & n = 0; \\ g(Y_{ik}^n, \{\tilde{Y}_{jk}^{n-1} | x_j^{n-1} \in [x_i^n]^{\downarrow}\}), & n \in [1, N-1], \end{cases} \tag{5.1}$$

where $g(\cdot, \cdot)$ denotes a problem-dependent function to balance the contributions from its first and second input terms and $[x_i^n]^{\downarrow}$ denotes the projection of x_i^n at level $n-1$. At the approximation level, $P_{i,k}^{N-1}$'s can be calculated using GRW. In order to pass information downwards in the MSR, $P_{i,k}^n$ needs to be updated as follows:

$$P_{i,k}^n = \begin{cases} P_{i,k}^n, & n = N-1; \\ h(Y_{ik}^n, \{P_{j,k}^{n+1} | x_j^{n+1} \in [x_i^n]^{\uparrow}\}), & n \in [0, N-2], \end{cases} \tag{5.2}$$

where $h(\cdot, \cdot)$ denotes a problem-dependent function to balance the contributions from its first and second input terms and $[x_i^n]^{\uparrow}$ denotes the projection of x_i^n at level $n+1$. The basic procedure

Algorithm 5.1 Basic steps of the multiscale random walks framework.

1: Apply MSD to the input data
2: Define problem-specific data compatibilities Y_{ik}^{n}'s at each level of the MSR
3: Define problem-specific neighborhood compatibilities W_{ij}^{N-1}'s at level $N-1$
4: **for** $n = 1$ to $N-1$ **do**
5: Update data compatibilities at level n using Equation (5.1)
6: **end for**
7: Obtain solution at level $N-1$ using GRW (Algorithm 3.1)
8: **for** $n = N-2$ to 0 **do**
9: Update the probability maps at level n using Equation (5.2)
10: **end for**

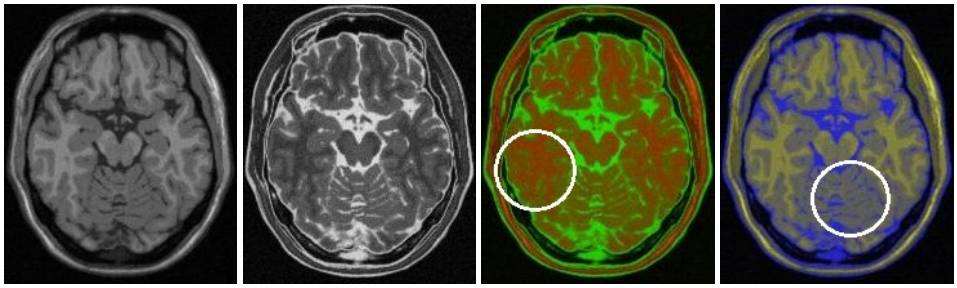
of MRW is summarized in Algorithm 5.1. In the following section, we will introduce a specific MRW-based algorithm for MSD-based medical image fusion.

5.2 Application of MRW in Volumetric Medical Image Fusion

5.2.1 Introduction

Medical imaging has become a vital component in routine clinical applications, such as diagnosis and treatment planning [124]. However, because each imaging modality only provides information in a limited domain, many studies prefer joint analysis of imaging data collected from the same patient using different modalities [36]. This requirement of joint analysis led to the introduction of image fusion into the medical field and the development of medical data-oriented fusion techniques [36, 175, 139]. The goal of image fusion is to generate a single composite image, which provides more accurate and reliable information than any individual source image and in which features may be more distinguishable [20]. Such an enhanced image facilitates visual perception (*e.g.,* by a radiologist) or further image processing (*e.g.,* by a computer-aided detection/diagnosis system, *i.e.,* a CAD system) [20].

Due to its compact and enhanced representation of information, image fusion has been employed in many medical applications. For instance, T_1-weighted (T1W) and T_2-weighted (T2W) magnetic resonance imaging (MRI) scans were fused to segment white matter lesions [187] or cerebral iron deposits [188] and to guide neurosurgical resection of epileptogenic lesions [122]. Computed tomography (CT) and MRI images were fused for neuronavigation in skull base tumor surgery [134]. Fusion of positron emission tomography (PET) and MRI images has proven useful for hepatic metastasis detection [47] and intracranial tumor diagnosis [22]. Single photon emission computed tomography (SPECT) and MRI images were fused for abnormality localization in patients with tinnitus [52]. Multiple fetal cardiac ultrasound scans were fused to reduce imaging artifacts [66]. In addition, the advantages of image fusion over side-by-side analysis of non-fused images have been demonstrated in lesion detection and localization in patients with neuroendocrine tumors [8] and in patients with pretreated brain tumors [9]. Even if image fusion is not performed explicitly, *e.g.,* by a CAD system, it is usually performed subconsciously by radiologists to compare images and better identify



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

Figure 5.2: Overlaying monochrome images using different color channels. Information is transferred from T1W and T2W MRI images into a single image at the cost of reduced image contrast. (a) T1W MRI image; (b) T2W MRI image; (c) Overlaid image using the red and green channels as in [187]; (d) Overlaid image using the yellow and blue channels as in [149].

abnormalities [82].

A straightforward multi-modal image fusion method is to overlay the source images by manipulating their transparency attributes [199, 64] or by assigning them to different color channels [187, 149]. This overlaying scheme is a fundamental approach in color fusion, a type of image fusion that uses color to expand the amount of information conveyed in a single image [173], but it does not necessarily enhance the image contrast or make image features more distinguishable. An example is given in Figure 5.2. The overlaying schemes transfer information from T1W and T2W MRI scans into a single image, but at the cost of reduced image contrast (*e.g.,* in the temporal lobe and cerebellum as indicated by the white circles). In this section, we propose a fusion rule that blends the pixel values in the monochrome source images to combine information while preserving or enhancing image contrast. In addition, we show how color fusion can benefit from the monochrome fusion results.

Image fusion can be performed at three different levels, *i.e.,* pixel/data level, feature/attribute level, and symbol/decision level, each of which serves different purposes [146, 20, 173]. Pixel-level fusion determines a pixel in the fused image from a set of pixels in the source images. Feature-level fusion applies the extracted image features (*e.g.,* segmented objects or edge maps) to guide the pixel merging process or form a joint feature vector. Symbol-level fusion combines abstract representations of the source images extracted using such techniques as classification. Among these three levels, pixel-level fusion and feature-level fusion are more closely related when both output a single fused image, while the output of symbol-level fusion is always a certain decision. Pixel-level fusion directly combines the original information in the source images and is computationally efficient, but it is more sensitive to image noise compared to feature-level fusion [146]. However, the fusion quality of feature-level methods largely depends on the performance of the feature extraction technique employed. In practice, a fusion task may also involve more than one level simultaneously [146].

According to whether MSD is used, pixel-level fusion methods can be classified as MSD-based or non-MSD based. Compared to the latter, MSD-based methods have the advantage of extract-

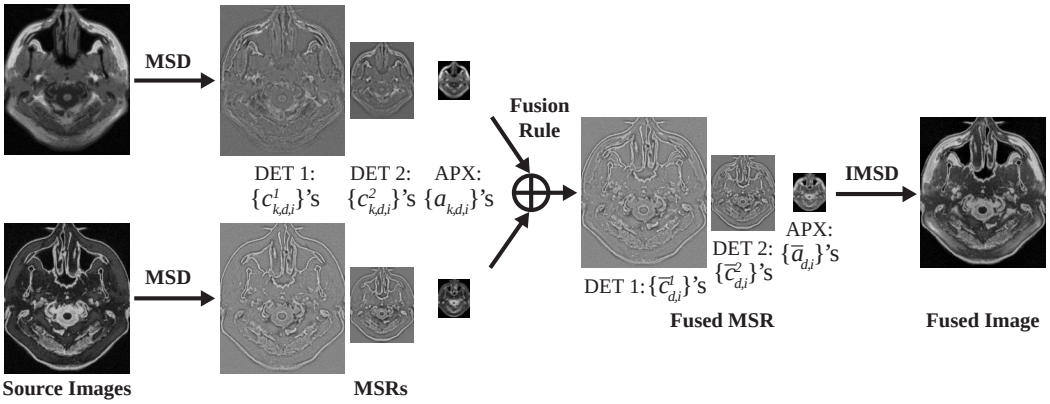


Figure 5.3: General procedure of MSD-based image fusion in a 2D case.

ing and combining salient features at different scales, and therefore normally produce images with greater information content [173]. The general procedure of MSD-based fusion is illustrated in Figure 5.3. First, the source images are transformed to multiscale representations (MSRs) using MSD. An MSR is a pyramidal structure with successively reduced spatial resolution; it usually has one approximation level storing low-pass coefficients and several detail levels storing high-pass or band-pass coefficients. Then, a certain fusion rule is applied to merge coefficients at different scales. Finally, an inverse MSD (IMSD) is applied to the fused MSR to generate the final image.

Two directions can be explored in MSD-based fusion to enhance the fusion quality: advanced MSD schemes and effective fusion rules. Here, we focus on the latter and propose a novel cross-scale (CS) fusion rule, where the belongingness/membership of each fused coefficient to each source image is calculated. Although we focus on pixel-level fusion, this proposed fusion rule may be extended to feature-level or symbol-level fusion as well. Unlike previous methods, our fusion rule calculates an optimal set of coefficients for each scale taking into account large neighborhood information, which guarantees intra- and inter-scale consistencies, so that coefficients with similar characteristics are fused in a similar way and artifacts (such as aliasing artifacts at object boundaries) are avoided in the results. The effectiveness of this new fusion rule is validated through experiments on 3D medical image fusion. Although it is possible to fuse individual 2D slices in 3D images/volumes separately, the results are not of the same quality as those of 3D fusion due to the lack of between-slice information in the fusion process [7]. Here, therefore, we apply MSD and our CS rule directly to the 3D volumes. An effective color fusion scheme utilizing the monochrome fusion results is also proposed.

5.2.2 Related Work

This section focuses on reviewing key MSD-based image fusion techniques. Although theoretically the decomposition of an image can be performed iteratively until there is only one pixel in each dimension at the approximation level, this will result in serious bias and inaccuracy in the feature

selection at low-resolution levels, which impairs the fusion quality [144]. Typically, only a few decomposition levels are therefore used in practice, and some analyses on the number of decomposition levels for different MSD schemes in some fusion tasks were conducted in [136, 109]. For applications of image fusion in medicine and other fields, please refer to the overview articles by Pattichis *et al.* [139] and by Hall and Llinas [75].

5.2.2.1 Fusion Rules

In addition to the MSD scheme, the other key factor affecting fusion results is the fusion rule. According to [20], a fusion rule is the processing that determines the formation of the fused MSR from the MSRs of the source images, and it normally consists of four key components, *i.e.,* activity-level measurement, coefficient grouping, coefficient combination, and consistency verification. In this section, we give a brief review of some representative schemes in these four steps. Please refer to [136, 20] for more detailed discussions and other types of fusion methods (*e.g.,* estimation theory-based method [203]).

Activity-Level Measurement The activity-level measurement reflects the salience of each coefficient in an MSR [109], and it can be categorized into three classes, *i.e.,* coefficient-based activity (CBA), window-based activity (WBA), and region-based activity (RBA) [20]. A CBA measure evaluates each coefficient independently and normally describes the activity level of a coefficient using its absolute value. A WBA measure uses the information within a window to evaluate the coefficient at the window center. A popular choice is the rank filter-based WBA, where the maximum value within a window is normally selected as in [107]. More sophisticated WBA measures also exist in the literature, such as image statistics-based WBA [136]. The concept of RBA is similar to that of WBA except that irregular-shaped regions are used instead of regular-shaped windows. RBA is one way to achieve feature-level fusion; it requires segmentation on (the MSRs of) the source images [146, 106]. In our CS rule, the activity-level measure is used to determine the salience at each coefficient and there is no restriction on the type of measures to be employed. The focus of our CS rule is to provide a unified framework combining the other three key components in a fusion rule, which were usually treated separately in previous methods.

Coefficient Grouping The coefficient grouping schemes can be roughly divided into three categories, *i.e.,* no grouping (NG), single-scale grouping (SG), and multiscale grouping (MG) [20]. NG means that each coefficient is fused independently; SG means that corresponding coefficients between different subbands at the same decomposition level are fused in the same way; and MG is more restrictive than SG because it also requires that corresponding coefficients between different scales take the same fusion decision. A cross-band SG (CBSG) scheme was proposed in [143], where the same fusion decision for every set of corresponding detail coefficients at the current scale is made based on the sum of their activity levels and their corresponding coefficients at a higher

scale. In [209], an MG scheme was proposed in which the fusion decision for every set of corresponding coefficients across all scales in an MSR is made based on the weighted average of their activity levels. Our CS rule performs similar to MG, but does not impose such a hard constraint on the fusion decision. Instead, the influence on each coefficient from their corresponding coefficients at adjacent scales is reflected in the membership calculation, and the fusion decision of a coefficient is determined based on its calculated membership. Please refer to Section 5.2.4 for a detailed description of the membership calculation.

Coefficient Combination One common coefficient combination scheme for the detail levels is the choose-max (CM) strategy, *i.e.,* selecting the coefficient with the highest activity level at each location from the MSRs of the source images as the coefficient at that location in the MSR of the fused image [20]. A common combination scheme for the approximation level is taking the average (AVG). Another popular scheme for the approximation level is the weighted average (WA) strategy proposed in [33]. A linear weighting function is applied when the local correlation between corresponding coefficients in a neighborhood in the MSRs of the source images is above a threshold. Some more complicated combination schemes were also proposed by previous researchers; please refer to [136] for more detailed discussions. Our CS rule does not apply combination schemes based directly on coefficient activity levels, but combines coefficients based on their memberships, which results in a more effective scheme utilizing inter- and intra-scale information.

Consistency Verification The consistency verification schemes ensure neighboring coefficients are fused in a similar manner [20]. A majority filter was used in [107] to apply window-based verification (WBV) at each individual scale. A cross-band verification (CBV) scheme was proposed in [143], where the detail coefficients at the current level of the fused MSR are recalculated if their corresponding coefficients at a lower level come from the MSR of a different source image. CBV was designed to comply with CBSG. It is also possible that no verification (NV) is applied. Our CS rule does not perform explicit verification, but embeds verification in the coefficient membership calculation process.

5.2.2.2 Medical Image Fusion

The overlaying schemes were discussed in Section 5.2.1, and here we discuss some activity-level measures and non-MSD-based methods proposed for medical image fusion. Please note that the MSD-based fusion methods discussed in the previous sections can be applied directly to medical image fusion; the DWT+CBA+NG+AVG+CM+NV method, for example, was used in [152] for quality enhancement of real-time 3D echocardiography. A multi-channel pulse coupled neural network was proposed in [190] for 2D medical image fusion. However, the fusion results suffered from loss of local contrast, which made details less distinguishable. DWT+WBA+NG+CM+CM+WBV

was applied to fuse 2D medical images in [205], where a visibility-based WBA and a local variance-based WBA were proposed for APX and DETs, respectively. In contrast, our focus here is a novel fusion rule rather than a specific activity-level measure.

In [7], a 3D shunting neural network was applied in order to achieve information decorrelation between source images, and the shunted images were then assigned to different color planes to achieve color fusion of 3D medical images. In our method, a more efficient scheme is proposed, where color fusion of two images is achieved as a natural extension to monochrome fusion. In addition, we emphasize both achromatic and chromatic color contrasts. In [189], the fused 3D medical image was obtained by minimizing a quadratic objective function defined using a gradient-based field. However, that method tends to cause artifacts in smooth image regions due to the form of the objective function. Such artifacts are less likely to appear in our method, where consistencies are emphasized.

5.2.3 Problem Formulation

The source images are assumed to be spatially registered, which is a common assumption in image fusion [20]. Various techniques [124, 166] can be applied to medical image registration. We follow the MSD-based fusion procedure, as illustrated in Figure 5.3. Let $c^n_{k,d,i}$ and $\bar{c}^n_{d,i}$ denote the i-th coefficients in the d-th subband at the n-th detail level of the MSR of the k-th source image and the fused image, respectively, where $n \in [0, N-1]$. Let $a_{k,d,i}$ and $\bar{a}_{d,i}$ denote the i-th coefficients in the d-th subband at the approximation level of the MSR of the k-th source image and the fused image, respectively. We assume that a subband at the approximation level has the same size as a subband at the N-th detail level. For PT schemes where the approximation level is at a higher level, applying an extra step of band-pass filtering can fulfill this assumption. Let $M : \{\bar{c}^n_{d,i}, \bar{a}_{d,i}\} \times \{c^n_{k,d,i}, a_{k,d,i}\} \rightarrow [0, 1]$ be a function representing the (partial) membership of $\bar{c}^n_{d,i}$ (or $\bar{a}_{d,i}$) to the MSR of the k-th source image, *i.e.,* the proportion of the contribution from $c^n_{k,d,i}$ (or $a_{k,d,i}$) to $\bar{c}^n_{d,i}$ (or $\bar{a}_{d,i}$) among all corresponding coefficients $\{c^n_{k,d,i}|k = 1, \ldots, K\}$ (or $\{a_{k,d,i}|k = 1, \ldots, K\}$). The memberships can be determined based on local and/or global information in the MSRs. To simplify notation, let $M^n_{k,d,i}$ and $M_{k,d,i}$ denote the coefficient memberships at the n-th detail level and the approximation level, respectively. We have $\sum_k M^n_{k,d,i} = 1$ and $\sum_k M_{k,d,i} = 1$.

For each subband of a detail level, where the corresponding coefficients among different MSRs are usually quite distinct from each other, a fused coefficient can be determined as the one with the highest membership:

$$\bar{c}^n_{d,i} = \arg\max_{c^n_{k,d,i}, k:1\ldots K} M^n_{k,d,i}. \tag{5.3}$$

For the approximation level, where the corresponding coefficients usually exhibit less diversity compared to those at a detail level, a fused coefficient can be determined as a weighted average of all of

its corresponding coefficients based on their memberships:

$$\bar{a}_{d,i} = \sum_{k=1}^{K} M_{k,d,i} a_{k,d,i}. \tag{5.4}$$

5.2.4 Cross-Scale Coefficient Selection

The proposed cross-scale fusion rule based on MRW aims to pass information within and between each decomposition level so that the fused image preserves the most details from the source images while exhibiting minimal artifacts. The basic steps are: 1) pass salient information from a lower level to a higher level in an MSR until the approximation level is reached; 2) calculate the memberships of each fused coefficient at the approximation level using the salient information that has been passed on; 3) use these memberships to guide the coefficient selection at the detail levels.

Let $A^n_{k,d,i}$ denote the activity level of $c^n_{k,d,i}$. In order to impose inter-scale consistency, the activity levels of coefficients at a lower decomposition level are passed to a higher level as follows:

$$\tilde{A}^n_{k,d,i} = \begin{cases} \operatorname{erf}(A^n_{k,d,i}), & n = 0; \\ \max(\operatorname{erf}(A^n_{k,d,i}), [\tilde{\mathbf{A}}^{n-1}_{k,d}]^{\downarrow 2}_i), & n \in [1, N-1], \end{cases} \tag{5.5}$$

where $\tilde{\mathbf{A}}^n_{k,d}$ denotes the vector containing all $\tilde{A}^n_{k,d,i}$'s in the d-th subband of the MSR of the k-th source image; $[\cdot]^{\downarrow 2}$ denotes downsampling by a factor of 2 in each dimension; and the subscript $[\cdot]_i$ denotes the i-th coefficient. The maximum function is used as a way to ensure inter-scale consistency by allowing the calculation at higher scales to access the most representative salient information at lower scales, which we take as those with high activity levels. $\operatorname{erf} : \mathbb{R} \to [-1, 1]$ is called the Gauss error function, a sigmoid-shaped function. The magnitudes of activity levels of coefficients across different detail levels can vary significantly, which makes it difficult to compare the relative importance of salient information across scales. This nonlinear function $\operatorname{erf}(\cdot)$ compresses the activity levels into the same range $[0, 1]$ for non-negative activity levels, which gives a more reasonable comparison of salient information. In addition, it also depresses very high activity levels, which sometimes may be caused by image noise.

