國 立 清 華 大 學

碩 士 論 文

立體匹配演算法使用階層式過度分割與可信度傳遞

Stereo Matching Algorithm using Hierarchical Over-segmentation and Belief Propagation

系別： 資訊工程學系

學號姓名： 9762512 金立軒 (Li-Hsuan Chin)

指導教授： 賴尚宏 教授 (Dr. Shang-Hong Lai)

中華民國 九十九 年 七 月

National Tsing Hua University

Master Thesis

Stereo Matching Algorithm using Hierarchical Over-segmentation and Belief Propagation

by Li-Hsuan Chin

Advisor: Prof. Shang-Hong Lai

July 2010

A Thesis

Submitted to the Department of Computer Science

National Tsing Hua University

Hsinchu, Taiwan, Republic of China

In Partial Fulfillment of the Requirements

For the Degree of Master

In

Computer Science

致謝

兩年多的碩士生涯，說長不長，說短也不短。在這條辛苦的路上，首先感謝我的指導老師賴尚宏教授在研究上給予我充分的指導；也因此讓我能夠對於我的研究有了明確的方向與深入的了解。其次感謝我的父母，雖然在我就讀研究所的日子中鮮少陪伴他們，但他們仍全力地支持我、鼓勵我、關心我。再來是實驗室永遠的大師兄—嘉岷學長，也是在研究上協助我最多的人。另外感謝曾經是小房間的成員：柏豪、奕麟、皓量、正宏、曉薇、幼臻等人陪伴著我，讓我在研究之路上不是孤單一人，亦增添了許多歡笑與溫馨。也感謝其餘的實驗室成員：思皓、家德、振國、宏任、書凡、東穎、德峰等博士班學長與宗昱、聖博、克駿、詩杰等與我同屆的碩士班兄弟們，還有已經畢業的淑娟、蕙如、依婷等學長姐和碩一的學弟志學及孝安。你們大家不僅構成了這個如大家庭般的實驗室，每個人更是包容著我，使我的生活更多采多姿而不至於乏味死板。特別要感謝的是這一路上始終如一持續給我加油打氣的女友—儀芳，除了要犧牲相處的時間讓我能專心做研究之外，更要擔任一個良好旁聽者的角色讓我傾訴一切。還要感謝我所飼養之動物界脊索動物門哺乳綱食肉目貓科貓屬亞種是家貓的寵物尼尼，我所累積一天的疲倦與困頓在看到他迎接我的可愛臉孔即全部消散了。最後再次感謝協助我完成論文與研究的所有人，分享一句話給你們—「我們秤不出生命中的黃金點滴、量不了愛河裡的迷人笑容，但只要指出哪次有比較多的歡樂那就夠了。」

摘要

在本篇論文中，我們提出了一個從校正好的一組影像得到視差圖的新演算法。我們首先使用影像過度分割來建立以內容為基礎的階層式馬可夫隨機場。這種影像表示方式包含了兩個利於做視覺應用的優點。第一個是階層式馬可夫隨機場的建立，而另一個是正規的圖形結構。前者已經被廣泛地應用於電腦視覺的問題來改進最佳化馬可夫隨機場的效率。後者可以簡化最佳化馬可夫隨機場技術其訊息的傳遞和硬體的實作。在建立完以內容為基礎的階層式馬可夫隨機場後，我們使用階層式可信度傳遞於對稱的立體匹配與遮蔽處理在提出的圖形模型上。最後，引進一個精煉視差圖的方法(例如平面擬合或雙向濾波器)來減少因遮蔽、無材質的區域或是影像上雜訊等等導致視差的錯誤估計。我們的實驗結果展現出我們可以有效率地獲得媲美大部分全局立體匹配演算法的準確度之視差圖。對於真實的影像序列，我們先使用強健的自身影像校正之前處理來準確地估計出每張影格的深度資訊。

Abstract

In this thesis, we present a novel algorithm to infer disparity map from given a pair of rectified images. We first employ image over-segmentation to construct a Content-based Hierarchical Markov Random Field (CHMRF). This image representation contains two advantages for vision applications. One is the hierarchical MRF construction, and the other is the regular graph structure. The former has been widely applied to computer vision problems to improve the efficiency in MRF optimization. The latter can simplify the message passing and hardware implementation of MRF optimization techniques. After the construction of CHMRF, we perform symmetric stereo matching and occlusion handing using Hierarchical Belief Propagation (HBP) based on the proposed graphical model. Finally, a refinement process for the disparity map is introduced (*e.g. plane fitting or bilateral filtering*) to reduce the disparity errors caused by occlusion, textureless region or image noise, etc. Our experimental results show that we can efficiently obtain disparity maps of comparable accuracy when compared to most global stereo algorithms. For real stereo video sequences, we are able to accurately estimate the depth information for each frame with the pre-processing of robust self image rectification.

Table of Contents

Figure List

Table List

Algorithm List

Chapter 1. Introduction

1.1 Motivation

Stereo matching has been one of the kernel topics in 3D computer vision. With the development of three-dimensional display industry, such as 3DTV, stereoscopic movies, and so on, stereo vision is becoming more and more mature. An extensive amount of work on this topic has been surveyed and evaluated by Scharstein and Szeliski [1]; afterwards, many stereo matching algorithms mushroomed like bamboo shoots after a spring rain. Thanks to the publically available performance testing datasets, such as the Middlebury benchmark [2], which allowed researchers to compare the performances of their algorithms against those of the others' algorithms.

If we study the state-of-the-art stereo methods on the Middlebury website [2], we can find 9 out of the top 10 methods share a common idea. That is, they perform some image segmentation, followed by some sort of global optimization. Segmentation, in a nutshell, is used as a way to reduce the effects of image noise and matching ambiguities on the stereo results. Therefore, most segmentation-based algorithms can acquire high-quality disparity maps. However, if they are combined with global

energy minimization approaches, e.g. *Belief Propagation (**BP**)* [3] *or Graph Cuts (**GC**)* [4], they may suffer from the time-consuming problem.

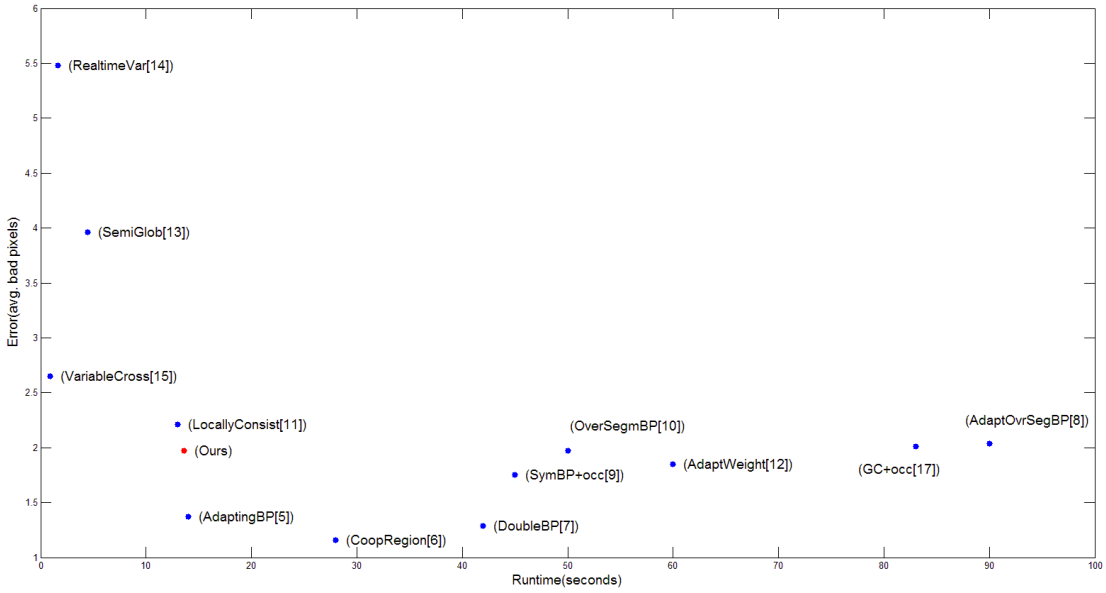


Figure 1 Runtime versus error for *Tsukuba* dataset. Red spot is the proposed method and blue ones are others.

We compare the proposed method with most global stereo algorithms which provided their execution time on the Middlebury website but without considering the computing platforms as shown in Figure 1. We can observe that our approach looks quite promising from both perspectives of accuracy and efficiency compared to most global stereo matching algorithms. This figure also justifies our expectation for the stereo matching technique to be developed.

1.2 Problem Description

Stereo vision infers 3D scene geometry from two images with different viewpoints.

Recent applications such as view synthesis and image-based rendering make stereo vision again an active research topic in computer vision. Classical dense stereo matching methods compute a dense disparity or depth map from a pair of images under known camera configuration.

Scharstein and Szeliski [1] pointed out that most stereo algorithms generally perform (subsets of) the following four steps:

1. matching cost computation;
2. cost (support) aggregation;
3. disparity computation / optimization; and
4. disparity refinement.

Depending on the actual sequence of steps taken, stereo research can be divided into two branches, *i.e. local* and *global* methods.

For *local* (window-based) approaches, the key factor is support windows. The disparity computation at a given point relies upon intensity values within a finite window or patch, and it usually makes implicit smoothness assumptions by aggregating these supports. Figure 2 shows the effect of different window size on disparity maps obtained by traditional sum-of-squared-differences (SSD) algorithm.

(a) (b) (c)

Figure 2 Effect on disparity maps. (a): input images. (b)and(c): stereo results generated by SSD with window size W = 3 and W = 20.

