



MARMARA UNIVERSITY  
INSTITUTE FOR GRADUATE STUDIES



# STEREO MATCHING BASED ON LOCAL AND GLOBAL AGGREGATION

---

---

MAKSAT YLYASOV

**MASTER THESIS**

Department of Electrical and Electronics Engineering

**Thesis Supervisor**

Prof. Dr. Cabir Vural

ISTANBUL, 2020

---

---



MARMARA UNIVERSITY  
INSTITUTE FOR GRADUATE STUDIES



# STEREO MATCHING BASED ON LOCAL AND GLOBAL AGGREGATION

---

---

MAKSAT YLYASOV

525014901

**MASTER THESIS**

Department of Electrical and Electronics Engineering

**Thesis Supervisor**

Prof. Dr. Cabir Vural

ISTANBUL, 2020

---

---

**MARMARA UNIVERSITY**  
**INSTITUTE FOR GRADUATE STUDIES IN**  
**PURE AND APPLIED SCIENCES**

Maksat YLYASOV, a Master of Science student of Marmara University Institute for Graduate Studies in Pure and Applied Sciences, defended his thesis entitled "Stereo matching based on local and global aggregation", on 08/01/2020 and has been found to be satisfactory by the jury members.

**Jury Members**

Prof.Dr. (Advisor)  
Marmara University ..... Cabir VURAL .....(SIGN) *Cabir Usel*

Assoc.Prof. Dr. (Jury Member)  
Marmara Üniversitesi ..... Küşat AYAN .....(SIGN) *Kuşat*

Assist.Prof. (Jury Member)  
Marmara Üniversitesi ..... Mustafa ONAT .....(SIGN) *Mustafa*

**APPROVAL**

Marmara University Institute for Graduate Studies in Pure and Applied Sciences Executive Committee approves that Maksat YLYASOV be granted the degree of Master of Science in Department of Electrical and Electronics Engineering Program on 22.01.2020. (Resolution no: 2020/03-02).

  
Director of the Institute  
Prof. Dr. Bülent EKİCİ



## **Acknowledgment**

I would first like to thank my thesis supervisor Prof. Dr. Cabir Vural for his support from day one till the end of my master journey. I learned what it really takes to be an academic researcher that the only way is through hard working, dedication, and commitment. Most importantly thanks for giving opportunity and helping me work on a topic which I really like. Finally, I must express my very profound gratitude to my family and especially my mother for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

# İÇİNDEKİLER / CONTENTS

Acknowledgment.....	i
ABSTRACT .....	vi
KISALTMALAR / ABBREVIATIONS .....	viii
ŞEKİL LİSTESİ / LIST OF FIGURES .....	ix
TABLO LİSTESİ / LIST OF TABLES.....	x
CHAPTER 1 INTRODUCTION .....	1
CHAPTER 2 RELATED WORKS AND BACKGROUND INFORMATION .....	4
2.1. Related studies .....	4
2.2. Principles of stereo matching.....	6
2.2.1. Epipolar geometry .....	7
2.2.2. Image pair rectification and triangulation .....	8
2.3. Matching cost functions.....	9
2.3.1. Truncated absolute difference (TAD).....	10
2.3.2. Birchfield and Tomasi's Dissimilarity Measure.....	11
2.3.3. Census Transform measure .....	12
2.4. Local and Global stereo methods.....	12
2.4.1. Image guided filtering cost aggregation .....	15
2.4.2. Segment tree filtering cost aggregation .....	17
2.4.3. GC stereo matching algorithm.....	20
2.4.4. Non-local cost aggregation .....	25
2.5. Learning of confidence measure .....	28
2.5.1. Matching Cost .....	29
2.5.2. Distance from Border (DB) .....	29
2.5.3. Maximum Margin (MM).....	29

2.5.4.	Attainable Maximum Likelihood (AML).....	29
2.5.5.	Left-Right Consistency (LRC) .....	30
2.5.6.	Left-Right Difference (LRD).....	30
2.5.7.	Distance from Discontinuity (DD) .....	30
2.5.8.	Difference with Median Disparity (MED) .....	31
CHAPTER 3 PROPOSED METHOD .....		32
3.1.	Mixture of local and global methods .....	32
3.2.	Ground control points added matching cost volume .....	37
3.3.	Energy function formulation.....	38
3.4.	Energy function minimization .....	40
3.5.	Post-processing (Disparity refinement) .....	42
3.5.1.	Hole filling (Densification process) .....	42
3.5.2.	Constant Time Weighted Median Filtering .....	43
3.5.3.	Left-right consistency check.....	44
CHAPTER 4 SIMULATION RESULTS AND DISCUSSION .....		45
CHAPTER 5 CONCLUSIONS AND FUTURE WORKS .....		62
REFERENCES .....		63
ÖZGEÇMİŞ.....		67

## ÖZET

### GLOBAL VE YEREL TOPLAMA YÖNTEMLERİNE DAYALI STEREO GÖRÜNTÜ İŞLEME

Stereo eşleştirme algoritmaları iki sınıfa ayrılabilir: yerel yöntemler ve global yöntemler. Global yöntemler düzgünlük varsayımına dayanmaktadır ve eşleşme haritası kestirimini enerji en küçükleme çatısı olarak ifade etmektedir. Yerel yöntemler, bir aday kümesinden en düşük eşleşme maliyetine sahip adayı seçerek derinlik haritasını kestirmektedir. Yerel yöntemler, yüksek frekanslı desen bölgelerinde iyi sonuç verirken kapanma oluşan bölgelerde güvenilir sonuçlar sağlamamaktadırlar. Global yöntemler, kalite bakımından yerel yöntemlere göre daha doğru sonuç üretmektedirler. Ancak, global yöntemler genelde yüksek hesap yüküne sahiptirler.

Yakın bir geçmişte stereo eşleşme için uzmanların bir karışımı yaklaşımı tanıtılmıştır. Yöntem, uyarlanabilir karıştırma katsayılı yeni bir yerel filtre oluşturmak için farklı filtreleri birleştirmektedir. Eşleşme hatasını azaltmada etkin olduğu gösterilmiştir. Ancak, yöntemin çeşitli eksiklikleri vardır. İlk olarak, çalışma farklı parametrelili görüntü kılavuzlu ve ağaç filtreleri kullanmaktadır. İkinci olarak, son işleme problemi ele alınmamaktadır. Bu nedenle, yöntem aykırı değerlerden etkilenmektedir. Üçüncüsü, yöntem yerel filtre havuzuna global bir yöntem eklenmesine izin vermemektedir. Yüksek doğruluklu sonuçlar genelde global yöntemler arafından sağlanmaktadır. Bu tezde, bu sınırlamaların gidermek için fikirler geliştirilmiştir.

Tezin temel amacı, Middlebury ve KITTI Vision Benchmark gibi iyi bilinen ve yaygın olarak kullanılan stereo görüntü veritabanlarında düşük ortalama stereo eşleşme hata oranı elde etmektir. Belirli bir parametre ayarına sahip bir filtre bir görüntü çifti için iyi sonuç verme potansiyeline sahip olabilir, ancak diğer görüntü çiftleri için yeterli performans sağlamayabilir. Bu sorunun üstesinden gelebilmek için, maliyet hacmi üzerinde türdeş olmayan filtre setinin uygulandığı ve sonuçların uyarlanabilir şekilde birleştirildiği bir uzmanlar karışımı modeli önerilmektedir. Özellikle kapanma oluşan bölgelerde global yöntemler daha iyi sonuç verme eğiliminde olduklarından, yerel filtreler havuzuna global bir metodu eklendiğinde iyileştirilmiş eşleştirme sonuçları elde etmeyi bekleriz. Ortalama eşleşme hatasını en küçükleme için son işleme de yöntemde dahil edilmiştir.

## **ABSTRACT**

### **STEREO MATCHING BASED ON LOCAL AND GLOBAL AGGREGATION**

Stereo matching algorithms can be divided into two major groups: global methods and local methods. Global methods rely on smoothness assumption and formulate the disparity map estimation as an energy-minimization framework. Local methods estimate the depth map by selecting the candidate with the smallest matching cost from a set of candidates. Local methods handle high-frequency texture areas well while they fail to deliver reliable results in occluded regions. In terms of quality, global methods generate more accurate results compared to local methods. However, global methods usually have expensive computation cost.

Recently, a mixture-of-experts approach for stereo matching have been introduced. The method combines different filters to produce a novel local filter with adaptive mixing coefficients. It was shown to be effective for reducing the matching error. However, the method has several limitations. First, it uses only image-guided and tree filters with different parameters. Second, the issue of post-processing is not handled. For this reason, the method suffers from outliers. Third, it does not allow to add global filter to the pool of local filters. High accuracy results are usually provided by global techniques. In this thesis, we will develop ideas to overcome these limitations.

