Smoothness-Transferred Random Field

by

Donglai Wei

B.S., Mathematics, Brown University, 2011

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science
in Electrical Engineering and Computer Science
at the Massachusetts Institute of Technology

September 2013

Signature of Author: ______________________________
Department of Electrical Engineering and Computer Science
August 30, 2013

Certified by: ______________________________
William T.Freeman
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: ______________________________
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Committee for Graduate Students

Smoothness-Transfer Random Field

by Donglai Wei

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of
Master of Science

Abstract

We propose a new random field (RF) model, smoothness-transfer random field (ST-RF) model, for image modeling.

In the objective function of RF models, smoothness energy is defined with compatibility function to capture the relationship between neighboring local regions, while data energy for the evidence from local regions. Usually, the smoothness energy is constructed in terms of a fixed set of filters or basis which can be learned from training examples and steered to local structures in test examples. ST-RF, on the other hand, takes the data-driven approach to nonparametrically model the compatibility function for smoothness energy.

The pipeline for our ST-RF model is as follows: first for each training example, we build a RF model with "ground truth smoothness energy", where the compatibility function is constructed from ground truth value. Then, for each test example, we use data-driven method to find its correspondence with training examples. Lastly, we construct the smoothness energy of ST-RF for each test example by transferring the compatibility function from matched region. After construction, we applies traditonal RF inference and learning algorithms to obtain the final estimation.

We demonstrate that with transferred ground truth smoothness, random field can achieve state-of-the-art results in stereo matching and image denoising on standard benchmark dataset.

Thesis Supervisor: William T.Freeman
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

This thesis is not possible without the support, guidance and love from many people around me.

I would like to especially thank my thesis advisor Prof. William T. Freeman. Bill has led me during my master thesis with his deep insights and unrelenting patience. I would also like to thank my former joint advisor Dr. John Fisher III for his limitless ideas and general enthusiasm in my research. I'm grateful to both Bill and John for their generous support in the past two years.

I want to thank my mentor and collaborator Dr. Ce Liu, who helped me closely on this master thesis project from engineering skills to result presentation. More importantly, I am thankful to have such a loving person to guide me in life.

I am grateful to Dr. Dahua Lin and Jason Chang for their guidance during the collaboration on my previous projects. I am also privileged to share office with Hyun Sung Chang, Roger Grosse, Neal Wadhwa and Andrew Owens. We have shared so much conversation and laughter altogether. To the rest of my friends at MIT, thank you for your support and the happy memories we have shared.

This thesis is impossible without my parents love and sacrifice. My gratitude to them is beyond words.

Finally, to the God who creates, loves and guides. I was lost in life when I first came to MIT. But God led me to His family to learn His Words and to see that Jesus is the light, the love and the life. "All flesh is like grass and all its glory like the flower of grass. The grass withers, and the flower falls, but the word of the Lord remains forever." (1 Peter 1:24-25)

Glory be to God.

Contents

Abstract 3

Acknowledgments 4

1 Introduction 9

1.1 Related Work 9

1.1.1 Smoothness Energy Modeling 9

1.1.2 Nonparametric Modeling 10

1.2 Outline 10

2 Background Material 11

2.1 Markov Random Field 11

2.1.1 Construction 11

2.1.2 Inference 12

2.1.3 Learning 13

2.2 Patch Correspondence 13

2.2.1 Matching between Patches 13

2.2.2 Matching between Images 14

2.3 Low Level Vision Task 14

2.3.1 Image Denoising 14

2.3.1.1 Problem Setup 14

2.3.1.2 Computation Pipeline 14

2.3.2 Stereo Reconstruction 15

2.3.2.1 Problem Setup 15

2.3.2.2 Computation Pipeline 15

3 Smoothness-Transfer Random Field (ST-RF) 19

3.1 ST-RF Construction 20

3.1.1 Construct Data Energy 20

3.1.2 Transfer Smoothness Energy 20

3.1.2.1 Scene Retrieval 20

3.1.2.2 Patch Correspondence 21
3.1.2.3 Patch Selection 21
3.1.2.4 Ground Truth Smoothness Energy 21
3.2 ST-RF Inference 22
3.3 ST-RF Learning 23
3.3.1 Discrimitive Learning 23
3.3.2 Gradient Calculation 24

4 Experiment I: Stereo Matching **27**
4.1 Pipeline 27
4.2 ST-RF Construction 27
4.2.1 Construct Data Energy 28
4.2.1.1 Modified SGM 28
4.2.2 Transfer Smoothness Energy 30
4.2.2.1 Greedy Strategy 30
4.3 ST-RF Inference 31
4.4 ST-RF Learning 32
4.4.1 Choice of Loss Function 32
4.4.2 Comparison of Loss Function 32
4.4.3 Training on KITTI 33
4.5 Results on KITTI Dataset 34
4.5.1 Image Level Evaluation 34
4.5.1.1 Training Data 34
4.5.1.2 Test Data 35
4.5.2 Patch Level Evaluation 36
4.5.2.1 Define Matching Metric 36
4.5.2.2 Metric Justification 37
4.5.2.3 Metric Evaluation 37

5 Task II: Image Denoising **39**
5.1 Pipeline 39
5.2 ST-RF Settings 40
5.3 Results on Berkeley Segmentation Dataset 40
5.3.1 Matching Performance 40
5.3.2 Image Denoising Comparison 41

6 Conclusion **43**
6.1 Contribution 43
6.2 Future Work 43
6.2.1 Correspondence Representation 44
6.2.2 Ground Truth Smoothness Representation 44

Bibliography **45**

Chapter 1

Introduction

RANDOM field (RF) models have been widely used in computer vision, particularly in problems of low-level vision such as image restoration, stereo reconstruction, and optical flow estimation. In these applications, Markov random field (MRF) impose spatial regularization with prior knowledge on the types of natural images, depth maps, flow fields, etc. In the last decade, much attention is received on the improvements in learning and inference algorithms. As pointed out in Roth and Black [22], most MRF models are limited in three different areas in regard to smoothness energy modeling: (1) simple neighborhood structures, e.g. pairwise graphs; (2) hand-defined and hand-tuned potentials; (3) lack of spatial adaptivity. The first two problems are well eplored in [21, 35] and Roth and Black [22] addresses the third limitation by aligning potentnal functions with local structure of the image. However, such parametric approach cannot well take the advantage of existing similar examples in the training data.

Nonparametric modeling, on the other hand, is by nature richly adaptive through its data-driven mechanism and scales better with respect to the training data size. It has obtained state-of-art performance on semantic labeling [17], image denoising [14], super-resolution [27] and depth transfer [11].

In this thesis, we propose to build a RF model with nonparametric modeling for smoothness energy. To our knowledge, it is not only the first attempt to define and transfer "ground truth smoothness energy" from database for building adaptive RF model but also the first data-driven system for vision task with high accuracy requirement, like stereo matching.

■ 1.1 Related Work

■ 1.1.1 Smoothness Energy Modeling

Traditionally, smoothness energy in RF is constructed parametrically in terms of filter response or basis assignment.

For image prior related low level vision tasks (e.g. denoising, inpainting, deblurring, super-resolution and texture synthesis), FRAME [35] and Field-of-Expert [21] are popular filter-based smoothness model, and sparse coding [18], kernel regression [29] and mixture of Gaussian [36] are recently proposed as basis-based methods.

For stereo and optical flow, however, the smoothness energy to capture the pattern of motion or disparity is mostly confined to gradient filter (e.g. $\nabla_x = 0$) with a pairwise grid or bilateral/median filter with higher-order connectivity.

■ 1.1.2 Nonparametric Modeling

Starting with the seminal work of Freeman et al. [7], nonparametric modeling has been widely used within RF model in low level vision. Such data-driven methods implicitly or explicitly learn the mapping between source image patches and its desired estimation. Thus, only data energy is transferred for the RF model.

In order to find the neartest neighbor of a patch to build its non-parametric distribution, [7] has a special data structure to store large amount of patches for fast retrieval. However, such approach doesn't scale well with the size of training data and matches can be ambiguous due to the lack of context. Alternatively, we can use scene alignment algorithms like SIFT flow [16] which are both efficient for computing correspondence and effective for preserving context.

■ 1.2 Outline

This thesis is organized as follows.

Chapter 2 begins with a brief discussion of background material and previous work essential in understanding the concepts presented in later chapters. Topics in this chapter include the construction, inference and learning of MRF models, correspondence computation in data-driven literature, and introduction of low level vision tasks for our smoothness-transfer Random Field (ST-RF) model.

Chapter 3 introduces the computation pipeline for an ST-RF. We first define "ground truth smoothness energy" for each training example. Then, we use data-driven method to find correspondence from training examples. Lastly, we construct the smoothness energy of ST-RF by transferring the ground truth smoothness energy from matched region. We also explain in detail, the MRF inference and learning algorithms that we used for an ST-RF.

Chapter 4 and Chapter 5 show results of using our ST-RF model in stereo matching and image denoising. We first show the task-specific construction of ST-RF, describing the design choices for each module. Then compare our ST-RF result with other methods on the standard benchmark dataset. Moreover, we show the relationship between matching quality and the ST-RF performance.

Finally, we conclude the thesis in Chapter 6 with the summary of contribution and future research directions.

Chapter 2

Background Material

IN this chapter, we will briefly discuss some background material for readers that are not familiar with the topics. We here introduce readers to the two main components of our ST-RF model: MRF model as the backbone for our vision task modeling, and data-driven techniques to find matched patches to transfer smoothness energy. We also cover problem formulation and computation pipeline for two vision applications: stereo matching and image denoising.