At the approximation level, the passed salient information $\tilde{A}^{N-1}_{k,d,i}$'s and the approximation coefficients $a_{k,d,i}$'s are used together to calculate the memberships $M_{k,d,i}$'s. One simple scheme is to directly take normalized $\tilde{A}^{N-1}_{k,d,i}$'s as $M_{k,d,i}$'s. However, this scheme does not utilize the visual information embedded in $a_{k,d,i}$'s, which is crucial for producing locally smoothed solutions. GRW (Section 3.1) is employed to calculate $M_{k,d,i}$'s, which we consider as the steady-state probabilities in the random walks context, by minimizing K similarly-defined energy functions. Let $\mathbf{M}_{k,d}$ denote the vector containing all $M_{k,d,i}$'s, *i.e.,* memberships of all of the approximation coefficients in the d-th subband of the fused MSR to the k-th source image. The solution to the k-th energy function is given by:

$$\mathbf{L}_d \mathbf{M}_{k,d} = \tilde{\mathbf{A}}^{N-1}_{k,d}. \tag{5.6}$$

The matrix $\mathbf{L}_d$ encodes the interactions between adjacent coefficients. The entry in the i-th row and j-th column of $\mathbf{L}_d$ is defined as:

$$\mathbf{L}_{d,ij} = \begin{cases} \sum_{\bar{a}_{d,s} \in \mathcal{N}_{d,i}} W_{d,is} + \sum_k \tilde{A}^N_{k,d,i}, & i = j; \\ -W_{d,ij}, & \bar{a}_{d,j} \in \mathcal{N}_{d,i}; \\ 0, & \text{otherwise}, \end{cases} \tag{5.7}$$

where $\mathcal{N}_{d,i}$ is the first-order neighborhood of $\bar{a}_{d,i}$. $W_{d,ij}$ represents the expected similarity between $\bar{a}_{d,i}$ and $\bar{a}_{d,j}$ based on the observed approximation coefficients in the MSRs of the source images. $W_{d,ij}$ is defined as follows:

$$W_{d,ij} = \gamma \prod_{k=1}^{K} \exp(-\frac{\|a_{k,d,i} - a_{k,d,j}\|}{\sigma}), \tag{5.8}$$

where γ and σ are weighting factors. Equation (5.8) assigns a higher penalty to a coefficient pair with greater similarity. Therefore, similar coefficients are more likely to be assigned similar memberships to ensure intra-scale consistency.

Once $M_{k,d,i}$'s are calculated for the approximation level ($n = N$) using Equations (5.6) to (5.8), they are passed down to guide the membership calculation at each detail level in order to impose inter-scale consistency:

$$M^n_{k,d,i} = \begin{cases} M_{k,d,i}, & n = N - 1; \\ \frac{1}{\alpha}[\phi * (\tilde{\mathbf{A}}^n_{k,d} \odot [\mathbf{M}^{n+1}_{k,d}]^{\uparrow 2})]_i, & n \in [0, N-2], \end{cases} \tag{5.9}$$

where α is a normalization factor rendering $\sum_k M^n_{k,d,i} = 1$; $[\cdot]^{\uparrow 2}$ denotes upsampling by a factor of 2 in each dimension followed by interpolation; $*$ denotes convolution; $\odot$ denotes component-wise multiplication; and ϕ is a low-pass filter that helps to achieve intra-scale consistency. In our current implementation, ϕ is taken as a $5 \times 5 \times 5$ Gaussian filter for each decomposition level of a volume. The filter is constructed using the separable generating kernel in [32] with parameter $a = 0.375$[1]. In the following, we give two specific examples of applying our CS rule with two popular MSD schemes, LPT and DWT. However, the concept may be extended to other MSD schemes as well, such as RoLP and DT-CWT.

5.2.4.1 LPT+CS Based Fusion

In order to combine our CS rule with LPT, an extra step of band-pass filtering at the approximation level is needed to produce a corresponding detail level. This detail level is only used in the coefficient membership calculation and is not involved in IMSD. Please note that there is only one subband at each decomposition level for LPT. The whole process of LPT+CS-based fusion is summarized in Algorithm 5.2.

5.2.4.2 DWT+CS Based Fusion

Although it is possible to apply the same scheme for LPT-based fusion to DWT-based fusion, each detail level for DWT contains $D = 2^S - 1$ subbands (S is the number of dimensions of the signal),

[1] This roughly corresponds to a filter generated by a Gaussian kernel with a standard deviation of 1.05.

Algorithm 5.2 Basic steps of the LPT+CS fusion scheme.

1: Apply N-level LPT to each source image
2: Apply band-pass filtering to the approximation levels
3: Compute $\tilde{A}^n_{k,1,i}$'s for detail level 0 to $N-1$ using Equation (5.5)
4: Compute $M_{k,1,i}$'s at the approximation level using Equations (5.6) to (5.8)
5: Compute $M^n_{k,1,i}$'s for detail level $N-1$ to 0 using Equation (5.9)
6: Select coefficients for the approximation level of the fused MSR using Equation (5.4)
7: Select coefficients for the detail levels of the fused MSR using Equation (5.3)
8: Apply inverse LPT to the fused MSR

Algorithm 5.3 Basic steps of the DWT+CS fusion scheme.

1: Apply N-level DWT to each source image
2: Compute $\tilde{A}^n_{k,1,i}$'s for detail level 0 to $N-1$ using Equations (5.10) and (5.5)
3: Compute $M_{k,1,i}$'s at the approximation level using Equations (5.6) to (5.8)
4: Compute $M^n_{k,1,i}$'s for detail level $N-1$ to 0 using Equation (5.9)
5: Assign $M^n_{k,1,i}$'s to their corresponding $M^n_{k,d,i}$'s
6: Select coefficients for the approximation level of the fused MSR using Equation (5.4)
7: Select coefficients for the detail levels of the fused MSR using Equation (5.3)
8: Apply inverse DWT to the fused MSR

which will result in significantly increased computational cost and poor consistency among subbands. Therefore, corresponding coefficients in different subbands at the same scale are evaluated together and the same membership is assigned to all of them. Hence, $A^n_{k,d,i}$ used in Equation (5.5) is substituted with the following:

$$\hat{A}^n_{k,d,i} = \max(A^n_{k,1,i}, \ldots, A^n_{k,D,i}). \tag{5.10}$$

The whole process of DWT+CS-based fusion is summarized in Algorithm 5.3.

5.2.5 Color Fusion

In this section, we introduce an efficient color fusion scheme for the case of two monochrome source images. The color fusion scheme, which utilizes the fusion result from the previous section to further enhance image contrast, is inspired by the color opponency theory in physiology [79], which states that human perception of achromatic and chromatic colors occurs in three independent dimensions, *i.e.,* black-white (luminance), red-green, and yellow-blue. Contrast sensitivity in these three dimensions has been studied by many researchers [131, 159, 182]. The contrast sensitivity function of luminance shows band-pass characteristics, while the contrast sensitivity functions of both red-green and yellow-blue show low-pass behavior. Therefore, luminance sensitivity is normally higher than chromatic sensitivity except at low spatial frequencies. Hence, the fused monochrome image, which provides combined information and good contrasts, should be assigned to the luminance channel to exploit luminance contrast. In addition, the color-fused image should also provide good contrasts in the red-green and/or yellow-blue channels in order to fully exploit human color perception. To achieve this, we can consider that red, green, yellow, and blue are arranged on a color circle as in [79], where the red-green axis is orthogonal to the yellow-blue axis and color (actually

its hue) transits smoothly from one to another in each quadrant. Then, in order to maximize color contrast/dissimilarity between an object and its surroundings in the color-fused image, their hues should come from two opposite quadrants, or at least from two orthogonal hues on the color circle. With these considerations in mind, we have developed the following scheme.

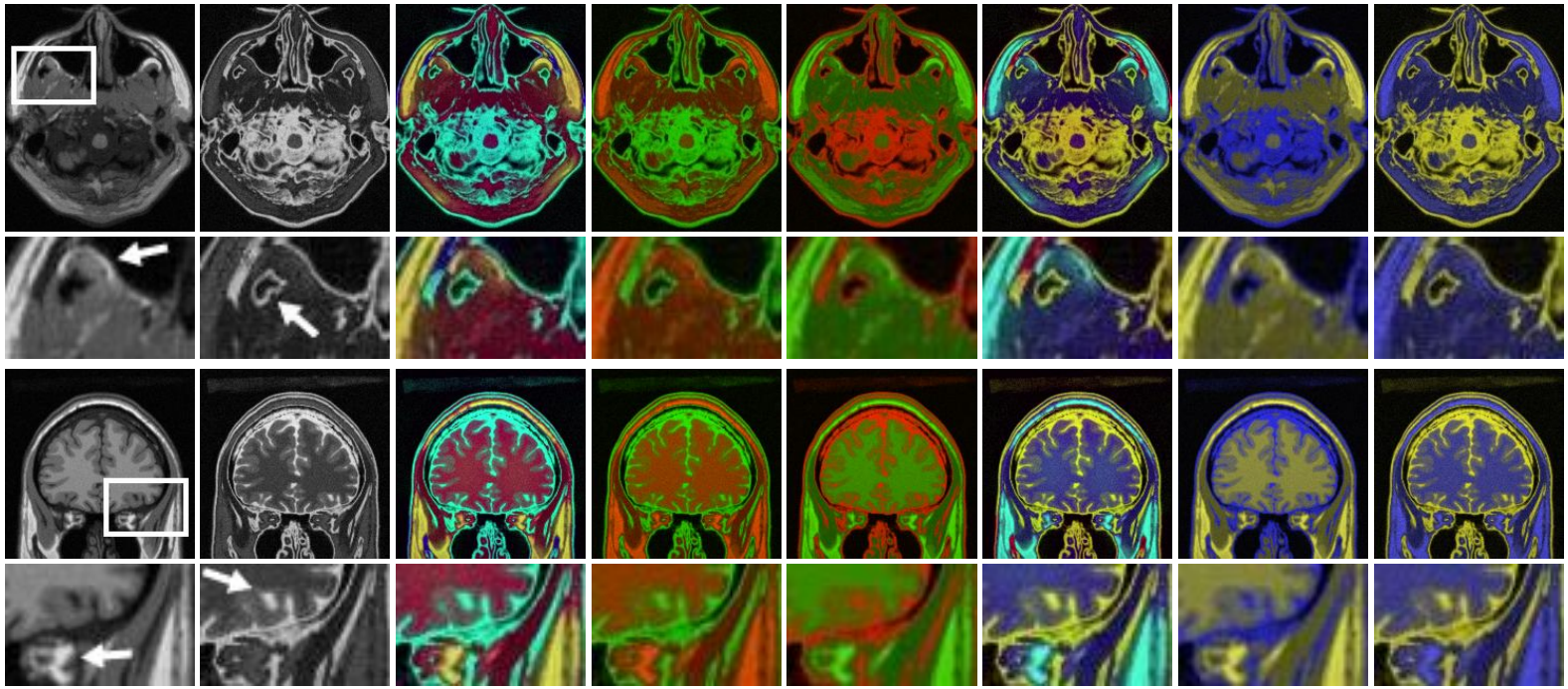
Let $\mathbf{I}_1$ and $\mathbf{I}_2$ denote the two source images and $\bar{\mathbf{I}}$ the monochrome fused image. $\bar{\mathbf{I}}$ is considered as the luminance image of the color-fused image $\bar{\mathbf{I}}^c$. Therefore, if we consider the YUV color space, $\bar{\mathbf{I}}$ is the Y component. Let $\bar{\mathbf{I}}^c_r$, $\bar{\mathbf{I}}^c_g$, and $\bar{\mathbf{I}}^c_b$ denote the red, green, and blue color planes of $\bar{\mathbf{I}}^c$, respectively. The source images are assigned to the red and blue planes in the RGB color space (*i.e.,* $\bar{\mathbf{I}}^c_r = \mathbf{I}_1, \bar{\mathbf{I}}^c_b = \mathbf{I}_2$ or $\bar{\mathbf{I}}^c_r = \mathbf{I}_2, \bar{\mathbf{I}}^c_b = \mathbf{I}_1$), and the green plane is derived by reversing the calculation of the Y component from the RGB color space:

$$\bar{\mathbf{I}}^c_g = (\bar{\mathbf{I}} - 0.299\bar{\mathbf{I}}^c_r - 0.114\bar{\mathbf{I}}^c_b)/0.587. \tag{5.11}$$

This scheme provides more contrast enhancement than the overlaying schemes, because it fully utilizes color opponency in human perception. A visual comparison of slices from two directions is provided in Figure 5.4. An inset is given below each slice, which clearly shows the improved contrast using our scheme, as indicated by the white arrows (*i.e.,* the sarcolemma shown in the T1W scan and the mastoid air cells shown in the T2W scan in the upper row, the orbital apex in the T1W scan and the sulcus in the T2W scan in the lower row). Researchers [7, 191, 6] have previously studied opponent-color fusion, which is essentially based on opponent processing. After intermediate fused and/or enhanced grayscale images are generated by opponent processing, they are either directly assigned to different color planes (in the case of two source images) or assigned in a way that emphasizes chromatic color contrast (in the case of three or more source images) to form the color-fused image. This is different from our scheme, which aims to maximize both achromatic and chromatic color contrasts in the color-fused image.

5.2.6 Experimental Results and Discussion

The performance of the proposed cross-scale fusion rule was evaluated on volumetric image fusion of T1W and T2W MRI scans using both synthetic and real data (Section 5.2.6.1). After this validation, we demonstrate the capability of our fusion rule to fuse other modalities (Section 5.2.6.3). In addition, we have consulted a neurosurgeon and a radiologist. In their opinion, our method not only provides enhanced representations of information, which is useful in such applications as diagnosis and neuronavigation, but also offers them the flexibility of combining modalities of their choice, which is important because the data types required are normally application-dependent.



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

Figure 5.4: Comparison between our color fusion scheme and the overlaying scheme. The first row shows a slice along the axial axis from each volume. The second row shows slices along the coronal axis. Our scheme provides more image contrast, which makes the combined details more distinguishable, such as those indicated by the arrows in the insets. (a) T1W MRI slices; (b) T2W MRI slices; (c) Our color fusion results with $\bar{\mathbf{I}}_r^c = \mathbf{I}_1, \bar{\mathbf{I}}_b^c = \mathbf{I}_2$; (d) Overlaid images with $\bar{\mathbf{I}}_r^c = \mathbf{I}_1, \bar{\mathbf{I}}_g^c = \mathbf{I}_2$; (e) Overlaid images with $\bar{\mathbf{I}}_r^c = \mathbf{I}_2, \bar{\mathbf{I}}_g^c = \mathbf{I}_1$; (f) Our color fusion results with $\bar{\mathbf{I}}_r^c = \mathbf{I}_2, \bar{\mathbf{I}}_b^c = \mathbf{I}_1$; (g) Overlaid images with $\bar{\mathbf{I}}_r^c = \bar{\mathbf{I}}_g^c = \mathbf{I}_1, \bar{\mathbf{I}}_b^c = \mathbf{I}_2$; (h) Overlaid images with $\bar{\mathbf{I}}_r^c = \bar{\mathbf{I}}_g^c = \mathbf{I}_2, \bar{\mathbf{I}}_b^c = \mathbf{I}_1$.

5.2.6.1 Validation of the Proposed Cross-Scale Fusion Rule

In our experiments, only CBA was employed as the activity-level measurement. This is because: 1) as evaluated in [136], CBA is one of the measures that give the best performance; 2) our CS rule places no restriction on the activity level and employing the same activity-level measure in all methods gives a fair comparison between our CS rule and existing fusion rules. Three grouping schemes (NG, CBSG and MG) were considered, as were two combining schemes for APX (AVG and WA). In addition, we considered one combining scheme for DETs (CM) and three verification schemes (NV, WBV, and CBV). Please note that only CM can be used for the APX when MG is used due to the nature of MG and that CBV can only be used with CBSG due to the nature of CBSG. For WA and WBV, we took a $5 \times 5 \times 5$ neighborhood/window. We used the suggested threshold value of 0.85 in WA [33]. The two suggested threshold values of 0.2 and 0.5 in [107] were used in CBV. Two MSD schemes were considered: LPT and DWT. For DWT, we used the wavelet package provided by [35] and employed the 2-band orthogonal near-symmetric filters with $K = 2, L = 6$ [1]. 5-level decomposition was applied in LPT and 4-level decomposition was applied in DWT. Please note that the APX in an N-level LPT has the same size as that in an $(N - 1)$-level DWT. All methods were implemented in Matlab, run on the same computer, and applied directly to the 3D volumes rather than 2D slices. Two free parameters (γ and σ) are in our CS rule. Taking $\gamma = 10, \sigma = 1$ produced the best results in the experiments on the synthetic data; we therefore also used these values in the experiments on real data.

Objective Evaluation Metric The objective metric $Q^{AB/F}$ [202] was employed in evaluating the fusion quality. This metric does not require an ideal composite image, which is difficult to get in practical cases, as a reference image. $Q^{AB/F}$ has been proven to correspond well with subjective tests among different metrics [142] and is widely used to assess fusion quality [117, 109]. $Q^{AB/F}$ measures the amount of edge information correctly transferred from source images to the fused image; a $Q^{AB/F}$ score is within the range $[0, 1]$, where a higher score indicates a better fusion result.

Evaluation Using Synthetic Data Our CS rule was evaluated on two sets of realistic simulated 3D MRI brain images from BrainWeb [98], which were constructed based on real scans. The scans in each set are spatially registered due to the nature of the simulation. Each scan has $181 \times 217 \times 181$ voxels with 12-bit precision, and the size of each voxel is 1 mm^3. One set contains images of a normal brain, and the other contains images of a brain with moderate multiple sclerosis lesions. The objective evaluation results for LPT- and DWT-based fusion are summarized in Tables 5.1 and 5.2 (first two rows), respectively. Our CS rule has the best performance in transferring edge information on both datasets, as indicated by the highest $Q^{AB/F}$ scores. LPT performs better than DWT for these datasets. When the other settings are the same, WBV performs better than NV and CBV;

and AVG performs better than WA. Therefore, for the sake of brevity, only NG+AVG+CM+WBV, CBSG+AVG+CM+WBV, and MG+CM+CM+WBV are visually compared with our CS rule on the lesion dataset in Figure 5.5. Only one slice along the axial axis from each volume is displayed. These slices are normalized to 8-bit precision only for the purpose of a visual comparison of fusion qualities. Please note that when viewed using a medical image visualization software (*e.g.,* VolView), the voxel values are usually not normalized, but instead, the display range of the voxel values and the image contrast can be interactively adjusted via the window/level setting. The formulation of image fusion as membership calculation, together with the consistency constraints imposed in our CS fusion rule, helps to ensure that associated coefficients are fused similarly in order to avoid fusion artifacts, and to ensure that salient information (*e.g.,* edge information) is correctly transferred from the source images to the fused image. Therefore, compared to other fusion rules, our CS rule not only correctly combined information with high consistency with the source images, but also provided good local contrasts (*e.g.,* between ventricles, grey matter and white matter). As shown in the insets below each slice (refer to the scalp, diploë, grey matter, and white matter indicated by the white arrows), our CS rule successfully eliminated the blocking artifacts shown in MG when coupled with LPT, and it eliminated the aliasing artifacts in NG, CBSG, and MG when coupled with DWT.

Evaluation Using Real Data We also tested our method on fifty real datasets of normal brains randomly selected from the NIH Pediatric MRI Data Repository [28]. Each dataset contains stereotaxically registered MRI scans of the same subject. We used the T1W and T2W scans in each dataset to evaluate the proposed fusion method. Each scan has $197 \times 233 \times 189$ voxels with 32-bit precision, and the size of each voxel is 1 mm^3. The objective evaluation results on five representative datasets for LPT- and DWT-based fusion are summarized in Tables 5.1 and 5.2 (last five rows), respectively. According to the $Q^{AB/F}$ scores, the fusion rules are ranked from 1 (best) to 13 (worst) for each dataset. The average rankings on all the 50 datasets are presented in Table 5.3. Due to the various image conditions in different datasets, a high $Q^{AB/F}$ score in one dataset may be low in a different dataset (*e.g.,* compare the last two rows in Table 5.2). Therefore, in order to give a relatively fair comparison of the average performance of all the 13 fusion rules on the 50 datasets, we used this average ranking method. Our CS rule has the best performance in transferring edge information on the average. With the other settings the same, AVG and WA have very close performance for both LPT and DWT; WBV has better performance than NV and CBV for LPT; NV and WBV have similar performance for DWT+NG; WBV has better performance than NV and CBV for DWT+CBSG; and NV has better performance than WBV for DWT+MG. The results on a representative case (dataset #1002) are visually compared in Figure 5.6. For the sake of brevity, only NG+AVG+CM+WBV, CBSG+AVG+CM+WBV, LPT+MG+CM+CM+WBV, and DWT+MG+CM+CM+NV are visually compared with our CS rule. For the same reason as for Figure 5.5, the slices are normalized to 8-bit

precision. As shown in the insets (refer to the cerebellum and the regions around the fourth ventricle as indicated by the white arrow), our CS rule produced better consistencies at region boundaries and gave better local details and contrasts. As Table 5.3 reveals, MG+CM+CM+NV performs, on average, at a level close to that of our CS rule in DWT-based fusion, and MG performs slightly better on some datasets as shown in Tables 5.1 and 5.2. However, this interpretation is based on the $Q^{AB/F}$ metric, which only considers edge information. Without correctly transferred neighborhood information, the fusion results (even with higher $Q^{AB/F}$ scores) tend to contain artifacts. These artifacts will impair further analysis either by a radiologist or by image processing software. Our CS rule effectively minimizes the occurrence of such artifacts. Figure 5.7 shows an example. Our CS rule produced better neighborhood consistencies in the white matter.