The main aim of the thesis is to achieve low average stereo matching error rate in well known and widely used stereo image pairs datasets like Middlebury and KITTI Vision Benchmark. A specific filter with a specific parameter setting may have a potential to work for an image pair, but may not provide satisfactory performance for other image pairs. To overcome this issue, a mixture-of-experts model in which a heterogeneous set of filters on the cost volume is applied and the results are adaptively combined is proposed. By adding a global filter to the pool of local filters, we expect to get improved matching results since global methods tend to give better results especially in the occluded areas. Postprocessing is also included to minimize average matching error.

## SEMOLLER/SYMBOLS

$E$	: Energy function
$C(x,y,d)$	: Cost volume
$\widehat{C}(x,y,d)$	: Combined cost volume
$g^m(v_{x,y})$	: Model mixing weight
$C^{-m}(x,y,d)$	: Aggregated cost volume
$f(v_{x,y})$	: Classifier output function
$V_{x,y,u,v}$	: Potential function
$v^h$	: Feature vector
$\tau_1, \tau_2$	: Truncation values

## KISALTMALAR / ABBREVIATIONS

<b>AD</b>	: Absolute Difference
<b>BMP</b>	: Bad Matching Pixels
<b>DME</b>	: Disparity Map Estimation
<b>GL</b>	: Graph Cut Optimization Stereo Model
<b>GCP</b>	: Ground Control Point
<b>GF</b>	: Guided Image Filter Stereo Model
<b>GT</b>	: Ground Truth
<b>LRC</b>	: Left Right Consistency Check
<b>MRF</b>	: Markov Random Field
<b>MST</b>	: Minimum Spanning Tree
<b>NCC</b>	: Normalized Cross-Correlation
<b>RGB</b>	: Red, Blue and Green Color Channel
<b>ST</b>	: Segment Tree Stereo Model
<b>TAD</b>	: Truncated Absolute Difference
<b>TF</b>	: Tree Filtering
<b>WMF</b>	:Weighted Median Filter
<b>WTA</b>	: Winner Takes All optimization

## ŞEKİL LİSTESİ / LIST OF FIGURES

<b>Figure 1.1.</b> Flowchart corresponding to the proposed method .....	3
<b>Figure 2.1.</b> Epipolar geometry.....	7
<b>Figure 2.2.</b> Triangulation for estimating the distance from observer .....	8
<b>Figure 2.3.</b> Importance of cost aggregation in disparity estimation. The top right image is the matching cost distribution without aggregation. Bottom images show the cost distribution for different aggregations including non-local, segment tree and image-guided filtering .....	14
<b>Figure 2.4.</b> Disparity estimation based on the guided filtering .....	15
<b>Figure 2.5.</b> Segment tree construction.....	19
<b>Figure 2.6.</b> One example for stereo image pair. ....	20
<b>Figure 2.7.</b> A simple example of network flow. (a) A network with max capacities. (b) For any node (except sink and source nodes) in the network, the incoming flow has to be equal to outgoing flow .....	24
<b>Figure 2.8.</b> Energy minimization algorithm .....	25
<b>Figure 2.9.</b> Cost aggregation using MST. (a) MST is aggregated from leaf to root, (b) aggregation is repeated from root to leaf.....	27
<b>Figure 3.1.</b> Graph cut expansion move algorithm .....	41
<b>Figure 4.1.</b> Results for Baby3, Cloth1, Lampshade1 and Flowerpots a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF,ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs. ....	51
<b>Figure 4.2.</b> Results for Lampshade2, Monopoly, Plastic and Rocks1 a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF,ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs.....	53
<b>Figure 4.3.</b> Results for Rocks2, Woods2, Books and Reindeer a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF,ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs. ....	55
<b>Figure 4.4.</b> Disparity maps obtained by different methods for Aloe image pair (a) left image, (b) the true disparity map, (c) through (f) disparity maps obtained by GF (32,50), ST (35,63), GL (38,66) and the proposed (32,58) methods, respectively. Numbers in brackets are percentage of BMPs.....	57
<b>Figure 4.5.</b> Disparity maps obtained by different methods for Baby2 image pair (a) left image, (b) the true disparity map, (c) through (f) disparity maps obtained by GF (28,74), ST (27,43), GL (33,08) and the proposed (28,44) methods, respectively. Numbers in brackets are percentage of BMPs. ....	58
<b>Figure 4.6.</b> Average classification error per method for Aloe and Baby2 image pairs. ....	59

## TABLO LİSTESİ / LIST OF TABLES

<b>Table 4.1</b> Average percentage of BMPs of Aloe, Baby2 and Teddy, (left images) with post-processing (WMF) (r: kernel size of WMF) .....	46
<b>Table 4.2</b> Overall error (left disparity) per method number for Baby2, Aloe and Teddy stereo pairs .....	46
<b>Table 4.3</b> Average error rate (%) of 17 image pairs from Middlebury stereo dataset provided by different methods .....	47
<b>Table 4.4</b> Error rates of GF, ST, GL and the proposed method for several stereo pairs .....	48
<b>Table 4.5</b> Error rates of the proposed method with GCPs are added for several stereo pairs ....	49
<b>Table 4.6</b> Error rates (left disparity) for Aloe, Baby2 and Teddy image pairs in occluded regions without post-processing.....	60
<b>Table 4.7</b> Error rates (left disparity) for Teddy and Baby2 image pairs in occluded regions without post-processing.....	61

## CHAPTER 1 INTRODUCTION

Stereo matching is used in many applications such as 3D reconstruction, robot navigation, driver assistance systems, etc. The key problem in stereo matching is depth map (also referred to as disparity map) estimation (DME). Left and right images of a scene are obtained by using a pair of cameras. If the cameras are arranged in a parallel configuration, then a point in a left image corresponds to a point in the same scan line in the right image. Given a point in the left image, finding its correspondence in the right image is called *disparity estimation*. Once disparity is known, the depth map can be constructed by using triangulation. The common goal of DME algorithms is to provide an acceptable estimation error for a given application. However, ambiguity arises in occluded and textureless regions. The accuracy of a DME method depends on many factors such as existence of occlusion, lack of texture, illumination of the scene whose depth map is required. Another bottleneck for DME algorithms is computational complexity. In order for a DME algorithm to be applicable in practice, it should have low computational complexity.

In the last few decades, several stereo matching algorithms have been developed. They can be divided into two major groups: global and local methods. Global methods rely on smoothness assumption and formulate the DME as an energy-minimization framework. On the other hand, local methods estimate the depth map by selecting the candidate with the smallest matching cost from a set of candidates. This process is known as winner-take-all (WTA) optimization. Thus, cost aggregation is the most important step in local stereo matching algorithms.

In this thesis, three contributions are made to the stereo matching research: (i) global stereo matching method is integrated into a general mixture of experts model, (ii) the most optimal post-processing is determined, (iii) diversity of methods or filters is increased and its effect is investigated. A general overview of the proposed method is given Figure 1.1. Initially, global stereo matching is computed [1], then confidence measures are extracted [2-4], final disparity map is obtained from the previous method. As in [2-4] we intend to create datasets by implementing Random Forrest classifier [5] or any other 2 class classifier. The reason why we are creating dataset is to calculate mixing coefficients [6]. After computing mixing coefficients of a global method, the next step is

how to integrate the global method into the cost aggregation over models framework proposed in [6]. For this purpose, cost aggregation is done over models. Output of global methods is a two-dimensional disparity map with final selected disparity values. On the other hand, output of a local method is three-dimensional cost volume (the third dimension being disparity range). In order to integrate local and global methods, output data of global method is converted into a three-dimensional cost volume. Furthermore, instead of using WTA strategy, Graph Cut (GC), a global optimization method, is used as shown in Figure 1.1. As a result, we need to make sure before optimization stage that all the terms in the objective (or cost) function are three-dimensional. Usually, maximum disparity range is assumed to be 60. Cost aggregation term generated from local methods has a dimension of  $Image_{height} \times Image_{width} \times Disparity\ Range$ , while that obtained from global methods has a dimension of  $Image_{height} \times Image_{width} \times 1$ . First, we calculate matching cost of a pixel for each disparity value in disparity map of the global method. Then, the remaining matching costs in three-dimensional cost volume are set to some higher constant level. The reason for doing so is that during the optimization stage GC will throw out the values that are set to high value and select the most optimal disparity level.

Main steps of the proposed method is summarized below;

*Matching cost selection:* it is the first step almost in all stereo matching algorithms.