■ 2.1 Markov Random Field

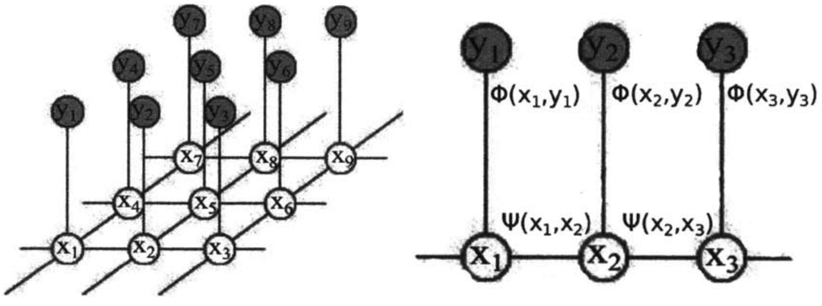
Markov Random Field (MRF) is an undirected graphical model and is widely used in image modeling to capture both the evidence from local regions and the relationship between neighboring local regions. Below, we go through basic elements for MRF model in order to better understand our ST-RF model as a variation. For detailed explanation and recent progress, we refer readers to Blake et.al. [2].

■ 2.1.1 Construction

As explained in [7], we denote the observed image data as y and the underlying scene to recover as x. In the MRF graphical model shown in Figure 2.1a, we have $P(x)$ as the prior distribution of the scene to estimate, and $P(y|x)$ as the likelihood to observe the image from such scene. Usually, these probabilities are defined with Boltzmann distribution as the nomalized expontenial of negative energy function. Conventionally, we denote the energy function for $P(x)$ as "smoothness energy" to capture the neighboring relationship among x: $E_{smoothness} = -\log(P(x)) = \sum_p \Psi(x_p)$, where x_p are groups of nodes and Ψ the **compatibility function** to prefer certain configuration of x_p. And that for $P(y|x)$ as "data energy", to model the generative process from scene to image data: $E_{data} = -\log(P(y|x)) = \sum_i \Phi(x_i)$, where Φ the **likelihood function** to evaluate the possibility to observe y_i from x_i.

In pair-wise 4-connected MRF model for example, we show the energy function defined on the edges of the graph in Figure 2.1b. The joint probability is defined as:

$$P(x,y) = P(x)P(y|x) = \exp\{-\sum_i \Phi(x_i, y_i) - \sum_{i,j} \Psi(x_i, x_j)\} \tag{2.1}$$



(a) (b)

Figure 2.1: (a) Graphical Model for pair-wise MRF with 4-connected neighborhood. Each x_i node describes a local patch of scene where y_i as the corresponding image observation. Edges in the graph indicate statistical dependencies between nodes. (b) Energy function defined on the graph

In order to capture longer range connectivity, high-order MRF models are proposed [21, 35].

2.1.2 Inference

In computer vision tasks, we aim to estimate the underlying scene from observed image data within the Bayesion framework by calculating the posterior probability of $P(x|y) \propto P(y|x)P(x)$. In general, there are two kinds of estimation we can make from the posterior probability. We can either find the mode of the probability function as the Maximum-A-Posteriori(MAP) estimator, or calculate the expectation of a certain loss function with respect to it as the Bayesian estimator. Due to the difficulty to integrate out the posterior probability, MAP estimator is commonly used in vision application. Below, we list the common methods for MRF optimization with x being either discrete or continuous.

a. Discrete Model

For the discrete formulation of stereo matching, where the disparity at each pixel is treated as a discrete variable, the state space grows exponentially with the range of the connectivity. In inference algorithms for discrete pairwise MRF, three major types can be seen: Polyhedral and Combinatorial Methods, Message Passing Methods, and Max-Flow and Move-Making Methods. The detailed description and comparasion of these inference algorithms can be found in [10]

However, these algorithms became cumbersome when it comes to high-order-MRF with big potential clique. Thus, tree approximation of the connectivity graph [26, 31] are proposed as an alternative.

b. Continuous Model

Compared to discrete models, continuous model has the drawback of having a continuos parametric form of data energy, which is usually hard to capture the non-convexity. However, it can well handle high-order smoothness energy since during optimization it only considers the local gradient at current estimation instead of searching over exponentially large joint state space.

■ 2.1.3 Learning

Besides the inference phase above, MRF need a learning phase for the paramter in the potential function. For example, the most common compatibility function in smoothness energy is the pair-wise potential $\Psi(x_p) = \Psi(x_i, x_j) = w_{ij}\rho(x_i - x_j)$, where ρ is a cost function and w_{ij} weight of the constraint of x_i, x_j being alike. For a homogenous MRF, w_{ij} is set to be the same value, which need to be learned during the MRF learning phase to balance the influence between smoothness energy and data energy.

For learning MRF, two main approaches are seen: Sampling-based methods [25, 34] and discriminative learning methods [15, 23]. The sampling-based methods utilize efficient sampling methods to learn image prior by fitting the statistics of ground truth. These methods are well founded on statistical theories and exhibit good performance in learning natural image prior in a general way. The discriminative learning methods learn the model parameters by constructing a loss function [13] between the inferred estimation from MRF model and the target.

■ 2.2 Patch Correspondence

Given a database of stereo images with ground truth depth map, we want to find nonparametric depth prior for test stereo images. In the following section, we briefly overview data driven matching methods and a recent application that creates such depth prior by finding correspondence between test image and the database.

■ 2.2.1 Matching between Patches

Early works build a fast data structrue (e.g. kd-tree) for training patches in order to find the nearest neighbor for each test patch efficiently. However, such method throws away the context information of images that training patches are extracted from. Good match for one test patch alone may not be misleading while good matches for mutiple neighboring patches from one test image to those from one training image may be more compelling.

■ 2.2.2 Matching between Images

Inspired by optical flow that is able to produce pixel level correspondence between two images, [16] replaces raw pixel value by SIFT descriptors for matching cost and follows the standard computational framework of optical flow. As shown in many applications in [16], the use of SIFT features allows robust matching across different scene/object appearances and the discontinuity-preserving spatial model allows matching of objects located at different parts of the scene.

Typically, there are three steps in most data-driven applications. First, retrieve a small set of similar images from the large corpus of training images. This can be done, for example,by comparing global image descriptors such as GIST [19]. Secondly, compute dense pixel mappings between images in the retrieval set and the target image. Some methods enforce smooth mappings [16], while others allow arbitrary mappings [32]. Lastly, transfer desired scene property, e.g. semantic label, local appearance, from the images in the retrieval set to the target image via the dense pixel mappings. To resolve the ambiguity and mistakes during transfer, a MRF is often used to aggregate local evidence from transfer and enforce smoothness constraints.

■ 2.3 Low Level Vision Task

To show the power of better capturing smoothness prior with our new proposed T-MRF model, we test it on two popular low level vision task: image denoising and stereo reconstruction. Below, we give brief overview of these vision tasks.

■ 2.3.1 Image Denoising

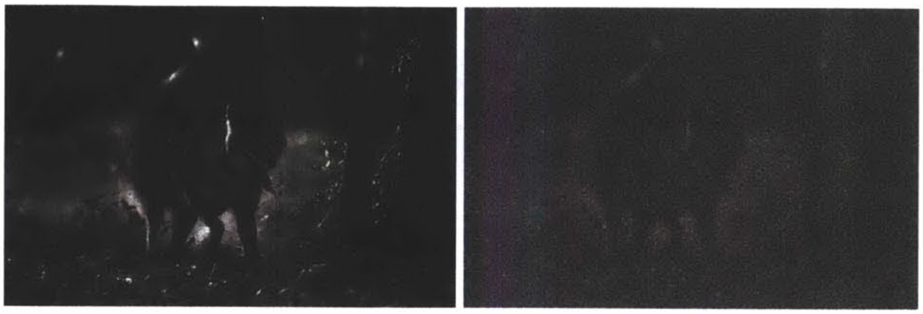
■ 2.3.1.1 Problem Setup

Natural image denoising is defined as the problem of estimating a clean version of a noise corrupted image (Figure 2.2b), given a priori knowledge that the original unknown signal is a natural image (Figure 2.2a). The main idea in this setting is that image denoising can be performed using not only the noisy image itself but rather using a suitable prior on natural image statistics.

■ 2.3.1.2 Computation Pipeline

Recently, impressive results have been obtained with non-parametric techniques such as non-local means [4] or BM3D [5], and sparse representation methods such as KSVD [6].

These methods share the same computation framework: first extract overlapping patches from the noisy image, then denoise each patch either individually or collaboratively, and lastly take the average or median of the pixel value from corresponding patches.



(a) Original Image (b) Add Gaussian Noisy

Figure 2.2: The generative process for image denoising (a) original image (b) corrupted by Gaussian noise

■ 2.3.2 Stereo Reconstruction

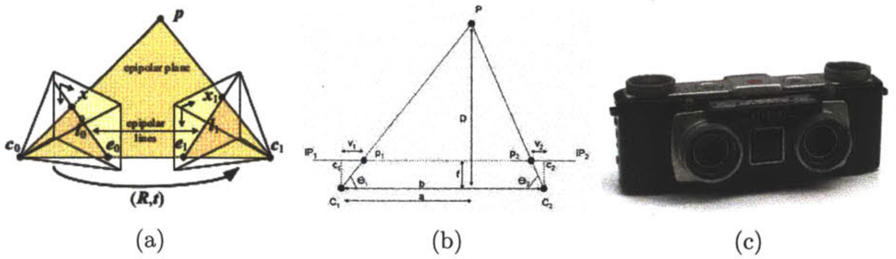
■ 2.3.2.1 Problem Setup

Calculating the distance of various points, or any other primitive, in a scene relative to the position of a camera is one of the important tasks of a computer vision system. Epipolar geometry provides tools in order to solve the stereo correspondence problem, i.e. to recognize the same feature in both images. If no rectification is performed, the matching procedure involves searching within two-dimensional regions of the target image, as shown in Figure 1.8(b). However, this matching can be done as a one-dimensional search if accurately rectified stereo pairs are assumed in which horizontal scan lines reside on the same epipolar line, as shown in Figure 1.8(a). A point P1 in one image plane may have arisen from any of points in the line C1P1, and may appear in the alternate image plane at any point on the epipolar line E2 (Jain et al. 1995). Thus, the search is theoretically reduced within a scan line, since corresponding pair points reside on the same epipolar line. The difference of the horizontal coordinates of these points is the disparity value. The disparity map consists of all the disparity values of the image. Fig 2.3a shows a generic formulation of two-view stereo problem using epipolar geometry where details can be found in [28]. In practice, stereo images are often taken by two side-by-side synchronized cameras shown in Fig 2.3c. Consequently, the relationship between 3D depth and the disparity on two image planes can be simplified as the function of focal length and the displacement between two cameras illustrated in Fig 2.3b.