We can observe that smaller window leads to not only more details but also more noises, and on the other hand, larger window causes both less details and noises. Kanade and Okutomi [17] presented an adaptive window-based algorithm to deal with this problem for obtaining better results.

For *global* algorithms, they make explicit smoothness assumptions and then solve an optimization problem. These algorithms, typically, do not perform an aggregation step, but rather seek a disparity assignment that minimizes a global energy function; the objective is to find a disparity function *d* that minimizes this energy,

$$E(d) = E_{data}(d) + \lambda E_{smooth}(d). \quad (1)$$

Most global methods solve the stereo matching problem based on the Markov Random Field (MRF) models. While the MRF formulation of these problems yields an energy minimization problem that is NP hard, some good approximation algorithms based on BP [3] and GC [4] have been developed and widely utilized.

1.3 System Overview

Our stereo reconstruction system contains three major parts: **CHMRF construction**, **stereo matching** and **refinement process**. As described above, we implement CHMRF structure to improve the performance of standard HBP. And the second part will generate both left and right disparity maps and occlusion maps via applying symmetric stereo matching based on HBP with the proposed graph structure and energy formulation. Finally, the refinement of disparity maps is applied optionally to suppress the disparity errors. Figure 3 is our system framework.

1.4 Main Contributions

1. We propose a novel image representation called CHMRF, and it can be used in stereo matching since it can be modeled as a labeling problem on a MRF graph.
2. The CHMRF graph structure used in HBP for solving the stereo matching problem can efficiently infer disparity maps of comparable precision with those of most segmentation-based global stereo algorithms.

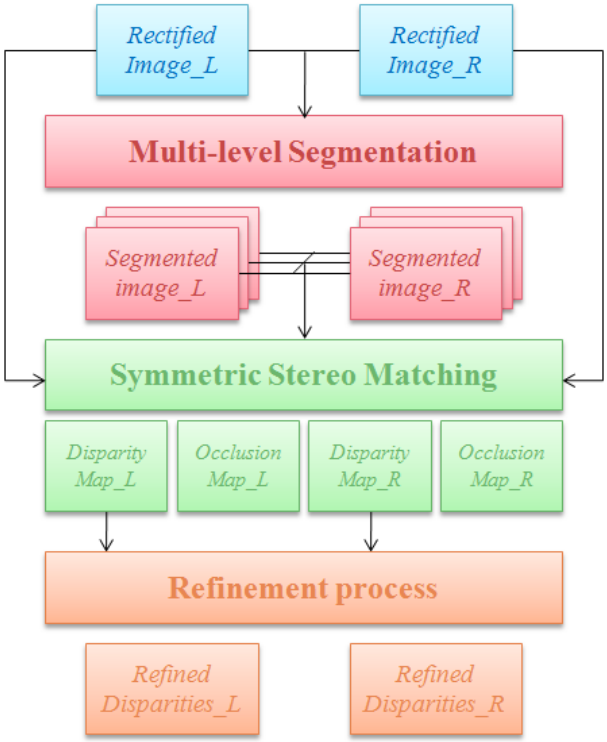


Figure 3 The system flowchart of the proposed stereo matching algorithm.

1.5 Thesis Organization

The rest of this dissertation is organized as follows: first in Chapter 2, several previous related works are reviewed. Chapter 3 describes the proposed strategy for constructing the CHMRF and solving the stereo matching problem using HBP based on this structure. Chapter 4 gives the experimental results by using the proposed algorithm. Finally, some conclusions are drawn in Chapter 5.

Chapter 2. Related Works

2.1 Segmentation-based Stereo Algorithms

As mentioned in Chapter 1, image segmentations are widely used by stereo matching algorithms for estimating depth information, such as [18], [19], [20][21][22][7], and they apparently dominate the top-ranking methods in the Middlebury benchmark [2]. The core of these segmentation-based algorithms is formed by the segmentation assumption, which claims that a segment should lie on the same surface in 3D space. Most of them adopt the following four-step procedure. In the first step, a color segmentation technique, for instance the mean-shift segmentation method [23], is applied to the images. Secondly, a stereo matching algorithm is applied to acquire initial disparities from the given image pair. Thirdly, each segment is assigned to one unique 3D disparity plane by applying the plane fitting to an initial disparity map. In the final optimization step, the extracted disparity planes are then propagated among segments to minimize a given energy function.

Except for segmentation-based approaches, [8] introduced an over-segmentation based method to infer high quality disparities. They modeled the stereo and

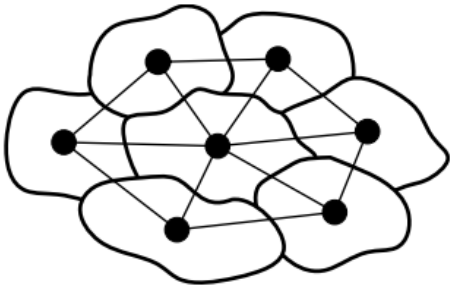


Figure 4 Stereo and smoothness constraints on the segment depths are modeled using a pairwise MRF, in which each segment corresponds to a node, and the nodes of neighboring segments are joined by edges.

smoothness constraints using a pairwise MRF of segments, as shown in Figure 4. Each node represents a segment *s* and shares common edges with neighboring segment *t* for (*s*, *t*) ∈ *N*, where *N* is the set of all adjacent segments. Then the initial depth of each segment is estimated using max-product belief propagation on their proposed stereo MRF model.

2.2 Performance Improvement of BP

However, most of these segmentation-based stereo algorithms are multi-stage algorithms, and they are usually combined with the global energy minimization approaches. In addition to generating the segmentation result, the optimization step takes most of the runtime in their stereo system. BP is a very popular inference algorithm for finding the most likely label of each node in MRF, especially for solving stereo matching problem. Here we will review some previous works related to BP, including performance analysis and improvement over some proposed schemes.

BP algorithm works via passing messages around the graph which is defined by the four-connected image grid. Let N be the image size, L be the number of disparity levels, and T be the number of iterations, so the computational complexity for computing the messages is $O(4TNL^2) = O(TNL^2)$ [3]. Felzenszwalb and Huttenlocher [24] showed that the complexity of BP can be reduced to $O(TNL)$ using min convolution method. They also used hierarchical structure to reduce the number of message passing iterations, and sped up the convergence of the BP algorithm at the same time.

In recent years, many applications related to stereo matching need to realize it on a hardware system, so the factor of efficiency must be taken into consideration. On account of this reason, Yu *et al.* [25] proposed two common schemes, predictive coding and linear transform coding (PCA), and then proposed a novel Envelope Point Transform (EPT) method for message compression.

On graphic processing unit (GPU) implementation, Yang *et al.* [26] presented the first BP based global method called fast-converging BP that runs at real-time speed. They reduced redundant computations involved in standard BP by only updating pixels that have not yet converged. And Liang *et al.* [27] designed a low-power VLSI circuit for

disparity estimation that generates high quality disparity maps nearly in real-time. The key idea is that a message can be well approximated from other faraway ones, thus they split the MRF into many tiles and perform BP within each one. An $O(L)$ message construction algorithm is applied for the robust functions commonly used for describing the smoothness terms in the energy function. Recently, Yang *et al.* [28] proposed a constant space $O(1)$ BP (**CSBP**) method to improve the standard HBP algorithm. Unlikely previous memory reduction methods which focus on the original spatial resolution, they hierarchically reduce the disparity range to be searched.

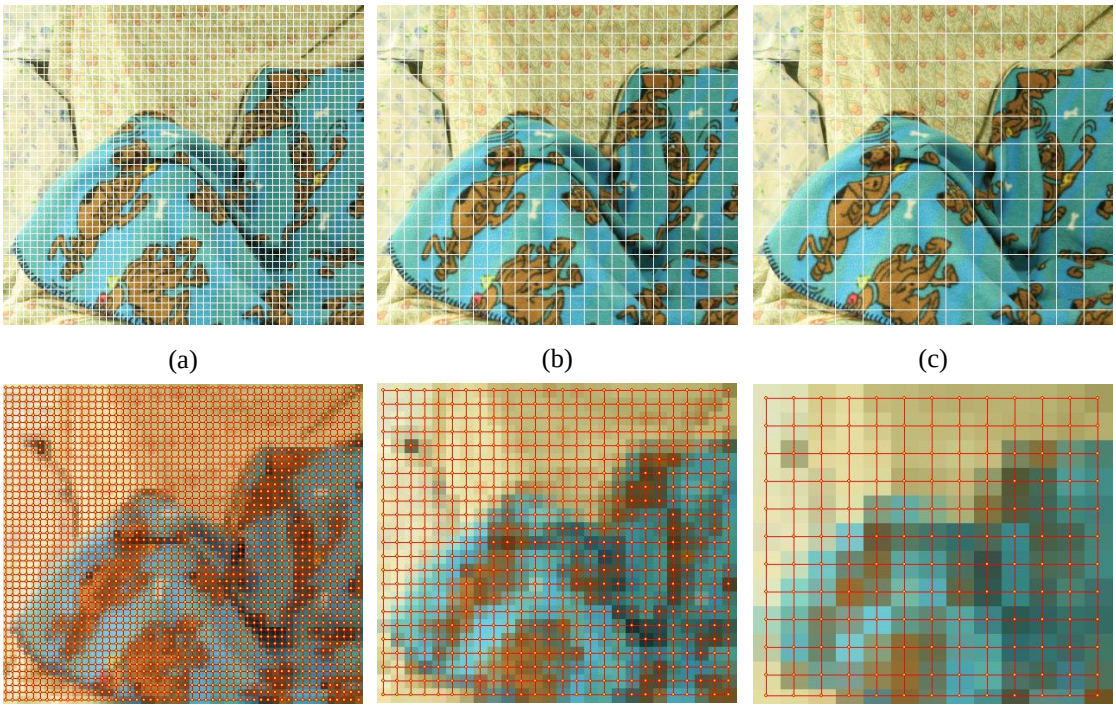
Chapter 3. Proposed Method

As mentioned in section 1.3, our stereo system can be divided into three major components : hierarchical over-segmentation, stereo algorithm and refinement process. This section will go into details of these three components in the proposed method.