5.2.6.2 Fusion Quality and the Number of Decomposition Levels

In this experiment, we analyzed the relationship between the fusion quality using our CS rule and the number of decomposition levels (NDL). The seven datasets in Table 5.1 were used in this analysis, and we considered four combinations: synthetic+LPT, synthetic+DWT, real+LPT, and real+DWT. For LPT, 3- to 7-level decompositions were considered; for DWT, 2- to 6-level decompositions were considered[2]. To better depict the relationship between fusion quality and NDL, for each MSD scheme on each dataset, we used the $Q^{AB/F}$ score obtained at the lowest NDL as the reference and computed the ratio between the $Q^{AB/F}$ score obtained at a higher NDL and the reference. A ratio larger/less than 1 means an increase/decrease in the score. Figure 5.8(a) shows the average ratios for each of the four combinations. On the horizontal axis, the size of the approximation level, rather than the NDL, is used to normalize between LPT and DWT. On this axis, $1/M$ means that the approximation level is $1/M$ of the original resolution in each dimension, which corresponds to a $(\log_2 M + 1)$-level LPT or a $(\log_2 M)$-level DWT. The blue line denotes synthetic+LPT, red line synthetic+DWT, green line real+LPT, and black line real+DWT. On average, the highest $Q^{AB/F}$ score is achieved when the approximation level size is $1/16$, and the performance of our CS rule is less affected by the NDL in LPT than in DWT. The average processing times (in seconds) of our CS rule for each of the four combinations are compared in Figure 5.8(b). It can be concluded that when the approximation level size is above $1/8$, there is little improvement on the time performance. The average MSD time of LPT on the seven datasets is about 3 seconds, while DWT takes about 11 seconds. The IMSD time of LPT is about 39% of its MSD time, while the IMSD time of DWT is about 54% of its MSD time. Considering MSD time, fusion time, and IMSD time together, LPT+CS takes only 36% of the time required by DWT+CS on the synthetic datasets, and 45% on the real datasets. From this analysis and the evaluations in Section 5.2.6.1, it can be concluded that when combined with the same fusion rule, LPT has better performance than DWT in terms of fusion quality and time performance under our current implementation environment and experimental setup.

[2]For a very low NDL, a large linear system needs to be solved, which is intractable in practice for 3D medical data; hence, these NDLs were selected.

Table 5.1: Objective evaluation results on LPT-based fusion of synthetic and real data using the $Q^{AB/F}$ metric.

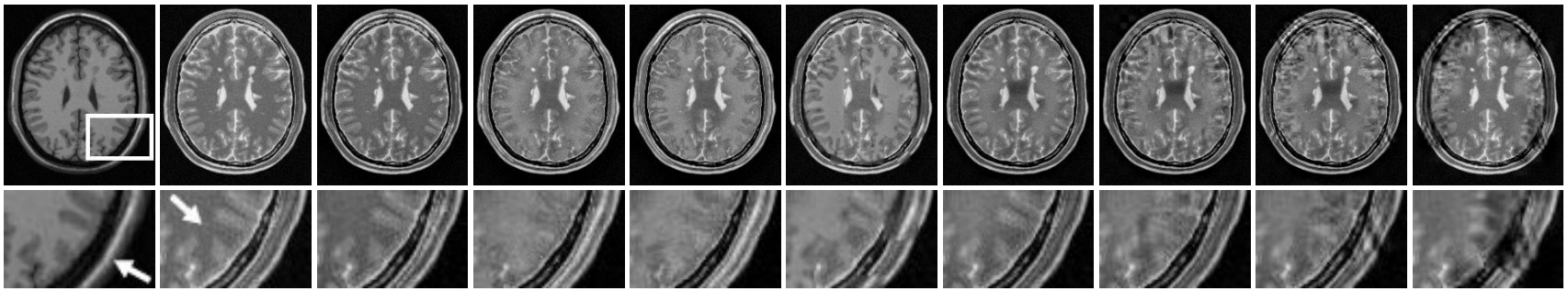
Grouping	CS	NG				CBSG						MG	
APX combining	CS	AVG		WA		AVG			WA			CM	
DET combining	CS	CM		CM		CM			CM			CM	
Verification	CS	NV	WBV	NV	WBV	NV	WBV	CBV	NV	WBV	CBV	NV	WBV
Synthetic normal	**0.7727**	0.6566	0.7022	0.6529	0.6965	0.6532	0.6928	0.5982	0.6492	0.6871	0.5974	0.6609	0.6675
Synthetic lesion	**0.7810**	0.6221	0.6867	0.6120	0.6729	0.6265	0.6853	0.5697	0.6163	0.6718	0.5683	0.6763	0.6837
Real #1002	**0.6662**	0.5661	0.6325	0.5656	0.6321	0.5615	0.6181	0.4758	0.5611	0.6177	0.4755	0.6092	0.6149
Real #1037	0.6301	0.5395	0.5941	0.5399	0.5944	0.54	0.5840	0.4601	0.5404	0.5898	0.4605	0.6243	**0.6305**
Real #1215	**0.6803**	0.5835	0.6521	0.5833	0.6520	0.5783	0.6313	0.4843	0.5781	0.6311	0.4843	0.6169	0.6233
Real #1344	**0.6338**	0.5115	0.5874	0.5103	0.5863	0.5117	0.5726	0.4365	0.5105	0.5714	0.4352	0.5748	0.5807
Real #1372	**0.6560**	0.5627	0.6144	0.5630	0.6147	0.5622	0.6123	0.4765	0.5626	0.6126	0.4771	0.6261	0.6313

Table 5.2: Objective evaluation results on DWT-based fusion of synthetic and real data using the $Q^{AB/F}$ metric.

Grouping	CS	NG				CBSG						MG	
APX combining	CS	AVG		WA		AVG			WA			CM	
DET combining	CS	CM		CM		CM			CM			CM	
Verification	CS	NV	WBV	NV	WBV	NV	WBV	CBV	NV	WBV	CBV	NV	WBV
Synthetic normal	**0.7217**	0.5610	0.6719	0.5481	0.6556	0.5579	0.6405	0.4840	0.5443	0.6248	0.4792	0.6115	0.6220
Synthetic lesion	**0.7195**	0.5343	0.6540	0.5133	0.6213	0.5480	0.6217	0.4659	0.5246	0.5898	0.4528	0.6106	0.5834
Real #1002	**0.6140**	0.4950	0.4896	0.4942	0.4887	0.5144	0.5265	0.4110	0.5136	0.5255	0.4110	0.5828	0.5606
Real #1037	0.5578	0.4688	0.4478	0.4690	0.4481	0.4802	0.5180	0.3917	0.4804	0.5183	0.3917	**0.5824**	0.5513
Real #1215	**0.6338**	0.5081	0.5314	0.5073	0.5309	0.5285	0.5508	0.4211	0.5278	0.5502	0.4211	0.5944	0.5740
Real #1344	**0.5638**	0.4347	0.4489	0.4336	0.4475	0.4583	0.5056	0.3765	0.4571	0.5039	0.3761	0.5598	0.5386
Real #1372	0.5908	0.4955	0.4976	0.4959	0.4981	0.5085	0.5722	0.4090	0.5088	0.5730	0.4089	0.5991	**0.6010**

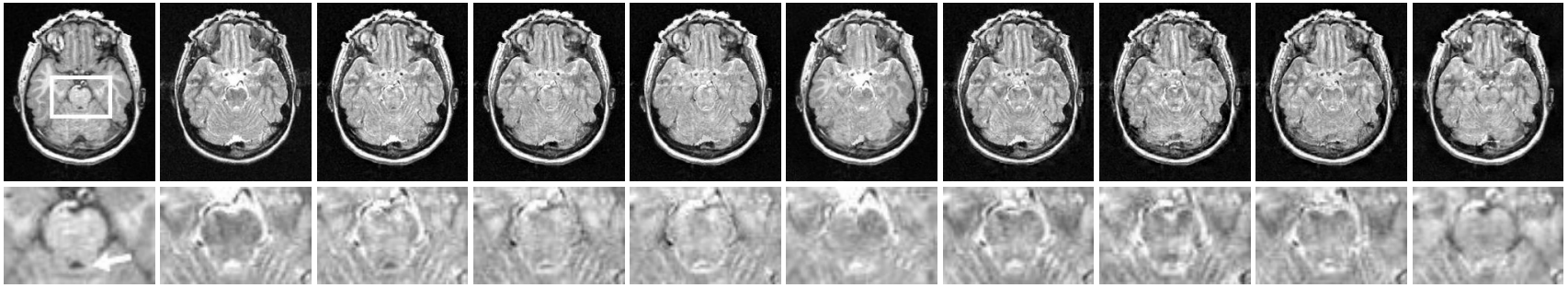
Table 5.3: Average rankings of different fusion rules on the real data using the $Q^{AB/F}$ metric.

Grouping	CS	NG				CBSG						MG	
APX combining	CS	AVG		WA		AVG			WA			CM	
DET combining	CS	CM		CM		CM			CM			CM	
Verification	CS	NV	WBV	NV	WBV	NV	WBV	CBV	NV	WBV	CBV	NV	WBV
LPT-based	**1.02**	9.58	4.04	8.58	3.2	10.42	6.18	12.88	9.42	5.36	12.12	4.7	3.5
DWT-based	**1.58**	9.82	9.76	8.98	8.96	7.12	4.84	12.76	6.34	4.16	12.24	1.66	2.78



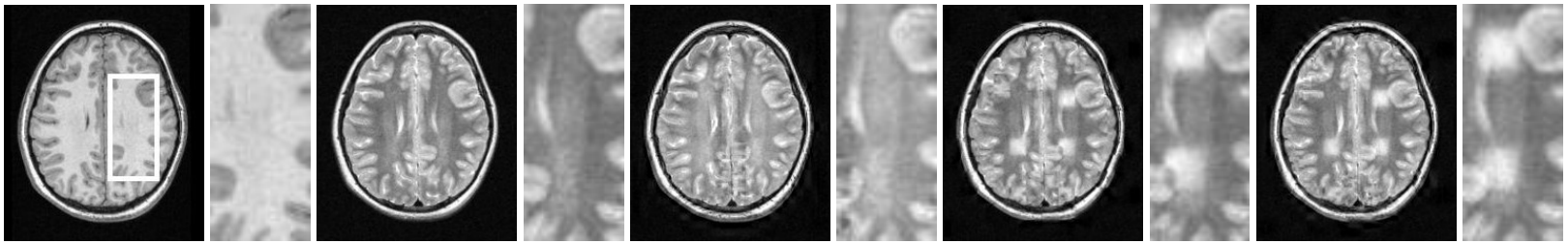
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 5.5: Comparison of different fusion rules using synthetic MRI brain images. Our CS rule provides better reproduction of local details. (a) T1W MRI image; (b) T2W MRI image; (c) LPT+CS; (d) LPT+NG+AVG+CM+WBV; (e) LPT+CBSG+AVG+CM+WBV; (f) LPT+MG+CM+CM+WBV; (g) DWT+CS; (h) DWT+NG+AVG+CM+WBV; (i) DWT+CBSG+AVG+CM+WBV; (j) DWT+MG+CM+CM+WBV.



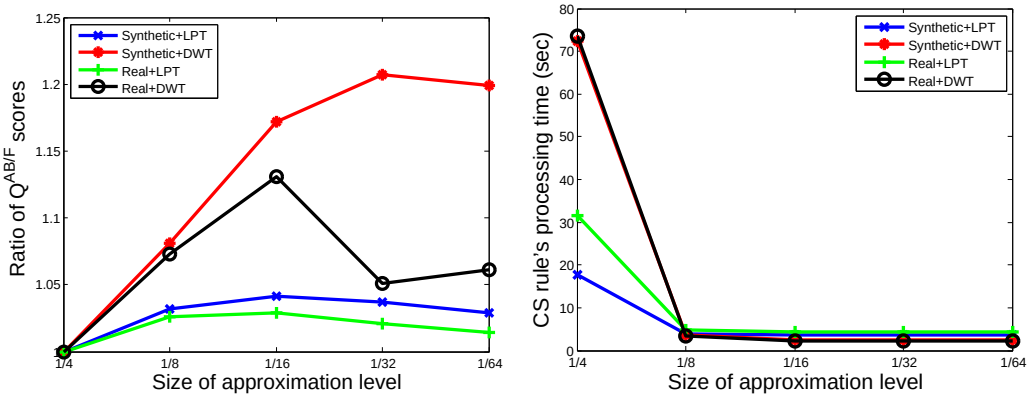
(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 5.6: Comparison of different fusion rules using real dataset #1002. Our CS rule provides better reproduction of local details with better consistencies at region boundaries. (a) T1W MRI image; (b) T2W MRI image; (c) LPT+CS; (d) LPT+NG+AVG+CM+WBV; (e) LPT+CBSG+AVG+CM+WBV; (f) LPT+MG+CM+CM+WBV; (g) DWT+CS; (h) DWT+NG+AVG+CM+WBV; (i) DWT+CBSG+AVG+CM+WBV; (j) DWT+MG+CM+CM+NV.



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

Figure 5.7: Comparison of different fusion rules using real dataset #1372. Our CS rule provides better neighborhood consistencies. The artifacts produced by the MG rule in the white matter do not appear in the result by our CS rule. (a) T1W MRI image; (b) T2W MRI image; (c) DWT+CS; (d) DWT+MG+CM+CM+NV; (e) DWT+MG+CM+CM+WBV.



(a) (b)

Figure 5.8: Analysis of the influence of the number of decomposition levels on the performance of our CS fusion rule. (a) Influence on the $Q^{AB/F}$ scores; (b) Influence on the processing times.

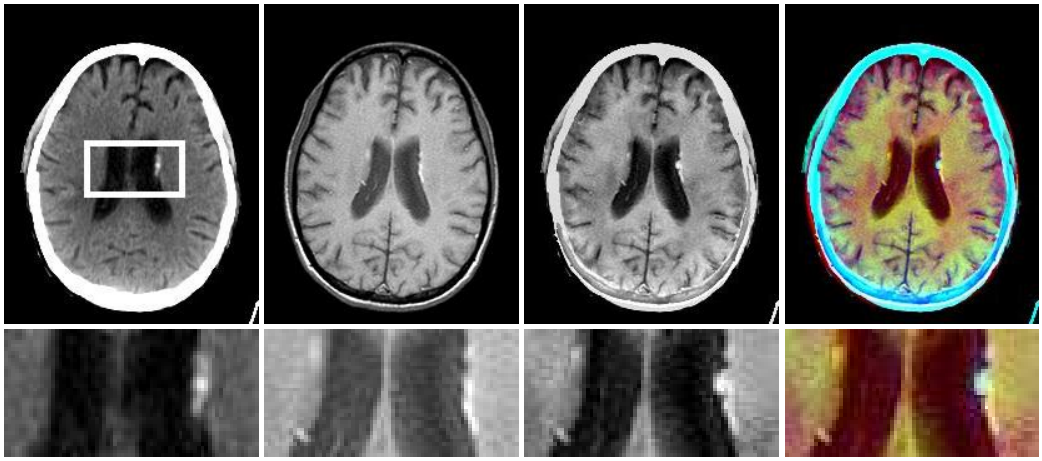
5.2.6.3 Application in Fusion of Other Imaging Modalities

Now that the performance of our cross-scale fusion rule has been validated, analyzed, and compared with other fusion rules on T1W/T2W MRI fusion, we further demonstrate its effectiveness in the fusion of other modalities. The registered 3D images used in the experiments in this section are retrieved from [83]. LPT+CS was applied with the same parameter setting as in Section 5.2.6.1 unless otherwise mentioned.

Fusion of CT/MRI This dataset contains one CT scan and one T1W MRI scan of a patient with cerebral toxoplasmosis. Each scan contains $256 \times 256 \times 24$ voxels with 8-bit precision. Four decomposition levels were applied because the depth of the third dimension is only 24 voxels. As displayed in Figure 5.9[3], the calcification captured in the CT scan and the soft tissue structures captured in the MRI scan are successfully transferred to the fused image. With our color fusion scheme applied, different features stand out even better.

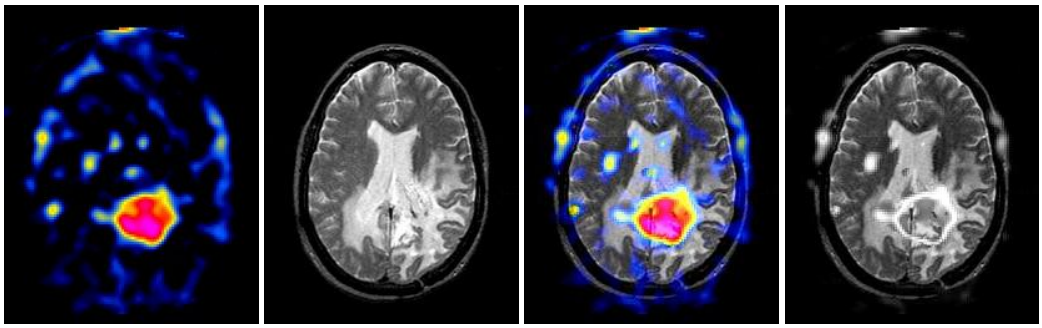
Fusion of SPECT/MRI This dataset contains one color-coded SPECT scan and one T2W MRI scan of a patient with anaplastic astrocytoma. Each scan contains $256 \times 256 \times 56$ voxels with 8-bit precision in the luminance channel. When one source image contains color (*e.g.,* the color-coded SPECT scan), a common procedure [201] is to fuse its luminance channel with the other monochrome source image using a monochrome fusion method. As displayed in Figure 5.10, our method combines the high Thallium uptake shown in the SPECT scan with the anatomical structures shown in the MRI scan in the fused image in order to achieve better determination of the extent of the tumor, while preserving high image contrast.

[3]Some blank pixels at the borders of the displayed images in Figures 5.9, 5.10, and 5.11 were clipped for more concise presentations.



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

Figure 5.9: Fusion of CT and T1W MRI images. (a) CT image; (b) T1W MRI image; (c) Fused; (d) Color-fused.



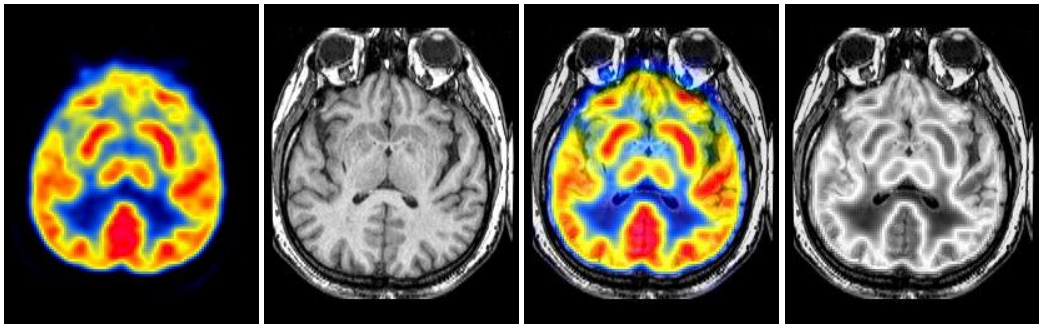
(a) (b) (c) (d)

Figure 5.10: Fusion of SPECT and T2W MRI images. (a) SPECT image (color-coded); (b) T2W MRI image; (c) Fused; (d) Fused (luminance channel).

Fusion of PET/MRI This dataset contains one color-coded PET scan and one T1W MRI scan of a normal brain. Each scan contains $256 \times 256 \times 127$ voxels with 8-bit precision in the luminance channel. As demonstrated in Figure 5.11, the metabolic activity revealed in the PET scan and the anatomical structures revealed in the MRI scan are combined in the fused image, providing better spatial relationships.

5.2.6.4 Noise Sensitivity

Noise sensitivity is a common concern for many pixel-level fusion methods [146]. In order to achieve robustness on noisy images such as ultrasound images, one could employ a denoising activity-level measure [120] or a denoising pre-processing step as in Section 3.3.7.4.



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

Figure 5.11: Fusion of PET and T1W MRI images. (a) PET image (color-coded); (b) T1W MRI image; (c) Fused; (d) Fused (luminance channel).

5.3 Summary

In this chapter, we proposed the multiscale random walks framework to incorporate multiscale techniques into the generalized random walks framework for robust performance based on both inter- and intra-scale information. We proposed a cross-scale fusion rule built on MRW. Our fusion rule selects an optimal set of coefficients for each decomposition level, and guarantees intra- and inter-scale consistencies. Experiments on volumetric medical image fusion demonstrated the effectiveness and versatility of our cross-scale fusion rule, which produced fused images with higher quality than existing rules. An efficient color fusion scheme effectively utilizing monochrome fusion results was also introduced.

It is worth mentioning that it is possible to also extend MGCRF to incorporate multiscale representations, which leads to a multiscale MGCRF model. However, the operators for constructing MGCRFs at different scales and for passing information between scales need to be defined in accordance with the employed multiscale decomposition scheme. This is different from HMGCRF, where a single set of operators can be defined.

Chapter 6

Conclusions and Future Work

The primary goal of this thesis is to develop efficient and effective frameworks for solving discrete pixel labeling problems in digital image processing and analysis. We focus on two representative applications, *i.e.,* stereo correspondence and image fusion (with a main focus on the latter), to verify the performance of the frameworks. In the following, we first summarize the major contributions of this thesis and then present several future research directions.

6.1 Contributions

Generalized Random Walks, Hierarchical Random Walks, and Multiscale Random Walks We proposed a series of random walks based frameworks for discrete pixel labeling problems. The generalized random walks framework converts a discrete labeling problem into steady-state probability calculation on a graph. The hierarchical random walks framework improves the computational efficiency of GRW by performing the calculation in a coarse-to-fine fashion. The multiscale random walks framework extends GRW to work with multiscale representations aiming for more robust results taking into account both inter- and intra-scale information.