■ 2.3.2.2 Computation Pipeline

a. Matching Cost Computation

In order to find for each pixel in the left image the corresponding pixel in the right one, we need to measure the similarity of these pixels. The pixel to be matched without any



(a) (b) (c)

Figure 2.3: (a) Illustration from [28] for the corresponding set of epipolar lines and their epipolar plane (b) simplified geometry relationship between depth and disparity on two image planes when there is only translation between two camera poses (c) image of a stereo camera from http://commons.wikimedia.org

ambiguity should have the lowest matching cost distinctly from its surrounding pixels. As described in [24], we first compute such cost for each pixel with all possible corresponding pixels. The feature to match for each pixel can be its intensity, gradient or patch-based features like census-transform and SIFT. The most common matching metrics are absolute differences (AD) and the squared differences (SD), p-norm of vectors with p=1,2. To limit the influence of noise, robust measures like truncated quadratics and contaminated Gaussians have been proposed. To be insensitive to image sampling, Birchfield and Tomasi compares pixel values in the reference image against a linearly interpolated function of the other image instead of the integral pixel positions.

b. Matching Cost Aggregation

To be robust against noisy observation, we aggregate within a small support . Aggregation with a fixed support region can be performed using 2D or 3D convolution:

$$C(x, y, d) = w(x, y, d)C_0(x, y, d) \tag{2.2}$$

where $C_0(x, y, d)$ is the initial cost volume computed above, $w(x, y, d)$ the weight and $C(x, y, d)$ the agregated cost volume. The weight can come from box filter, gaussian filter, bilateral filter and guided image filter.

c. Spatial Regularization

In addition to the local evidence for each pixel, we need to add spatial regularization to obtain a globally consistent stereo estimation. Here, we show the formulation of different stereo matching methods using Markov Random Field (MRF) Model. In general, we want to minimize the energy function E, the total matching cost E_{data} with spatial regularization E_{smooth}.

$$E(d) = E_{data}(d) + \lambda E_{smooth}(d) \tag{2.3}$$

$$E_{data}(d) = \sum_{(x,y)} C(x, y, d(x, y)) \tag{2.4}$$

Local methods set $E_{smooth} = 0$ and the optimal disparity d can be calculated for each pixel indenpendently, known as winner-take-all (WTA) optimization. For global methods,

$$E_{smooth}(d) = \sum_{(x,y)} \rho(\nabla^p d(x,y)) \tag{2.5}$$

$$E_{smooth}(d) = \sum_{(x,y)} \rho(w(x',y',d')d(x',y') - d(x,y)) \tag{2.6}$$

$$\tag{2.7}$$

E_{smooth} wants the dispairty at each pixel to be similar to that predicted by the neighouring pixels. Similar to the metric in cost computation, we use robust function ρ to capture such difference. In terms of the prediction, it is usually made by simple gradient or regression. For first-order MRF model, where only pair-wise spatial regularization is imposed, Belief Propagation is used. Tree approximation of the MRF model is used for efficient Dynamic Programming. In general, the optimization of high-order MRF model is hard.

d.Disparity Refinement

After obtaining the disparity estimation from the model above, we can further improve the result with post-procesing steps. To refine an initial disparity estimation, we first remove unconfident or invalid estimation and then fill in the holes. The confidence of the estimation is defined as the ratio between the top two lowest cost and the invalid estimation is detected using cross-checking. Usually, holes are filled by surface fitting or by distributing neighboring disparity estimates. A median filter can be applied to clean up spurious mismatches,

Chapter 3

Smoothness-Transfer Random Field (ST-RF)

IN this chapter, we describe in details the computation pipeline of our ST-RF model. Similar to other pipelines with MRF modeling, we need to first construct the MRF model with data energy and smoothness energy. Then we learn the parameters of the MRF model to better tune the model to fit the training data. Last, we do inference on the MRF model for the test data to obtain the estimation for underlying scenes. The novelty of our ST-RF lies in the smoothness energy construction, Sec. 3.1.2, where we use data-driven techniques to find correspondence from database and to transfer the compatibility function from the matched region.

The summary of the pipeline is shown in Algo 1.

```
input  : Test image data ($y_{test}$), Training image data and scene data ($\vec{y}_{train}, \vec{x}_{train}$)
output: Test scene ($x_{test}$)
a) Learn MRF parameter:                                    Sec.3.3
b) Construct Data Driven MRF:                              Sec.3.1
      Construct Data Energy:                               Sec.3.1.1
      Transfer Smoothness Energy:                          Sec.3.1.2
c) Infer MRF variable:                                     Sec.3.2
```

Algorithm 1: Pipeline of Smoothness Energy Transfer Framework

3.1 ST-RF Construction

Below, we show the comparison of the energy function of the standard and the proposed energy function.

$$E_0(x;y,\theta) = \underbrace{\sum_i \rho(x_i - y_i)}_{\text{Data Energy}} + \underbrace{\sum_p \rho(\phi(x_p))}_{\text{Smoothness Energy}} \tag{3.1}$$

$$E_1(x;y,\theta) = \underbrace{\sum_i \rho(x_i - y_i)}_{\text{Data Energy}} + \underbrace{\sum_p \rho(\phi(x_p) - \phi(t_p))}_{\text{Transferred Smoothness Energy}} \tag{3.2}$$

where x denotes the current state of the MRF, ρ the penalty function, ϕ the potential function to capture the statistics of patches, and t_p the matched patches from the training.

3.1.1 Construct Data Energy

For image denoising, the data energy is defined straight-forwardly from the generative model. It is the probability of the Gaussian distribution $N(Y - X; 0, \sigma^2)$ where Y is the noisy observation, X the estimated pixel value, and σ the known noise level. For stereo reconstruction, however, the data energy need to be approximated with a continuous function. Various relaxation methods and coarse-to-fine scheme are used to address such issue. We take the approach implemented in [30], where the continuos function is a Gaussian distribution centered at the initial estimation. We'll discuss the detail in Sec 4.2.1.

3.1.2 Transfer Smoothness Energy

For each patch p from the test image, we need to first find matched patches t_p in the database, and then transfer the compatibility function from them.

3.1.2.1 Scene Retrieval

Like label transfer system, we need to first find a small set of images for patch level correspondence. Here, there are two variation of the method:

1. Distance metric: we need to quantify the distance between matched images. We can calculate the difference between two images in either intensity space (I) or SIFT feature space (S). Moreover, for stereo, we can also compute the distance between the initial disparity estimation of test image and that of training images with DC component removed (D) Also, we can design new feature by combining both appearance and disparity. (not implemented)

2. Number of neighbor: we can use a simple K-NN approach or a generalized $< K, \epsilon >$ approach to adaptively choose the number of neighbors, where ϵ is the threshold

parameter to remove neighbors with distance further than $(1+\epsilon)$ times the smallest distance. .

3.1.2.2 Patch Correspondence

Traditionally, [7] builds fast querying data structure for all patches, which is still expensive considering the size of training patches. Recently, image alignment through SIFT feature matching (e.g. SIFT flow) and approximate Nearest-Neighbor-Field method (e.g. PatchMatch) have shown great promise for fast patch correspondence while preserving image context. Here, we consider two popular methods: SIFT flow [16] and PatchMatch [1].

3.1.2.3 Patch Selection

Given K_i matches for each patch i in the query image, we can either build a mixture model as for image prior, or choose a simple estimator from these matches. Here, we only consider three intuitive factors for patch selection, where ϵ is a threshold parameter (different for each case):

1. number of matches: (Count> ϵK) if only few matches are found from neighboring images, then the patch is less confident about its disparity matching distance

2. standard deviation of the matched smoothness: (Std< ϵ) if the matched smootheness for one patch agree with each other, then this patch has strong confidence to have good matches

3. deviation from initial estimation: (SM-L1< ϵ) since the disparity estimation from SGBM is still valuable, we can reject outlier matches through checking its difference from the transferred estimation

3.1.2.4 Ground Truth Smoothness Energy

Ever since the advent of the MRF model, people have been devising various forms of smoothness energy to better capture the correlation among nodes in the graph. However, the answer to the question, "What is the ground truth smoothness energy", is still not clear. We here explicitly define the ground truth smoothness energy as a function form that leads to ground truth estimation during inference.

Take stereo matching for example, one trivial ground truth smoothness energy can be defined as $\rho(x_p(center) - x_p(j) - (g_p(center) - g_p(j)))$, where $x_p(center), x_p(j)$ are nodes at center and j^{th} position in the patch p in RF, $g_p(center), g_p(j)$ the disparity value at center and j^{th} position in the ground truth disparity map. With reasonable data energy, we can get the infered disparity map same as the ground truth.

For the convenience of transfer, we define ground truth smoothness for each patch of the image. The final form of the ground truth smoothness energy is:

$$E(x;y,\theta) = \underbrace{\sum_i \rho(x_i - y_i)}_{\text{Data Energy}} + \underbrace{\sum_p \sum_j \theta_j \rho(x_p(center) - x_p(j) - (g_p(center) - g_p(j))}_{\text{Transferred Smoothness Energy}} \tag{3.3}$$

3.2 ST-RF Inference

As discussed in Sec.2.1.2, the discrete variable x has the advantage of capturing the non-linearity of the energy while the continous version of the variable can incorporate nonlocal smoothness more efficiently. We use a continuous MRF model to capture long range correlation within a matched patch. Our inference problem is the same as optical flow. Specifically for our nonlocal smoothness energy, we follow the inference technique in [12]. The outer loop is a standard coarse-to-fine scheme, in which the disparity is estimated at increasing resolutions. Below, we focus on disparity estimation during a single step of this multiresolution refinement. At a given resolution, we estimate the disparity D between the warped images iteratively, initializing it with the previous estimate D0. At step k + 1, for k $\geq$ 0, we express the disparity as $x^{k+1} = x^k + \Delta x^k$. Our goal is to find a displacement Δx^k that satisfies

$$\nabla_{\Delta x^k} E(x^k + \Delta x^k) = 0 \tag{3.4}$$

We linearize the gradient around x^k, reducing it to a linear system, which can be solved efficiently using the conjugate gradient algorithm with Jacobi preconditioning.