3.1 Hierarchical Over-segmentation

To construct the CHMRF structure suitable for the subsequent MRF optimization, we employ an over-segmentation technique at each level, and there are several steps. In the initial step, we layout regular blocks on the input images hierarchically. The block size b_k at level k is set to 2^L and $b_k = 2b_{k-1}$; at first level, $b_0 = 2^{L_{ini}}$ is given by user. For example, we set $L_{ini} = 3$ which indicates the initial block size is $2^3 = 8$, and we set $L_{tot} = 3$ which denotes the total level of this hierarchical structure is 3. For the representation of a regular graph, each node corresponds to the center of each block. Figure 5 shows the results and its corresponding graph at this step.

In the next step, we apply over-segmentation to each level via computing the geodesic



(a) (b) (c)

(d) (e) (f)

Figure 5 Initialization from level 1 to level 3 in the hierarchical over-segmentation. (a)~(c): Regular blocks; (d)~(f): corresponding graph structures for different levels.

distance transform [29] efficiently between nodes and centers. The geodesic distance $D_{geo}(\boldsymbol{p}, \boldsymbol{c})$ between a node p and the block's center c is defined as the shortest path that connects $\boldsymbol{p}$ and $\boldsymbol{c}$:

$$D_{geo}(\boldsymbol{p}, \boldsymbol{c}) = min_{P \in \mathcal{P}_{\boldsymbol{p},\boldsymbol{c}}} d(P) \tag{2}$$

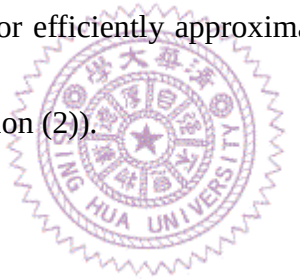
Here, $\mathcal{P}_{\boldsymbol{p},\boldsymbol{c}}$ denotes the set of all possible path between $\boldsymbol{p}$ and $\boldsymbol{c}$. A path P is defined as a sequence of spatially neighboring nodes in 8-connectivity $\{\boldsymbol{p_1}, \boldsymbol{p_2}, \dots, \boldsymbol{p_n}\}$. The cost of a path P is computed by

$$d(P) = \sum_{i=2}^{n} diff(\boldsymbol{p_i}, \boldsymbol{p_{i-1}}) \tag{3}$$

where $diff()$ is a function that determines the spatial and color differences. This function is defined by

$$diff(\boldsymbol{p},\boldsymbol{q}) = \sqrt{\|(\boldsymbol{p}-\boldsymbol{q})\|^2 + \lambda_c \sum_{i=1}^{3} |I_i(\boldsymbol{p}) - I_i(\boldsymbol{q})|^2} \tag{4}$$

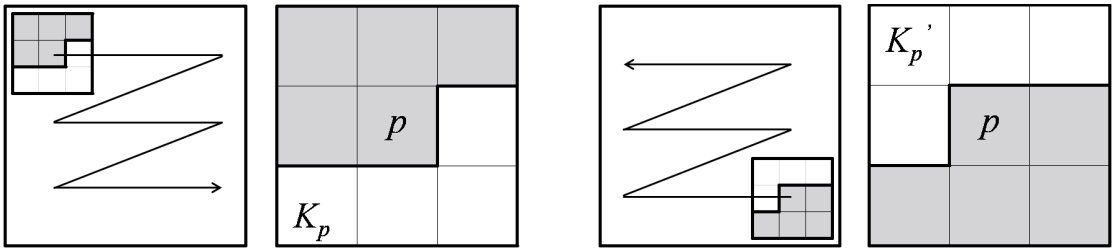
where $I_i(\boldsymbol{p})$ denotes the value of the i^{th} color channel at pixel $\boldsymbol{p}$. In our implementation, we represent color using the RGB system. And λ_c is the weight for the color difference, which can controls the cost of geodesic distance. However, there exists a direct path between any two nodes, which makes the problem more difficult. We apply the method of [30] for efficiently approximating the geodesic distances of each node to the centers (Equation (2)).



Each node $\boldsymbol{p}$ inside the window is assigned to the cost $C(\boldsymbol{p})$. Initially, the cost of the center node is set to 0, while the costs of all other nodes are set to a large constant value. In the forward pass of the algorithm, we traverse the support window in a row major order. The cost of a node $\boldsymbol{p}$ is thereby updated by

$$C(\boldsymbol{p}) := \min_{q \in K_p}(C(\boldsymbol{q}) + diff(\boldsymbol{p},\boldsymbol{q})) \tag{5}$$

with the kernel K_p being a set of nodes consisting of $\boldsymbol{p}$ itself as well as its north-west, north, north-east and west neigjbors (see Figure 6(a)).



(a) (b)

Figure 6 Approximation of the geodesic distances inside a window. (a): Forward pass. (b): Backward pass.

The cost update is thereby performed immediately so that the new costs already affect the cost computation of the next node. Once the forward pass is completed, we invoke the backward pass. This pass traverses the window in the reverse direction (see Figure 6(b)). It thereby updates the costs using Equation (5) in conjunction with the kernel K_p' of Figure 6(b).

Forward and backward passes proceed iteratively at each level. In our experiments, we found two iterations to be sufficient for giving reasonable results. After the final costs $C(\boldsymbol{p})$ are computed, we assign each node to a center if the geodesic distance between them is the smallest.

Algorithm 1 summarizes the method in this step, and Figure 7 shows our multi-level over-segmentation results and the CHMRF graph structure. Especially, in the clustering procedure, we only consider the block center of a node and its four neighboring centers. This indicates that any two edges will not be crossed. Due to the

segmentation results used in the stereo matching step, this constraint can reduce additional checks.

Algorithm 1 Multi-level over-segmentation method

Input : An image I to be segmented.

Output : Segmentation results $\{S\}$.

1: Choose a L_{ini} value for initial block size and a L_{tot} value for total level.

2: Layout regular blocks on the input images hierarchically according to L_{ini} and L_{tot}.

3: **for** l = 0 to $L_{tot} - 1$ **do**

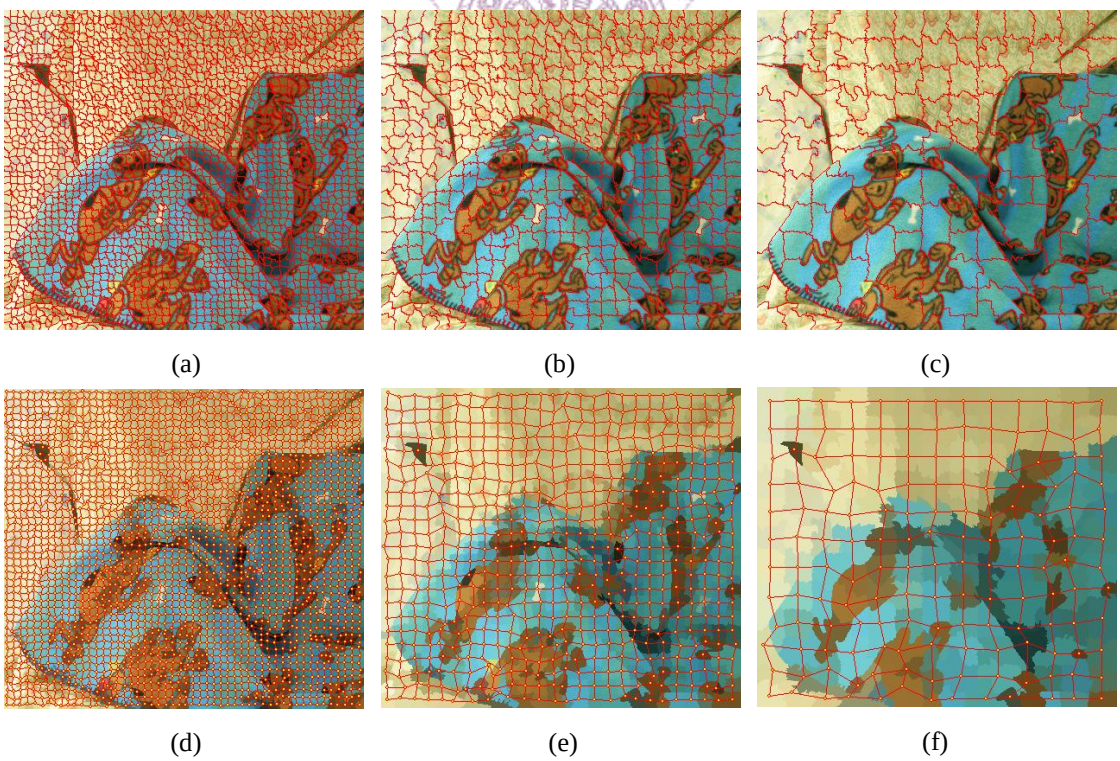
4: **for** t = 1 to 2 **do**

5: Compute the geodesic distances between nodes and centers according to Equation (4) by the two-pass algorithm.

6: **end for**

7: Assign each node to a center if the final cost $C(\boldsymbol{p})$ is the smallest.

8: **end for**



(a) (b) (c)

(d) (e) (f)

Figure 7 Segmentation results from level 1 to level 3 in the hierarchical over-segmentation. (a)~(c): Regular blocks; (d)~(f): corresponding graph structures for different levels.