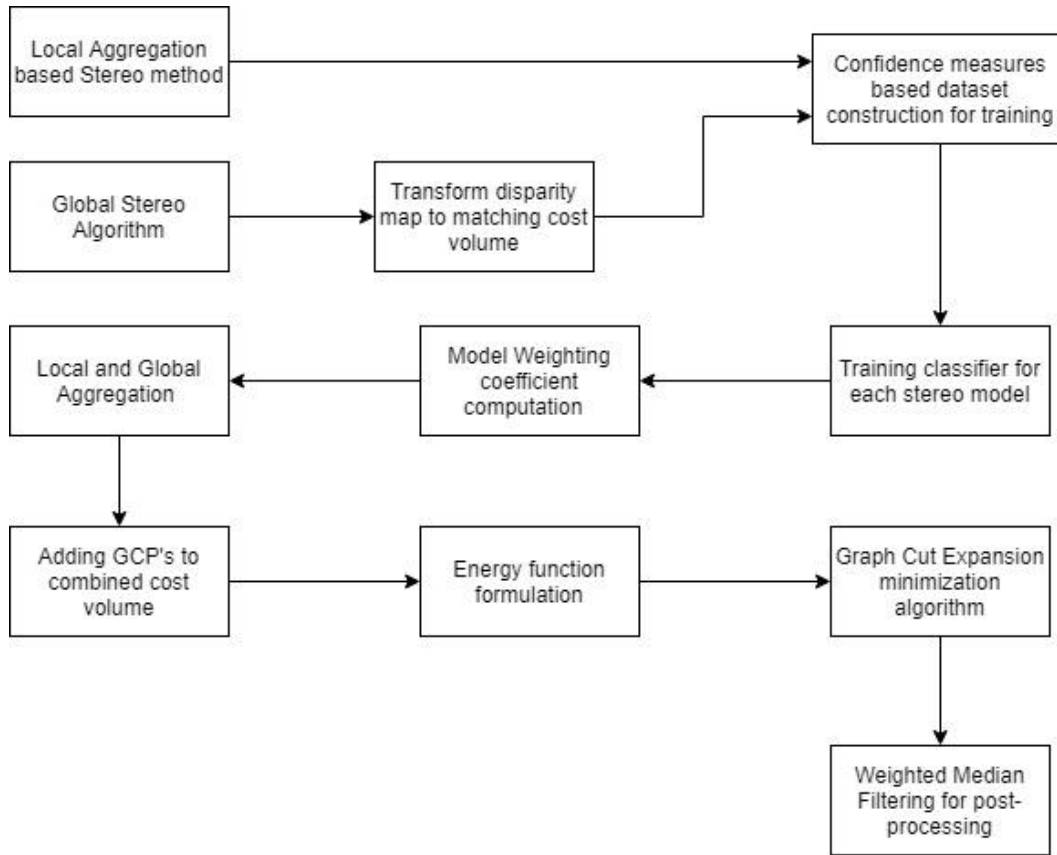
*Cost aggregation or global optimization:* two main matching cost methods for disparity estimation.

*Confidence measure based feature vector construction* [2-4]: mixing coefficients are computed from classification results of confidence measures. Datasets are created with the help of true disparities.

*Mixing coefficients computing:* based on [6], the mixing coefficient that weights each method.

*Cost aggregation over models:* mixture of experts that calculates general aggregation of all methods [6].

*Detect ground control points:* detect points for which disparity is known [7-10] and have high confidence [3], [10] in order to get a more reliable result in optimization.



**Figure 1.1.** Flowchart corresponding to the proposed method

When the classifier decides that the disparity for the pixel is reliable, the other disparities for the pixel are set to a high constant value.

*Energy function formulization:* it has two terms, data term regulated and modified by Ground Control Points (GCP's) and smoothness term that is a common term in most optimization based matchings. It uses color dissimilarity for the soft constraint.

*GC based minimization:* selects minimum disparity values over cost volumes. Weighted median filtering (WMF) is used [11]. Before applying the weighted median filter, left-right check is performed. Then, WMF whose parameters are image-guided weights are applied. The WMF is capable of capturing the strong edges, sharp corners, and thin structures in the image.

As in real time applications, the image pairs used in stereo matching are assumed to be ideal. Fusing various methods increase diversity and may provide satisfactory results.

## CHAPTER 2 RELATED WORKS AND BACKGROUND INFORMATION

### 2.1. Related studies

Local methods handle high-frequency texture areas well while they fail to deliver reliable results in occluded regions [12-14]. Recently, accurate image-guided local stereo matching methods have been developed [16-17]. The guidance image provides enhanced content for the stereo matching problem. In [18], an edge-preserving guided filter (EGF) was proposed extending the guided filter (GF) by introducing an edge-aware term. Furthermore, based on EGF a cost aggregation method using adaptive edge-preserving guided filter (AEGF) was developed. The AEGF can achieve proper cost volume filtering as well as edge preserving. In [19], a novel concept called two-level local adaptation was introduced. A parallel algorithm speeding up the basic computing element of adaptive guided filtering was designed. It improves the efficiency of disparity estimation significantly.

Global methods generate more accurate results compared to local methods in terms of quality [20-24]. However, computational cost of global methods is high. Hence, the main purpose of global methods is to reduce computational complexity while minimizing the negative effects on quality. For that purpose, GC based global methods were proposed [25-26]. First, reliable points of initial disparity maps are extracted and the disparity values of unreliable points are estimated. Then, the scheme of reliable points is introduced into a region-based GC framework leading to robust results. In a similar manner, sets of non-overlapping templates are derived from the reference image (left or right) to represent the scene structure in [27]. A disparity value is assigned to each template in order to obtain an energy function used to construct a graph. Another work presents a depth estimation approach based on segment-based GC [28]. The method is able to obtain a high-quality dense disparity map of a scene from its disparity plane estimation. It solves each image segment explicitly and assigns an independent disparity value to each segment. It uses improved hierarchical clustering algorithm together with the geometrical relationship of adjacent planes such as parallelism and intersection to

merge segments. Finally, applying GC approximation generates high-quality disparity image.

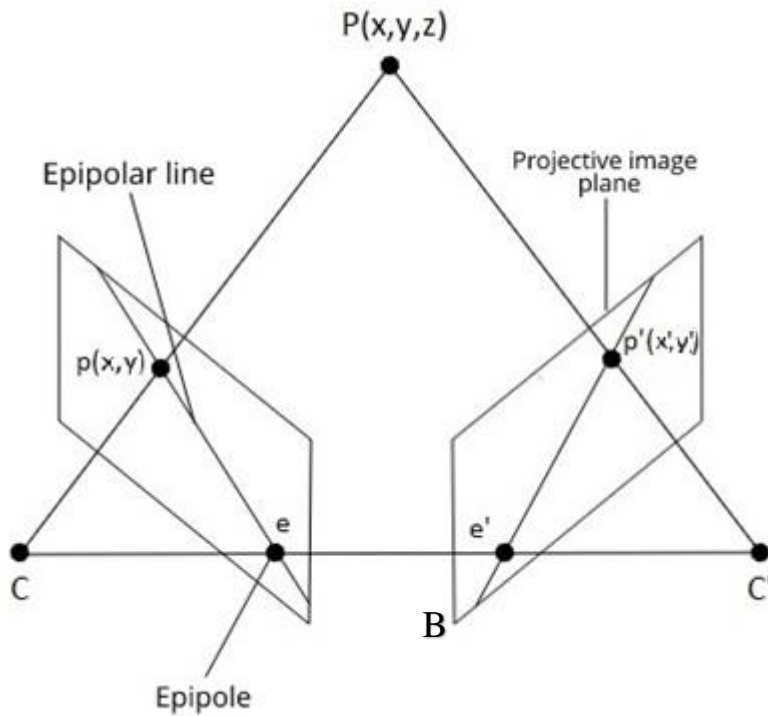
Local methods provide accurate results and they have low computational complexity [15-19]. Local methods suffer from outliers especially in occluded regions if post-processing is ignored. Consequently, the quality of depth map worsens without post-processing. Hence, selecting the right post-processing method with optimal parameters is an important issue in local methods. It is a somewhat complicated process because of variations in the input image pairs. Rather than depending on a single method, the concept of cross-scale cost aggregation for stereo matching was introduced in [29]. In the study, the scale space behavior of various cost aggregation methods was investigated in detail. Even with simple box filtering, the cross-scale framework results in satisfying results. Robustness of energy minimization approach to stereo matching in the presence of occlusions was improved by incorporating cost filtering reformulated as a energy minimization problem on a fully connected graph into the global optimization in [30-31].

Recently, a mixture-of-experts approach for stereo matching has been introduced [6]. The method combines different filters to produce a novel local filter with adaptive mixing coefficients. It was shown to be effective for reducing the matching error with an average error at the same level as the guided filtering and tree filtering [12], [22]. However, the method has several limitations. First, it uses only image-guided and tree filters with different parameters. In a mixture of experts, modeling the same filter with different parameter is considered to be an independent expert, though. Second, the issue of post-processing is not handled. As a result, the method suffers from outliers since it is local. Third, it does not allow to add global filter to the pool of filters. High accuracy results are usually provided by global techniques. In this thesis, we develop ideas to overcome these limitations. It is a challenging task since developing a global methods is an ill-posed problem. For details, please see Chapter 3.