To linearize the gradient ∇E_N, we begin with a simple variable substitution $\rho_N(x) = \phi_N(x^2)$. This yields the following expression for the components of the gradient

$$\frac{\partial}{\partial \Delta x_i^k} E_N(x^k + \Delta x^k) = 2 \sum_{j \neq i} w_{ij}(x_i^k + \Delta x_i^k - x_j^k - \Delta x_j^k) \tag{3.5}$$

$$\psi_N'((x_i^k + \Delta x_i^k - x_j^k - \Delta x_j^k)^2) \tag{3.6}$$

To linearize the gradient with general penalty functions $\rho_N(x)$, we follow Papenberg et al. [11] and use a fixed point iteration to compute Δu^k. We initialize this inner iteration with $\Delta u^k = 0$. At each step $l + 1$, for $l \geq 0$, we compute a displacement vector $\Delta u^{k,l+1}$ that satisfies

$$\nabla_{\Delta x^{k,l+1}} E(x^k + \Delta x^{k,l}) = 0 \tag{3.7}$$

where

$$\frac{\partial}{\partial \Delta x_i^{k,l+1}} E_N(x^k + \Delta x^{k,l+1}) = 2 \sum_{j \neq i} w_{ij}(x_i^k + \Delta x_i^{k,l+1} - x_j^k - \Delta x_j^{k,l+1}) \tag{3.8}$$

$$\psi_N'((x_i^k + \Delta x_i^{k,l} - x_j^k - \Delta x_j^{k,l})^2) \tag{3.9}$$

This assumes that the derivative terms $\psi'_N(.)$ are approximately constant for small changes in the flow displacement [11]. The terms $\psi'_N(.)$ in Equation 5 are constant with respect to $\Delta u^{k,l+1}$, thus Equation 4 is a linear system. Specically, we can express Equation 4 as

$$(A + B)\Delta u^{k,l+1} = w_A + w_B \tag{3.10}$$

where $B\Delta u^{k,l+1} - w_B$ is the sum of the linearized gradients of E_D and E_S, and A and w_A are defined as follows:

$$A_{ij} = -w_{ij}\phi'_N((x_i^k + \Delta x_i^{k,l} - x_j^k - \Delta x_j^{k,l})^2) \;\; for\ i \neq j \tag{3.11}$$

$$A_{ii} = \sum_{j \neq i} w_{ij}\phi'_N((x_i^k + \Delta x_i^{k,l} - x_j^k - \Delta x_j^{k,l})^2) \;\; for\ i \neq j \tag{3.12}$$

$$w_A = -Au^k \tag{3.13}$$

3.3 ST-RF Learning

There are two approaches in MRF learning: generative and discrimitive. In generative learning, it is often hard to compute the partition function. We here take the discrimitive approach and adapt the image prior model in [23] for our ST-RF model.

3.3.1 Discrimitive Learning

We rewrite the energy function of ST-RF model with nonlocal smoothness defined in Equation 3.3 with the following notation: x_m as the m^{th} variable, θ_{ij} as the weight vector on the edge between i^{th} and j^{th} variable, y_m the m^{th} observation, and $\Delta x_{gt}(i,j)$ as the disparity difference between i^{th} and j^{th} variable.

$$E(x; y, \theta) = \sum_i \rho(x_i - y_i) + \sum_i \sum_j \theta_{ij}\rho(x_i - x_j - \Delta x_g t(i,j)) \tag{3.14}$$

$$x^*(\theta) = \arg\min_x E(x; y, \theta) \tag{3.15}$$

,where x^* is the MAP estimation of the MRF model.

For discrimitive learning, we want to minimize the distance between the MAP estimation and the ground truth x_{gt}, where the distance metric is based on the evaluation methods. Stereo, for example, needs thresholded L_0-norm while image denoising needs L_2-norm. For simplicity, we here demonstrate the derivation of our learning algorithm with L_2-norm. Thus, the objective function and the gradient w.r.t. parameter θ are:

$$L(x^*(\theta), x_{gt}) = (x^*(\theta), x_{gt})^2 \tag{3.16}$$

$$\frac{\partial L(x^*(\theta), x_{gt})}{\partial \theta} = 2(x^*(\theta) - x_{gt})\frac{\partial x^*(\theta)}{\partial \theta} \tag{3.17}$$

Although it is hard to write down the analytical form of x^*, we can make use of the implicit derivative trick described in [23]. We first define the auxilary function $g(x^*, \theta)$ and calculate its total derivative.

$$g(x, \theta) = \frac{\partial E(x; y, \theta)}{\partial x} \tag{3.18}$$

$$\frac{\partial E(x; y, \theta)}{\partial x}|_{x^*(\theta)} = g(x^*, \theta) = 0 \tag{3.19}$$

$$\frac{dg(x^*, \theta)}{d\theta} = \frac{\partial g(x^*, \theta)}{\partial x^*}\frac{\partial x^*}{\partial \theta} + \frac{\partial g(x^*, \theta)}{\partial \theta} = 0 \tag{3.20}$$

```
input : MRF parameter $\theta^0$, Observation $y$
output: $\theta^*$ s.t. MAP estimation $x^*(\theta^*)$ is closest to the ground truth $x_{gt}$ in L2
for $t \leftarrow 1$ to $T$ do
    MRF inference: Compute $x^*(\theta^{t-1})$  3.2
    MRF learning: Compute $\theta^t$           3.3
end
```

Algorithm 2: Pipeline for discrimitive MRF learning

3.3.2 Gradient Calculation

Now with the implicit derivative trick, we get the desired $\frac{\partial x^*(\theta)}{\partial \theta}$ as

$$\frac{\partial x^*(\theta)}{\partial \theta} = -\left(\frac{\partial g(x^*, \theta)}{\partial x^*}\right)^{-1}\frac{\partial g(x^*, \theta)}{\partial \theta} \tag{3.21}$$

we explicitly write out each part of Eq. as

$$\begin{aligned} g_m(x^*, \theta) &= \frac{\partial E(x; y, \theta)}{\partial x_m}|_{x^*(\theta)} \\ &= \rho'(x^*_m - y_m) + \sum_j \sum_k \theta_j \rho'(x^*_m - x^*(j, k) - \Delta x_{gt}(m, jk)) \end{aligned} \tag{3.22}$$

$$\frac{\partial g_m(x^*, \theta)}{\partial x^*_m} = \rho''(x^*_m - y_m) + \sum_j \sum_k \theta_j \rho''(x^*_m - x^*_{p_{jk}} - \Delta x_{gt}(m, jk)) \tag{3.23}$$

$$\frac{\partial g_m(x^*, \theta)}{\partial x^*_n} = -\sum_j \sum_k \theta_j \rho''(x^*_m - x^*_{jk} - \Delta x_{gt}(m, jk)) \tag{3.24}$$

$$\frac{\partial g_m(x^*, \theta)}{\partial \theta_k} = \sum_j \sum_k \rho'(x^*_m - x^*_{jk} - \Delta x_{gt}(m, jk)) \tag{3.25}$$

Thus, the final update θ is as following, where s is the step size for the gradient descend to find the minimum energy.

$$\begin{aligned}\theta^t &= \theta^{t-1} - s\frac{\partial L(x^*(\theta^{t-1}), t)}{\partial \theta^{t-1}} \\ &= \theta^{t-1} - 2s(x^*(\theta^{t-1}) - t)\frac{\partial x^*(\theta^{t-1})}{\partial \theta^{t-1}} \\ &= \theta^{t-1} + 2s(x^*(\theta^{t-1}) - t)\left(\frac{\partial g(x^*, \theta^{t-1})}{\partial x^*}\right)^{-1} \frac{\partial g(x^*, \theta^{t-1})}{\partial \theta^{t-1}} \end{aligned} \tag{3.26}$$

Chapter 4

Experiment I: Stereo Matching

Traditionally, stereo matching dataset contains few training and test examples which makes it hard to apply our framework. However, thanks to the recent KITTI dataset which provides around 200 training and test examples, we can find matches to transfer the smoothness information to demonstrate the usefulness of matching approach. Below, we first adapt our general SM-MRF framework to stereo matching task. Then we make use KITTI training data to make the design decision for each module during MRF construction and to test our MRF inference and learning algorithm. For evaluation, we show comparison result not only on overall performance but also the patch level correlation between estimation error and the matching quality.

■ 4.1 Pipeline

In Figure 4.1, we visualize the pipeline of ST-RF for stereo matching. Since these images are approximately aligned, SIFT flow algorithm doesn't make much change during scene alignment.

We define the ground truth smoothness energy as the difference between disparity value at each position and that in the center:

$$\phi(x_p(i)) = x_p(i) - x_p(center) \tag{4.1}$$

■ 4.2 ST-RF Construction

To build data energy, we need the initialization result from modified SGBM, which is used in [30] to achieve state-of-the-art stereo matching result on KITTI. To transfer smoothness energy, we need to finalize the design of our data-driven module since it has many different choices in how to find and select matched patches. In Sec. 4.2.2, we used a greedy search approach to make these decisions based on the disparity error of the estimation with leave-one-out cross validation method.