2D and 3D Image Fusion in the Random Walks Context We formulated image fusion as a labeling problem in the random walks context. We applied GRW and HRW to 2D image fusion and MRW to 3D image fusion. The derived fusion algorithms demonstrated better performance than other state-of-the-art algorithms. In addition, we introduced perceptual metrics to multi-exposure fusion, which further improved the fusion quality; and we also introduced an efficient color fusion scheme based on color opponency theory.

Stereo Correspondence in the Random Walks Context We converted stereo correspondence to a labeling problem in the random walks context and applied GRW to solve it. With basic distance metrics, the derived algorithm demonstrated better performance than belief-propagation-based and graph-cuts-based algorithms coupled with basic distance metrics. With basic disparity refinement

schemes applied, the GRW-based stereo correspondence algorithm showed improved disparity estimation.

Multivariate Gaussian Conditional Random Fields and Hierarchical MGCRFs We proposed the multivariate Gaussian conditional random field model and the hierarchical MGCRF model for more general multi-label problems. GRW and HRW can be considered special cases of MGCRF and HMGCRF, respectively, in terms of steady-state probability calculation, when the pairwise precision matrices are defined as identity matrices subject to one data-dependent and one data-independent scaling factor.

6.2 Directions for Future Work

Effective and Compact Distance Metrics To date, we have been focusing primarily on the development of frameworks/models for solving labeling problems in digital image processing and analysis. As noted above, proper distance metrics are another important factor that affects the solution quality. An example of this is the perceptual metrics that improved the image quality in multi-exposure fusion. Therefore, developing effective and compact distance metrics is also crucial for deriving high-quality solutions at a low computational cost. This is especially useful for real-time applications and for applications where image acquisition conditions are not ideal.

Multi-Dimensional Distance Metrics In our current applications, each dimension of the MGCRF or HMGCRF is assumed to be independent from the others, which results in diagonal precision matrices for each variable pair. While this assumption simplifies the calculation (to the GRW or HRW model) and leads to satisfactory results in stereo correspondence and image fusion, employing non-diagonal precision matrices, *i.e.,* exploring multi-dimensional distance metrics, may be useful for more complicated labeling problems.

Generalization of HMGCRFs Although the MGCRF model that we introduced does not place restrictions on the types of graphs or neighborhoods used, the HMGCRF model that we introduced here considers only lattice graphs with first-order neighborhoods. This model can be generalized to graphs with arbitrary topologies and with higher-order neighborhoods using a properly-defined node grouping/clustering scheme and a properly-defined distance metric for neighboring node groups/clusters. This generalized HMGCRF model can be applied to labeling problems with irregular sites (*e.g.,* extracted features or segmented regions).

6.3 Publications

Refereed Publications Directly Arising from or Closely Related to Thesis Work

[1] R. Shen, I. Cheng, and A. Basu. Cross-scale coefficient selection for volumetric medical image fusion. *IEEE Transactions on BioMedical Engineering*, 2012.

[2] R. Shen, I. Cheng, J. Shi, and A. Basu. Generalized random walks for fusion of multi-exposure images. *IEEE Transactions on Image Processing*, vol. 20, no. 12, pages 3634-3646, 2011.

[3] G. Gou, R. Shen, Y. Wang, and A. Basu. Temporal-spatial face recognition using multi-atlas and Markov process model. In *ICME '11: Proceedings of the 2011 IEEE International Conference on Multimedia and Expo*, pages 1-4, 2011.

[4] R. Shen, I. Cheng, and A. Basu. A hybrid knowledge-guided detection technique for screening of infectious pulmonary tuberculosis from chest radiographs. *IEEE Transactions on BioMedical Engineering*, vol. 57, no. 11, pages 2646-2656, 2010.

[5] R. Shen, I. Cheng, X. Li and A. Basu. Stereo matching using random walks. In *ICPR '08: Proceedings of the 19th International Conference on Pattern Recognition*, pages 1-4, 2008.

Other Refereed Publications during Ph.D. Study

[1] R. Shen. Animated volume texture mapping for complex scene generation and editing. In *SG '10: Proceedings of the 10th International Symposium on Smart Graphics*, pages 33-43, 2010.

[2] R. Shen, I. Cheng, and A. Basu. Multi-camera calibration using a globe. In *OMNIVIS '08: Proceedings of ECCV Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, 2008.

Book Chapter Published during Ph.D. Study

[1] R. Shen, I. Cheng, and A. Basu. Automatic difficulty level estimation of multimedia math test items. In *Multimedia in Education: Adaptive Learning and Testing*, World Scientific Publishing Co., 2010.

Bibliography

[1] A. F. Abdelnour and I. W. Selesnick. Design of 2-band orthogonal near-symmetric CQF. In *ICASSP '01: Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 6, pages 3693–3696, 2001.

[2] E. H. Adelson and J. Y. A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, 1992.

[3] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. In *SIGGRAPH '04: Proceedings of the 31th International Conference on Computer Graphics and Interactive Techniques*, pages 294–302, 2004.

[4] M. Aggarwal and N. Ahuja. High dynamic range panoramic imaging. In *ICCV '01: Proceedings of the 8th IEEE International Conference on Computer Vision*, pages 2–9, 2001.

[5] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li. Removing photography artifacts using gradient projection and flash-exposure sampling. In *SIGGRAPH '05: Proceedings of the 32th International Conference on Computer Graphics and Interactive Techniques*, pages 828–835, 2005.

[6] M. Aguilar, D. A. Fay, W. D. Ross, A. M. Waxman, D. B. Ireland, and J. P. Racamato. Real-time fusion of low-light CCD and uncooled IR imagery for color night vision. In *Proceedings of SPIE*, volume 3364, pages 124–135, 1998.

[7] M. Aguilar and J. R. New. Fusion of multi-modality volumetric medical imagery. In *ICIF '02: Proceedings of the 5th International Conference on Information Fusion*, volume 2, pages 1206–1212, 2002.

[8] H. Amthauer, T. Denecke, T. Rohlfing, J. Ruf, M. Bohmig, M. Gutberlet, U. Plokinger, R. Felix, and A. J. Lemke. Value of image fusion using single photon emission computed tomography with integrated low dose computed tomography in comparison with a retrospective voxel-based method in neuroendocrine tumours. *European Radiology*, 15(7):1456–1462, 2005.

[9] H. Amthauer, R. Wurm, D. Kuczer, J. Ruf, R. Michel, J. Eisenacher, F. Stockhammer, T. Denecke, R. Felix, and M. Plotkin. Relevance of image fusion with MRI for the interpretation of I-123 iodo-methyl-tyrosine scans in patients with suspected recurrent or residual brain tumor. *Clinical Nuclear Medicine*, 31(4):189–192, 2006.

[10] B. Appleton and H. Talbot. Globally minimal surfaces by continuous maximal flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):106–118, 2006.

[11] T. O. Aydin, R. Mantiuk, K. Myszkowski, and H.-P. Seidel. Dynamic range independent image quality assessment. In *SIGGRAPH '08: Proceedings of the 35th International Conference and Exhibition on Computer Graphics and Interactive Techniques*, number 69, pages 1–10, 2008.

[12] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, 91(1):1–31, 2011.

[13] Y. Bando, B.-Y. Chen, and T. Nishita. Extracting depth and matte using a color-filtered aperture. *ACM Transactions on Graphics*, 27(5):1–9, 2008.

[14] M. R. Banham and A. K. Katsaggelos. Digital image restoration. *IEEE Signal Processing Magazine*, 14(2):24–41, 1997.

[15] A. Banno and K. Ikeuchi. Disparity map refinement and 3D surface smoothing via directed anisotropic diffusion. In *3DIM '09: Proceedings of ICCV Workshop on 3-D Digital Imaging and Modeling*, 2009.

[16] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, 36(2):192–236, 1974.

[17] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, 48(3):259–302, 1986.

[18] D. Bickson, D. Dolev, O. Shental, P. H. Siegel, and J. K. Wolf. Linear detection via belief propagation. In *Proceedings of the 45th Annual Allerton Conference on Communication, Control, and Computing*, 2007.

[19] R. S. Blum and Z. Liu, editors. *Multi-Sensor Image Fusion and Its Applications*. CRC Press, 2005.

[20] R. S. Blum, Z. Xue, and Z. Zhang. *Multi-Sensor Image Fusion and Its Applications*, chapter An overview of image fusion, pages 1–36. CRC Press, 2005.

[21] L. Bogoni and M. Hansen. Pattern-selective color image fusion. *Pattern Recognition*, 34(8):1515–1526, 2001.

[22] A. Boss, S. Bisdas, A. Kolb, M. Hofmann, U. Ernemann, C. D. Claussen, C. Pfannenberg, B. J. Pichler, M. Reimold, and L. Stegger. Hybrid PET/MRI of intracranial masses: Initial experiences and comparison to PET/CT. *Journal of Nuclear Medicine*, 51(8):1198–1205, 2010.

[23] C. Bouman and K. Sauer. A generalized Gaussian image model for edge-preserving MAP estimation. *IEEE Transactions on Image Processing*, 2(3):296–310, 1993.

[24] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. In *ICCV '01: Proceedings of the 8th IEEE International Conference on Computer Vision*, volume 1, pages 105–112, 2001.

[25] 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.

[26] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *CVPR '98: Proceedings of the 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 648–655, 1998.

[27] 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.

[28] Brain Development Cooperative Group and A. C. Evans. The NIH MRI study of normal brain development. *NeuroImage*, 30(1):184–202, 2006. https://nihpd.crbs.ucsd.edu/nihpd/info/.

[29] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM, 2nd edition, 2000.

[30] B. Bringier, N. Richard, M.-C. Larabi, and C. Fernandez-Maloigne. No-reference perceptual quality assessment of color image. In *EUSIPCO '06: Proceedings of the 14th European Signal Processing Conference*, 2006.

[31] P. Burt. *Multiresolution Image Processing and Analysis*, chapter The pyramid as a structure for efficient computation, pages 6–35. Springer-Verlag, 1984.

[32] P. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, 1983.

[33] P. J. Burt and R. J. Kolczynski. Enhanced image capture through fusion. In *ICCV '93: Proceedings of the 4th IEEE International Conference on Computer Vision*, pages 173–182, 1993.

[34] M. Cadík, M. Wimmer, L. Neumann, and A. Artusi. Evaluation of HDR tone mapping methods using essential perceptual attributes. *Computers & Graphics*, 32(3):330–349, 2008.

[35] S. Cai, K. Li, and I. Selesnick. Wavelet software at Polytechnic University, Brooklyn, NY. http://eeweb.poly.edu/iselesni/WaveletSoftware/. accessed April 2010.

[36] V. D. Calhoun and T. Adali. Feature-based fusion of medical imaging data. *IEEE Transactions on Information Technology in Biomedicine*, 13(5):711–720, 2009.

[37] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz. Adaptive perceptual color-texture image segmentation. *IEEE Transactions on Image Processing*, 14(10):1524–1536, 2005.

[38] Y. Chen, T. A. Davis, W. W. Hager, and S. Rajamanickam. Algorithm 887: CHOLMOD, supernodal sparse Cholesky factorization and update/downdate. *ACM Transactions on Mathematical Software*, 35(3):1–14, 2008.

[39] I. Cheng and A. Basu. Contrast enhancement from multiple panoramic images. In *OMNIVIS '07: Proceedings of ICCV Workshop on Omnidirectional Vision, Camera Networks and Nonclassical Cameras*, pages 1–7, 2007.

[40] W.-H. Cho and K.-S. Hong. Extending dynamic range of two color images under different exposures. In *ICPR '04: Proceedings of the 17th International Conference on Pattern Recognition*, volume 4, pages 853–856, 2004.

[41] C. Couprie, L. Grady, L. Najman, and H. Talbot. Power watershed: A unifying graph-based optimization framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1384–1399, 2011.

[42] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiway cuts. In *STOC '92: Proceedings of the 24th Annual ACM Symposium on Theory of Computing*, pages 241–251, 1992.

[43] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994.

[44] T. A. Davis and W. W. Hager. Dynamic supernodes in sparse Cholesky update/downdate and triangular solves. *ACM Transactions on Mathematical Software*, 35(4):1–23, 2009.

[45] D. A. Dawson, L. G. Gorostiza, and A. Wakolbinger. Degrees of transience and recurrence and hierarchical random walks. *Potential Analysis*, 22(4):305–350, 2005.

[46] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH '97: Proceedings of the 24th International Conference on Computer Graphics and Interactive Techniques*, pages 369–378, 1997.

[47] O. F. Donati, T. F. Hany, C. S. Reiner CS, G. K. von Schulthess, B. Marincek, B. Seifert, and D. Weishaupt. Value of retrospective fusion of PET and MR images in detection of hepatic metastases: Comparison with ^{18}F-FDG PET/CT and Gd-EOB-DTPA-enhanced MRI. *Journal of Nuclear Medicine*, 51(5):692–699, 2010.

[48] P. G. Doyle and J. L. Snell. *Random Walks and Electric Networks*. Mathematical Association of America, 1984.

[49] F. Drago, K. Myszkowski, T. Annen, and N. Chiba. Adaptive logarithmic mapping for displaying high contrast scenes. *Computer Graphics Forum*, 22(3):419–426, 2003.

[50] R. D. Falgout. An introduction to algebraic multigrid. *Computing in Science and Engineering*, 8(6):24–33, 2006.

[51] Z. Farbman, R. Fattal, and D. Lischinski. Diffusion maps for edge-aware image editing. In *SIGGRAPH Asia '10: Proceedings of the 3rd ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques in Asia*, number 145, pages 1–10, 2010.

[52] M. Farhadi, S. Mahmoudian, F. Saddadi, A. R. Karimian, M. Mirzaee, M. Ahmadizadeh, K. Ghasemikian, S. Gholami, E. Ghoreyshi, S. Beyty, A. Shamshiri, S. Madani, V. Bakaev, S. Moradkhani, and G. Raeisali. Functional brain abnormalities localized in 55 chronic tinnitus patients: Fusion of SPECT coincidence imaging and MRI. *Journal of Cerebral Blood Flow and Metabolism*, 30(4):864870, 2010.

[53] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. Wiley, 3rd edition, 1968.

[54] P. F. Felzenszwalb and D. R. Huttenlocher. Efficient belief propagation for early vision. In *CVPR '04: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 261–268, 2004.

[55] P. F. Felzenszwalb and D. R. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1):41–54, 2006.

[56] S. Ferradans, M. Bertalmio, E. Provenzi, and V. Caselles. An analysis of visual adaptation and contrast perception for tone mapping. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2002–2012, 2011.

[57] A. Firouzmanesh, I. Cheng, and A. Basu. Perceptually guided fast compression of 3-D motion capture data. *IEEE Transactions on Multimedia*, 13(4):829–834, 2011.

[58] J. M. Foley and W. Schwarz. Spatial attention: Effect of position uncertainty and number of distractor patterns on the threshold-versus-contrast function for contrast discrimination. *J. Opt. Soc. Am. A*, 15(5):1036–1047, 1998.

[59] J. E. Fowler. The redundant discrete wavelet transform and additive noise. *IEEE Signal Processing Letters*, 12(9):629–632, 2005.

[60] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000.

[61] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.

[62] M. A. García-Pérez and R. Alcalá-Quintana. The transducer model for contrast detection and discrimination: Formal relations, implications, and an empirical test. *Spatial Vision*, 20(1-2):5–43, 2007.

[63] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

[64] F. L. Giesel, A. Mehndiratta, J. Locklin, M. J. McAuliffe, S. White, P. L. Choyke, M. V. Knopp, B. J. Wood, U. Haberkorn, and H. von Tengg-Kobligk. Image fusion using CT, MRI and PET for treatment planning, navigation and follow up in percutaneous RFA. *Experimental Oncology*, 31(2):106–114, 2009.

[65] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice-Hall, 3rd edition, 2006.

[66] M. J. Gooding, K. Rajpoot, S. Mitchell, P. Chamberlain, S. H. Kennedy, and J. A. Noble. Investigation into the fusion of multiple 4-D fetal echocardiography images to improve image quality. *Ultrasound in Medicine and Biology*, 36(6):957–966, 2010.

[67] A. Goshtasby. Fusion of multi-exposure images. *Image and Vision Computing*, 23(6):611–618, 2005.

[68] L. Grady. Multilabel random walker image segmentation using prior models. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 763–770, 2005.

[69] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[70] L. Grady. A lattice-preserving multigrid method for solving the inhomogeneous poisson equations used in image analysis. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 252–264, 2008.

[71] L. Grady and G. Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In *Proceedings of ECCV Workshop on Computer Vision Approaches to Medical Image Analysis and Mathematical Methods in Biomedical Image Analysis*, pages 230–245, 2004.

[72] L. Grady, T. Tasdizen, and R. Whitaker. A geometric multigrid approach to solving the 2D inhomogeneous laplace equation with internal dirichlet boundary conditions. In *ICIP '05: Proceedings of the 2005 IEEE International Conference on Image Processing*, volume 2, pages 642–645, 2005.

[73] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H.-P. Seidel, and H. P. A. Lensch. Optimal HDR reconstruction with linear digital cameras. In *CVPR '10: Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition*, pages 215–222, 2010.

[74] 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*, 51(2):271–279, 1989.

[75] D. L. Hall and J. Llinas. An introduction to multisensor data fusion. *Proceedings of IEEE*, 85(1):6–23, 1997.

[76] J. M. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. 1971.

[77] R. I. Hartley. Theory and practice of projective rectification. *International Journal of Computer Vision*, 35(2):115–127, 1999.

[78] S. W. Hasinoff and K. N. Kutulakos. Confocal stereo. *International Journal of Computer Vision*, 81(1):82–104, 2009.

[79] E. Hering. *Outlines of a Theory of the Light Sense*, chapter The natural color system, pages 25–65. Harvard University Press, 1964.

[80] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.

[81] H. Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1333–1336, 2003.

[82] A. Jackson, N. A. Thacker, and S. M. Stivaros. *Image Processing in Radiology: Current Applications*, chapter 3D image fusion, pages 101–122. Springer, 2007.

[83] K. A. Johnson and J. A. Becker. The whole brain altas. http://www.med.harvard.edu/aanlib/.

[84] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. In *SIGGRAPH '10: Proceedings of the 37th International Conference and Exhibition on Computer Graphics and Interactive Techniques*, number 30, pages 1–9, 2010.

[85] W. Kim and K. M. Lee. A hybrid approach for MRF optimization problems: Combination of stochastic sampling and deterministic algorithms. *Computer Vision and Image Understanding*, pages 1623–1637, 2011.

[86] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[87] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 15–18, 2006.

[88] P. Kohli and P. H. S. Torr. Dynamic graph cuts for efficient inference in Markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2079–2088, 2007.

[89] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(7):1274–1279, 2007.

[90] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *ICCV '01: Proceedings of the 8th IEEE International Conference on Computer Vision*, volume 2, pages 508–515, 2001.

[91] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision*, pages 82–96, 2002.

[92] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[93] G. Krawczyk, K. Myszkowski, and H.-P. Seidel. Lightness perception in tone reproduction for high dynamic range images. *Computer Graphics Forum*, 24(3):635–645, 2005.

[94] J. Kuang, G. M. Johnson, and M. D. Fairchild. iCAM06: A refined image appearance model for HDR image rendering. *Journal of Visual Communication and Image Representation*, 18(5):406–414, 2007.

[95] J. Kuang, H. Yamaguchi, C. Liu, G. M. Johnson, and M. D. Fairchild. Evaluating HDR rendering algorithms. *ACM Transactions on Applied Perception*, 4(2), 2007.

[96] M. Kumar and S. Dass. A total variation-based algorithm for pixel-level image fusion. *IEEE Transactions on Image Processing*, 18(9):2137–2143, 2009.

[97] S. Kumar and M. Hebert. Discriminative random fields. *International Journal of Computer Vision*, 68(2):179–201, 2006.

[98] R. K.-S. Kwan, A. C. Evans, and G. B. Pike. MRI simulation-based evaluation of image-processing and classification methods. *IEEE Transactions on Medical Imaging*, 18(11):1085–1097, 1999. http://www.bic.mni.mcgill.ca/brainweb/.

[99] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01: Proceedings of the 18th International Conference on Machine Learning*, pages 282–289, 2001.

[100] E. S. Larsen, P. Mordohai, M. Pollefeys, and H. Fuchs. Temporally consistent reconstruction from multiple video streams using enhanced belief propagation. In *ICCV '07: Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[101] G. W. Larson, H. Rushmeier, and C. Piatko. A visibility matching tone reproduction operator for high dynamic range scenes. *IEEE Transactions on Visualization and Computer Graphics*, 3(4):291–306, 1997.

[102] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen. Evaluation of tone mapping operators using a high dynamic range display. In *SIGGRAPH '05: Proceedings of the 32th International Conference on Computer Graphics and Interactive Techniques*, pages 640–648, 2005.

[103] C. Lei, J. Selzer, and Y.-H. Yang. Region-tree based stereo using dynamic programming optimization. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2378–238, 2006.

[104] C. Lei and Y.-H. Yang. Optical flow estimation on coarse-to-fine region-trees using discrete optimization. In *ICCV '09: Proceedings of the 12th IEEE International Conference on Computer Vision*, 2009.