## 2.2. Principles of stereo matching

Biological capability of human eyes for stereo vision has inspired researchers to develop stereo matching algorithms. By taking at least two or more images from different angles, it is possible to estimate the actual 3D structure of the image scene. An ultimate goal here is to find an accurate correspondence between those images. If exact correspondences are known, 3D location of any pixel in one image can be determined. Specifically speaking, for every pixel in the image a disparity is found based on user-defined measures (usually difference in image intensities) and with known calibrations. One can easily narrow down the matching space to an epipolar line (see Figure 2.1), under normal settings where both camera centers are colinear, the distance from any point in the world space are inversely proportional to disparity value.

There has been a huge progress in real-time stereo applications during the last few decades making it possible for robust and fast utilization. Stereo matching is still an active research area. Stereo matching algorithms usually consists of the following steps: 1) pre-processing to remove noises, 2) modifying camera angles and distances between them referred to as rectification, 3) searching for correspondence across image pairs resulting in disparity map containing disparities among pixels of stereo pairs, and 4) assuming cameras are calibrated, calculation of distance of a real world point to the camera center using triangulation process shown in Figure 2.2.

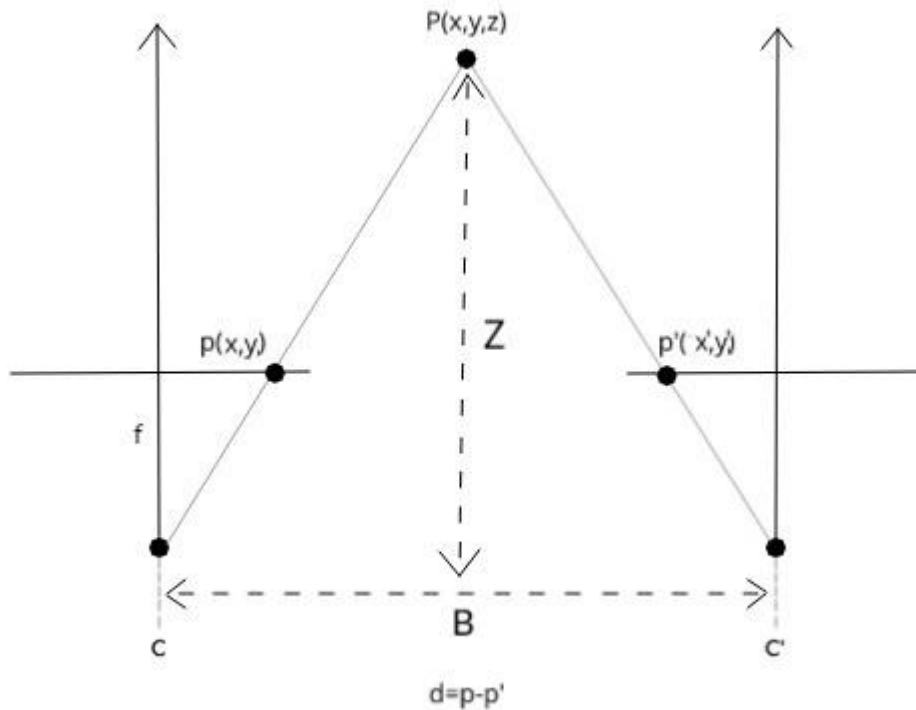


**Figure 2.1.** Epipolar geometry

### 2.2.1. Epipolar geometry

A general stereo imaging geometry referred to as epipolar geometry in the literature is shown in Figure 2.1.  $C$  and  $C'$  are called camera centers and the real world point  $P$  is projected to the image planes as points  $p(x, y)$  and  $p'(x', y')$ .  $B$  is the baseline distance between camera centers. The baseline intersects the imaging planes at  $e$  and  $e'$  called epipoles. The plane formed by connecting  $P$ ,  $C$  and  $C'$  is called epipolar plane. Epipolar lines are the lines connecting projected points  $p(x, y)$  and  $p'(x', y')$  with the corresponding epipoles  $e$  and  $e'$ .

Some useful properties of epipolar geometry in stereo matching perspective can be summarized as follows. Any world point  $P$  viewed from one camera has an epipolar plane containing an epipolar line. A projected point in the left image plane must have a corresponding point in the right image plane. This is referred to epipolar constraint. In



**Figure 2.2.** Triangulation for estimating the distance from observer

the case of parallel cameras configuration the epipolar constraint hugely reduces the two-dimensional matching workload to just one epipolar line. This not only increases efficiency but also avoids potential mismatches. Finally, ordering of points in one image plane must be preserved in the other image plane.

### 2.2.2. Image pair rectification and triangulation

Stereo camera calibration can be performed from seven sample matchings by constructing a fundamental matrix. After successful calibrations, the epipolar geometry constraint can be utilized to narrow down correspondence searching space as mentioned in the previous section. In the beginning, stereo image pairs are rectified so that the searching scanline

becomes an epipolar line. Then, by shifting a pixel in the left image across the right image scanline, related matching scores are obtained.

The idea behind rectification is to rotate the two cameras so that they become perpendicular to the baseline. Rectification changes the epipolar line to the horizontal direction in respect with the baseline and neglects the infinite disparities. Details can be found in [32].

A general structure of rectified stereo geometry is given in Figure 2.2. The final rectified stereo image pairs need to be calibrated. Once the calibration problem is solved, relationship between disparity  $d$  and distance of the real world point to the baseline denoted by  $Z$  is given by,

$$d=f\frac{B}{Z} \quad (2.1)$$

where  $f$  is the focal length of the camera,  $B$  is the distance between two camera centers. Under this configuration, correspondence between pixels  $p(x, y)$  and  $p'(x', y')$  becomes

$$x'=x+d(x, y), y'=y. \quad (2.2)$$

As it is clear, given  $d$ ,  $Z$  can be determined from (2.1).

### 2.3. Matching cost functions

Studies on stereo vision use two types of cost functions, namely parametric and nonparametric. Usually, parametric cost functions are formed by manipulating the pixel of interest. For example, calculating intensity difference or scaling intensity values to get some offset information about image pairs is one approach. Doing so, one can get a basic knowledge for stereo correspondence. However, these kinds of measures are sensitive to noise and pixelwise calculation solely relying on intensities can lead to erroneous results. With recently developed novel filters configurations, noise can be reduced significantly. Nonparametric cost functions rely on ordering of magnitude of pixels in a predefined neighborhood.

### 2.3.1. Truncated absolute difference (TAD)

A parametric cost measure is absolute difference (AD) that assumes brightness consistency between a pixel in the left image with the corresponding pixel in the right image. This type of measure a starting point for many algorithms. It important to mention that all the stereo image pairs are assumed perfectly rectified meaning a match of a pixel in the left image can be found exactly in the right image bounded to certain scanline. Global algorithms use pixelwise absolute differences since they reduce error in the iterated optimization step. On the other hand, local methods rely on the accuracy of the pixel and neighborhood around the pixel. Neighborhood region is usually referred to as window. Selecting proper window size is crucial. Before describing frequently used AD matching measures, some useful notations will be introduced  $I_L^{r,g,b}(x,y)$  and  $I_R^{r,g,b}(x',y')$  denote left and right color image pairs, respectively  $\tau_1$  and  $\tau_2$  are truncation values for color and gradient absolute difference,  $(x,y)$  and  $(x',y')$  represent left and right image pixel spatial coordinates. AD can be defined for grayscale and color images AD measure for color image given in (2.3) has the potential to reduce mismatches in duplicating texture regions. For robust matching costs, AD averaged over three color channels (RGB) are calculated as

$$N_{\text{color}}(x,y,x',y') := \frac{1}{3} \sum_{i \in \{R,G,B\}} |I_L^i(x,y) - I_R^i(x',y')|, \quad (2.3)$$

Then, truncated color AD with a threshold  $\tau_1$  is given by

$$N_{\text{color}}^{\tau_1}(x,y,x',y') := \min(N_{\text{color}}(x,y,x',y'), \tau_1) \quad (2.4)$$

Addition of two directional gradient measures given in (2.6) to the cost function in (2.8) can lead to a better measure compared to AD defined. Gradient is computed from grayscale image that provides detailed structural information about the scene. Grayscale image  $\tilde{I}_L(x,y)$  is obtained by averaging three color channels. In addition, gradient

measures are invariant to illumination. For this reason, gradient measures integrated with widely used color based AD are found to be a useful approach for getting a better understanding of image structure. The studies were done over the effect of truncation of matching costs to generate more accurate results for both truncated gradient AD given in (2.7) and truncated color AD. The relevant equations for determining truncated gradient AD are given below

$$\nabla_x \tilde{I}_L(x, y) := \frac{\tilde{I}_L(x+1, y) - \tilde{I}_L(x-1, y)}{2} \quad (2.5)$$

$$N_{\text{gradient}}(x, y, x', y') := |\nabla_x \tilde{I}_L(x, y) - \nabla_x \tilde{I}_R(x', y')| \quad (2.6)$$

$$N_{\text{gradient}}^{\tau_2}(x, y, x', y') := \min(N_{\text{gradient}}(x, y, x', y'), \tau_2) \quad (2.7)$$

$$N(x, y, x', y') := (1-\alpha) \cdot N_{\text{color}}^{\tau_1}(x, y, x', y') + \alpha \cdot N_{\text{gradient}}^{\tau_2}(x, y, x', y'). \quad (2.8)$$

where  $\alpha$  is a constant used to balance gradient and color AD values and  $N_{\text{gradient}}^{\tau_2}(x, y, x', y')$  is the truncated gradient AD.