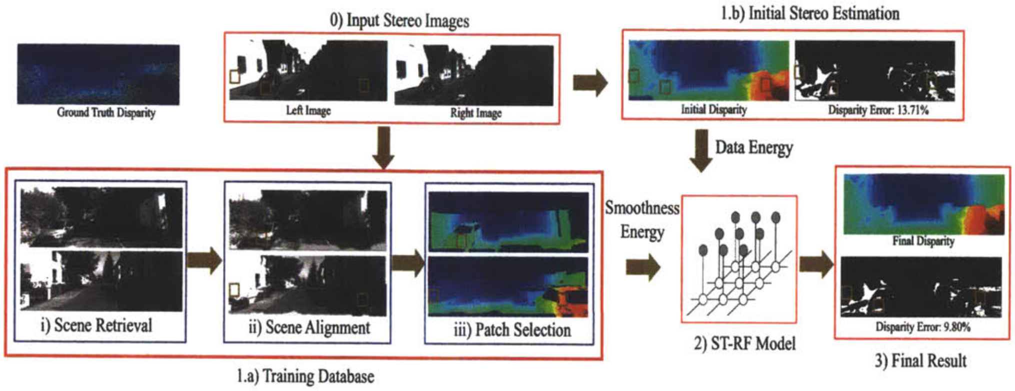


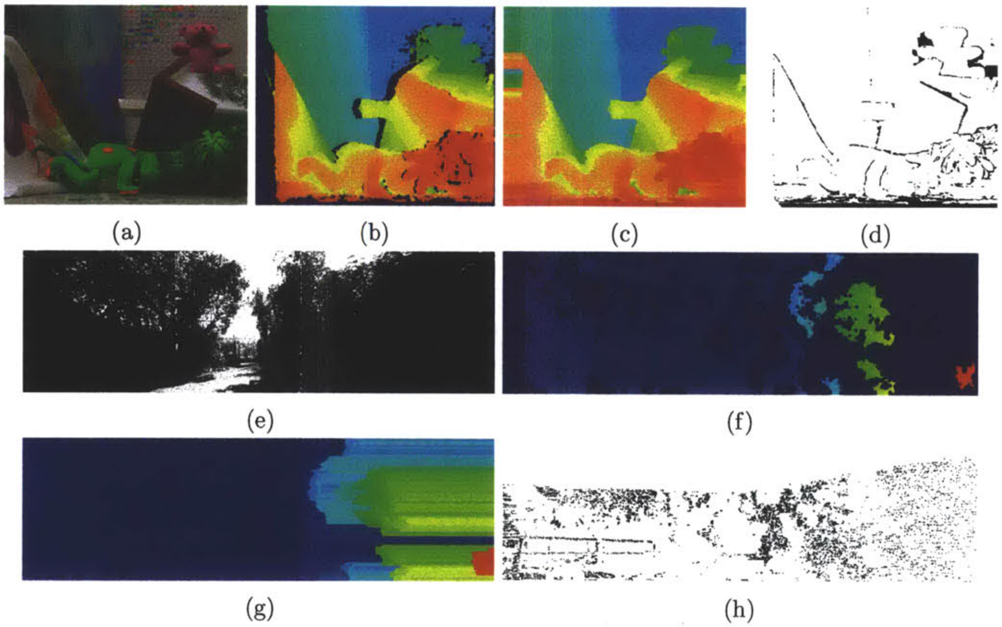
Figure 4.1: ST-RF Pipeline for Stereo Matching

■ 4.2.1 Construct Data Energy

In the discrete formulation of stereo model, the data energy, aggregated matching cost explained in Sec 2.3.2.2, is calculated at each possible state of the variable (distinct disparity values). But for continouse MRF model, we need to make a functional approximation of this discrete data energy function. Recently, [33] implements the adaptive convex relaxation and [20] applies the coarse to fine scheme to approximate the original data energy. However, these approaches suffer from large disparity value and highly non-convex data energy distribution. Inspired by [30], we take an alternative route which treats the initial disparity estimation from Semi-Global-Matching (SGM) stereo algorithm [9] as the i.i.d. observation for the ground truth disparity.

■ 4.2.1.1 Modified SGM

Following [30], we use the opencv implementation [3] of SGM. Usually, SGM uses pixel intensity as the feature to calculate the matching cost. It works well for the Middlebury dataset which is obtained in a controlled environment (Figure 4.2a- 4.2d). However, for the KITTI dataset, some image have large regions of low contrast as shown in Figure 4.2e. The original SGM won't produce estimation for these regions due to matching uncertainty and the post-processing method naively treats them as occluded regions leading to the result in Figure 4.2g. Instead of doing sophisticated preprocessing, [30] used the combination of gradient feature and census transformation feature to calculate the matching cost. We modified upon opencv SGM code and reproduced similar result to [30], 6.65% disparity error rate.



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

(e) (f)

(g) (h)

Figure 4.2: Stereo matching result from SGM [9] on examples from Middlebury and KITTI dataset. (a,e) show the original left image (b,f) are the output from SGM after noise removal (c,g) are the result using standard interpolation method to fill in the unestimated region (d,h) show the error map and we can see that error in (d) is mostly around boundary while error in (g) has large regions of wrong estimation due to the simple interpolation method

4.2.2 Transfer Smoothness Energy

4.2.2.1 Greedy Strategy

Considering the system design discussed in 3.1.2, we need to select parameters for the following modules on training data: (a) image feature and nearest neighbor set for scene retrival, (b) patch correspondence algorithm, and (c) patch smoothness tranfer methods.

We start from a reasonable initial setting (NS+5-NN, SIFT Flow, SM-L1) and greedily select the best setting for each module in the following order: (c)→(b)→(a).

(c) Patch Smoothness Transfer As described in Sec. 3.1.2.3, we test three different methods to transfer patch smoothness under three different parameter settings.

Count		Std		SM-L1	
ϵ	error	ϵ	error	ϵ	error
0.25	6.62	0.5	6.28	0.5	5.87
0.5	6.41	1	6.33	1	**5.76**
0.75	6.41	1.5	6.29	1.5	5.79

Table 4.1: Error rate on training data with different NN strategies in terms of number of images and descriptor to calculate the matching cost.

(b) Patch Correspondence From observation, we found that correspondence found by SIFT Flow can better preserve context information due to the smoothness energy. PatchMatch, on the other hand, may match low textured patches from the wall to that on the road as long as they have simliar appearance. This is undesirable, since patches from the wall has horizontal smoothness and that on the road has vertical smoothness. Our experimental results confirmed such intuition.

SIFT Flow	PatchMatch
5.76	6.23

Table 4.2: Error rate on training data with different patch correspondence algorithm

(a) Scene Retrieval we compare nearest neighbor strategies with differnet image retrieval features and numbers.

$< K, \epsilon >$-NN/Descriptor	SIFT	GIST	NS
5-NN	5.81	5.85	5.76
11-NN	5.83	5.86	**5.72**
<5,1.05>-NN	5.82	6.37	6.03
<11,1.05>-NN	5.82	6.37	6.01

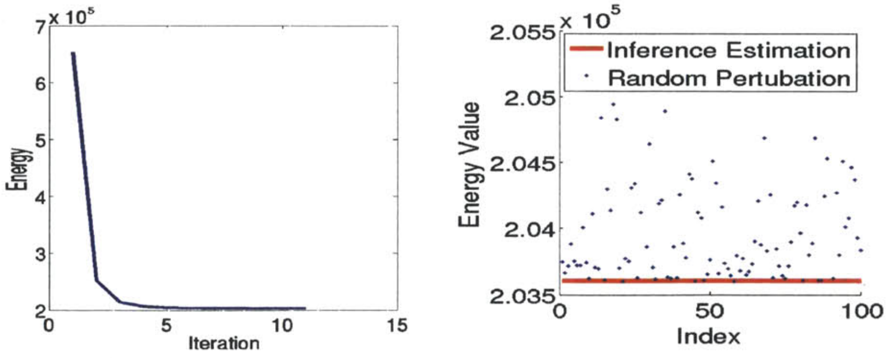
Table 4.3: Error rate on training data with different NN strategy in terms of number of images and descriptor to calculate matching cost.

■ 4.3 ST-RF Inference

For efficient inference, we not only adopt the coarse-to-fine optimization scheme but also update values for certain variables instead of all of them. We first lower SGBM's uncertainty threshold for assigning bad estimation, which leads to around 1% error rate for regions with good estimation, accounting for around 70% of the image. Thus, we can fix the values of variables which are classified as good estimation and focus on re-estimating the values for the rest variables.

To show the correctness of our implementation of the gradient descend MRF inference algorithm described in Sec 3.2, we plot the during each iteration in Figure 4.3a. As expected, we can see that the energy value does go down monotonically at each iteration.

Since the energy function we are using is convex, we can check the optimality of the inference by checking the energy value at the small perturbation of the estimation. of with random added. We can see from Figure 4.3b that the random pertubation does not further lower the energy, suggesting that the inference result does correspond to the optima of the model energy.



(a) Checking Monoticity

(b) Checking Optimality

Figure 4.3: Checking our MRF inference algorithm: (a) energy value of ST-RF model at each iteration (b) energy value of random perturbation of the inference estimation.

4.4 ST-RF Learning

4.4.1 Choice of Loss Function

As described in Sec 3.3, we can choose various loss function L for our discriminative learning. For evaluation of our disparity estimation, we use 0-1 loss $1 - \delta(|err| < 3)$, where errors within the threshold is counted as 0 and 1 otherwise. However, such loss function is not differentiable and we here choose to use to two approximations: quadratic loss $(err/3)^2$ and exponential loss $e^{-(err/2)^2}$. We visualize these three loss functions in Figure 4.4.