3.2 Stereo Matching Algorithm

After the hierarchical over-segmentation, we apply our stereo matching algorithm using the modified HBP on the proposed CHMRF graph structure to estimate disparity maps. Most local and global stereo algorithms only compute the disparity and occlusion maps in the reference view (say, the left input image), so here we employ a symmetric stereo model similar to the one proposed by Sun *et al*. [9]. In the global algorithm, stereo matching can be formulated in an energy minimization framework, i.e. to minimize the following energy function

$$E(D_L : I) = \lambda_d \cdot E_d(D_L; I_L, I_R) + E_s(D_L), \tag{6}$$

where the data term $E_d(D_L : I)$ measures how well the disparity D_L fits the given stereo image pair I, the smoothness term $E_s(D_L)$ encodes a smoothness assumption on disparity, and λ_d is a weight to balance these two terms. The occlusion is either considered implicitly or treated as an outlier process. The disparity D_R of the right image is computed independently. Here, consistency between two views is not enforced.

In the symmetric stereo model, given a stereo image pair $\{I_L, I_R\}$, we want to estimate the disparities $\{D_L, D_R\}$ and occlusion maps $\{O_L, O_R\}$ for left view I_L and right view I_R. Now we formulate the stereo matching problem as follows:

$$E(D_L, D_R, O_L, O_R; I_L, I_R) = E_d(D_L; I_L, I_R, O_L) + E_d(D_R; I_L, I_R, O_R)$$

$$+E_s(D_L, O_L) + E_s(D_R, O_R) \quad (7)$$

Since there exists symmetric relationship between $\{D_L, O_L\}$ and $\{D_R, O_R\}$ in the above formulation, it is called a symmetric stereo model.

3.2.1 Data term

In the stereo configuration, pixel x in I_L corresponds to pixel $x - d_L$ in I_R by disparity d_L. Similarly, x in I_R corresponds to $x + d_R$ in I_L. All possible disparity values for d_L and d_R are integers between 0 and $\mathcal{L}$, where $\mathcal{L}$ is the maximum positive disparity value. The color of pixel x in I_L (or I_R) is denoted as $I_L(x)$ (or $I_R(x)$). We define the data term $E_d(D_L, O_L; I_L, I_R)$ on the left (reference) image as

$$E_d(D_L; I_L, I_R, O_L) = \sum_x \big(1 - O_L(x)\big)\rho_d(C_{BT}(x, d_L)) + O_L(x)\eta_o \quad (8)$$

where $O_L(x) \in \{0,1\}$ is a binary variable, $\rho_d(s)$ is a truncated L_1 norm function that is robust to outliers due to noise, occlusions, specularities, and so on:

$$\rho_d(s) = -ln\left((1 - e_d)\, exp\left(-\frac{|s|}{\sigma_d}\right) + e_d\right) \quad (9)$$

where parameters σ_d and e_d control the shape of robust function. For the matching cost, we use the sampling-insensitive absolute difference of Birchfield and Tomasi's pixel dissimilarity [31] $C_{BT}(x, d_L)$, which computes the absolute distance between

the extrema of linear interpolations of the corresponding pixels of interest with their neighbors. It can be easily obtained by computing the following terms:

$$C_{BT}(x, d_L) = min\ (A, B)$$

$$A = max\ (0, I_L(x) - I_R^{max}(x - d_L), I_R^{min}(x - d_L) - I_L(x))$$

$$B = max\ (0, I_R(x - d_L) - I_R^{max}(x), I_L^{min}(x) - I_R(x - d_L))$$

$$I^{min}(x) = min\ (I^-(x), I(x), I^+(x))$$

$$I^{max}(x) = max\ (I^-(x), I(x), I^+(x))$$

$$I^-(x) = \frac{1}{2}(I(x-1) + I(x))$$

$$I^+(x) = \frac{1}{2}(I(x+1) + I(x)) \quad (10)$$

The cost η_o is the penalty for occlusion labeing. This term is necessary to prevent the whole scene from being labeled as occlusion. On the right image, the data term $E_d(D_R, O_R; I_L, I_R)$ can be computed in the similar way.

3.2.2 Smoothness term

Like many other stereo matching methods, [8] assumes that depth varies smoothly over the entire image except at object boundaries, and [9] only enforces smoothness within occluded and unoccluded regions. We combine these two schemes, and the smoothness (discontinuity) cost becomes

$$E_s(D_L, O_L) = \sum_{s,t \in C \setminus B} min\ (|d_s - d_t|, T_{st}) \quad (11)$$

where $N(s)$ is the set of neighbors of the pixel s and $C = \{s, t | s < t, t \in N(s)\}$ is the set of all adjacent pixel pairs. $B = \{s, t | O_L(s) \neq O_L(t), s, t \in C\}$ is the set of discontinuities at the boundaries between occluded and unoccluded pixels. Here we consider the situation at the finest level (pixel level) in HBP because the occlusion map is generated only at this level. For other levels, all adjacent nodes will be taken into consideration to compute the smoothness terms. Note that T_{st} is the truncation point for the L_2 norm function. For each neighboring pair of nodes (segments), the truncation point is set such that pairs with large color differences have a small impact on the discontinuity cost, and pairs with small differences have a large impact. We use

$$T_{st} = max\left(T_{max} \cdot exp\left(-\frac{\|\mu_s - \mu_t\|^2}{2\sigma_c^2}\right), T_{min}\right) \quad (12)$$

where μ_s and μ_t are the mean color of nodes (segments) s and t. The parameter σ_c controls the influence of the segments' color difference. T_{min} is chosen to be a small value to ensure that each node has at least some influence on its neighboring nodes.

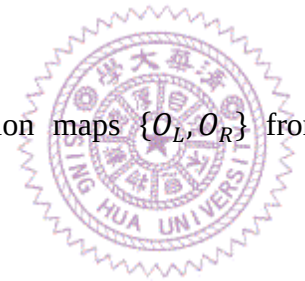
3.2.3 Inference procedure

This subsection describes the details of our inference process using the symmetric stereo models introduced in the previous subsections. The disparity maps are estimated using HBP [24] on our stereo CHMRF model.

At the first iteration, the value of occlusion $\{O_L, O_R\}$ are set to zeros, i.e. all pixels are initially unoccluded. Once the disparities have been estimated, we can obtain the corresponding occlusion maps by warping each pixel in the reference view to the target view according to its disparity. If the corresponding pixel in the target view exists, set the value of occlusion to 0, otherwise it is set to 1. We also apply the median filter to remove small noises on the occlusion maps. In summary, our itarative optimization algorithm alternates between the following two steps:

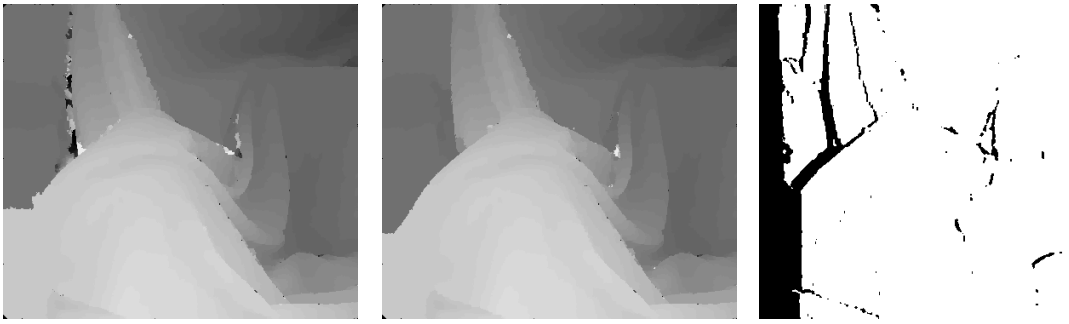
1. Estimate disparity maps $\{D_L, D_R\}$ with the given current occlusion maps $\{O_L, O_R\}$.

2. Compute the occlusion maps $\{O_L, O_R\}$ from the current disparity maps $\{D_L, D_R\}$.



The number of levels in HBP depends on the total level L_{tot} of our CHMRF structure, and we ran five message update iterations at each level.

In our experiments, two iterations of optimization is sufficient for obtaining reasonable disparity maps. Algorithm 2 summarizes the method in this step, and Figure 8 shows that our algorithm nicely fills in appropriate disparity values at the boundaries between occluded (black) and unoccluded (white) pixels.



(a) (b) (c)

Figure 8 Refine the disparity map with the occlusion map. (a) Initial disparity map. (b) Refined disparity map. (c) Corresponding occlusion map of (a).

Algorithm 2 Stereo matching algorithm

Input : An image pair I_L and I_R.
Corresponding CHMRF structure from Algorithm 1.

Output : Disparity maps D_L and D_R.

1: Set the parameters which will be used.
2: Initialize O_L and O_R to be zeros.
3: **For** t = 1 to 2 **do**
4: Construct the data cost volume according to Equation 8.
5: Estimate disparities using HBP to minimize the energy function defined in Equation 6.
6: Generate occlusion maps O_L and O_R.
7: **end for**

3.3 Refinement Process

Estimated disaprity maps may still have some errors in textureless regions because it is hard to find the actual correspondence. In these regions, we apply plane fitting to disparity maps. The basic idea is that each segment represents a 3D plane, so the disparity plane corresponding to a segmented region is expressed by the following function :

$$D(x, y) = a \cdot x + b \cdot y + c \tag{13}$$

where x and y are image coordinates, and a, b and c are plane parameters. The refinement procedure contains two major steps :

1. Color segmentation,
2. Plane fitting.

In the color segmentation step, we merge small segments from our over-segmentation result by adopting the mean-shift segmentation method [23] to our over-segmentation result of level 1. The first reason is based on the assumption – similar pixels are likely to have close disparity values; in other words, they are likely to belong to the same plane. The second reason is that the outliers in a small segment might dominate the final result of plane fitting.