[105] H. Levkowitz and G. T. Herman. GLHS: A generalized lightness, hue, and saturation color model. *CVGIP: Graphical Models and Image Processing*, 55(4):271–285, 1993.

[106] J. J. Lewis, R. J. O'Callaghan, S. G. Nikolov, D. R. Bull, and N. Canagarajah. Pixel- and region-based image fusion with complex wavelets. *Information Fusion*, 8(2):119–130, 2007.

[107] H. Li, B. S. Manjunath, and S. K. Mitra. Multisensor image fusion using the wavelet transform. *Graphical Models and Image Processing*, 57(3):235–245, 1995.

[108] S. Li, J. T.-Y. Kwok, I. W.-H. Tsang, and Y. Wang. Fusing images with different focuses using support vector machines. *IEEE Transactions on Neural Networks*, 15(6):1555–1561, 2004.

[109] S. Li, B. Yang, and J. Hu. Performance comparison of different multi-resolution transforms for image fusion. *Information Fusion*, 12(2):74–84, 2011.

[110] S. Z. Li. Markov random field models in computer vision. In *ECCV '94: Proceedings of the 3rd European Conference on Computer Vision*, pages 361–370, 1994.

[111] S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer, 3rd edition, 2009.

[112] S. Z. Li, H. Wang, K. L. Chan, and M. Petrou. Minimization of MRF energy with relaxation labeling. *Journal of Mathematical Imaging and Vision*, 7(2):149–161, 1997.

[113] Y. Li, L. Sharan, and E. H. Adelson. Compressing and companding high dynamic range images with subband architectures. In *SIGGRAPH '05: Proceedings of the 32th International Conference on Computer Graphics and Interactive Techniques*, pages 836–844, 2005.

[114] W. Lin and C.-C. Jay Kuo. Perceptual visual quality metrics: A survey. *Journal of Visual Communication and Image Representation*, 22(4):297–312, 2011.

[115] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski. Interactive local adjustment of tonal values. In *SIGGRAPH '06: Proceedings of the 33th International Conference and Exhibition on Computer Graphics and Interactive Techniques*, pages 646–653, 2006.

[116] I. Lissner and P. Urban. Toward a unified color space for perception-based image processing. *IEEE Transactions on Image Processing*, 21(3):1153–1168, 2012.

[117] Z. Liu, E. Blasch, Z. Xue, J. Zhao, R. Laganiere, and W. Wu. Objective assessment of multiresolution fusion algorithms for context enhancement in night vision: A comparative study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):94–109, 2011.

[118] M. Livingstone and D. Hubel. Segregation of form, color, movement, and depth: Anatomy, physiology, and perception. *Science*, 240(4853):740–749, 1988.

[119] L. Lovasz. Random walks on graphs: A survey. *Combinatorics*, 2:353–398, 1993.

[120] A. Loza, D. Bull, N. Canagarajah, and A. Achim. Non-Gaussian model-based fusion of noisy images in the wavelet domain. *Computer Vision and Image Understanding*, 114(1):54–65, 2010.

[121] E. Lübbe. *Colours in the Mind - Color Systems in Reality*. Books on Demand GmbH, 2010.

[122] M. Mahvash, R. König, H. Urbach, J. von Ortzen, B. Meyer, J. Schramm, and C. Schaller. FLAIR-/T1-/T2-co-registration for image-guided diagnostic and resective epilepsy surgery. *Neurosurgery*, 58(1):69–75, 2006.

[123] Z. Mai, H. Mansour, R. Mantiuk, P. Nasiopoulos, R. Ward, and W. Heidrich. Optimizing a tone curve for backward-compatible high dynamic range image and video compression. *IEEE Transactions on Image Processing*, 20(6):1558–1571, 2011.

[124] J. B. A. Maintz and M. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, 1998.

[125] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[126] H. Mannami, R. Sagawa, Y. Mukaigawa, T. Echigo, and Y. Yagi. High dynamic range camera using reflective liquid crystal. In *ICCV '07: Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[127] R. Mantiuk, S. Daly, and L. Kerofsky. Display adaptive tone mapping. In *SIGGRAPH '08: Proceedings of the 35th International Conference and Exhibition on Computer Graphics and Interactive Techniques*, number 68, pages 1–10, 2008.

[128] M. Meila and J. Shi. A random walks view of spectral segmentation. In *AISTATS '01: Proceedings of the 8th International Workshop on Artificial Intelligence and Statistics*, 2001.

[129] T. Mertens, J. Kautz, and F. Van Reeth. Exposure fusion. In *PG '07: Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 382–390, 2007.

[130] T. Mertens, J. Kautz, and F. Van Reeth. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum*, 28(1):161–171, 2009.

[131] K. T. Mullen. The contrast sensitivity of human colour vision to red-green and blue-yellow chromatic gratings. *Journal of Physiology*, 359:381–400, 1985.

[132] K. P. Murphy. An introduction to graphical models. 2001.

[133] S. K. Nayar and T. Mitsunaga. High dynamic range imaging: Spatially varying pixel exposures. In *CVPR '00: Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 472–479, 2000.

[134] S. F. Nemec, M. A. Donat, S. Mehrain, K. Friedrich, C. Krestan, C. Matula, H. Imhof, and C. Czerny. CT-MR image data fusion for computer assisted navigated neurosurgery of temporal bone tumors. *European Journal of Radiology*, 62(2):192–198, 2007.

[135] C. O'Sullivan, S. Howlett, Y. Morvan, R. McDonnell, and K. O'Conor. Perceptually adaptive graphics. In *Proceedings of Eurographics State of the Art Reports*, 2004.

[136] G. Pajares and J. M. de la Cruz. A wavelet-based image fusion tutorial. *Pattern Recognition*, 37(9):1855–1872, 2004.

[137] Y. Pan, I. Cheng, and A. Basu. Quality metric for approximating subjective evaluation of 3-D objects. *IEEE Transactions on Multimedia*, 7(2):269–279, 2005.

[138] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, 1998.

[139] C. S. Pattichis, M. S. Pattichis, and E. Micheli-Tzanakou. Medical imaging fusion applications: An overview. In *Proceedings of the 35th Asilomar Conference on Signals, Systems and Computers*, volume 2, pages 1263–1267, 2001.

[140] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1st edition, 1988.

[141] E. Peli. Contrast in complex images. *Journal of the Optical Society of America A*, 7(10):2032–2040, 1990.

[142] V. Petrović. Subjective tests for image fusion evaluation and objective metric validation. *Information Fusion*, 8(2):208–216, 2007.

[143] V. Petrović and C. Xydeas. Cross-band pixel selection in multiresolution image fusion. In *Proceedings of SPIE*, volume 3719, pages 319–326, 1999.

[144] V. S. Petrović and C. S. Xydeas. Gradient-based multiresolution image fusion. *IEEE Transactions Image Processing*, 13(2):228–237, 2004.

[145] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. In *SIGGRAPH '04: Proceedings of the 31th International Conference on Computer Graphics and Interactive Techniques*, pages 664–672, 2004.

[146] G. Piella. A general framework for multiresolution image fusion: from pixels to regions. *Information Fusion*, 4(4):259–280, 2003.

[147] G. Piella. Image fusion for enhanced visualization: A variational approach. *International Journal of Computer Vision*, 83(1):1–11, 2009.

[148] T. Poggio, V. Torre, and C. Koch. Computational vision and regularization theory. *Nature*, 317(216):314–319, 1985.

[149] B. C. Porter, D. J. Rubens, J. G. Strang, J. Smith, S. Totterman, and K. J. Parker. Three-dimensional registration and fusion of ultrasound and MRI using major vessels as fiducial markers. *IEEE Transactions on Medical Imaging*, 20(4):354–359, 2001.

[150] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*, 12(11):1338–1351, 2003.

[151] R. F. Quick. A vector-magnitude model of contrast detection. *Biological Cybernetics*, 16(2):65–67, 1974.

[152] K. Rajpoot, J. A. Noble, V. Grau, C. Szmigielski, and H. Becher. Multiview RT3D echocardiography image fusion. In *FIMH '09: Proceedings of the 5th International Conference on Functional Imaging and Modeling of the Heart*, volume 5528, pages 134–143, 2009.

[153] S. Raman and S. Chaudhuri. A matte-less, variational approach to automatic scene compositing. In *ICCV '07: Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–6, 2007.

[154] S. Raman and S. Chaudhuri. Bilateral filter based compositing for variable exposure photography. In *Proceedings of Eurographics Short Papers*, 2009.

[155] E. Reinhard. Parameter estimation for photographic tone reproduction. *Journal of Graphics Tool*, 7(1):45–52, 2002.

[156] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda. Photographic tone reproduction for digital images. In *SIGGRAPH '02: Proceedings of the 29th International Conference on Computer Graphics and Interactive Techniques*, pages 267–276, 2002.

[157] E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, W. Heidrich, and K. Myszkowski. *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting*. Morgan Kaufmann, 2nd edition, 2010.

[158] A. Roorda. *Encyclopedia of Imaging Science and Technology*, chapter Human visual system - image formation. Wiley, 2002.

[159] J. M. Rovamo, M. I. Kankaanpaa, and H. Kukkonen. Modelling spatial contrast sensitivity functions for chromatic and luminance-modulated gratings. *Vision Research*, 39(14):2387–2398, 1999.

[160] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman and Hall/CRC, 2005.

[161] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, 2nd edition, 2003.

[162] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[163] H. Seetzen, W. Heidrich, W. Stuerzlinger, G. Ward, L. Whitehead, M. Trentacoste, A. Ghosh, and A. Vorozcovs. High dynamic range display systems. In *SIGGRAPH '04: Proceedings of the 31th International Conference on Computer Graphics and Interactive Techniques*, pages 760–768, 2004.

[164] L. Seiler, D. Carmean, E. Sprangle, T. Forsyth, M. Abrash, P. Dubey, S. Junkins, A. Lake, J. Sugerman, R. Cavin, R. Espasa, E. Grochowski, T. Juan, and P. Hanrahan. Larrabee: A many-core x86 architecture for visual computing. In *SIGGRAPH '08: Proceedings of the 35th International Conference and Exhibition on Computer Graphics and Interactive Techniques*, number 18, pages 1–15, 2008.

[165] I. W. Selesnick, R. G. Baraniuk, and N. C. Kingsbury. The dual-tree complex wavelet transform. *IEEE Signal Processing Magazine*, 22(6):123–151, 2005.

[166] R. Shams, P. Sadeghi, R. Kennedy, and R. Hartley. A survey of medical image registration on multicore and the GPU. *IEEE Signal Processing Magazine*, 27(2):50–60, 2010.

[167] Q. Shan, J. Jia, and M. S. Brown. Globally optimized linear windowed tone mapping. *IEEE Transactions on Visualization and Computer Graphics*, 16(4):663–675, 2010.

[168] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[169] S. E. Shimony. Finding MAPs for belief networks is NP-hard. *Artificial Intelligence*, 68(2):399–410, 1994.

[170] E. P. Simoncelli and W. T. Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *ICIP '95: Proceedings of the 1995 IEEE International Conference on Image Processing*, volume 3, pages 444–447, 1995.

[171] D. Singaraju, L. Grady, and R. Vidal. P-brush: Continuous valued MRFs with normed pairwise distributions for image segmentation. In *CVPR '09: Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1303–1310, 2009.

[172] A. K. Sinop and L. Grady. A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In *ICCV '07: Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[173] M. I. Smith and J. P. Heather. A review of image fusion technology in 2005. In *Proceedings of SPIE*, volume 5782, pages 29–45, 2005.

[174] P. Smyth, D. Heckerman, and M. I. Jordan. Probabilistic independence networks for hidden Markov probability models. *Neural Computation*, 9(2):227–269, 1997.

[175] B. Solaiman, R. Debon, F. Pipelier, J.-M. Cauvin, and C. Roux. Information fusion: Application to data and model fusion for ultrasound image segmentation. *IEEE Transactions on Biomedical Engineering*, 46(10):1171–1175, 1999.

[176] R. N. Strickland, C.-S. Kim, and W. F. McDonnell. Digital color image enhancement based on the saturation component. *Optical Engineering*, 26:609–616, 1987.

[177] J. Sun, N.-N. Zheng, and H.-Y. Shum. Stereo matching using belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(7):787–800, 2003.

[178] R. Szeliski. Bayesian modeling of uncertainty in low-level vision. *International Journal of Computer Vision*, 5(3):271–301, 1990.

[179] 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.

[180] M. F. Tappen and W. T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters. In *ICCV '03: Proceedings of the 9th IEEE International Conference on Computer Vision*, volume 2, pages 900–906, 2003.

[181] M. F. Tappen, C. Liu, E. H. Adelson, and W. T. Freeman. Learning Gaussian conditional random fields for low-level vision. In *CVPR '07: Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[182] J. E. Thornton and E. N. Pugh. Red/green color opponency at detection threshold. *Science*, 219(4581):191–193, 1983.

[183] S. Todorovic and N. Ahuja. 3D texture classification using the belief net of a segmentation tree. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, volume 4, pages 33–36, 2006.

[184] A. Toet. Hierarchical image fusion. *Machine Vision and Applications*, 3(1):1–11, 1990.

[185] J. Tumblin, A. Agrawal, and R. Raskar. Why i want a gradient camera. In *CVPR '05: Proceedings of the 2005 IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 103–110, 2005.

[186] J. Tumblin and H. Rushmeier. Tone reproduction for realistic images. *IEEE Computer Graphics and Applications*, 13(6):42–48, 1993.

[187] M. C. Valdés Hernández, K. Ferguson, F. Chappell, and J. Wardlaw. New multispectral MRI data fusion technique for white matter lesion segmentation: method and comparison with thresholding in FLAIR images. *European Radiology*, 20(7):1684–1691, 2010.

[188] M. C. Valdés Hernández, T. H. Jeong, C. Murray, M. E. Bastin, F. Chappell, I. J. Deary, and J. Wardlaw. Reliability of two techniques for assessing cerebral iron deposits with structural magnetic resonance imaging. *Journal of Magnetic Resonance Imaging*, 33(1):54–61, 2011.

[189] C. Wang and Z. Ye. First-order fusion of volumetric medical imagery. *IEE Proceedings - Vision, Image and Signal Processing*, 153(2):191–198, 2006.

[190] Z. Wang and Y. Ma. Medical image fusion using m-PCNN. *Information Fusion*, 9(2):176–185, 2008.

[191] A. M. Waxman, M. Aguilar, D. A. Fray, D. B. Ireland, J. P. Racamato Jr., W. D. Ross, J. E. Carrick, A. N. Gove, M. C. Seibert, E. D. Savoye, R. K. Reich, B. E. Burke, W. H. McGonagle, and D. M. Craig. Solid-state color night vision: Fusion of low-light visible and thermal infrared imagery. *Lincoln Laboratory Journal*, 11(1):41–60, 1998.

[192] H. Wechsler and M. Kidode. A random walk procedure for texture discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(3):272–280, 1979.

[193] C. Wehrhahn and B. Dresp. Detection facilitation by collinear stimuli in humans: Dependence on strength and sign of contrast. *Vision Research*, 38(3):423–428, 1998.

[194] Y. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12(1):1–41, 2000.

[195] Y. Weiss and W. T. Freeman. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neural Computation*, 13(10):2173–2200, 2001.

[196] Y. Weiss and W. T. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Transactions on Information Theory*, 47(2):736–744, 2001.

[197] H. R. Wilson. A transducer function for threshold and suprathreshold human vision. *Biological Cybernetics*, 38(3):171–178, 1980.

[198] S. Winkler. *Vision models and quality metrics for image processing applications*. PhD thesis, EPFL, 2000.

[199] T. Z. Wong, T. G. Turkington, T. C. Hawk, and R. E. Coleman. PET and brain tumor image fusion. *Cancer Journal*, 10(4):234–242, 2004.

[200] F. Y. Wu. The Potts model. *Reviews of Modern Physics*, 54(1):235–268, 1982.

[201] Z. Xue and R. S. Blum. Concealed weapon detection using color image fusion. In *ICIF '03: Proceedings of the 6th International Conference on Information Fusion*, volume 1, pages 622–627, 2003.

[202] C. S. Xydeas and V. Petrović. Objective image fusion performance measure. *Electronics Letters*, 36(4):308–309, 2000.

[203] J. Yang and R. S. Blum. Image fusion using the expectation-maximization algorithm and a hidden Markov model. In *VTC '04: Proceedings of the 60th IEEE Vehicular Technology Conference*, volume 6, pages 4563–4567, 2004.

[204] U. M. Yang. Parallel algebraic multigrid methods - high performance preconditioners. In *Numerical Solution of Partial Differential Equations on Parallel Computers*, pages 209–236. Springer, 2006.

[205] Y. Yang, D. S. Park, S. Huang, and N. Rao. Medical image fusion via an effective wavelet-based approach. *EURASIP Journal on Advances in Signal Processing*, 2010(579341):1–13, 2010.

[206] J. S. Yedidia, W. T. Freeman, and Y. Weiss. Understanding belief propagation and its generalizations. *Exploring Artificial Intelligence in the New Millennium*, pages 239–269, 2003.

[207] T. Yu, R.-S. Lin, B. Super, and B. Tang. Efficient message representations for belief propagation. In *ICCV '07: Proceedings of the 11th IEEE International Conference on Computer Vision*, pages 1–8, 2007.

[208] W. Zhang and W.-K. Cham. Gradient-directed multi-exposure composition. *IEEE Transactions on Image Processing*, 2011.

[209] Z. Zhang and R. S. Blum. Region-based image fusion scheme for concealed weapon detection. In *Proceedings of the 31st Annual Conference on Information Sciences and Systems*, pages 168–173, 1997.

[210] H. Zhao, Q. Li, and H. Feng. Multi-focus color image fusion in the HSI space using the sum-modified-laplacian and a coarse edge map. *Image and Vision Computing*, 26(9):1285–1295, 2008.

[211] S. Zheng, W.-Z. Shi, J. Liu, G.-X. Zhu, and J.-W. Tian. Multisource image fusion method using support value transform. *IEEE Transactions on Image Processing*, 16(7):1831–1839, 2007.

[212] X. Zhu, Z. Ghahramani, and J. D. Lafferty. Semi-supervised learning using Gaussian field and harmonic functions. In *ICML '03: Proceedings of the 20th International Conference on Machine Learning*, pages 912–919, 2003.

[213] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM Transactions on Graphics*, 23(3):600–608, 2004.

Appendix A

Belief Propagation for Bayesian Networks

Pearl [140] proposed two belief propagation algorithms for Bayesian networks based on message passing between neighboring nodes. The first algorithm is called *belief update* (also known as sum-product belief propagation, or BP-S). When iterative message updating in belief update reaches equilibrium, an MPM solution is found. The correctness of this algorithm is guaranteed for singly connected networks. The belief (posterior probability) $b_i(f_i)$ at node x_i reflects the conditional probability that the scene variable x_i at location i takes label f_i given all evidence $\mathbf{e}$ received so far:

$$b_i(f_i) \triangleq P(x_i = f_i|\mathbf{e}) \triangleq P(f_i|\mathbf{e}), \tag{A.1}$$

where $\mathbf{e}$ consists of all of the instantiated variables. A belief vector $\mathbf{b_i} = (b_i(l_1), \ldots, b_i(l_K))^T$ is stored for each node x_i to represent the beliefs of that node taking different labels. Then, the MPM labeling problem is equivalent to finding the label with the highest belief for each variable:

$$f_i^* = \arg\max_{f_i \in \mathcal{L}} b_i(f_i). \tag{A.2}$$

Suppose node x_i has n immediate parents $\mathbf{U} = \{u_1, u_2, \ldots, u_n\}$ (*i.e.,* there is a direct link from each u_i to x_i in the network) and m immediate children $\mathbf{Y} = \{y_1, y_2, \ldots, y_m\}$ (*i.e.,* there is a direct link from x_i to each y_i in the network). Note that $\mathbf{U} \cup \mathbf{Y} = \mathcal{N}_i$. Then, $\mathbf{e}$ can be split into two sets: $\mathbf{e_U}$, which represents the evidence provided through $\mathbf{U}$, and $\mathbf{e_Y}$, which represents the evidence provided through $\mathbf{Y}$, with $\mathbf{e} = \mathbf{e_U} \cup \mathbf{e_Y}$ and $\mathbf{e_U} \cap \mathbf{e_Y} = \emptyset$. Therefore, using the Bayes' theorem, $b_i(f_i)$ defined in Equation (A.1) can be expressed as:

$$b_i(f_i) = P(f_i|\mathbf{e_U}, \mathbf{e_Y}) = \alpha P(\mathbf{e_Y}|\mathbf{e_U}, f_i) P(f_i|\mathbf{e_U}) = \alpha P(\mathbf{e_Y}|f_i) P(f_i|\mathbf{e_U}), \tag{A.3}$$

where $\alpha = 1/P(\mathbf{e_Y}|\mathbf{e_U})$ serves as a normalizing constant rendering $\sum_{i=1}^{K} b_i(l_i) = 1$. Let $m_{u_j x_i}(f_{u_j})$ denote the message that x_i receives from its parent u_j given $u_j = f_{u_j}$, and $m_{y_j x_i}(f_i)$ the message that x_i receives from its child y_j given $x_i = f_i$. These messages are actually conditional probabilities given the evidence provided through those nodes:

$$m_{u_j x_i}(f_{u_j}) \triangleq P(u_j = f_{u_j}|\mathbf{e_{u_j}}), m_{y_j x_i}(f_i) \triangleq P(\mathbf{e_{y_j}}|x_i = f_i). \tag{A.4}$$

Then, $b_i(f_i)$ can be further factorized as:

$$b_i(f_i) = \alpha[\prod_{j=1}^{m} m_{y_j x_i}(f_i)]\{\sum_{f_{\mathbf{U}} \in \mathcal{L}^n} [P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} m_{u_j x_i}(f_{u_j})]\}, \tag{A.5}$$

where $P(f_i|f_{\mathbf{U}}) \triangleq P(x_i = f_i|\mathbf{U} = f_{\mathbf{U}})$ is one entry in the fixed $K \times \mathcal{L}^n$ transition matrix $P(x_i|\mathbf{U})$, which relates x_i to its immediate parents $\mathbf{U}$.