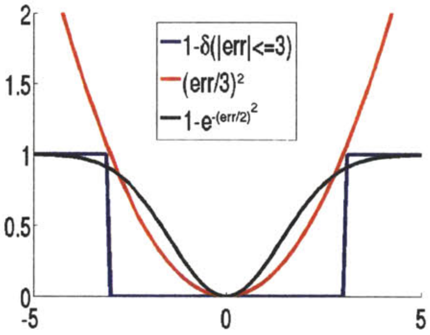
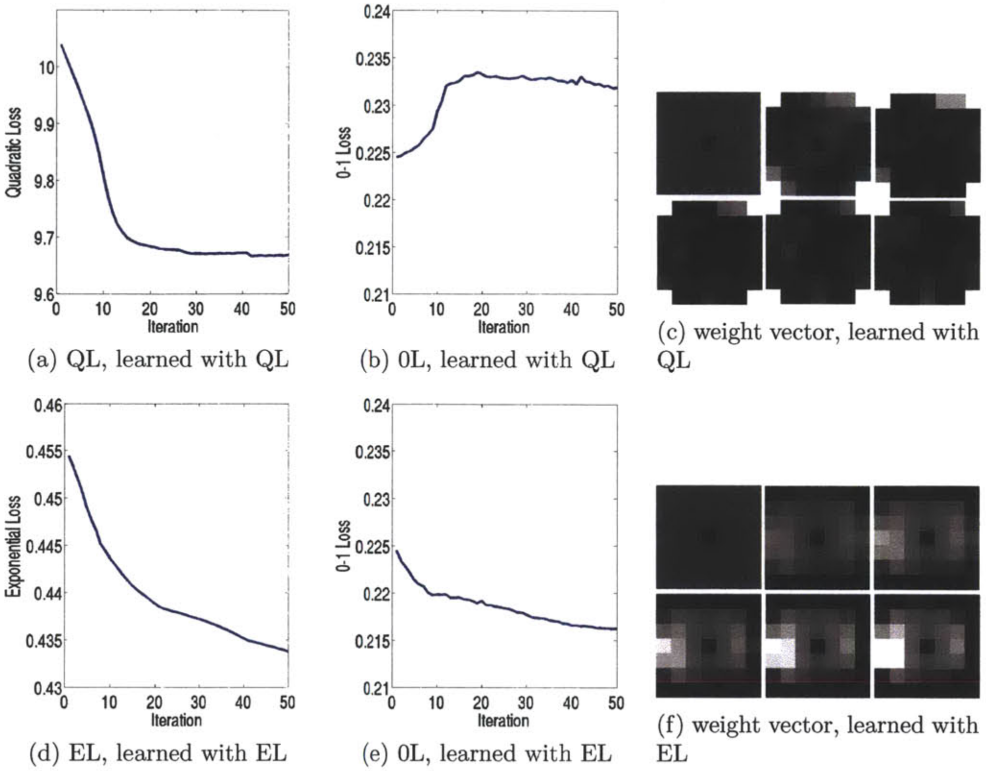


Figure 4.4: Visualization of the 0-1 loss(0L) function for disparity evaluation and two approximated loss functions , quadratic loss(QL) and exponential loss(EL), that are easier for computation.

4.4.2 Comparison of Loss Function

In order to determine which loss function to use between the two approximation, we perform the following experiment on the training data from KITTI. We pick the first 10 examples with ground truth disparity and run 50 iterations of our MRF learning algorithm for each loss function described in Sec 3.3. For quadratic loss function for example, we show in Figure 4.5a the values of the loss function during each iteration, in Figure 4.5b the value of 0-1 loss function during each iteration, and in Figure 4.5c the visualization of the weight within the patch.

We can see that our learning algorithm converges within a reasonable number of iterations for both loss function. The quadratic loss function is a bad approximation since the learned weight vector leads to higher 0-1 loss. The weight vector learned with exponential loss function, on the other hand, has lower 0-1 loss.



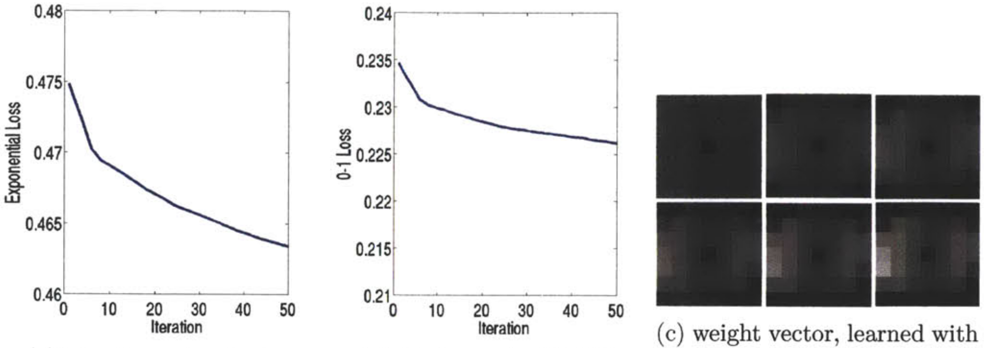
(a) QL, learned with QL (b) 0L, learned with QL (c) weight vector, learned with QL

(d) EL, learned with EL (e) 0L, learned with EL (f) weight vector, learned with EL

Figure 4.5: MRF learning result. (a,d) Evaluation of approximated loss function during learning with itself; (b,e) Evaluation of 0-1 loss function learned with approximated loss function; (c,f) weight vector learned by approximated loss function at iteration 0,10,20,30,40,50.

■ 4.4.3 Training on KITTI

We test our data driven MRF model with learned parameter on KITTI dataset. To see the improvement of the training with respect to the number of the training example, we train the parameter with {10,97} examples separately. In Figure 4.6, we visualize the learned weight vector which is similar in to that trained with 10 examples in Figure 4.5f. and in Table 4.4, we find that more training examples only improve the final performance slightly.



(a) QL, learned with QL (b) 0L, learned with QL (c) weight vector, learned with QL

Figure 4.6: MRF learning result on all the training images. (a) Evaluation of approximated loss function during learning with itself; (b) Evaluation of 0-1 loss function learned with approximated loss function; (c) weight vector learned by approximated loss function at iteration 0,10,20,30,40,50.

Number of training	97	10
Error Rate	**5.55**	5.61

Table 4.4: Error rate of FoE-type MRF on KITTI training data.

4.5 Results on KITTI Dataset

In Sec. 4.5.1, we show results on the recent KITTI benchmark dataset with comparison to MRF models with various smoothness modeling. To deeper analyze the performance of our framework, we need to show the correlation between disparity estimation error and the disparity matching error. In Sec. 4.5.2, we first design and justify the disparity matching metric and then show such desired relation.

4.5.1 Image Level Evaluation

For image level evaluation, we use the error rate for the whole disparity map with 3 pix threshold, which is standard on the KITTI benchmark dataset. Below, we compare our results on both the training and test data.

4.5.1.1 Training Data

Due to the constraint of evaluation on KITTI test data from the server, we here compare our transfer smoothness approach with homogeneous MRF methods on the training data where dense ground truth disparity maps are available.

1. Filter MRF: In stereo matching, pairwise MRF and second-order MRF are widely used, and they correspond to using ∇, ∇^2 filter in the FoE MRF. Recently, bilateral filter is used in this framework. In Table 4.5, we compare the result without using MRF model (data energy only) and that using Filter MRF with the filters mentioned above. .is used to model

Filter Type	None	∇	∇^2	Bilateral
Error Rate	8.97	7.89	7.43	**7.25**

Table 4.5: Comparison of different Filter MRF on KITTI training data.

2. Basis MRF: We use EPLL [36] algorithm for disparity map inpainting. We first learn a Mixture-of-Gaussian (MoG) model on the held-out ground truth disparity map, and then apply EPLL algorithm to improve upon the initial disparity estimation. In Table 4.6, we show the result with varying K components and the oracle performance with ground truth component assignment of MoG.

K/Algo	EPLL	Oracle EPLL
100	6.37	5.58
200	**6.30**	5.40
500	6.40	**5.31**

Table 4.6: Comparison of EPLL MRF with oracle result on KITTI training data.

3. Slanted Plane MRF [30](Segmentation Smoothness): 5.94

4. ST-RF (Matching Smoothness): 5.72/**5.55**, the latter one is using learned weight vector

4.5.1.2 Test Data

The state-of-the-art algorithm on KITTI benchmark is PCBP, a slanted-plane MRF model with good initialization. The initial algorithms are SGBM (6.54%, rank 10) and StereoSLIC (5.17%, rank 2) and the PCBP improve them into (5.45%, rank 4) and (4.79%, rank 1).

Shown in Table 4.7, our ST-RF is initialized with similar SGBM result, and the error rate drops to 5.71% with default parameter and 5.44% (rank 3) with learned parameter. Although the improvement of ST-RF over PCBP is not significant, our inference time is 30x fasterthan the authors' implementation of PCBP.

	Initial	Inference	Inference Time	Rank
Slanted Plane MRF [30]	~6.50	5.45	5 min	4
ST-RF (ours)	6.54	5.71/5.44	10 sec	3

Table 4.7: Comparison of ST-RF with state-of-the-art algorithms on KITTI testing data.

■ 4.5.2 Patch Level Evaluation

Data-driven matching module is vital in our ST-RF performance and we here show the relationship between the matching quality and the disparity accuracy in our ST-RF system. We want to see whether regions with good matches in the database will have better disparity estimation, which verifies whether data-driven module is the bottleneck of the current system.

In image appearance related applications, e.g. denoising, the matching error between two matched patches is defined by the L2 distance of image intensity. In stereo, however, we can match features from image appearance and/or initial disparity of the query patch. For example, Karsch et.al. [11] uses the SIFT feature of the query patch to find matches in the database. As explained later in 4.5.2.2, there are many other options to define the matching metric and we want to find one that better capture matching quality in terms of disparity improvement.

Thus, in the following paragraphs, we first propose three possible matching metrics, then justify to use one of them, and lastly plot the correlation between the matching quality and the accuracy of disparity inferred by our ST-RF.

■ 4.5.2.1 Define Matching Metric

The problem statement here is that, given two patches left stereo images with their corresponding disparity map and the matching between patches from them, we want to quantify the matching distance between every pair of corresponding patches.

Below, we propose three different metric to approach this problem.

1. Intensity Difference:(PIX)
   The simplest approach is to calculate the difference in intensity between pair of patches.

2. SIFT Descriptor:$(SIFT)$
   As widely used in label transfer literature, SIFT feature matching is believed to relate patches with similar local geometry structure.

3. Nonlocal Smoothness Descriptor:$(NL\text{-}SM)$
   Instead of using appearance information, we use the disparity value after removing DC component as the feature vector. Thus, we are expecting to encourage patches with similar smoothness and orientation

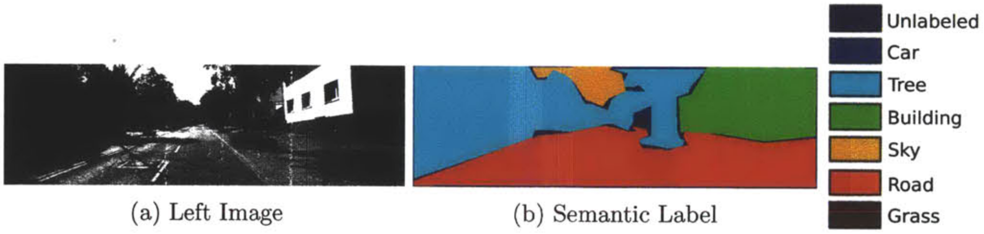
4.5.2.2 Metric Justification

In order to select from the proposed metric to define a "good" match, we evaluate their ability to capture semantic labels.