### 2.3.2. Birchfield and Tomasi's Dissimilarity Measure

In real situations where a three-dimensional world point is projected into two stereo cameras; ideally or theoretically, intensities for the same point must be equal given that the calibration of stereo cameras were done accurately. However, in practice that is not the case. Due to physical limitations, different reflected rays from the point are inputs to the cameras based on their bias. Additionally camera hardware related noise is generated. Since reflected rays from a world point are summed up to form a pixel, aforementioned processes cause cameras to have different input intensities.

To overcome previously mentioned shortcomings, linearly interpolated intensity dissimilarity was proposed taking surrounding pixels into consideration unlike conventional pixelwise dissimilarity measures [33]. However, the computation time is more expensive than it takes to compute an absolute difference. The biggest benefit is that this kind of measure is insensitive to sampling.

Stereo pairs are rectified to produce narrow search scanlines that are equal to epipolar lines. In other words, we look up for correspondence only in one horizontal epipolar line. The objective is to find dissimilarity between the pixel  $I_L$  in the left scanline with pixel  $I_R$  in the right scanline. Initially, we compute linearly interpolated function within the pixels in the right scanline. Then, by comparing the intensity of left pixel  $I_L$  with the interpolated region around  $I_R$ ; the lowest possible difference defined in (2.9) is sought. The procedure is repeated for the right pixel and corresponding lowest possible difference is computed from (2.10). Final disparity for pixel  $(x,y)$  is obtained from (2.11).

$$\bar{d}_L(x, y, x', y', I_L, I_R) = \min_{(x', y') - \frac{1}{2} \leq i \leq (x', y') + \frac{1}{2}} |I_L(x, y) - I_R(i)|, \quad (2.9)$$

$$\bar{d}_R(x, y, x', y', I_R, I_L) = \min_{(x, y) - \frac{1}{2} \leq i \leq (x, y) + \frac{1}{2}} |I_L(i) - I_R(x', y')|, \quad (2.10)$$

$$d(x, y, x', y') = \min\{\bar{d}_L(x, y, x', y', I_L, I_R), \bar{d}_R(x, y, x', y', I_R, I_L)\} \quad (2.11)$$

### 2.3.3. Census Transform measure

An alternative way to calculate correspondence is using Census filter [34]. Census uses a binary array whose elements represent each neighbor of the pixel of interest, and the values depend whether neighbor pixel intensity has a higher or lower magnitude. In case if a pixel of interest has lower value, then array's element for this pixel is set, otherwise cleared. Rank Census filter also figures out the structure of corresponding regions. After finishing construction of binary array, next step is computation the Hamming distance between arrays.

## 2.4. Local and Global stereo methods

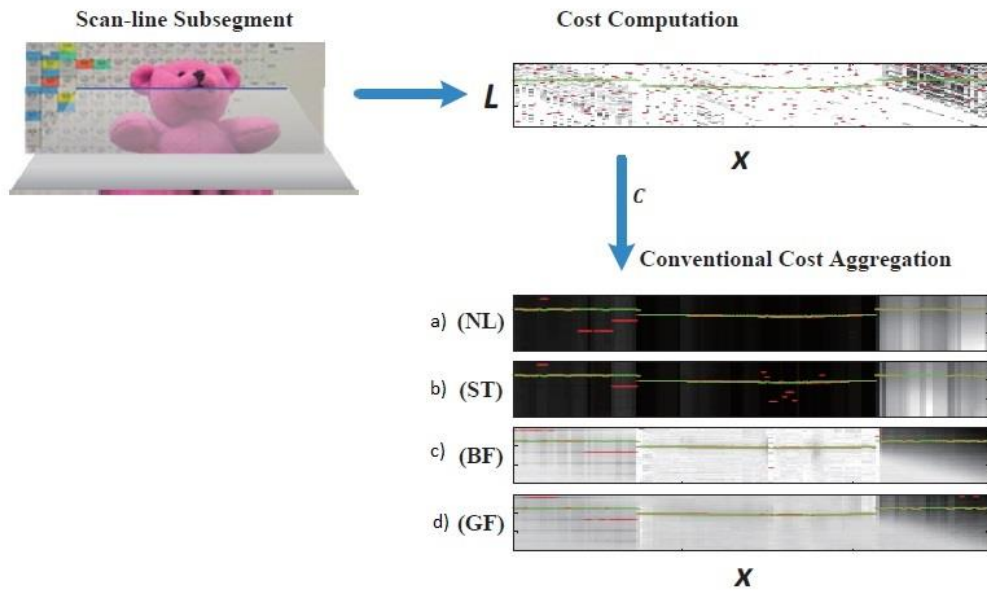
A typical stereo vision is defined as pixel labeling (disparity) problem. 3D cost volume is created using input stereo image pairs. For the costs, stereo developers use cost measures described in the previous sections. In an attempt to solve the problem, methods usually focus on the next three sub-objectives:

- 1) consistency of disparity values with costs,
- 2) disparity should maintain a local spatial smoothness,
- 3) disparity should change in object boundaries or image edges.

Global methods establishing the problem with Markov Random Field is usually referred to as Markov Random Field (MRF) optimization. Energy function consist of two terms. Matching costs are used for data term and smoothness term checks whether neighbor pixels have equal disparity values based on the assumption that images are most likely to have piecewise smooth structures. After constructing the energy function, the next task is to minimize it using popular algorithms like graph cut or belief propagation. Global methods often generate accurate results at the expenses of higher computation time. Moreover, processing high-resolution images introduce additional difficulties. Generally speaking, global methods have good performance for some limited low-resolution image pairs.

As the demand for high-resolution vision systems increases, researchers have focused mainly on alternative filter based local methods. Local methods filter the cost volume on the user-specified window. They are fast and practical. Recently, local stereo approaches have gained more popularity because of their high accuracy and fast performance. One of the local methods is image -guided filter stereo algorithm, having the most accurate results. Its complexity is independent from window size. Moreover, it does not require a trade-off between accuracy and efficiency.

Conventional stereo algorithms initially construct a three-dimensional cost volume with coordinates  $(x, y, d)$ , the last dimension is referred to disparity. Figure 2.3 shows the importance of cost aggregation. In the figure, top-right image is a matching cost distribution or  $x - d$  slice of cost volume for a horizontal scanline. The stereo problem is solved by selecting disparity with the lowest matching cost for every pixel in scanline. Red dots are disparities with the lowest costs. Top-right image shows us that the final results contain too many mismatches. The cause for such a failure is the fact that the cost is not aggregated. We can aggregate the selected slice of cost volume with the help of the support window, then we can apply WTA to get the final disparity map. This kind of approach is known as local stereo methods. For cost aggregation step, any



**Figure 2.3.** Importance of cost aggregation in disparity estimation. The top right image is the matching cost distribution without aggregation. Bottom images show the cost distribution for different aggregations including non-local, segment tree and image-guided filtering.

filter (even a box filter) does a better job than raw cost volume. Filtering is done for every disparity level on  $(x, y)$  dimensions. Clearly, results are more satisfying (smoother) than the raw matching cost even though there exist some errors in image edges. Even the quality of the disparity map is poor, the benefit of cost aggregation is that filtering has a practical and fast implementation with the help of a sliding window. Let  $C_o(x,y,d)$  and  $w(x,y,d)$  denote the initial matching cost distribution and 3-D window function. Then, the aggregated cost denoted by  $C(x,y,d)$  is given by

$$C(x,y,d)=w(x,y,d)*C_o(x,y,d). \quad (2.12)$$

where  $*$  denote the discrete-time convolution operator.