In the iterative message updating procedure, a node x_i sends a message $m^t_{x_i u_j}(f_i)$ to each of its parents given $u_j = f_{u_j}$ and $m^t_{x_i y_j}(f_i)$ to each of its children given $x_i = f_i$ based on the received messages from the previous iteration, and it updates the beliefs $b_i(f_i)$. The three belief updating rules are as follows:

$$m^t_{x_i u_j}(f_{u_j}) \leftarrow \beta \sum_{f_i \in \mathcal{L}} \{[\prod_{k=1}^{m} m^{t-1}_{y_k x_i}(f_i)]\{\sum_{f_{\mathbf{U} \setminus u_j}} [P(f_i|f_{\mathbf{U} \setminus u_j}) \prod_{k \neq j} m^{t-1}_{u_k x_i}(f_{u_k})]\}\}. \tag{A.6}$$

$$m^t_{x_i y_j}(f_i) \leftarrow \alpha[\prod_{k \neq j} m^{t-1}_{y_k x_i}(f_i)]\{\sum_{f_{\mathbf{U}}} [P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} m^{t-1}_{u_j x_i}(f_{u_j})]\}. \tag{A.7}$$

$$b^t_i(f_i) \leftarrow \alpha[\prod_{j=1}^{m} m^{t-1}_{y_j x_i}(f_i)]\{\sum_{f_{\mathbf{U}}} [P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} m^{t-1}_{u_j x_i}(f_{u_j})]\}. \tag{A.8}$$

α and β are normalizing constants. The procedure is initialized with all messages set to 1. An observed node sends out the same messages and does not receive messages. For singly connected networks, the computed beliefs are guaranteed to be exact at convergence, and therefore the exact MPM configuration can be found. However, the exact computation either of beliefs or of the MPM configuration cannot be guaranteed for networks with loops.

The second BP algorithm proposed by Pearl [140] is called *belief revision* (also known as max-product belief propagation, or BP-M), for producing the MAP configuration. It differs from BP-S in that, in order to get the MAP configuration, the probabilities of the other unobserved nodes need to be maximized rather than summed. Therefore, the belief $\hat{b}_i(f_i)$ at node x_i is defined as:

$$\hat{b}_i(f_i) \triangleq \max_{f} P(\mathcal{X} = f|\mathbf{e}) \triangleq \max_{f \setminus f_i} P(x_i = f_i, \mathcal{X} \setminus x_i = f \setminus f_i|\mathbf{e}). \tag{A.9}$$

Then, the MAP labeling problem expressed in Equation 2.5 is equivalent to finding the label with the highest belief for each variable:

$$f_i^* = \arg\max_{f_i \in \mathcal{L}} \hat{b}_i(f_i). \tag{A.10}$$

Differing from BP-S, $\hat{b}_i(f_i)$ may not sum to unity over f_i in BP-M. Let $\hat{m}_{u_j x_i}(f_{u_j})$ denote the message that x_i receives from its parent u_j given $u_i = f_{u_j}$, $\hat{m}_{y_j x_i}(f_i)$ the message that x_i receives from its child y_j given $x_i = f_i$. These messages can be expressed as maximization over probabilities:

$$\hat{m}_{u_j x_i}(f_{u_j}) \triangleq \max_{f_{\mathbf{U} \setminus u_j}} P(u_j = f_{u_j}, \mathbf{U} \setminus u_j = f_{\mathbf{U} \setminus u_j}), \hat{m}_{y_j x_i}(f_i) \triangleq \max_{f_{y_j}} P(y_j = f_{y_j}|x_i = f_i). \tag{A.11}$$

Therefore, $\hat{b}_i(f_i)$ can be factorized as:

$$\hat{b}_i(f_i) = \alpha \max_{f_{\mathbf{U}}} [\prod_{j=1}^{m} \hat{m}_{y_j x_i}(f_i) P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} \hat{m}_{u_j x_i}(f_{u_j})]. \tag{A.12}$$

The belief updating rules for BP-M are similar to those in BP-S:

$$\hat{m}^t_{x_i u_j}(f_{u_j}) \leftarrow \max_{f_i, f_{\mathbf{U}\setminus u_j}} [\prod_{j=1}^{m} \hat{m}^{t-1}_{y_j x_i}(f_i) P(f_i|f_{\mathbf{U}}) \prod_{k\neq j}^{n} \hat{m}^{t-1}_{u_k x_i}(f_{u_k})]. \tag{A.13}$$

$$\hat{m}^t_{x_i y_j}(f_i) \leftarrow \alpha \max_{f_{\mathbf{U}}} [\prod_{k\neq j}^{m} \hat{m}^{t-1}_{y_k x_i}(f_i) P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} \hat{m}^{t-1}_{u_j x_i}(f_{u_j})]. \tag{A.14}$$

$$\hat{b}^t_i(f_i) \leftarrow \alpha \max_{f_{\mathbf{U}}} [\prod_{j=1}^{m} \hat{m}^{t-1}_{y_j x_i}(f_i) P(f_i|f_{\mathbf{U}}) \prod_{j=1}^{n} \hat{m}^{t-1}_{u_j x_i}(f_{u_j})]. \tag{A.15}$$

Like BP-S, the procedure is initialized with all messages set to 1, and an observed node only sends out the same messages. For singly connected networks, the correct MAP configuration can be found at convergence. However, the correctness of the MAP configuration cannot be guaranteed for networks with loops.

Appendix B

Random Walks with Label Priors

The algorithm reviewed in Section 2.4.2 requires some of the variables to be instantiated. This requirement is relaxed in [68] by incorporating label priors and approaching the problem from a Bayesian perspective[1]. Given the likelihood probability (referred to as label prior in [68]) $\lambda_{i,k} \triangleq P(\mathcal{D}|x_i = l_k)$ of observing $\mathcal{D}$ given node x_i being assigned label l_k, and assuming that the prior probabilities $P(x_i = l_k)$ of x_i being assigned with different labels are equal, the posterior probability $P_k(x_i)$ can be expressed as follows using the Bayes theorem:

$$P_k(x_i) = \frac{\lambda_{i,k}}{\sum_{l_k \in \mathcal{L}} \lambda_{i,k}}. \tag{B.1}$$

Let Λ_k denote the diagonal matrix whose i-th diagonal entry is $\lambda_{i,k}$. Equation (B.1) can then be rewritten in a vector form:

$$(\sum_{l_k \in \mathcal{L}} \Lambda_k)\mathbf{P}_k = \lambda_k, \tag{B.2}$$

where $\mathbf{P}_k$ is a column vector representing the PMF $P_k(\cdot)$ defined on $\mathcal{X}$ and λ_k is a column vector whose i-th entry is $\lambda_{i,k}$. $\mathbf{P}_k$ satisfying Equation B.2 minimizes the following aspatial energy[2]:

$$E_{aspatial}(P_k) = (\mathbf{1} - \mathbf{P}_k)^T \Lambda_k (\mathbf{1} - \mathbf{P}_k) + \sum_{m \neq k} {\mathbf{P}_k}^T \Lambda_m \mathbf{P}_k. \tag{B.3}$$

Let the spatial energy be defined proportional to the Dirichlet integral (Equation 2.48):

$$E_{spatial}(P_k) = {\mathbf{P}_k}^T \mathbf{L} \mathbf{P}_k. \tag{B.4}$$

Then the total energy can be expressed as a combination of the spatial and aspatial energies via a free parameter γ:

$$E(P_k) = E_{spatial}(P_k) + \gamma E_{aspatial}(P_k), \tag{B.5}$$

The minimization of $E(P_k)$ is equivalent to solving the following linear system:

$$(\mathbf{L} + \gamma \sum_{l_m \in \mathcal{L}} \Lambda_m)\mathbf{P}_k = \gamma \lambda_k. \tag{B.6}$$

[1] Please note: This derivation in [68] is different from the Bayesian inference on MRFs.

[2] The Equations (B.3), (B.6), and (B.7) mentioned here are corrected versions for those printed in [68]. The corrected paper is available on Grady's personal webpage: http://cns.bu.edu/~lgrady/publications.html.

If there are labeled nodes, then only $\mathbf{P}_{U,k}$ needs to be calculated for each $l_k \in \mathcal{L}$. Therefore, Equation B.6 becomes:

$$(\mathbf{L}_{UU} + \gamma \sum_{l_m \in \mathcal{L}} \Lambda_{U,m})\mathbf{P}_{U,k} = \gamma\lambda_{U,k} - \mathbf{L}_{UL}\mathbf{P}_{L,k}. \tag{B.7}$$

Appendix C

Conditional Density Derivation in MGCRFs

As introduced in Section 3.4.2, the conditional density $p(\tilde{\mathbf{x}}|\tilde{\mathcal{B}}, \mathcal{D})$ follows a multivariate Gaussian distribution:

$$p(\tilde{\mathbf{x}}|\tilde{\mathcal{B}}, \mathcal{D}) \propto \exp(-(\frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}{\Delta_{ik}^{\tilde{\mathcal{B}}}}^{T}\tilde{\mathbf{T}}_{ik}\Delta_{ik}^{\tilde{\mathcal{B}}} + \frac{1}{4}\sum_{e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}}{\Delta_{ij}^{\tilde{\mathcal{X}}}}^{T}\tilde{\mathbf{S}}_{ij}\Delta_{ij}^{\tilde{\mathcal{X}}})). \tag{C.1}$$

Let us denote the first term (*i.e.,* unary term) in the negative exponent of Equation (C.1) as E_{unary} and the second term (*i.e.,* pairwise term) as $E_{pairwise}$. Now, we derive the two terms separately.

$$\begin{aligned}
E_{unary} &= \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}{\Delta_{ik}^{\tilde{\mathcal{B}}}}^{T}\tilde{\mathbf{T}}_{ik}\Delta_{ik}^{\tilde{\mathcal{B}}} \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}(\mathbf{x}_i - \mathbf{b}_k)^T\tilde{\mathbf{T}}_{ik}(\mathbf{x}_i - \mathbf{b}_k) \\
&= \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{N}\begin{pmatrix}\mathbf{x}_i\\ \mathbf{b}_k\end{pmatrix}^T\begin{pmatrix}\tilde{\mathbf{T}}_{ik} & -\tilde{\mathbf{T}}_{ik}\\ -\tilde{\mathbf{T}}_{ik} & \tilde{\mathbf{T}}_{ik}\end{pmatrix}\begin{pmatrix}\mathbf{x}_i\\ \mathbf{b}_k\end{pmatrix} \\
&= \frac{1}{2}\sum_{k=1}^{K}\begin{pmatrix}\tilde{\mathbf{x}}\\ \mathbf{b}_k\end{pmatrix}^T\begin{pmatrix}\tilde{\mathbf{T}}_{1k} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\tilde{\mathbf{T}}_{1k}\\ \mathbf{0} & \tilde{\mathbf{T}}_{2k} & \mathbf{0} & \mathbf{0} & -\tilde{\mathbf{T}}_{2k}\\ \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \vdots\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \tilde{\mathbf{T}}_{Nk} & -\tilde{\mathbf{T}}_{Nk}\\ -\tilde{\mathbf{T}}_{1k} & -\tilde{\mathbf{T}}_{2k} & \cdots & -\tilde{\mathbf{T}}_{Nk} & \sum_{i=1}^{N}\tilde{\mathbf{T}}_{ik}\end{pmatrix}\begin{pmatrix}\tilde{\mathbf{x}}\\ \mathbf{b}_k\end{pmatrix} \\
&= \frac{1}{2}\begin{pmatrix}\tilde{\mathbf{x}}\\ \tilde{\mathbf{b}}\end{pmatrix}^T\begin{pmatrix}\sum_{k=1}^{K}\tilde{\mathbf{T}}_{1k} & \mathbf{0} & \mathbf{0} & -\tilde{\mathbf{T}}_{11} & \cdots & -\tilde{\mathbf{T}}_{1K}\\ \mathbf{0} & \ddots & \mathbf{0} & \vdots & \vdots & \vdots\\ \mathbf{0} & \mathbf{0} & \sum_{k=1}^{K}\tilde{\mathbf{T}}_{Nk} & -\tilde{\mathbf{T}}_{N1} & \cdots & -\tilde{\mathbf{T}}_{NK}\\ -\tilde{\mathbf{T}}_{11} & \cdots & -\tilde{\mathbf{T}}_{N1} & \sum_{i=1}^{N}\tilde{\mathbf{T}}_{i1} & \mathbf{0} & \mathbf{0}\\ \vdots & \vdots & \vdots & \mathbf{0} & \ddots & \mathbf{0}\\ -\tilde{\mathbf{T}}_{1K} & \cdots & -\tilde{\mathbf{T}}_{NK} & \mathbf{0} & \mathbf{0} & \sum_{i=1}^{N}\tilde{\mathbf{T}}_{iK}\end{pmatrix}\begin{pmatrix}\tilde{\mathbf{x}}\\ \tilde{\mathbf{b}}\end{pmatrix} \\
&= \frac{1}{2}\begin{pmatrix}\tilde{\mathbf{x}}\\ \tilde{\mathbf{b}}\end{pmatrix}^T\begin{pmatrix}\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}} & \tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}}\\ \tilde{\mathbf{T}}_{\tilde{\mathcal{B}}\tilde{\mathcal{X}}} & \tilde{\mathbf{T}}_{\tilde{\mathcal{B}}\tilde{\mathcal{B}}}\end{pmatrix}\begin{pmatrix}\tilde{\mathbf{x}}\\ \tilde{\mathbf{b}}\end{pmatrix} \\
&= \frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}}\tilde{\mathbf{x}} + \tilde{\mathbf{x}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}}\tilde{\mathbf{b}} + \epsilon,
\end{aligned} \tag{C.2}$$

where $\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}}$ is an $N \times N$ diagonal matrix; $\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}} = \tilde{\mathbf{T}}^T_{\tilde{\mathcal{B}}\tilde{\mathcal{X}}}$ is an $N \times K$ matrix; $\tilde{\mathbf{T}}^1_{\tilde{\mathcal{B}}\tilde{\mathcal{B}}}$ is a $K \times K$ matrix; and $\epsilon = \frac{1}{2}\tilde{\mathbf{b}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{B}}\tilde{\mathcal{B}}}\tilde{\mathbf{b}}$ is a constant.

$$\begin{aligned} E_{pairwise} &= \frac{1}{4}\sum_{e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}} \Delta_{ij}^{\tilde{\mathcal{X}}^T}\tilde{\mathbf{S}}_{ij}\Delta_{ij}^{\tilde{\mathcal{X}}} \\ &= \frac{1}{4}\sum_{e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}} (\mathbf{x}_i - \mathbf{x}_j)^T\tilde{\mathbf{S}}_{ij}(\mathbf{x}_i - \mathbf{x}_j) \\ &= \frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{P}}\tilde{\mathbf{x}}, \end{aligned} \tag{C.3}$$

where each $K \times K$ element of $\tilde{\mathbf{P}}$ is defined as:

$$\tilde{\mathbf{P}}_{ij} = \begin{cases} \sum_{\mathbf{x}_j\in\mathcal{N}_i}\tilde{\mathbf{S}}_{ij}, & i = j; \\ -\tilde{\mathbf{S}}_{ij}, & e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}; \\ 0, & \text{otherwise.} \end{cases} \tag{C.4}$$

If we substitute Equations (C.2) and (C.3) into Equation (C.1), we have:

$$\begin{aligned} p(\tilde{\mathbf{x}}|\tilde{\mathcal{B}},\mathcal{D}) &\propto \exp(-E_{unary} - E_{pairwise}) \\ &= \exp(-\frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}}\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}}\tilde{\mathbf{b}} - \epsilon - \frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{P}}\tilde{\mathbf{x}}) \\ &\propto \exp(-\frac{1}{2}\tilde{\mathbf{x}}^T(\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}} + \tilde{\mathbf{P}})\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T\tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}}\tilde{\mathbf{b}}) \\ &= \exp(-\frac{1}{2}\tilde{\mathbf{x}}^T\tilde{\mathbf{Z}}\tilde{\mathbf{x}} - \tilde{\mathbf{x}}^T\tilde{\mathbf{R}}\tilde{\mathbf{b}}), \end{aligned} \tag{C.5}$$

where $\tilde{\mathbf{Z}} = \tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{X}}} + \tilde{\mathbf{P}}$, *i.e.,* each $K \times K$ element of $\tilde{\mathbf{Z}}$ is defined as:

$$\tilde{\mathbf{Z}}_{ij} = \begin{cases} \sum_{k=1}^{K}\tilde{\mathbf{T}}_{ik} + \sum_{\mathbf{x}_m\in\mathcal{N}_i}\tilde{\mathbf{S}}_{im}, & i = j; \\ -\tilde{\mathbf{S}}_{ij}, & e_{ij}\in\mathcal{E}_{\tilde{\mathcal{X}}}; \\ 0, & \text{otherwise.} \end{cases} \tag{C.6}$$

and $\tilde{\mathbf{R}} = \tilde{\mathbf{T}}_{\tilde{\mathcal{X}}\tilde{\mathcal{B}}}$, *i.e.,* each $K \times K$ element of $\tilde{\mathbf{R}}$ is defined as:

$$\tilde{\mathbf{R}}_{ik} = -\tilde{\mathbf{T}}_{ik}. \tag{C.7}$$

Appendix D

Experimental Results of Perception-Guided Multi-Exposure Fusion

D.1 Comparison with Other Methods

Our perception-guided fusion method was compared with two state-of-the-art multi-exposure fusion (MEF) methods and two state-of-the-art tone-mapping (TM) methods on six different scenes (Figures D.1 to D.6). PGF denotes our method that takes final probabilities directly as fusion weights. PGF-A denotes our method that uses a linear fusion weighting function defined on the final probabilities. PF denotes the probabilistic fusion method proposed in Section 3.3. EF denotes the exposure fusion method proposed by Mertens *et al.* [130]. PTR denotes the photographic tone reproduction TM operator proposed by Reinhard *et al.* [156]. iCAM06 denotes the TM operator proposed by Kuang *et al.* [94]. The results of PF, EF, PTR, and iCAM06 were generated by the programs provided by their respective authors. PGF, PGF-A, PF, EF, and iCAM06 are Matlab implementations; PTR is a C implementation. The same parameter setting introduced in Section 4.2.8 was used in PGF and PGF-A for all experiments unless otherwise mentioned. The default parameter settings in PF, EF, and iCAM06 were used. The parameters in PTR were generated using the estimation technique in [155]. The high dynamic range (HDR) images for TM methods were generated using HDR reconstruction [46] from the corresponding source sequences.

D.1.1 Objective Evaluation Using DRIVDP

The dynamic range independent visible difference predictor (DRIVDP) [11] was used to assess the visual distortions between a test image and each source image. Three distortions were considered: loss of visible contrast; amplification of invisible contrast; and reversal of visible contrast. These distortions were reflected in a distortion map, where green, blue, red, and gray pixels indicated contrast loss, amplification, reversal, and no distortion, respectively. We assumed that the images were viewed on a typical LCD with a max luminance equivalent to $100cd/m^2$, a gamma value of

2.2, and a visual resolution of 30 pixels per degree at a viewing distance of 0.5 meter, and that the peak contrast sensitivity of the viewer was 0.25%.

The evaluation result using two images from the Memorial Church sequence is given in Figure D.7 (please refer to Figure D.1 for a visual comparison). The source image on the left offers good clarity for the ceiling and wall that appear in the right portion of the image. The source image on the right offers good clarity for the lower left window. For the ceiling and wall, PF, PGF, and PGF-A show relatively less distortion, followed by EF, iCAM06, and PTR. For the lower left window, iCAM06 shows the least distortion, followed by PGF, PGF-A, PTR, PF, and EF.

The evaluation result on the National Cathedral sequence is given in Figure D.8 (please refer to Figure D.2 for a visual comparison). The two source images with good clarity respectively for the windows and the walls are given in Figure D.8(a). The contrast distortion maps generated using DRIVDP for each method are given in Figure D.8(b)-(g). PGF performs best in preventing contrast distortions, followed by PGF-A, PF, EF, PTR, and iCAM06.