We use the first 20 KITTI training examples and manually labeled the semantic segmentation with the following six categories: car, tree, building, sky, road and grass. In Table 4.8, we show the category statistics of our evaluation database and in Figure 4.7, we show one example of our manual semantic labeling.

Category	car	tree	building	sky	road	grass
Number	54	40	28	27	26	8

Table 4.8: Object statistics from the semantically labeled dataset



(a) Left Image

(b) Semantic Label

Figure 4.7: Visualization of one example of manually labeled semantic segmentation of images from KITTI.

In Figure 4.8, we compare the semantic label accuracy for SIFT Flow algorithm with the three proposed metric. We can see that there is stronger positive correlation between the NL-SM descriptor matching metric and the semantic labeling accuracy.

4.5.2.3 Metric Evaluation

We here plot the relationship between patch matching distance when using NL-SM descriptor metric and error of disparity inferred by ST-RF.

In Fig 4.9a, we plot the mean and std of disparity error as a function of disparity matching error. With the increase matching distance, the disparity error of our data-driven stereo algorithm is increasing in both mean and std.

In Fig 4.9b, we plot the error rate of disparity difference bigger than 3 pix (standard for KITTI evaluation) as a function of disparity matching error. We also draw the error rate for traditional min-interpolation in red. When the matching distance is smaller than 0.2, our data driven stereo algorithm outperform the baseline pairwise MRF, which assumes matched patches is fronto-parallel.

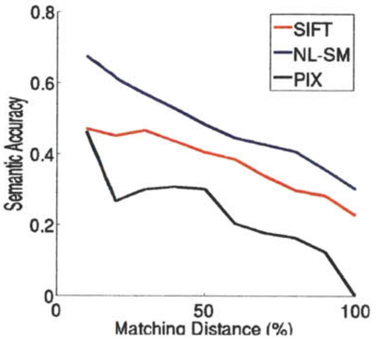
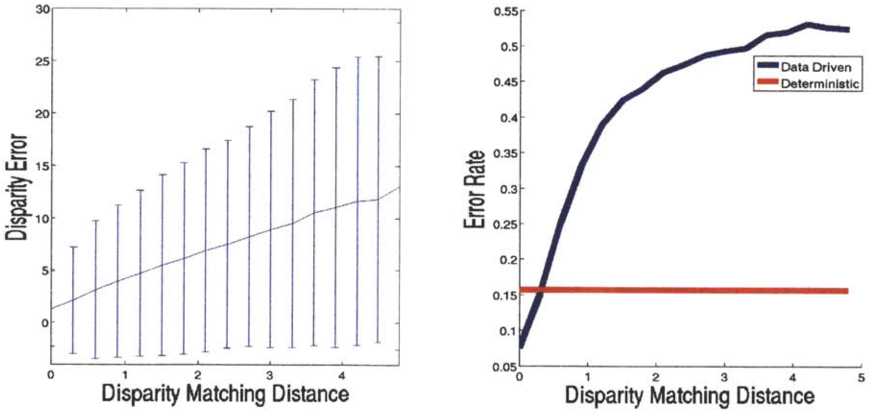


Figure 4.8: Justification for using NL-SM descriptor for matching distance. We show comparison of the relationship between semantic prediction accuracy and the matching distance for three different metrics.



(a)

(b)

Figure 4.9: Relationship between disparity matching metric and disparity improvement (a) Mean and Std of disparity error as a function of disparity matching error, (b) Error rate of disparity error as a function of disparity matching error

Chapter 5

Task II: Image Denoising

5.1 Pipeline

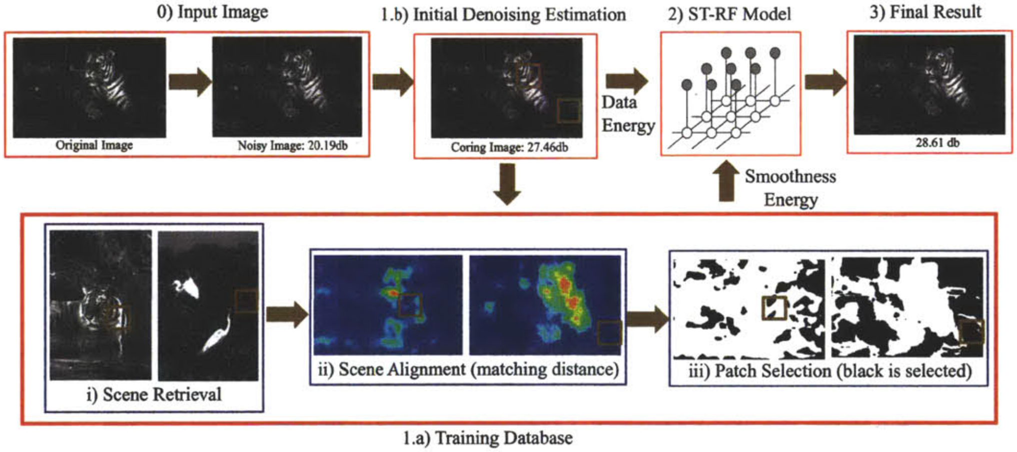


Figure 5.1: Pipeline for image denoise with ST-RF

In Figure 5.1, we show the pipeline of the image denoising:

1. **Image Coring:** Our observation is noisy but our training data is clean. In order to find better correspondence between them, we need image coring to pre-process the noisy image.

2. **Database Matching:**
   Instead of doing nearest patch search [14], we do coring image to training image matching (e.g. PatchMatch) which taking patch context information into account during matching. For scalability, we can prune the training images to match with various metric (e.g. L2 distance in pixel space).

3. **ST-RF:**

- ST-RF construction: For data energy, we use the standard Gaussian distribution from the generative model. For smoothness energy, we do the following: Now, for each patch in the coring image, we have a pile of matched patches. We set a threshold for the distance of the matches to define inliers, with which we build non-parametric kernel distribution.
- ST-RF inference: We use similar algorithm to that in EPLL, where the assignment of the inlier for each patch is inferred iteratively during annealing.

5.2 ST-RF Settings

With no intention to further improve the result, we use almost the same setting as that for stereo matching in Chapter 4. The only difference lies in (1) data energy modeling, where it is defined as $-log(\mathcal{N}(x, y; \sigma^2))$ for image denoising; (2) loss function for MRF learning, where we use directly the evaluation loss function, quadratic loss.

5.3 Results on Berkeley Segmentation Dataset

In all experiments below, we use 100 test images from Berkeley segmentation dataset and generate Gaussian noise corrupted images with noise level $\sigma = \{15, 25, 50, 100\}$. We use the sum-absolute-distance (SAD) metric in pixel space to measure the distance between patches.

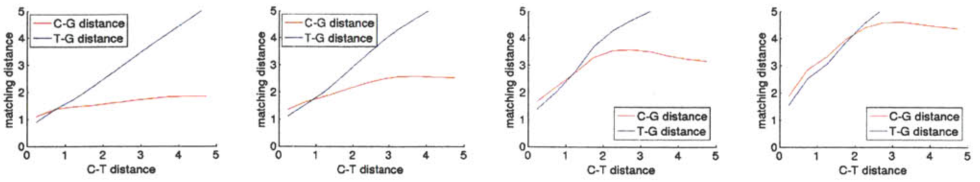
We want to show three things:

1. The closer distance between patches from coring image (C) and matched patches from training image (T), the closer T is than C in terms of the distance from the ground truth patches (G).
2. Incorporating these "better" patches T into smoothness energy in ST-MRF model can achieve better denoising result.

We repeat the same experiment set up except that we train on the training data from Berkeley segmentation dataset (200 images) instead of the ground truth image.

5.3.1 Matching Performance

Shown in Figure 5.2, unlike the performance for matching ground truth image, the improvement of retrieved patches is small due to the diversity of training images and lack of similarity to test images.



(a) σ=15 (b) σ=25 (c) σ=50 (d) σ=100

Figure 5.2: Train on training images from Berkeley segmentation dataset. Comparing the distance from matched training patches to ground truth (T-G) and that from coring patches to ground truth (C-G)

■ 5.3.2 Image Denoising Comparison

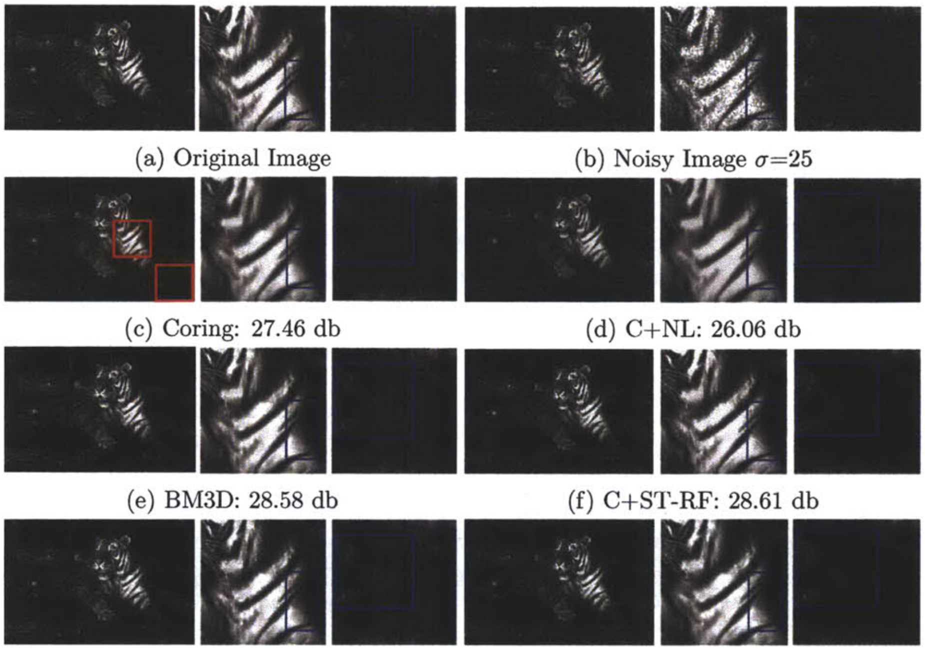
For comparison, we re-trained EPLL and KSVD method on the sampled patches from all the training images. For ST-RF, we use the BM3D-coring and test with default parameters and learned parameters respectively. The learning and inference is similar to that presented in Sec 4.4, 4.3, and we here omit the detailed testing result.