---

```

Input: Stereo image pairs  $I_l, I_r$ ;
Output: Disparity map  $d'(x, y)$ ;
Calculate mean  $\mu_k$  and variance  $\sum_k$ ;
1 foreach  $d$  in set of  $[d_{min}, d_{max}]$  do
2   Create  $d_{th}$  slice of cost volume;
3   foreach pixel  $i$  in the left image do
4     Compute color difference matching cost (2.2);
5     Compute gradient difference measure (2.4);
6     Compute balanced matching cost (2.6);
7     Assign balanced matching cost to  $C(i, d)$ ;
   end
8   Filtering;
9   Apply guided filter on  $d_{th}$  slice using the weights in (2.16);
  end
10 Disparity selection;
11 foreach pixel  $i(x, y)$  in the left cost volume do
12   Select the  $d$  with minimum matching cost;
13   Set the disparity value  $d'(x, y) = d$  for pixel  $i(x, y)$ ;
  end

```

---

**Figure 2.4.** Disparity estimation based on the guided filtering

#### 2.4.1. Image guided filtering cost aggregation

A weighted filter can eliminate edge fattening effect. Selecting proper filter weights smooth cost volume without blurring depth discontinuity. For example, a bilateral filter weights take into consideration both color image clues and also spatial clues. Implementation of such a filter leads to a spatially smooth result which preserves image edges. One drawback of this method is a high amount of computation time making it inadequate for real-time stereo applications. Filter runtime depends on the size of a window, that can vary based on image pairs.

Guided filtering (GF) stereo method shown in Figure 2.4 is another stereo method based on filtering that generates high-quality disparity maps at fast speed and has edge preserving property [12]. This filter just like bilateral filter works in the same over the cost volume processing principle. GF stereo approach outperforms most of the local methods. The key advantage of this method is that it works as iterative box filtering which is reason behind its fast implementation.

Consider a general stereo matching problem with the assumption that each pixel  $(x, y)$  should have some disparity equivalence  $d$  from a predefined disparity set  $D =$

$\{d_{min}, \dots, d_{max}\}$ . Image-guided stereo matching has the following steps: 1) building cost volume, 2) cost volume aggregation using an GF, 3) Disparity selection. A typical cost volume contains three-dimensional pixelwise matching information for each disparity level  $d$ . In the aggregation step, slices of the cost-volume are filtered, and to clear out the previous statement there are  $D$  slices of cost volume whose pixel's disparity values are computed using the weighted neighboring pixels in the same slice as:

$$C'_{x,y,d} = \sum_{w_k} W_{x,y,u,v}(I_L) C_{u,v,d} \quad (2.13)$$

where  $C'_{x,y,d}$  is the aggregated cost volume,  $(x, y)$  is the pixel of interest and  $(u, v)$  is the adjacent pixel. Aggregation is done within window of  $w_k$ . Filter weights  $W_{x,y,u,v}$  are calculated from input image pairs.

After aggregation step is over the disparity map is constructed by applying WTA optimization, and the disparity with the lowest matching for each pixel is selected:

$$d_{x,y} = \min_{d \in D} C'_{x,y,d}. \quad (2.14)$$

Intensity values of input image pairs are the main factor in the determination of filter weights  $W_{x,y,u,v}$ , that are calculated from

$$W_{x,y,u,v} = \frac{1}{|w|^2} \sum_{k:(x,y,u,v) \in w_k} (1 + (I_L(x, y) - \mu_k)^T (\Sigma_k + \epsilon U)^{-1} (I_L(u, v) - \mu_k)) \quad (2.15)$$

where  $I$  is the guiding color image,  $\mu_k$  and  $\Sigma_k$  are mean vector and covariance of the input image in a window  $w_k$  with size  $(2r + 1) \times (2r + 1)$ . The window is centered at pixel  $k$ ,  $U$  is the identity matrix,  $|w|$  is the total number of pixels in the window and  $\epsilon$  is a smoothness constant.

There are plenty of cost measures for stereo correspondence. They try to figure out the mismatches between image pairs. Any pixel  $(x, y)$  in the left image has to have correspondence in right image with some offset  $d$  and offset occurs only in the horizontal dimension. For an image-guided stereo method, a pixelwise AD of both color and gradient

image are used, because they are robust to illumination variation. A truncated AD is given in (2.16):

$$C_{x,y,d} = (1-\alpha) \cdot \min[\|I_L(x,y) - I_R(x,y - d)\|, \tau_1] + \alpha \cdot \min[\|\nabla_x I_L(x,y) - \nabla_x I_R(x,y - d)\| + \|\nabla_y I_L(x,y) - \nabla_y I_R(x,y - d)\|, \tau_2] \quad (2.16)$$

where  $\nabla_x$  and  $\nabla_y$  are the gradients of grayscale images in the  $x$  and  $y$  directions, constant  $\alpha$  balances color and gradient terms,  $\tau_1$  and  $\tau_2$  are truncation values.

#### 2.4.2. Segment tree filtering cost aggregation

What really makes any method a local stereo method is the fact that cost for a pixel is aggregated around local support window by summing or averaging cost of neighbor pixels. These types of window-based approaches are similar to fast Gaussian or box filters, but they are unable to generate accurate results in image edges. Unlike previously mentioned window-based methods, a non-local stereo aggregation was proposed in [22]. In this study, a non-local aggregation is processed over the whole image using a tree. On the other hand, local methods have to limit themselves on certain window size to avoid unnecessary blurring. In nonlocal method input stereo image is converted into a 4-connected undirected graph whose nodes denote every pixel in the image. Edges linking nodes in the graph are weights between neighboring pixels. A graph contains excessive amount of nodes decreasing efficiency. To overcome this issue, Minimum Spanning Tree (MST) is derived from the graph. This method aggregates constructed tree from the root node to the leaf node, and repeats aggregating reversibly from the leaf node to the root node, an each pixel would get some information from all neighboring pixels. It was shown to be efficient and it outperforms most stereo methods on the Middlebury benchmark.

Segment Tree (ST) algorithm which is another non-local cost aggregation method is shown in Figure 2.5. Image graph is divided into smaller coherent segments, for each segment a sub-tree is created, and in the end all the sub-trees are combined to create merged Segment Tree. ST has two advantages: 1) ST is more robust than MST, (ST uses

local edge weights and benefits from all non-local sub-segments), 2) useful information retrieved from the segmentation is added to the cost aggregation phase as a soft constraint. Note that ST encourages the pixels in the same segment to have similar disparities. Moreover, analyzing connections of pixels in the segment and giving them higher weights can lead to more accurate aggregation results. During aggregation, a geodesic distance of sub-tree is used to calculate the weights of edges. This may cause large variations inside the segment. Evaluation results are competitive with MST and image-guided filter on the Middlebury dataset. Further investigation can be done to improve tree structure with a color-disparity graph. One drawback of this method is that the computation time is higher than that of other methods. The reason behind this is the addition of segmentation.

ST stereo method has the following three steps:

- 1) Input image is segmented.
- 2) Each segment gets its own sub-tree.
- 3) All sub-trees are merged at the end to form the final tree.

Step 1 is segmentation of the image. Any segmentation algorithm such as mean-shift will work for this step. After the segmentation step, step 2 builds sub-tree for the corresponding segment and step 3 links all segments to form one final tree which is used to aggregate through nodes.

The tree building process is shown in Figure 2.5. Input stereo image is converted into a graph  $G = (V, E)$ . Then, a subset is derived from the graph as  $G' = (V, E')$ . Algorithm in Figure 2.5 mainly contains Initialization, Grouping, Linking stages. Lines from 1 to 5 correspond to initialization, line 1 sorts edges from highest to lowest, lines 3-5 create sub-trees for every node of the graph. Lines 6 to 13 constitute grouping. The algorithm checks whether each edge

---

**Input:** Graph  $G = (V, E)$  with  $n$  vertices and  $m$  edges. Each edge  $e \in E$  has the weight  $w_e$ ;

**Output:** Tree  $T = (V, E')$ , where  $E' \subset E$ ;

- 1 Sort edge weights in increasing order  $w_{e_1} < w_{e_2} < \dots < w_m$ ;
- 2 Initialize  $E' \leftarrow \emptyset$ ;
- 3 **foreach** node  $v_{(x,y)} \in V$  **do**
- 4 |   Setup tree  $T_{(x,y)} = (V_{(x,y)}, E_{(x,y)}) : T_{(x,y)} \leftarrow \{v_{(x,y)}\}, E_{(x,y)} \leftarrow \emptyset$ ;
- end**
- 5 **foreach** edge  $e_k \in E$  **do**
- 6 |   Check whether nodes  $v_{(x,y)}$  and  $v_{(u,v)}$  are connected with  $e_k$ ;
- 7 |   **if**  $T_{(x,y)} \neq T_{(u,v)}$  **and**  $e_k$  satisfies eq. in (2.18) **then**
- 8 |   |   Merge  $T_{(x,y)}$  and  $T_{(u,v)}$  to form new tree  $T_{(x,y),(u,v)} = (V_{(x,y),(u,v)}, E_{(x,y),(u,v)})$ ;
- 9 |   |    $V_{(x,y),(u,v)} = V_{(x,y)} \cup V_{(u,v)}, E_{(x,y),(u,v)} = E_{(x,y)} \cup E_{(u,v)} \cup \{e_k\}$ ;
- 10 |   |   Update  $E' = E' \cup \{e_k\}$ ;
- end**
- end**
- 11 Update  $E = E - E'$ ;
- 12 **foreach** edge  $e_k \in E$  **do**
- 13 |   Check whether nodes  $v_{(x,y)}$  and  $v_{(u,v)}$  are connected with  $e_k$ ;
- 14 |   **if**  $T_{(x,y)} \neq T_{(u,v)}$  **then**
- 15 |   |   Merge  $T_{(x,y)}$  and  $T_{(u,v)}$  to form new tree  $T_{(x,y),(u,v)} = (V_{(x,y),(u,v)}, E_{(x,y),(u,v)})$ ;
- 16 |   |    $V_{(x,y),(u,v)} = V_{(x,y)} \cup V_{(u,v)}, E_{(x,y),(u,v)} = E_{(x,y)} \cup E_{(u,v)} \cup \{e_k\}$ ;
- 17 |   |   Update  $E' = E' \cup \{e_k\}$ ;
- end**
- 18 |   End for loop if  $|E'| = |V| - 1$ ;
- end**
- 19 **Output:** Segmented tree  $T = (V, E')$ ;