The evaluation result using two images from the House sequence is given in Figure D.9 (please refer to Figure D.3 for a visual comparison). The source image at the top offers good clarity for the outdoor scene, and the source image at the bottom offers good clarity for the bookshelf. For these two source images, PGF-A and EF perform similarly and best, followed by PGF, PF, iCAM06, and PTR.

The evaluation result using two images from the Chateau sequence is given in Figure D.10 (please refer to Figure D.4 for a visual comparison). The source image on the left offers good clarity for the balcony and trees. For these regions, PGF, PGF-A, and iCAM06 perform similarly, followed by PTR, PF, and EF. The source image on the right offers good clarity for the indoor scene. For the indoor scene, PGF, PGF-A, PF, and EF perform similarly, followed by PTR and iCAM06.

The evaluation result using two images from the Lamp sequence is given in Figure D.11 (please refer to Figure D.5 for a visual comparison). The two source images offer good clarity for the bulb and the books, respectively. For the bulb, PGF-A shows the least distortion, followed by PGF, PTR, PF, EF, and iCAM06. For the books, PGF-A shows the least distortion, followed by EF, PF, PGF, iCAM06, and PTR.

The evaluation result using two images from the Belgium House sequence is given in Figure D.12 (please refer to Figure D.6 for a visual comparison). The two source images offer good clarity for part of the outdoor scene and the indoor scene, respectively. For the outdoor scene, iCAM06 shows the least distortion, followed by PGF-A, PGF, PF, EF, and PTR. For the indoor scene, PGF-A and EF show the least distortion, followed by PGF, PF, PTR, and iCAM06.

On the average, for the twelve source images considered in this evaluation, the ranking of the six methods are: PGF-A $>$ PGF $>$ EF $\approx$ PF $>$ iCAM06 $>$ PTR.

D.1.2 Subjective Evaluation

The four criteria in our evaluation, *i.e., global contrast*, *details*, *colors*, and *overall appearance*, are a subset of those used by Čadík *et al.* [34]. The two criteria not used are *brightness* and *artifacts*. The *brightness* criterion is not used because its impact already spreads into the other criteria and therefore reveals itself indirectly [34]. The reason for not using the *artifacts* criterion is two-fold. 1) The subjects were non-experts with (almost) no prior experience in MEF or TM. Therefore, their understanding of artifacts can vary from one individual to another. It is difficult for them to identify artifacts solely based on their own judgement. 2) The methods selected for our experiment did not produce major artifacts except some minor color distortion in some results, but this can be covered under the *colors* criterion.

The subjective evaluation results are summarized in Figure 4.12. Our PGF-A received the highest average ranking scores for all four criteria on five scenes, and performs similarly to our PGF on the National Cathedral sequence. Considering all six scenes, the ranking from high to low for the six methods is: PGF-A $>$ EF $>$ PGF $>$ PF $>$ iCAM06 $>$ PTR for the *global contrast* criterion; PGF-A $>$ PGF $>$ EF $>$ iCAM06 $\approx$ PF $>$ PTR for the *details* criterion; PGF-A $>$ EF $>$ PGF $>$ PF $>$ iCAM06 $>$ PTR for the *colors* criterion; PGF-A $>$ PGF $\approx$ EF $>$ PF $>$ iCAM06 $>$ PTR for the *overall appearance* criterion. On average, the ranking of the six methods on the six scenes from high to low is: PGF-A $>$ EF $\approx$ PGF $>$ PF $>$ iCAM06 $>$ PTR.

The experimental results also suggest that there is no single factor that dominates the user preference, but rather that it is the combination of different factors (*e.g.,* global contrast and details) that determines a user's choice. Below are two examples:

- Although some images may suffer some loss of subtle details due to the enhanced contrast, high global contrast and good color scheme can compensate for that in the visual impression, which results in higher ranking scores. For the Lamp scene (Figure D.5), PGF received higher rating than EF under the *details* criterion, but EF received higher ratings under the *global contrast* and the *colors* criteria. Under the *overall appearance* criterion, EF received a higher rating than PGF.

- On the contrary, good detail reproduction may also compensate for low global contrast or less vivid color scheme in the overall impression. For the Chateau scene (Figure D.4), EF received higher ratings than PF under the *global contrast* and the *colors* criteria, but PF received a slightly better *details* rating. Both methods received similar ratings under the *overall appearance* criterion. Also for the Chateau scene, PGF received similar ratings with EF under the *global contrast* and the *colors* criteria, but received a much higher rating under the *details* criterion. Under the *overall appearance* criterion, PGF has higher rating than EF.

Because user interaction is usually exploited in TM to generate images of different appearances, we also tested different parameter settings of iCAM06. In iCAM06, the parameter p controls the

brightness of the tone-mapped image. The default value of p in the implementation of iCAM06 provided by the authors of [94] is 0.7, which is the value that we took in the above experiments. A larger p results in darker image and a smaller p results in brighter image. The suggested range of p in [94] is $[0.6, 0.85]$. We also tested $p = 0.6$ and $p = 0.4$ for the six test scenes. The results are given in Figure D.13, along with comparisons with our PGF and PGF-A. For each scene, the results by iCAM06 with $p = 0.6$, iCAM06 with $p = 0.4$, PGF, and PGF-A are given on the left most, left, right, and far right, respectively. Our PGF and PGF-A produce better details, contrasts, and colors.

D.1.3 Comparison with Perception-Based Tone Mapping Methods

Our fusion methods (PGF and PGF-A) were compared with two state-of-the-art TM methods (PTR and iCAM06) in the previous sections. Among the two TM methods, iCAM06 is perception-based. To the best of our knowledge, our fusion scheme is the first one to introduce contrast perception into MEF. Although MEF and TM have different workflows, our fusion methods are also perception-based. Therefore, in this section, we compare our methods with two more perception-based TM methods, the display adaptive TM (DATM) operator [127] and the two-stage TM (TSTM) operator [56]. The C implementations of these two operators are provided by their respective authors. The input HDR images were generated using HDR reconstruction from their corresponding source sequences.

For DATM, we used the parameter setting for a typical office LCD. The gamma value of the display is 2.2; the peak luminance of the display is $100cd/m^2$; the black level of the display is $0.8cd/m^2$; the reflectivity of the display is 1%; and the ambient illumination is 400 lux. This setting corresponds to the setting in our objective evaluation using DRIVDP. Figure D.14 gives the comparison between DATM, EF and our PGF-A. For each scene, the results of DATM, EF, and PGF-A are given on the left, middle, and right, respectively. DATM produces images with quality close to EF on average. Our PGF-A produces better details, contrasts, and colors.

For TSTM, there are two free parameters in their program. We set the Gaussian kernel radius parameter σ to the value reported in the paper, *i.e.,* 30 percent of the number of pixels along the longer dimension of the input HDR image. The other parameter ρ controls the brightness of the tone-mapped image. A smaller ρ results in a brighter image. We tested the performance of TSTM with the default ρ value of 0, and also with the setting $\rho = -5$. The results are given in Figure D.15, along with a comparison to those of our PGF and PGF-A. A smaller ρ in TSTM helps to reveal more details in five scenes except in National Cathedral, where over-exposure appears in the window region. Our PGF and PGF-A produce better details, contrasts, and colors.

D.2 Evaluation of Different Transducer Functions

Our fusion method currently employs the transducer function proposed by Foley and Schwarz [58]. Different transducer functions were proposed by other researchers, *e.g.,* [62, 197]. A comparison

between different transducer functions is given in Figure D.16. With the exception of the transducer function, all of other settings in our fusion algorithm were the same when generating the fusion results. The parameters in the individual transducer functions were all set to the reported values from their respective papers. García-Pérez's [62] transducer function does not produce satisfactory results for multi-exposure fusion, where over- and under-exposures occur in the fusion results. Wilson's [197] transducer function for threshold and suprathreshold vision produces fusion results with quality very close to Foley's [58].

D.3 Evaluation of Different Saturation Measures

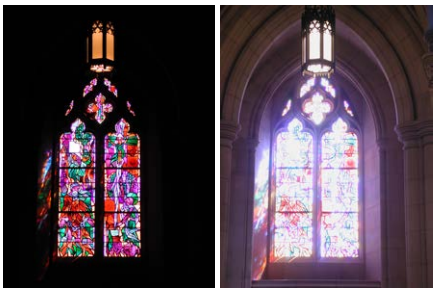
Our fusion method currently employs the saturation measure defined in the LHS (luminance, hue, and saturation) color space [176]. Other saturation measures, such as the saturation measure defined in the HSV (hue, saturation, and value) color space, Lübbe's [121] saturation measure, and Mertens' [130] saturation measure, generate similar results in our fusion scheme. A comparison between different saturation measures is given in Figure D.17. With the exception of the saturation calculation scheme, all of other settings in our fusion algorithm were the same when generating the fusion results. The saturation measures in the LHS and HSV color spaces, and Lübbe's saturation measure, generate very similar fusion results. The fused images using Mertens' saturation measure show slight over-exposure when applied in PGF-A.

(a)

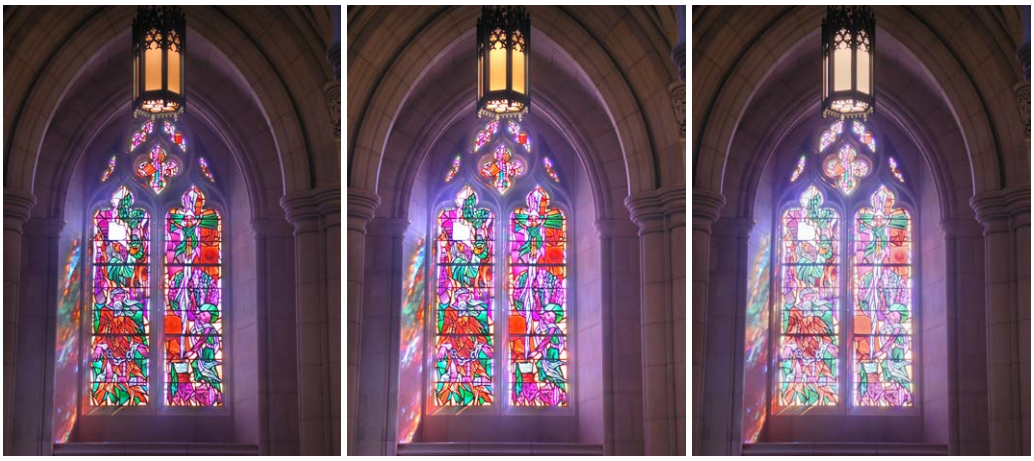
(b) (c) (d)

(e) (f) (g)

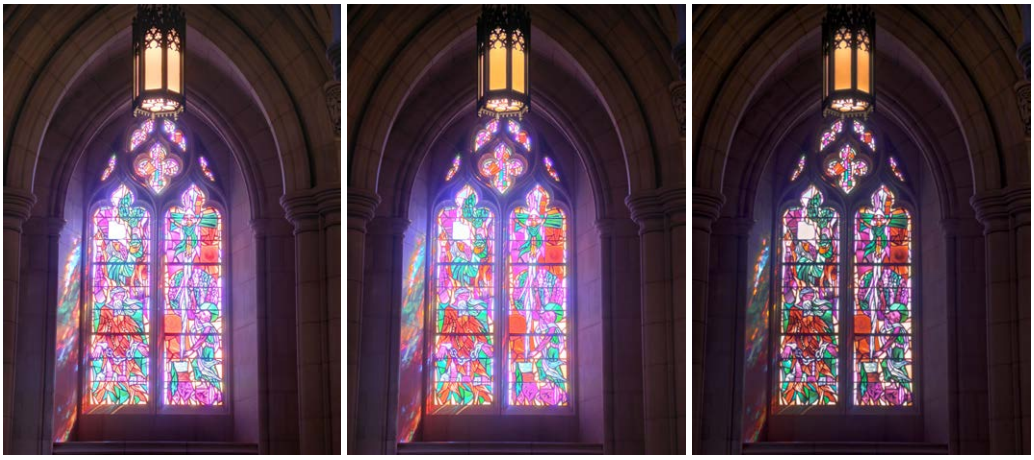
Figure D.1: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Memorial Church sequence. PGF and PGF-A preserve more details than the others, especially in the window regions. PGF-A produces higher global contrast than PGF, but may cause the loss of subtle details, *e.g.,* in the window regions. Our PGF-A has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.



(a)



(b) (c) (d)



(e) (f) (g)

Figure D.2: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the National Cathedral sequence. PGF and PGF-A generate the best results, where details in all regions are preserved with high local contrasts and saturated colors. iCAM06 preserves as many details as ours in the window regions but with less vivid colors. Our PGF has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation, followed by our PGF-A. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

(a)

(b) (c)

(d) (e)

(f) (g)

Figure D.3: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the House sequence. Our PGF-A has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

(a)

(b) (c)

(d) (e)

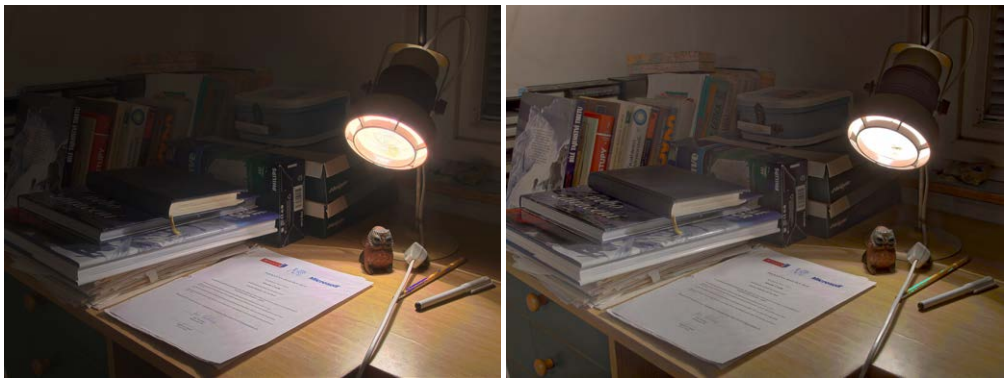
(f) (g)

Figure D.4: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Chateau sequence. Our PGF-A has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

(a)

(b) (c)

(d) (e)



(f) (g)

Figure D.5: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Lamp sequence. Our PGF-A has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

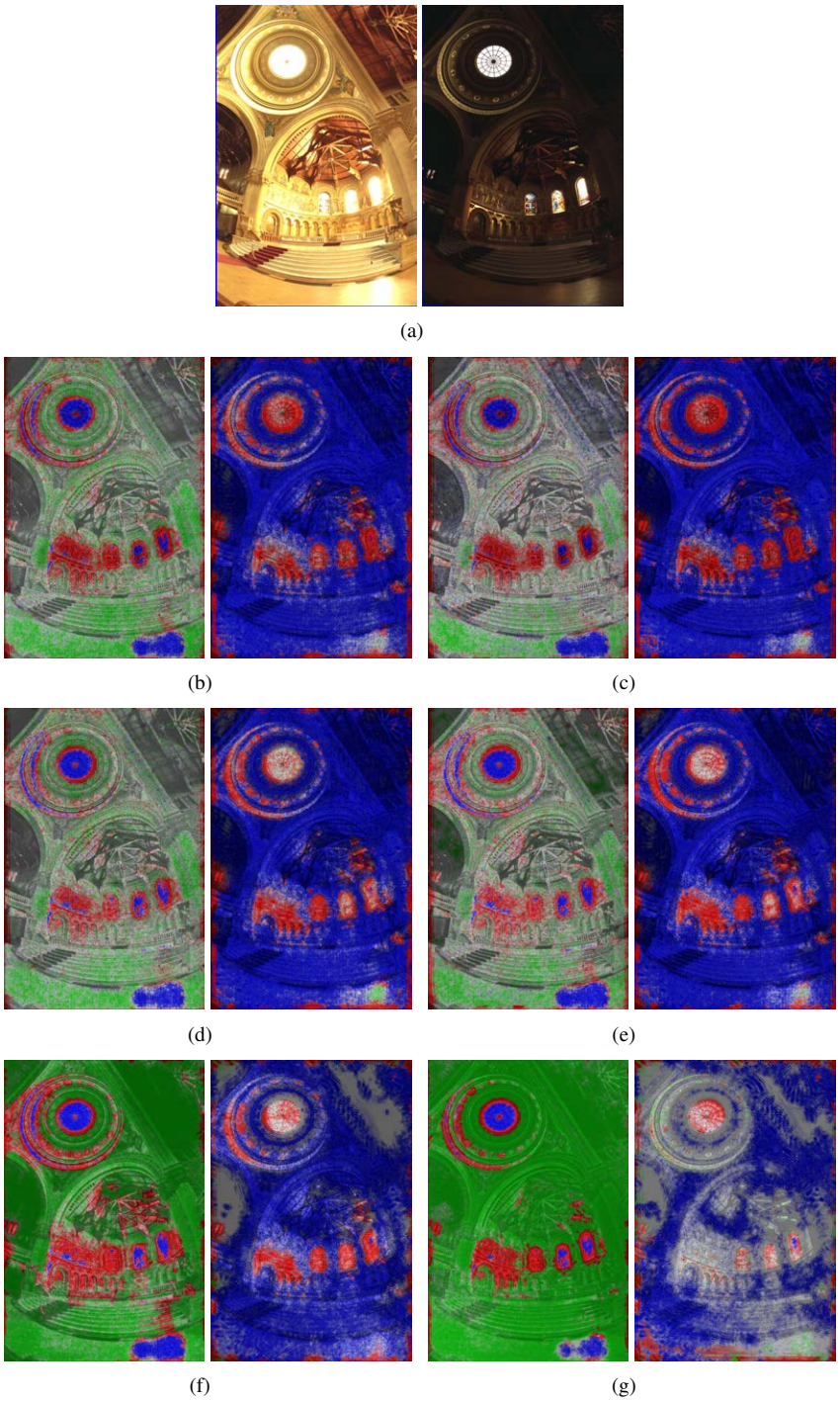
(a)

(b) (c)

(d) (e)

(f) (g)

Figure D.6: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Belgium House sequence. Our PGF-A has the highest average ranking scores under all four criteria on this sequence in the subjective evaluation. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.



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

Figure D.7: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Memorial Church sequence using DRIVDP. The source image on the left offers good clarity for the ceiling and wall, which appear in the right portion of the image. The source image on the right offers good clarity for the lower left window. In a distortion map, green, blue, red, and gray pixels indicate contrast loss, amplification, reversal, and no distortion, respectively. For the ceiling and wall, PF, PGF, and PGF-A show relatively less distortion, followed by EF, iCAM06, and PTR. For the lower left window, iCAM06 shows the least distortion, followed by PGF, PGF-A, PTR, PF, and EF. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