In the plane fitting step, for each segment, we estimate these parameters by using "Iteratively Reweighted Least Squares (IRLS)" algorithm to solve the linear system defined as follows:

$$WAX = Wb$$

$$W = diag(w_1, w_{2,\dots,} w_n), A = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix}, X = \begin{bmatrix} a \\ b \\ c \end{bmatrix}, b = \begin{bmatrix} D(x_1, y_1) \\ D(x_2, y_2) \\ \vdots \\ D(x_n, y_n) \end{bmatrix}. \quad (14)$$

For each iteration, the diagonal matrix W is updated according to the current residuals as follows :

$$w_i = \frac{2\sigma^2}{2\sigma^2 + r_i^2}, i = 1 \sim n \quad (15)$$

where r_i is the residual computed from the currently estimated parameters :

$$r_i = (a \cdot x_i + b \cdot y_i + c) - D(x_i, y_i) \quad (16)$$

and σ is chosen to be $1.4826 \times r_{mid}$, where r_{mid} is the median of the sorted residuals.

After several iterations, the parameters of each plane have been estimated. Afterwards, the disparity value of each pixel belongs to ths same plane is updated by Equation (13). This algorithm is robust against some outliers because pixels with large residual values will have small weights. Algorithm 3 summarizes the plane fitting procedure.

Algorithm 3 Plane fitting procedure

Input : Segmented image *S* and disparity map *D*.

Output : Refined disparity map *D'*.

1: Initialize weighting matrix to be an identity matrix.
2: Construct matrix *A* and vector *b* according to *S* and *D*.
3: **For** t = 1 to T **do**
4: **For** each segment **do**
5: Solve the linear system defined in Equation 14.
6: **if** t < T **then**
7: Compute all the residuals according to Equation 16.
8: Update the weighting matrix according to Equation 15.
9: **else**
10: Output *D'* according to Equation 13.
11: **end if**
12: **end for**
13: **end for**

3.4 Efficiency Improvement

BP works by passing messages within the graph defined by direct-connected neighboring nodes. Each message is a vector of dimension the same as the total number of depth labels/values considered in the problem. Message computation at each iteration can be summarized as follows:

$$m_{pq}^{t}(f_q) = \min_{f_p}(V(f_p, f_q) + D_p(f_p) + \sum_{s \in N(p) \setminus q} m_{sp}^{t-1}(f_p)) \tag{17}$$

where m_{pq}^{t} is the message that node p sends to node q at time t. $D_p(f_p)$ is the cost of assigning label f_p to node p, which is referred to as data cost (see 3.2.1). $V(f_p, f_q)$ is the cost of assigning labels f_p and f_q to two neighboring nodes, which is normally referred to as the discontinuity cost (see 3.2.2).

After T iterations, a belief vector is computed for each node,

$$b_q(f_q) = D_q(f_q) + \sum_{s \in N(q)} m_{pq}^{T}(f_q). \tag{18}$$

And the label of each node is fixed, i.e.

$$f_q^* = arg\min_{f_q} b_q(f_q). \tag{19}$$

It is well-known that the standard BP algorithm is too slow to be used in practice. Felzenszwalb and Huttenlocher [24] proposed a hierarchical algorithm which runs much faster than the previous algorithms while maintaining comparable accuracy.

However, after the first iteration in our optimization step, we only modified the data cost of nodes which are occluded. According to Equation (17), if two adjacent nodes are both unoccluded, the value will be the same as that in the former iteration. Here we propose a scheme to inactivate some nodes if they are not occluded so that some redundant computation is saved while passing messages.

Chapter 4. Experimental Results

All of our experiments were performed on a PC equipped with Pentium(R) Dual-Core 2.50 GHz GPU and 2GB RAM.

4.1 Data Sets

We evaluate our algorithm on the Middlebury data sets. The data sets used in our experiments contain four rectified stereo image pairs, which are the *Tsukuba* image pair of size 384×288 with disparity range from 0 to 16, the *Venus* image pair of size 434×383 with disparity range from 0 to 20, the *Teddy* image pair of size 450×375 with disparity range from 0 to 60 and the *Cones* image pair of size 450×375 with disparity range from 0 to 60. Figure 10, 12, 14 and 17 show the disparity maps and over-segmentation results generated by the proposed method.

4.2 Parameter Settings

There are several parameters used in our algorithm. For the multi-level over-segmentation, L_{ini} is the initial block size, L_{tot} is the total level of this hierarchical structure, n_{sg} is the number of iterations to approximate geodesic distances in each

level. For HBP, η_o is the penalty for occlusion labeing, ($\sigma_{d,}$ e_d) are parameters of the robust function as defined in Equation (9), and λ_d is a constant weight factor applied to the data term after the robust function. The parameters ($\sigma_c, T_{max}, T_{min}$) determine the threshold used in smoothness term as defined in Equation (11). The number of iterations in HBP for each level is set to 5. Finally, plane fitting is employeed only for *Teddy* and *Cones* data sets to modify wrong disparity estimations which appeal in textureless regions.

Table 1 Parameter settings for different image pairs.

Image	**Multi-level Over-segmentation**				**HBP**							**Plane fitting**
	L_{ini}	L_{tot}	n_{sg}	λ_c	η_o	σ_d	e_d	λ_d	σ_c	T_{max}	T_{min}	
Tsukuba, Venus	1	3	2	1	0.0001	4	0.0001	0.7	12	8	0.9	N
Teddy, Cones	2											Y

4.3 Results and Evaluation

The stereo result on each data set is computed by measuring the percentage of pixels with an incorrect disparity estimate. This measure is computed for three subsets of the image:

- The subset of non-occluded pixels, denoted by “nonocc”.
- The subset of pixels being either non-occluded or half-occluded, denoted by

"all".

- The subset of pixels near the occluded regions, denoted by "disc".

Note that all occluded pixels are excluded from the evaluation. Take *Tsukuba* image pair for example: errors are only evaluated in the three different white regions as shown in Figure 9.

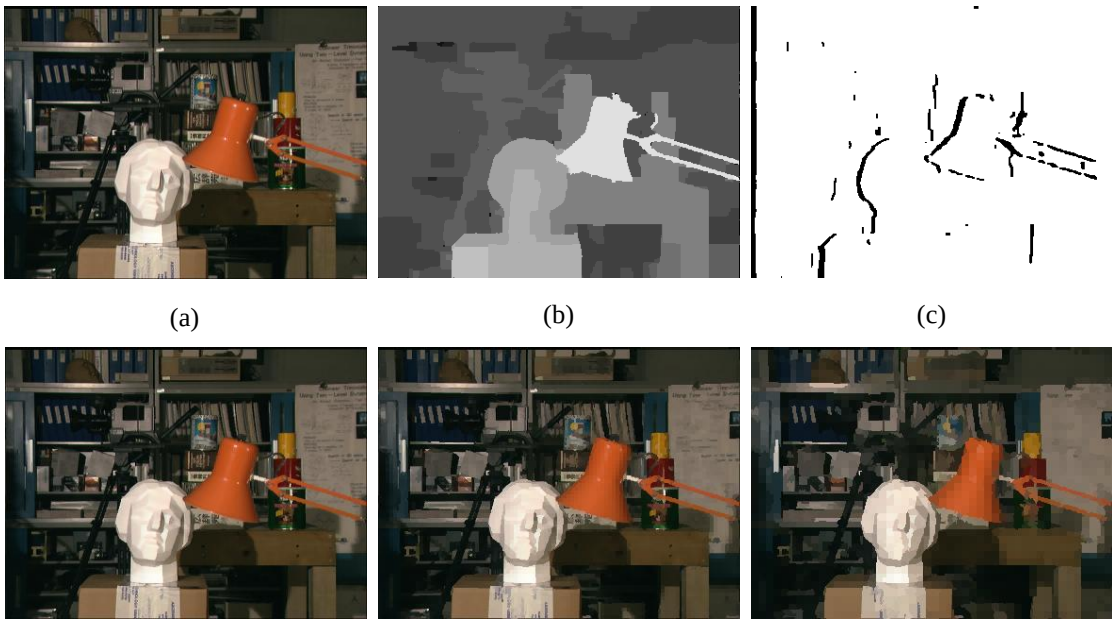
(a) (b) (c)

Figure 9 Three different regions for evaluation. (a) Non-occluded regions (white) and occluded and border regions (black). (b) All (including half-occluded) regions (white) and border regions (black). (c) Regions near depth discontinuities (white), occluded and border regions (black), and other regions (gray).

In Figure 11, 13, 16 and 19, we compare each of our disparity map for the left input image with the ground truth. If the absolute disparity error of a pixel is larger than 1, it will be marked “black”. The quantitative comparison is summarized in Table 2.

Table 2 Results on the Middlebury stereo evaluation (Threshold = 2).