---

**Figure 2.5.** Segment tree construction

( $e_k \in E$ ) satisfies the condition in equation (2.17) or not. If the condition is satisfied, corresponding nodes ( $V_{x,y}, V_{u,v}$ ) of  $e_k$  are merged to form a new segment ( $V_{x,y,u,v}$ ). After step 3, all the segments are merged to create on tree. As expected edge  $e_k$  is added to  $E'$ . The criterion which controls the edge weights is computed from

$$(w_{e_k} \leq \min \left( \text{Int}(T_{x,y}) + \frac{1}{|T_{x,y}|}, \text{Int}(T_{u,v}) + \frac{1}{|T_{u,v}|} \right)). \quad (2.17)$$

where  $T_{x,y}$  and  $T_{u,v}$  are trees for nodes  $V_{x,y}$  and  $V_{u,v}$ .



**Figure 2.6.** One example for stereo image pair.

### **2.4.3. GC stereo matching algorithm**

In stereo vision, two mutually calibrated cameras are capable of constructing the 3D structure of the world scene where a point is projected to two different camera plane. Having the same baseline for both of the cameras and rectifying image pairs leads to the simpler stereo matching problem. Such image pairs with different angle of view are called stereo images. One example is given in Figure 2.6. Basically, each 3D point is projected to the left image which is also located in the same horizontal line with the right image and in the stereo literature these lines are referred to as epipolar lines.

The method proposed in [1] solves correspondence problem using energy minimization. The objective is to minimize the given energy function according to disparity levels. The exact or approximated solution can be found by implementing GC optimization. Moreover, this method includes occlusion handling term. An algorithm detects occluded pixels and marks them as unmatchable or in other words those pixels do not have any corresponding pixels in the second view.

As explained in Section 2.2.2 Thales's theorem says that disparity values are inversely proportional to the distance between camera centers. Disparity map is enough to calculate a distance from the image scene to the observer.

Calculating the depth map is a difficult task. As two cameras have different view angles, some pixels are visible only in one view and these pixels are defined as occluded. The most important challenge in stereo vision is how to handle occlusion.

Notations used in GC method will be introduced first. Let  $(x, y)$  denote the pixel in the left image  $I_L$  and pixel  $(x', y')$  in the right image  $I_R$ . Assuming that the image pairs are rectified, disparity takes values in the one-dimensional range  $D_{disp} = [d_{min}, d_{max}]$ . Let  $A$  be a set of matching pixels pairs  $a = (x, y, x', y')$  that has a potential of being correct match. Additionally, for every matching pair  $a = (x, y, x', y')$ , there is a disparity  $d(a) = (x, y) - (x', y')$ . For two matching pairs  $a_1$  and  $a_2$ , their corresponding disparities are equal  $d(a_1) = d(a_2)$ .

For a function  $f: f(a) = 1$  means the matching between  $(x, y)$  and  $(x', y')$  are active (correct) matching. But, when  $f(a) = 0$ , this is referred to as inactive matching. If for pixels  $(x, y; x', y')$  there is only one active matching then the matching is unique. Finally, all matchings are inactive means that a pixel  $(x, y)$  does not have any correspondence and, it is called occluded matching. The energy function is defined as

$$E(f) = E_{data}(f) + E_{occlusion}(f) + E_{smoothness}(f) + E_{uniqueness}(f). \quad (2.18)$$

It consists of four terms. The first term measures the accuracy of matching pairs, third term forces the image to have a piecewise smooth structure, the second term detects occluded regions, the last term forces pixels to have one correspondence.

Data term forces energy function to have better matches. The more accurate the matches (means smaller matching costs), the smaller would be energy function. A data term is calculated from

$$E_{data}(f) := \sum_{a, f(a)=1} D(a) = \sum_a D(a) \cdot 1(f(a)=1) \quad (2.19)$$

where the indicator function  $1(\cdot)$  equals 1 when matching  $a$  is active, otherwise it is 0. Function  $D(a)$  contains matching cost values and measures the similarity between pixels  $(x, y)$  and  $(x', y')$ . Only active matchings are added to data term. Difference based matching cost for grayscale images is computed from

$$D_d(x, y, x', y') := T(|I_L(x, y) - I_R(x', y')|)^d \quad (2.20)$$

In (2.20), when  $d=1$  absolute difference is obtained while  $d=2$  leads to squared difference. Similarly, for a color image, matching cost is defined as

$$D_d(x, y, x', y') := \frac{T\left(|I_L^r(x, y) - I_R^r(x', y')|^d + |I_L^g(x, y) - I_R^g(x', y')|^d + |I_L^b(x, y) - I_R^b(x', y')|^d\right)^d}{3} \quad (2.21)$$

where  $T(\cdot)$  is the potential function.

The typical potential function that evaluates truncation according to similarity measure is given by

$$T(x) := \min(\text{CUTOFF}, x). \quad (2.22)$$

where CUTOFF is user specified truncation constant.

Occlusion term is designed to maximize the number of correct correspondences and to minimize the number of occluded pixels. The number of occluded pixels is directly proportional to inactive matchings, and the effect of these matches is reduced by constant giving rise to

$$E_{\text{occlusion}}(f) := \sum_{a, f(a)=0} K = \sum_a K \cdot 1(f(a)=0) = K \times \#A - \sum_a K \cdot 1(f(a)=1) \quad (2.23)$$

Multiplying the occlusion term with  $K$  gives the number of wrong matches.

Smoothness term tries to maintain piecewise smoothness in the disparity map. The idea behind this is that neighboring pixels are likely to have similar disparities. Speaking mathematically, if matchings of pixels  $(x_1, y_1)$  and  $(x_2, y_2)$  have equal disparities, then both matchings tend to be correct (active) or wrong (inactive). The smoothness term involving two neighboring matches is defined in (2.24):

$$E_{\text{smoothness}}(\mathbf{f}) := \sum_{a_1 \sim a_2} V_{a_1, a_2} \cdot 1(\mathbf{f}(a_1) \neq \mathbf{f}(a_2)) \quad (2.24)$$

where the regularizer  $V_{a_1, a_2}$  is computed from

$$V_{a_1, a_2} := \begin{cases} \lambda_1 = 3\lambda & \text{if } \max(|I_L(x_1, y_1) - I_L(x_2, y_2)|, |I_R(x'_1, y'_1) - I_R(x'_2, y'_2)|) < 8 \\ \lambda_2 = \lambda & \text{otherwise} \end{cases} \quad (2.25)$$

When  $(x_1, y_1)$  and  $(x_2, y_2)$  have equal disparities, the smoothness term going to takes the minimum possible value. The equation in (2.25) penalizes adjacent pixels with respect to their disparity values, and if the difference is large, smaller smoothness energy is assigned for these pair otherwise, a high value is assigned.