Shown in Table 5.1, ST-RF result with the default parameter is slightly worse than the-state-of-art algorithm EPLL.

As expected, nonlocal means will further blur out the detail from the coring image by averaging in the pixel domain. Collaborative Wiener filtering obtains better estimation by taking the spectrum energy estimated from the coring image. Our ST-RF is on par with BM3D by retriving and incorporating clean patches which add back the detail of estimated patches. We have a closer look at one example in Figure.

σ/Algo	Coring	ST-RF	EPLL	KSVD	BM3D
15	30.51	30.96/**31.07**	30.97	30.69	30.87
25	27.96	28.10/**28.61**	28.46	28.12	28.35
50	24.92	25.31/25.43	**25.49**	25.01	25.45
100	22.18	22.87/**23.17**	22.94	22.31	23.13

Table 5.1: Image Denoising result for algorithms trained on Berkeley segmentation training dataset. Our new ST-RF is comparable to the state-of-the-art methods.



(a) Original Image

(b) Noisy Image σ=25

(c) Coring: 27.46 db

(d) C+NL: 26.06 db

(e) BM3D: 28.58 db

(f) C+ST-RF: 28.61 db

(g) EPLL: 28.33 db

(h) KSVD: 28.19db

Figure 5.3: Comparison of Image denoising results and closer look at patch level. Notice that C+ST-RF better preserves the texture around the tiger's neck and has less artifact on the grass, due to contraints of matched patches in the training data. C+NL and BM3D only have the input from the corruped image and consequently have hard time recovering the detail on patches. EPLL and KSVD, on the other hand, learn a parametric patch model, which may lead to unnatural results due to the expressiveness of the model.

Chapter 6

Conclusion

IN this thesis, we introduce the idea of using scene level correspondence to construct smoothness energy in the domain of spatially discrete random field models of image. In contrast to previous RF models, which construct smoothness energy using fixed set of filters or basis with or without locally steering of the orientation, the proposed Smoothness Transfer Random Field model (ST-RF) transfers the ground truth smoothness energy from the training data.

On standard benchmark datasets, KITTI dataset for stereo matching and Berkeley Segmentation dataset for image denoising, our ST-RF framework achieves state-of-the art performance.

In the following sections, we summarize the contribution of this thesis and point out future directions to pursue.

■ 6.1 Contribution

In terms of RF model construction, we are the first to build data-driven smoothness to model the heterogeneous property of an image. Local adaptation is an active research area in random field model, and previous work focuses on finding better parametric form to orientation or weighting.

In terms of data-driven application, we work on stereo matching which has high accuracy requirement. Most nonparametric modeling explore to improve performance for enhancing image appearance like scene completion [8] and super resolution [27]. Beyond transferring pixel intensity, semantic classification [17] and depth transfer [11] are made possible by scene alignment algorithms. However, the complexity of the semantic labeling space is substantially limited and the accuracy evaluation for depth transfer is only qualitative.

■ 6.2 Future Work

There are two modules in our ST-RF: data-driven module and RF module. On one hand, our RF module uses pixel level grid representation and standard MRF learning and inference algorithm. One future direction is to apply our ST-RF framework to more sophisticated RF module with the goal of achieving more significant improvement in specific vision tasks. On the other hand, our data-driven module uses scene alignment algorithm for correspondence calculation and nonlocal smoothness to capture ground truth smoothness. Another future direction is to consider alternative choices depending on the property of the vision task and the database. Below, we list our thoughts on possible representation for data-driven module.

6.2.1 Correspondence Representation

Object Level

Due to the size of the training dataset, many images do not have close enough matches during scene alignment. Thus, object level correspondence appear more favorable for its repetitivity and flexibility.

Segmentation Level

Recently, [27] built the-state-of-the-art super-resolution algorithm upon the of segmentation level correspondence.

6.2.2 Ground Truth Smoothness Representation

Beyond Nonlocal Smoothness

Given the matched patch, we now transfer the difference between the center pixel and the rest as the ground truth smoothness. However, such smoothness is only invariant to the shift of DC (e.g. value at the center pixel). Thus, we can not only have higher-order smoothness (e.g. bilateral smoothness) but also smoothness which has other types of invariance (e.g.rotation) For example, let x_0 be the value at center pixel, $x_1, x_1^{'}$ the value at two x_0-symmetric pixel. Then the smoothness function $x_1 + x_1^{'} - 2 * x_0$ is a second-order smoothness with rotation invariance. We also desire a compatibility function that can explicitly capture the occluding boundary.

Learning Smoothness

Instead of the manually design of smoothness form, we can try to build a model to learn the right type of smoothness automatically.

Bibliography

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.

[2] Andrew Blake, Pushmeet Kohli, and Carsten Rother. *Markov random fields for vision and image processing.* The MIT Press, 2011.

[3] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[4] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*, 4(2):490–530, 2005.

[5] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, Karen Egiazarian, et al. Image denoising with block-matching and 3 d filtering. In *Proceedings of SPIE*, volume 6064, pages 354–365, 2006.

[6] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *Image Processing, IEEE Transactions on*, 15(12):3736–3745, 2006.

[7] William T Freeman, Egon C Pasztor, and Owen T Carmichael. Vista. *International journal of computer vision*, 40(1):25–47, 2000.

[8] James Hays and Alexei A Efros. Scene completion using millions of photographs. In *ACM Transactions on Graphics (TOG)*, volume 26, page 4. ACM, 2007.

[9] Heiko Hirschmuller. Accurate and efficient stereo processing by semi-global matching and mutual information. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 807–814. IEEE, 2005.

[10] Jörg H Kappes, Bjoern Andres, Fred A Hamprecht, Christoph Schnörr, Sebastian Nowozin, Dhruv Batra, Sungwoong Kim, Bernhard X Kausler, Jan Lellmann, Nikos Komodakis, et al. A comparative study of modern inference techniques for

discrete energy minimization problems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[11] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth extraction from video using non-parametric sampling. In *Computer Vision–ECCV 2012*, pages 775–788. Springer, 2012.

[12] Philipp Krähenbühl and Vladlen Koltun. Efficient nonlocal regularization for optical flow. In *Computer Vision–ECCV 2012*, pages 356–369. Springer, 2012.

[13] Yann LeCun and Fu Jie Huang. Loss functions for discriminative training of energy-based models. AIStats, 2005.

[14] Anat Levin, Boaz Nadler, Fredo Durand, and William T Freeman. Patch complexity, finite pixel correlations and optimal denoising. In *Computer Vision–ECCV 2012*, pages 73–86. Springer, 2012.

[15] Yunpeng Li and Daniel P Huttenlocher. Learning for stereo vision using the structured support vector machine. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[16] Ce Liu, Jenny Yuen, Antonio Torralba, Josef Sivic, and William T Freeman. Sift flow: dense correspondence across different scenes. In *Computer Vision–ECCV 2008*, pages 28–42. Springer, 2008.

[17] Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1972–1979. IEEE, 2009.

[18] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2272–2279. IEEE, 2009.

[19] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[20] René Ranftl, Stefan Gehrig, Thomas Pock, and Horst Bischof. Pushing the limits of stereo using variational stereo estimation. In *Intelligent Vehicles Symposium (IV), 2012 IEEE*, pages 401–407. IEEE, 2012.

[21] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 860–867. IEEE, 2005.

[22] Stefan Roth and Michael J Black. Steerable random fields. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[23] Kegan GG Samuel and Marshall F Tappen. Learning optimized map estimates in continuously-valued mrf models. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 477–484. IEEE, 2009.

[24] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1-3):7–42, 2002.

[25] Uwe Schmidt, Qi Gao, and Stefan Roth. A generative perspective on mrfs in low-level vision. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1751–1758. IEEE, 2010.

[26] Brandon M Smith, Li Zhang, and Hailin Jin. Stereo matching with nonparametric smoothness priors in feature space. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 485–492. IEEE, 2009.

[27] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *Computational Photography (ICCP), 2012 IEEE International Conference on*, pages 1–12. IEEE, 2012.

[28] Richard Szeliski. *Computer vision: algorithms and applications*. Springer, 2010.

[29] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Kernel regression for image processing and reconstruction. *Image Processing, IEEE Transactions on*, 16(2): 349–366, 2007.

[30] Koichiro Yamaguchi, Tamir Hazan, David McAllester, and Raquel Urtasun. Continuous markov random fields for robust stereo estimation. In *Computer Vision–ECCV 2012*, pages 45–58. Springer, 2012.

[31] Qingxiong Yang. A non-local cost aggregation method for stereo matching. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1402–1409. IEEE, 2012.

[32] Honghui Zhang, Tian Fang, Xiaowu Chen, Qinping Zhao, and Long Quan. Partial similarity based nonparametric scene parsing in certain environment. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2241–2248. IEEE, 2011.

[33] Shengqi Zhu, Li Zhang, and Hailin Jin. A locally linear regression model for boundary preserving regularization in stereo matching. In *Computer Vision–ECCV 2012*, pages 101–115. Springer, 2012.

[34] Song Chun Zhu and David Mumford. Prior learning and gibbs reaction-diffusion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(11):1236–1250, 1997.

[35] Song Chun Zhu, Yingnian Wu, and David Mumford. Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling. *International Journal of Computer Vision*, 27(2):107–126, 1998.

[36] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 479–486. IEEE, 2011.