Algorithm	*Tsukuba*			*Venus*			*Teddy*			*Cones*		
	non occ	all	disc	non occ	all	disc	non occ	all	disc	non occ	all	disc
Proposed method	1.40	1.76	5.77	0.23	0.34	2.35	5.12	10.5	12.3	5.11	10.9	13.2
AdaptOvrSegBP [8]	1.20	1.44	4.36	0.12	0.15	1.33	4.07	6.16	10.0	2.13	6.96	6.12
Segm+visib [20]	1.14	1.37	6.09	0.35	0.48	4.16	2.39	3.82	6.35	2.36	6.70	6.89
OutlierConf [22]	0.74	1.23	4.00	0.15	0.20	2.13	2.05	3.89	6.12	1.89	6.80	5.15
CoopRegion[6]	0.77	1.00	4.14	0.11	0.18	1.53	2.14	3.41	6.61	2.10	5.95	6.24



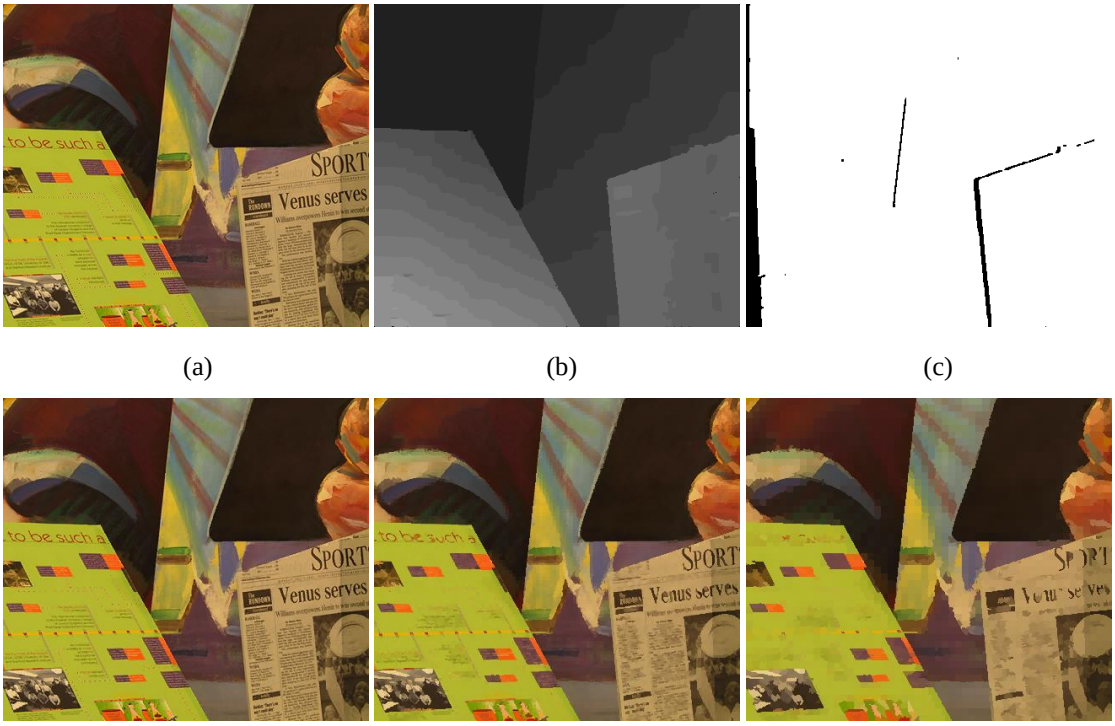
(a) (b) (c)

(d) (e) (f)

Figure 10 Experimental results for the *Tsukuba* image pair. (a) Reference image. Disparity map and (c) occlusion map determined by the proposed algorithm. (d-f) Over-segmentation results from level1 to level 3.

(a) (b) (c)

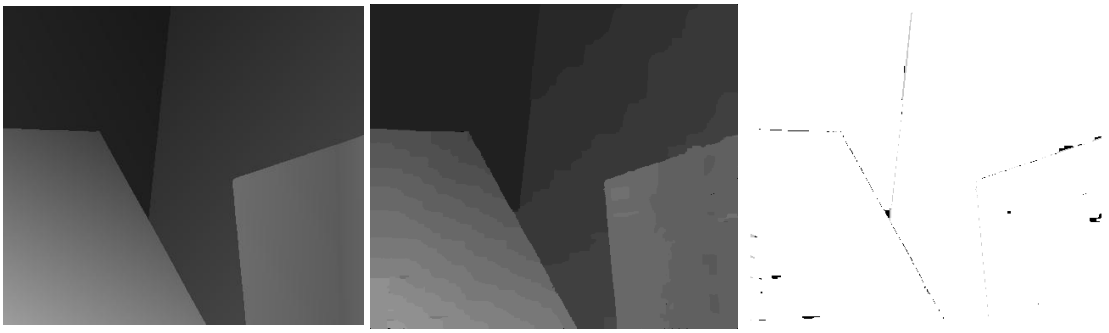
Figure 11 Evaluation results. (a) Ground truth. (b) The disparity map estimated by the proposed algorithm. (c) Black pixels indicate that the absolute disparity error is larger than 1.



(a) (b) (c)

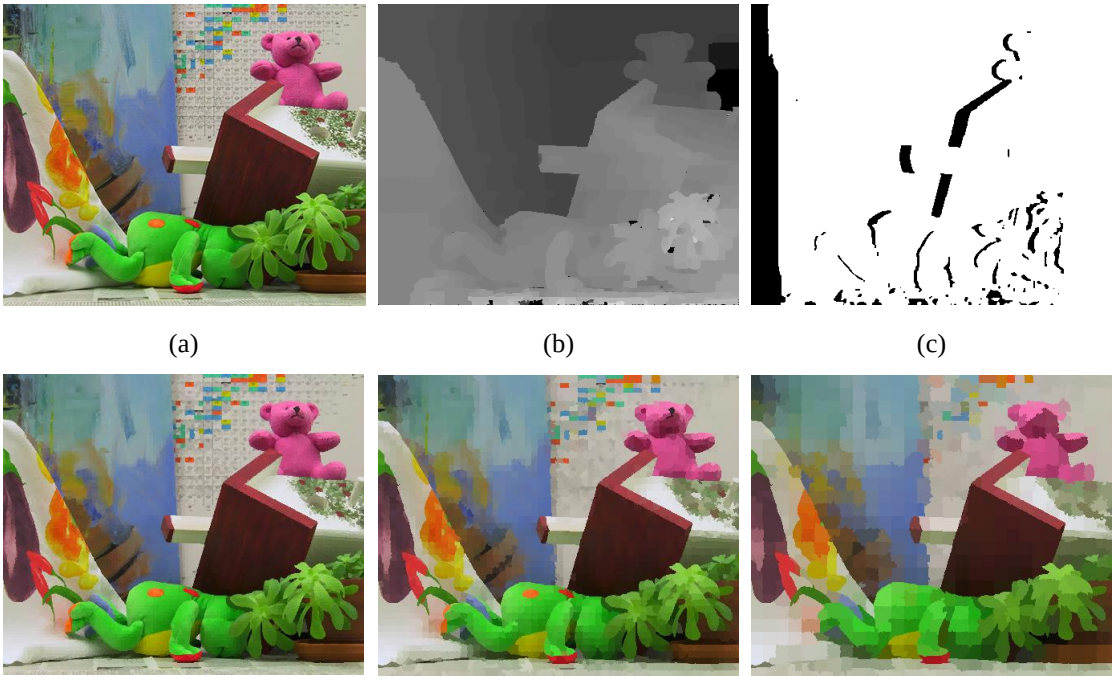
(d) (e) (f)

Figure 12 Experimental results for the *Venus* image pair. (a) Reference image. (b) Disparity map and (c) occlusion map determined by the proposed algorithm. (d-f) Over-segmentation results from level1 to level 3.



(a) (b) (c)

Figure 13 Evaluation results. (a) Ground truth. (b) The disparity map estimated by the proposed algorithm.. (c) Black pixels indicate the absolute disparity error is larger than 1.



(a) (b) (c)

(d) (e) (f)

Figure 14 Experimental results for the *Teddy* image pair. (a) Reference image. (b) D Disparity map and (c) occlusion map determined by the proposed algorithm. (d-f) Over-segmentation results from level1 to level 3.

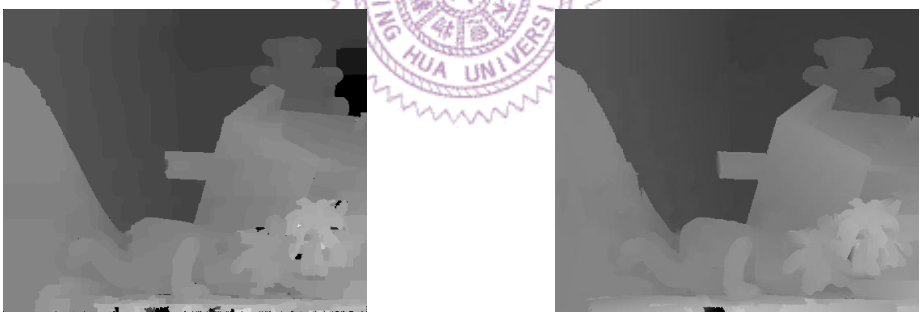
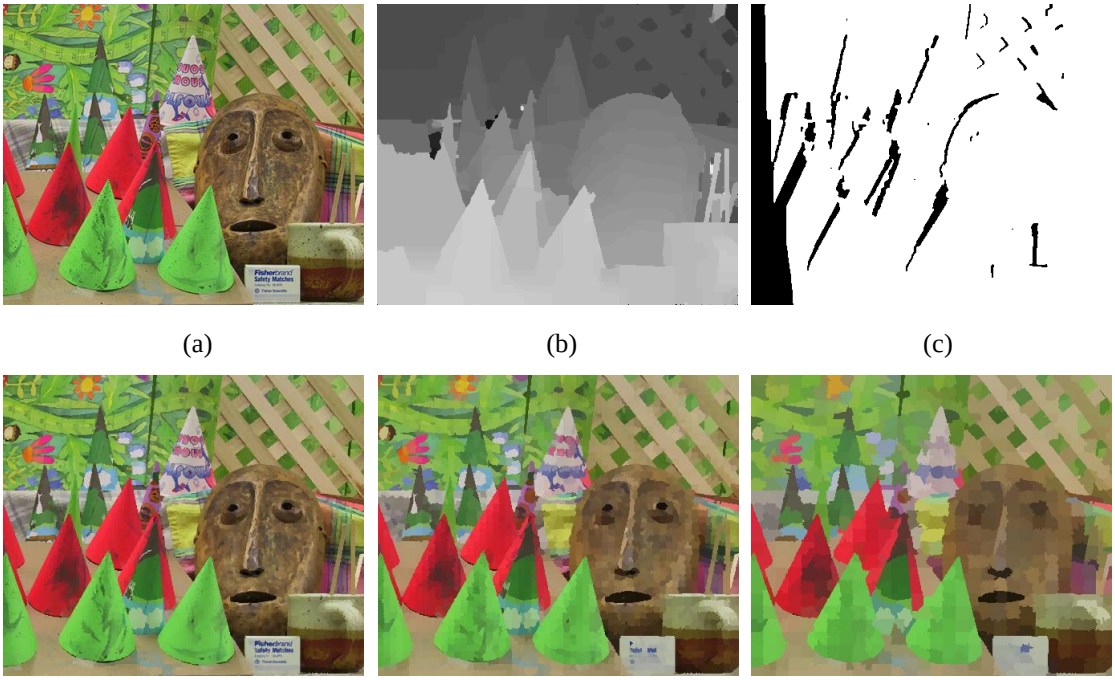