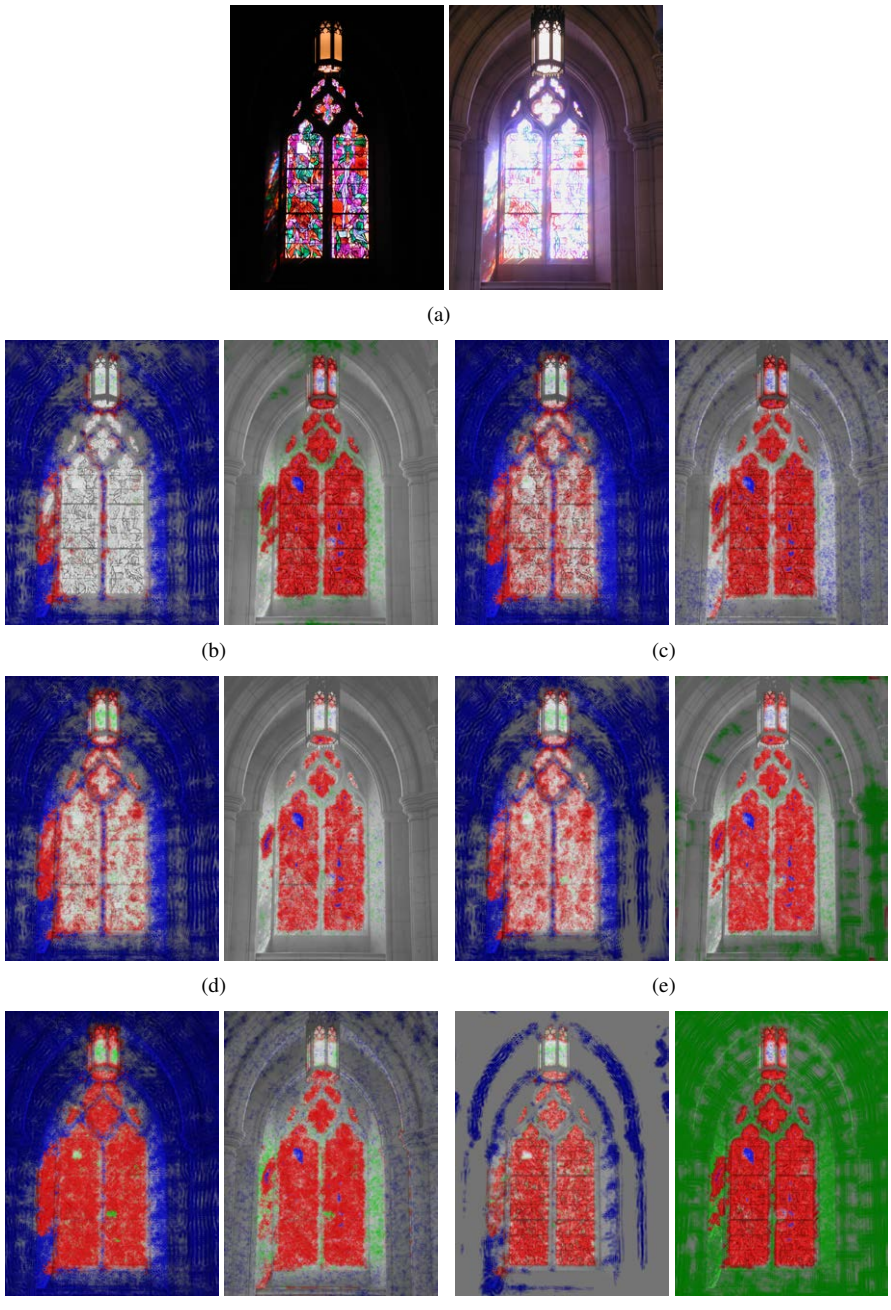
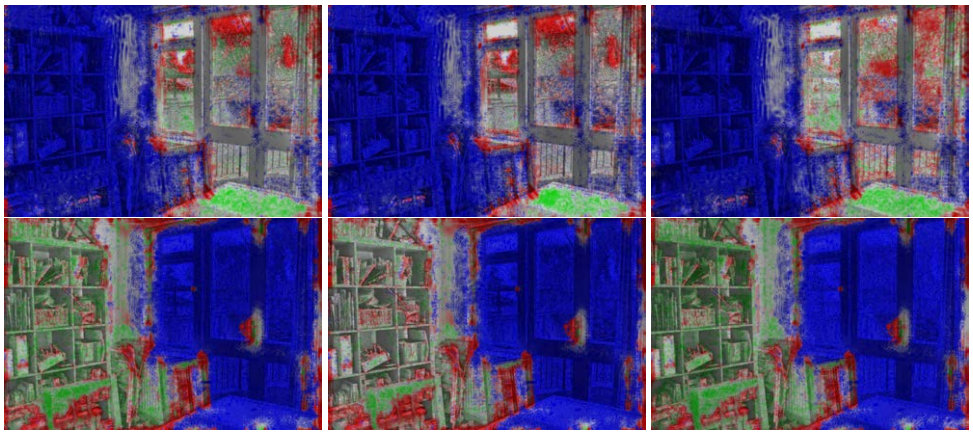
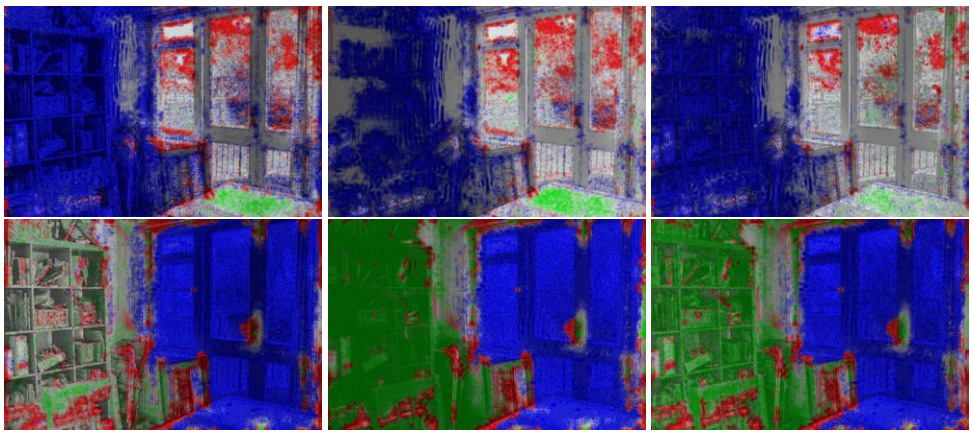


Figure D.8: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the National Cathedral sequence using DRIVDP. The two source images offer good clarity for the windows and the wall, respectively. PGF performs best in preventing contrast distortions, followed by PGF-A, PF, EF, PTR, and iCAM06. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

(a)



(b) (c) (d)



(e) (f) (g)

Figure D.9: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the House sequence using DRIVDP. The two source images offer good clarity for the outdoor scene and the bookshelf, respectively. For these two source images, PGF-A and EF performs similarly and best, followed by PGF, PF, iCAM06, and PTR. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

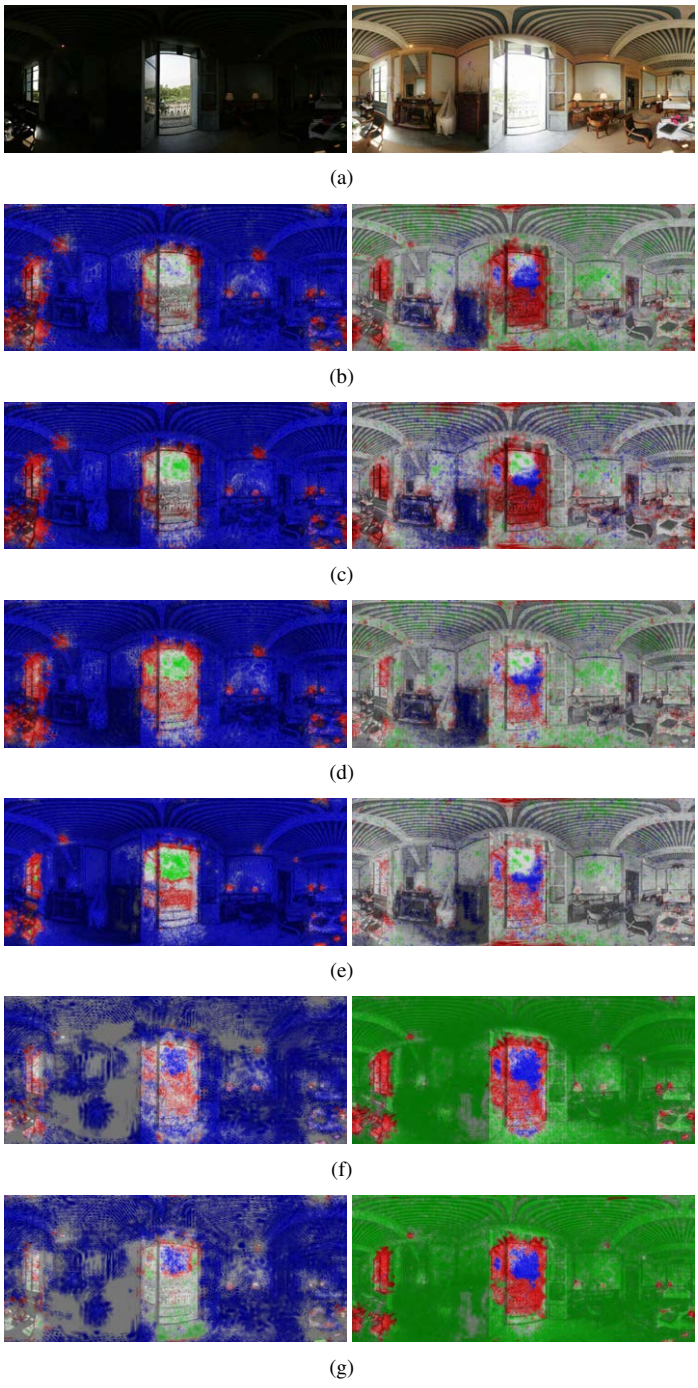
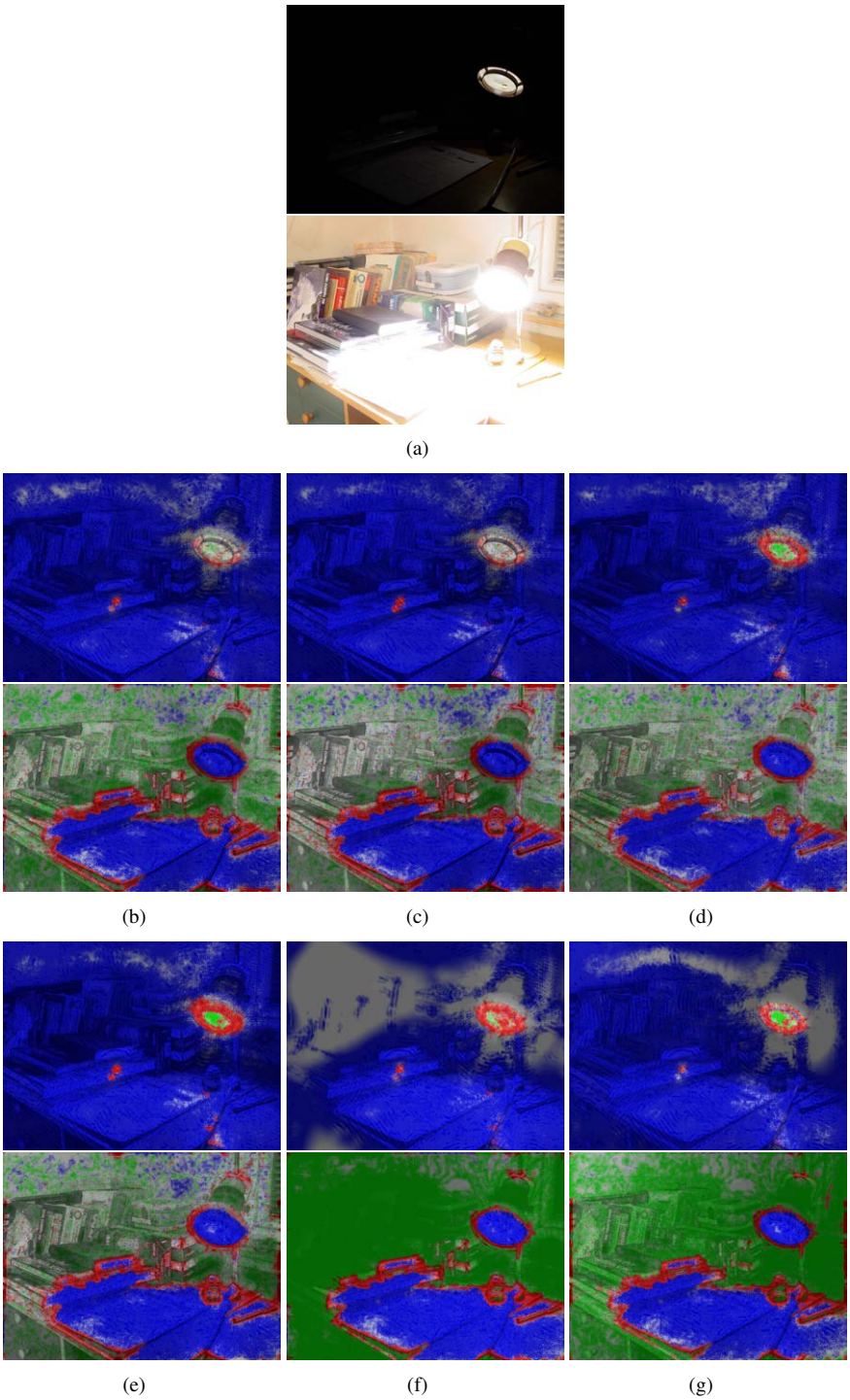
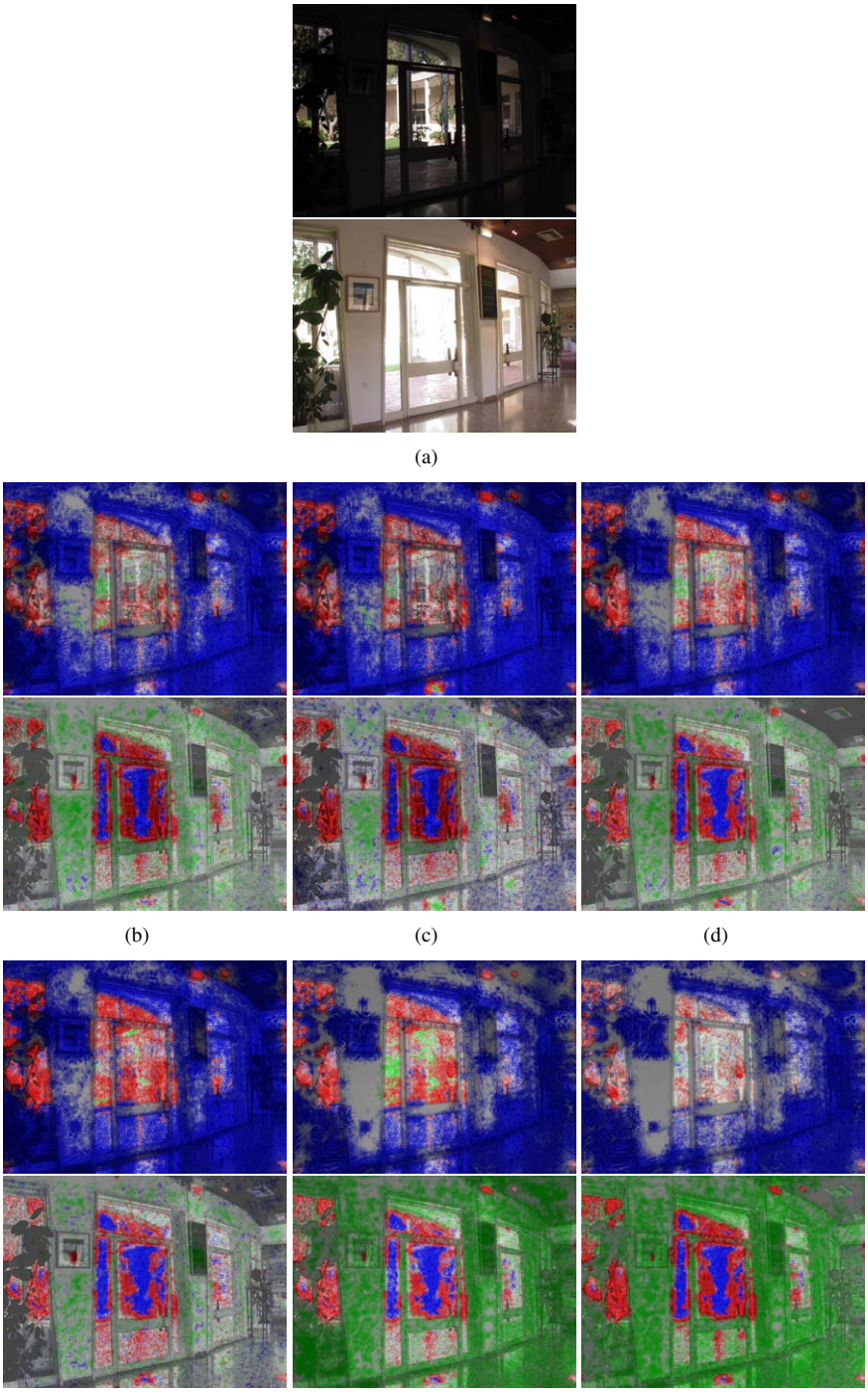


Figure D.10: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Chateau sequence using DRIVDP. The source image on the left offers good clarity for the balcony and trees. For these regions, PGF, PGF-A, and iCAM06 perform similarly and best, followed by PTR, PF, and EF. The source image on the right offers good clarity for the indoor scene. For the indoor scene, PGF, PGF-A, PF, and EF perform similarly and best, followed by PTR and iCAM06. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.



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

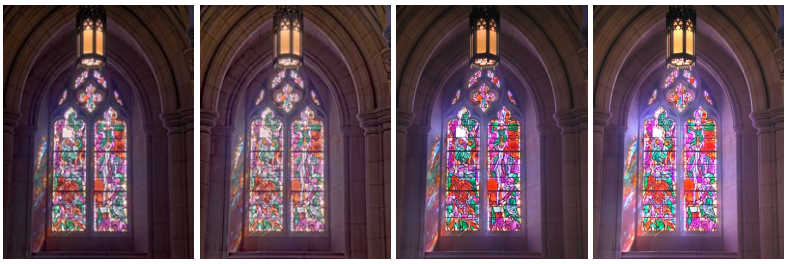
Figure D.11: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Lamp sequence using DRIVDP. The two source images offer good clarity for the bulb and the books, respectively. For the bulb, PGF-A shows the least distortion, followed by PGF, PTR, PF, EF, and iCAM06. For the books, PGF-A shows the least distortion, followed by EF, PF, PGF, iCAM06, and PTR. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.



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

Figure D.12: Comparison of PGF and PGF-A with PF, EF, PTR, and iCAM06 on the Belgium House sequence using DRIVDP. The two source images offer good clarity for part of the outdoor scene and the indoor scene, respectively. For the outdoor scene, iCAM06 shows the least distortion, followed by PGF-A, PGF, PF, EF, and PTR. For the indoor scene, PGF-A and EF show the least distortion, followed by PGF, PF, PTR, and iCAM06. (a) Source sequence. (b) PGF. (c) PGF-A. (d) PF. (e) EF. (f) PTR. (g) iCAM06.

(a)



(b)

(c)

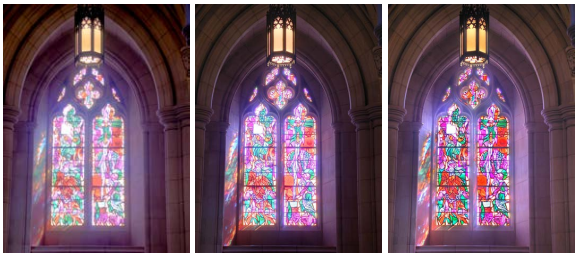
(d)

(e)

(f)

Figure D.13: Comparison of iCAM06 with $p = 0.6$, iCAM06 with $p = 0.4$, our PGF, and our PGF-A. For each scene, the results by iCAM06 with $p = 0.6$, iCAM06 with $p = 0.4$, PGF, and PGF-A are given at the left most, left, right, and right most, respectively. PGF and PGF-A produce better details, contrasts, and colors. (a) Results on the Memorial Church sequence. (b) Results on the National Cathedral sequence. (c) Results on the House sequence. (d) Results on the Chateau sequence. (e) Results on the Lamp sequence. (f) Results on the Belgium House sequence.

(a)



(b)

(c)

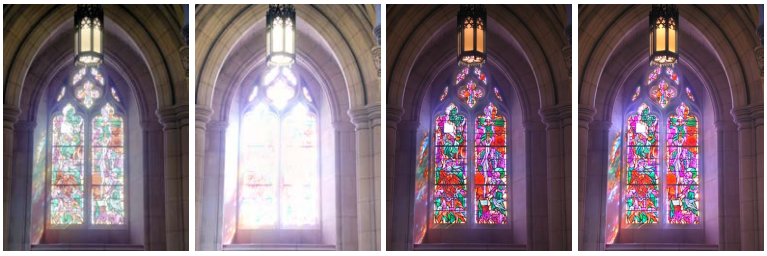
(d)

(e)

(f)

Figure D.14: Comparison of DATM, EF and our PGF-A. For each scene, the results by DATM, EF, and PGF-A are given at the left, middle, and right, respectively. DATM produces images with quality close to EF on average. PGF-A produces better details, contrasts, and colors. (a) Results on the Memorial Church sequence. (b) Results on the National Cathedral sequence. (c) Results on the House sequence. (d) Results on the Chateau sequence. (e) Results on the Lamp sequence. (f) Results on the Belgium House sequence.

(a)



(b)

(c)

(d)

(e)

(f)

Figure D.15: Comparison of TSTM with $\rho = 0$, TSTM with $\rho = -5$, our PGF, and our PGF-A. For each scene, the results by TSTM with $\rho = 0$, TSTM with $\rho = -5$, PGF, and PGF-A are given at the left most, left, right, and right most, respectively. A smaller ρ in TSTM helps to reveal more details in five scenes, but not in National Cathedral, where over-exposure appears in the window region. PGF and PGF-A produce better details, contrasts, and colors. (a) Results on the Memorial Church sequence. (b) Results on the National Cathedral sequence. (c) Results on the House sequence. (d) Results on the Chateau sequence. (e) Results on the Lamp sequence. (f) Results on the Belgium House sequence.

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

Figure D.16: Comparison of different transducer functions on the Lamp sequence. García-Pérez's transducer function does not produce satisfactory results for multi-exposure fusion, where over- and under-exposures appear in the fusion results. Wilson's transducer function produces results very close to those of Foley's transducer function. (a) PGF with Foley's transducer function. (b) PGF-A with Foley's transducer function. (c) PGF with Wilson's transducer function. (d) PGF-A with Wilson's transducer function. (e) PGF with García-Pérez's transducer function. (f) PGF-A with García-Pérez's transducer function.

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

Figure D.17: Comparison of different saturation measures on the Belgium House sequence. The saturation measures in the LHS and HSV color spaces, and Lübbe's saturation measure generate very close fusion results. The fused images using Mertens' saturation measure show a little over-exposure when applied in PGF-A. Overall, these different measures generate similar results in our fusion scheme. (a) PGF with saturation defined in the LHS color space. (b) PGF-A with saturation defined in the LHS color space. (c) PGF with saturation defined in the HSV color space. (d) PGF-A with saturation defined in the HSV color space. (e) PGF with Lübbe's saturation measure. (f) PGF-A with Lübbe's saturation measure. (g) PGF with Mertens' saturation measure. (h) PGF-A with Mertens' saturation measure.