Figure 16 Left and right are disparity maps before and after plane fitting, respectively.

(a) (b) (c)

Figure 15 Evaluation results. (a) Ground truth. (b) The disparity map estimated by the proposed algorithm. (c) Black pixels indicate the absolute disparity error is larger than 1.

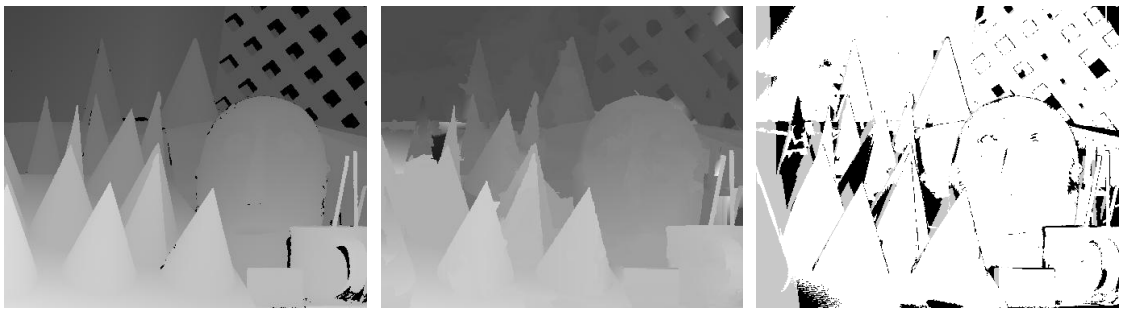


(a) (b) (c)

(d) (e) (f)

Figure 17 Experimental results for the *Cones* image pair. (a) Reference image. (b) Disparity map and (c) occlusion map determined by the proposed algorithm. (d-f) Over-segmentation results from level1 to level 3.

Figure 18 Left and right are disparity maps before and after plane fitting, respectively.



(a) (b) (c)

Figure 19 Evaluation results. (a) Ground truth. (b) The disparity map estimated by the proposed algorithm. (c) Black pixels indicate the absolute disparity error is larger than 1.

4.4 Execution Time

Table 3 shows the execution time of our method for the four different image pairs, which is divided into three parts. The first part is the total run time of constructing the CHMRF structure, including geodesic distances computation, over-segmentation and so on. The second part is the elapsed time of the stereo matching. The last part includes the execution time of segmentation and plane fitting if the refinement process is included into the algorithm.

Table 3 The execution time of each step for four different image pairs.

Image	Multi-level Over-segmentation	Stereo matching	Plane fitting	Total
Tsukuba	5.656 s	7.812 s	-	13.568 s
Venus	8.532 s	14.156 s	-	22.688 s
Teddy	6.829 s	28.625 s	4.906 s	40.360 s
Cones	7.078 s	29.422 s	4.454 s	40.954 s

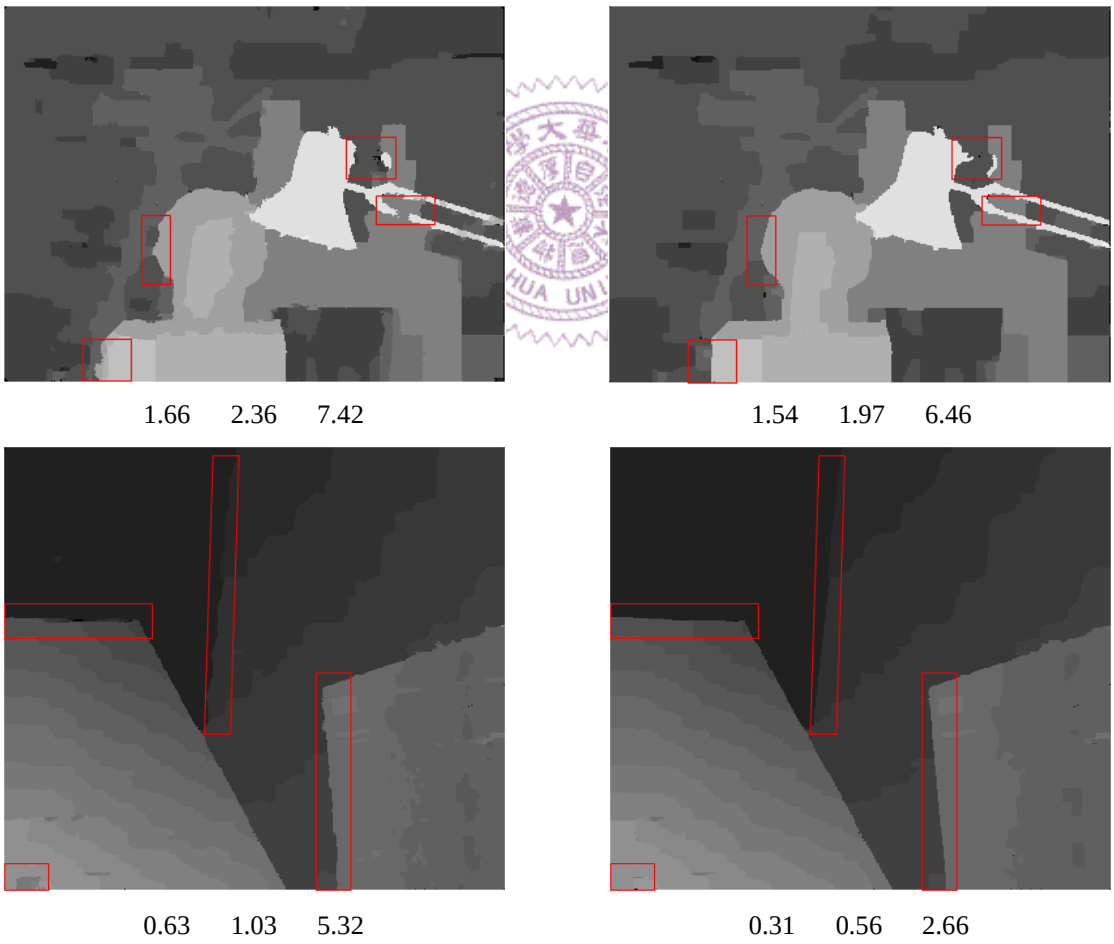
We also analyze the computation time for the stereo matching step on *Tsukuba* data set. Constructing the cost volume costs about 15% of the total runtime. And over 70% of the time is used to execute HBP algorithm. Especially in level 0, this part is the main factor that reduces the whole efficiency.

Table 4 Computation time analysis of stereo matching on *Tsukuba* image pair

Image	Cost volume construction	Belief Propagation				Others
		level 3	level 2	level 1	Level 0	
Tsukuba	1.187 s	0.063 s	0.266 s	1.064 s	4.219 s	1.013 s
	15.2%	0.8%	3.4%	13.6%	54%	13%

4.5 Verification of CHMRF Structure

For the consideration of efficiency, Felzenszwalb and Huttenlocher [24] showed that HBP computes a low energy solution in just a few iterations per level, while the standard algorithm takes hundreds of iterations to obtain a similar result. Since our CHMRF structure is similar to theirs, we can also reach a lower energy faster than other segment-based algorithms which used standard belief propagation algorithm based on single-scale MRF graph structure.



1.66 2.36 7.42 1.54 1.97 6.46

0.63 1.03 5.32 0.31 0.56 2.66

Figure 20 Comparison of stereo results using two different MRFs. Left column used the standard one, right column used the proposed CHMRF. The numbers under images are the percentages of bad pixels evaluated in three different regions as mentioned before.

We compare the stereo matching results on *Tsukuba* and *Venus* data sets using the same energy formulation with the standard hierarchical MRF and the proposed CHMRF graphical models as shown in Figure 20. Our disparity maps contain fewer bad pixels and keep more complete structure and sharper edge in disparity discontinuity regions, as shown inside the red rectangles in Figure 20.

4.6 Efficiency Improvement

Table 5 shows the comparison of execution time in our stereo matching step before and after efficiency improvement. The average speed-up factor on the four image pairs is 1.36. Table 6 further reveals that we improved the efficiency of BP in level 0 on *Tsukuba* image pair with speed-factor 1.86.

Table 5 Computational efficiency of stereo matching algorithm with and without speed-up

Image	Efficiency improvement scheme		Speed-up Factor
	No	Yes	
Tsukuba	7.891 s	5.922 s	1.33
Venus	14.156 s	10.390 s	1.36
Teddy	28.625 s	20.656 s	1.38
Cones	29.422 s	21.375 s	1.37

Table 6 Computational efficiency of BP in level 0 on *Tsukuba* image pair with and without speed-up

Image	Efficiency improvement scheme		Speed-up Factor
	No	Yes	
Tsukuba	4.219 s	2.265 s	1.86

4.7 Real Datasets

In this subsection, we perform the proposed stereo matching algorithm on real videos. In the initial step, we utilize a self image rectification algorithm proposed by Cheng *et al.* [32] to rectify the stereo images for un-calibrated stereo video sequences with temporally varying camera motions and zooming in/out effects. Then, we estimate the disparity maps by using the proposed algorithm. Finally, we apply the bilateral filter to reduce some noise on stereo results. The sampled results of our stereo system are shown in Figure 21.

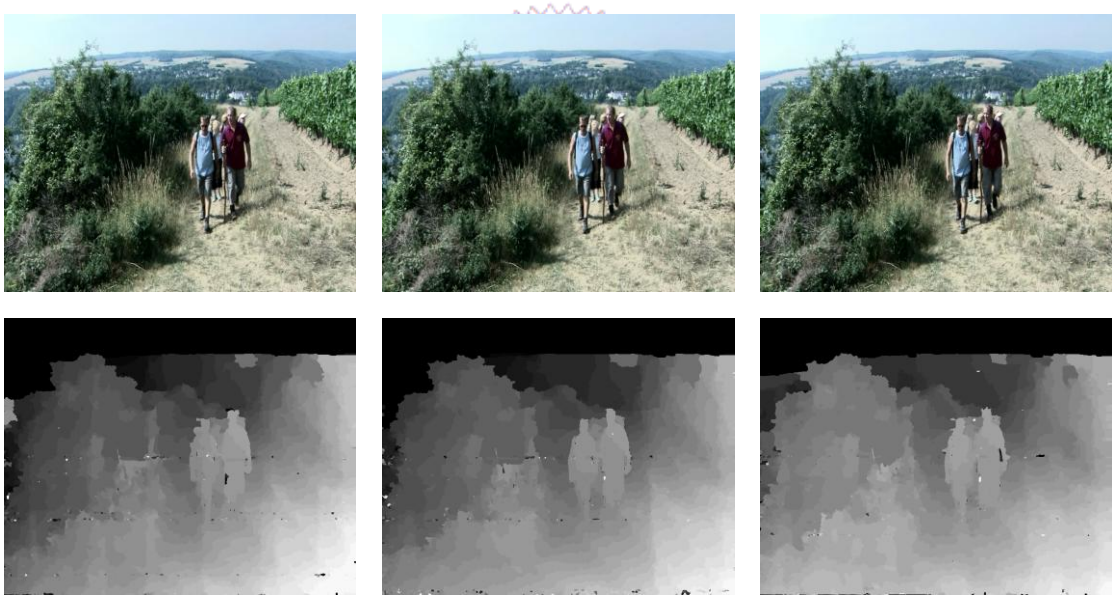


Figure 21 Stereo results on real video. The first row depicts the original input images. The second row shows the estimated disparity maps.

Chapter 5. Conclusion

5.1 Summary

In this thesis, we propose a segmentation-based stereo algorithm on our CHMRF graphical model. The CHMRF is constructed via computing the image geodesic distance. For stereo matching, we perform HBP on this graph with defined energy to estimate disparity maps of image pair. To eliminate some disparity errors, we apply the plane fitting technique based on iteratively reweighted least squares estimation. Our experimental results show the proposed algorithm can provide acceptable depth estimation and multi-level segmentation results efficiently. The proposed CHMRF is suitable for other problems that can be modeled as a graph labeling problem in MRF.

The main limitation of our algorithm is that the energy formulation is a little simple, since we want to estimate disparities in a short time. However, more clues or constraints can help to obtain more accurate disparity maps, but it will take more computational time. This is a trade-off between efficiency and accuracy. This thesis tries to find a good balance for the stereo matching problem.

5.2 Future Work

We plan to implement the major parts of our stereo system on GPU to significantly improve the computational performance as the future research direction. For the hierarchical over-segmentation step, our approach and implementation to establish this structure is very suitable for realization in parallel. This is because the geodesic distance computation and clustering steps can be executed for each node individually. In the stereo matching step, there have been many works proposed to improve the efficiency of HBP by taking advantage of GPU.

Except for efficiency improvement, we plan to modify the structure of our CHMRF. Our current graph is restricted to be 4-connected, this will sacrifice the accuracy of segmentation results in some way. A possible modification is to employ over-segmentation scheme via computing geodesic distance for each level first, and then we construct the MRF like [8] but with a hierarchical relationship between levels. This should generate more accurate depth estimation and keep the high efficiency at the same time.

References

[1] D. Scharstein and R. Szeliski. “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms”. In *IJCV*, vol. 47, no. 1, pp. 7-42, 2002.

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

[3] J. Sun, H. Y. Shum, and N. N. Zheng. “Stereo matching using belief propagation”. In A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors, *Computer Vision* - ECCV 2002, number 2351 in Lecture Notes in Computer Science, pages 510–524, 2002.

[4] Y. Boykov, O. Veksler, and R. Zabih. “Fast Approximate Energy Minimization via Graph Cuts”. In IEEE Trans. *PAMI*, 23(11), Nov. 2001.

[5] A. Klaus, M. Sormann and K. Karner. “Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure”. In *ICPR* 2006.

[6] Z. Wang and Z. Zheng. “A region based stereo matching algorithm using cooperative optimization”. In *CVPR* 2008.

[7] Q. Yang, L. Wang, R. Yang, H. Stewnius, and D. Nistér. “Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling”. In *TPAMI*, 31(3):492–504, 2009.

[8] Y. Taguchi, B. Wilburn and C. L. Zitnick. “Stereo Reconstruction with Mixed Pixels Using Adaptive Over-Segmentation”. In *CVPR* 2008.

[9] J. Sun, Y. Li, S. B. Kang and H.-Y. Shum. “Symmetric stereo matching for occlusion handling”, In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. II:399-406, 2005.

[10] L. Zitnick and S.B. Kang. “Stereo for image-based rendering using image over-segmentation”. In *IJCV* 2007.

[11] S. Mattoccia. “A locally global approach to stereo correspondence”. In *3DIM* 2009.

[12] K.-J. Yoon and I.-S. Kweon. “Adaptive support-weight approach for correspondence search”. In *PAMI* 28(4):650-656, 2006.

[13] H. Hirschmüller. “Accurate and efficient stereo processing by semi-global matching and mutual information”. In *PAMI* 30(2):328-341, 2008.

[14] S. Kosov, T. Thormählen, and H.-P. Seidel. “Accurate real-time disparity estimation with variational methods”. In *ISVC* 2009.

[15] K. Zhang, J. Lu, and G. Lafruit. “Cross-based local stereo matching using orthogonal integral images”. In *IEEE TCSVT* 2009.

[16] V. Kolmogorov and R. Zabih. “Computing visual correspondence with occlusions using graph cuts”. In *ICCV* 2001.

[17] T. kanade and M. Okutomi. “A stereo matching algorithm with an adaptive window: Theory and experiment”. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, P. 920 September 1994.

[18] H. Tao, H. S. Sawhney, and R. Kumar. “A global matching framework for stereo computation”. In *ICCV*, Vol I:532–539, 2001.

[19] L. Hong and G. Chen. “Segment-based stereo matching using graph cuts”. In *CVPR*, Vol I:74–81, 2004.

[20] M. Bleyer and M. Gelautz. “A layered stereo matching algorithm using image segmentation and global visibility constraints”. In *ISPRS Journal*, 59(3):128–150, 2005.

[21] A. Klaus, M. Sormann, and K. Karner. “Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure”. In *ICPR*, pages

15–18, 2006.

[22] L. Xu and J. Jia. “Stereo matching: An outlier confidence approach". In *ECCV*, 2008.

[23] C. Christoudias, B. Georgescu, and P. Meer. ”Synergism in low-level vision”. In *ICPR*, volume 4, pages 150–155, 2002.

[24] P. Felzenszwalb and D. Huttenlocher. “Efficient belief propagation for early vision”. In *IJCV*, 70(1):41–54, 2006.

[25] T. Yu, R. S. Lin, B. J. Super, and B. Tang, “Efficient message representations for belief propagation”. In *ICCV*, 2007, pp. 1–8.

[26] Q. Yang, L. Wang, R. Yang, S. Wang, M. Liao and D. Nistér. “Real-time global stereo matching using hierarchical belief propagation”. In *BMCV* 2006.

[27] C. Liang, C. Cheng, Y. Lai, L. Chen and H. Chen. “Hardware-efficient belief propagation”. In *CVPR* 2009.

[28] Q. Yang, L. Wang and N. Ahuja. “A constant-space belief propagation algorithm for stereo matching”. In *CVPR* 2010.

[29] G. Borgefors. “Distance transformations in digital images”. In *Computer Vision, Graphics and Image Processing*, 34(3):344–371, 1986.

[30] A. Criminisi, T. Sharp, and A. Blake. “Geos: Geodesic image segmentation”. In *ECCV*, pages 99–112, 2008.

[31] S. Birchfield and C. Tomasi. “A pixel dissimilarity measure that is insensitive to image sampling”. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 4, April 1998.

[32] C.-M. Cheng, S.-H. Lai, and S.-H. Su. “Self Image Rectification for Un-calibrated Stereo Video with Varying Camera Motions and Zooming Effects”. In *MVA09*, Japan, May 20-22, 2009.