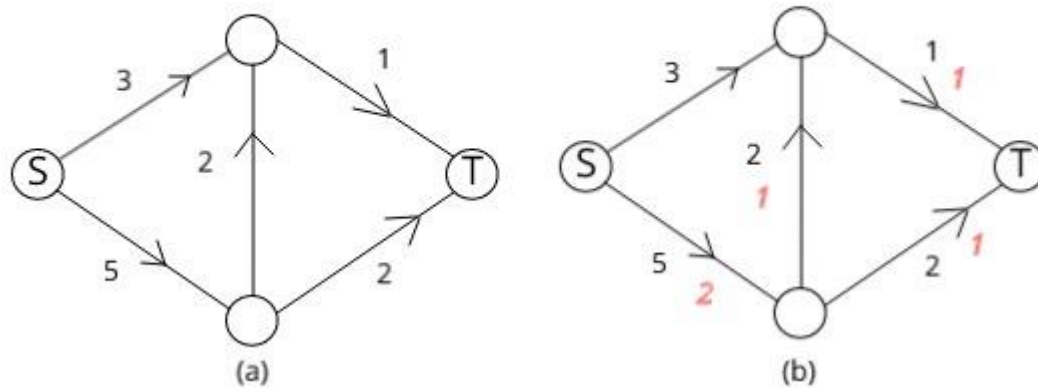
The last term forces uniqueness of the matchings. Energy takes a high value if the matching is not unique, zero otherwise as shown in (2.26):

$$E_{\text{uniqueness}}(\mathbf{f}) := \sum_{\substack{a_1=(x, y, x'_1, y'_1) \\ a_2=(x, y, x'_2, y'_2) \\ (x'_1, y'_1) \neq (x'_2, y'_2)}} \infty \cdot 1(\mathbf{f}(a_1) = \mathbf{f}(a_2) = 1) + \sum_{\substack{a_1=(x_1, y_1, x', y') \\ a_2=(x_2, y_2, x', y') \\ (x_1, y_1) \neq (x_2, y_2)}} \infty \cdot 1(\mathbf{f}(a_1) = \mathbf{f}(a_2) = 1). \quad (2.26)$$

Figure 2.8 illustrates the energy minimization algorithm for a graph  $G = (V, E)$  that consist of vertices  $V$  and edges  $E$  with positive weights. Let  $s$  and  $t$  be a source and sink nodes of the graph. Cut between nodes  $s$  and  $t$  contains nodes vertices  $s \in V$  and  $t \in V$ . Consequently, total weights from vertice  $s$  to  $t$  is referred to as a cut. If  $V_1$  and  $V_2$  are two nodes, edges connecting them is denoted by  $e_{12}$ . Cost of the cut of the graph  $G$  is defined as

$$c_G(V^s, V^t) = \sum_{\substack{e_{12} \in E \\ V_1 \in V^s, V_2 \in V^t}} c_G(V_1, V_2), \quad (2.27)$$

$$\forall e=(x, y) \in E, \quad 0 \leq \Phi(e) \leq c_G(x, y), \quad (2.28)$$



**Figure 2.7.** A simple example of network flow. (a) A network with max capacities. (b) For any node (except sink and source nodes) in the network, the incoming flow has to be equal to outgoing flow

where  $c_G$  indicates the capacity of relative nodes and  $\Phi(e)$  is capacity of edge. Figure 2.7 shows the capacities and flow of the graph. A possible maximum flow of the graph is the sum of edges with the highest flow. That is

$$\sum_{e=(s,x) \in E} \Phi(e) = \sum_{e=(x,t) \in E} \Phi(e). \quad (2.29)$$

Maximum flow and minimum cut of the graph are related to each other as stated in Theorem 1 [1].

**Theorem 1 (Max-Flow/Min-Cut)** The cost of a minimum cut of a graph is the value of maximum flow.

---

```

Input: Matching function  $f$ , disparity sets  $[d_{min}, d_{max}]$ , array  $[done]$  containing expansion info;
Output: Updated matching  $f$  with lower or equal energy function;
foreach  $\alpha$  do
  if not  $done[\alpha]$  then
    Find a better  $\alpha$  matching  $f^*$  from  $f$  that decrease the energy;
     $f^* \leftarrow \operatorname{argmin}_{f', \alpha\text{-expansion}} E(f')$ ;
    if  $E(f^*) < E(f)$  then
       $f \leftarrow f^*$ ;
       $done[:] = false$ ;
    end
     $done[\alpha] = true$ ;
    if  $done[:] = true$  then
      return  $f$ ;
    end
  end
end

```

---

**Figure 2.8.** Energy minimization algorithm

#### 2.4.4. Non-local cost aggregation

The study in [20] proposed a non-local cost aggregation method also called as Tree Filtering (TF) that has a similar technique to filtering methods except the fact that it does not need any window for aggregation. The method aggregates cost values that are obtained from the pixel similarity measure along with the tree structure. Also a tree is created using the input stereo image. The tree consists of nodes and edges. Nodes represent image pixels and edges are similarity weights between neighboring pixels. A shorter distance between tree nodes means pixels are more likely to have a similar disparity. In the aggregation step, since every node takes some contribution from all other nodes, the method is considered as non-local. Experimental results has shown that the method can be competitive with other local methods on Middlebury dataset. Not only the quality of correspondence but also the computation time of the method are better compared to those of other local methods.

The technique has a similar idea to conventional aggregation based stereo vision, and the input stereo image is used to calculate pixel dissimilarity. An input image is

transformed into the undirected graph, where image is represented in terms of vertices and edges. Vertices express image pixels, and edges give information about the connection between neighboring pixels. Aggregation is similar to bilateral filtering; the matching cost values are processed according to pixel similarity, while MST is created from the undirected graph. As a similarity measure between vertices, the shortest distance on MST is calculated.

The main advantage of the approach is that it shows more accurate pixel dissimilarity with MST. Every pixel on the image has some affection during the aggregation step.

Unlike local methods that need a predefined window only pixels inside the window are responsible for the outcome. The method tends to perform better in low texture image regions than other filters. The main reason why local methods provide poor results is that support window is limited to certain boundaries, and it does not cover all low texture parts. Experimental results on Middlebury dataset has shown that more accurate disparities are obtained compared to other local stereo methods.

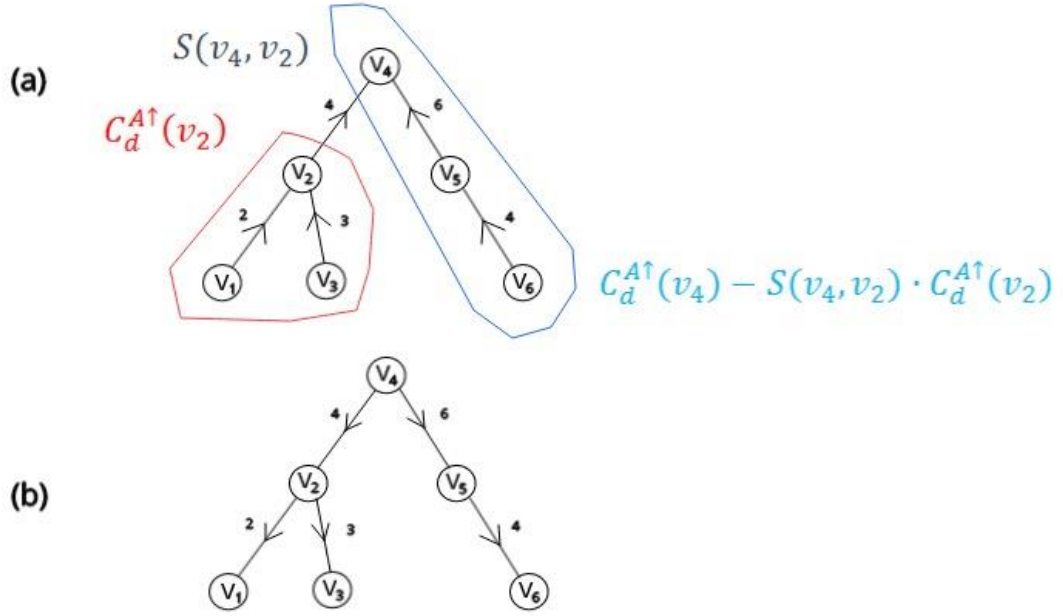
The main contributions of the method to the stereo problem are the following: 1) It is a new way of aggregation for solving a stereo problem, giving rise to better performance in low texture regions. 2) it has low computational complexity with only 2 add/subs and 3 multiplications for each pixel for one disparity level.

An input stereo image is converted into an undirected 4-connected, planar graph  $G(V, E)$ . Weights of edges are computed from

$$w(x,y,u,v)=w(x,y,u,v)=|I_L(x,y)-I_L(u,v)|. \quad (2.30)$$

where  $(x,y)$  and  $(u,v)$  are two neighboring pixels.

It's important to cut unwanted edges in order to eliminate some parts of the graph, otherwise, the workload of the graph itself is impractical. To construct MST; edges with high weights are removed from the graph. A final truncated tree contains vertices with the least linking weights among all spanning trees, and this makes it possible to find similarity between two pixels on MST.



**Figure 2.9.** Cost aggregation using MST. (a) MST is aggregated from leaf to root, (b) aggregation is repeated from root to leaf.

Sum of edge weights connecting two nodes defined in (2.31) is the distance  $D(x,y,u,v)$  on the MST.  $D$  represents the similarity between pixel  $(x,y)$  and  $(u,v)$ ,  $\sigma$  is a constant to adjust similarity, and  $C_d(u,v)$  denotes matching cost of pixel  $(u,v)$ . The equation for aggregation  $C_d^A(x,y)$  is given in (2.32)

$$S(x,y,u,v)=S(u,v,x,y)=\exp\left(-\frac{D(x,y,u,v)}{\sigma}\right), \quad (2.31)$$

$$C_d^A(x,y)=\sum_{\substack{\text{adjacent} \\ \text{pixels}}} S(x,y,u,v)C_d(u,v)=\sum_{\substack{\text{adjacent} \\ \text{pixels}}} \exp\left(-\frac{D(x,y,u,v)}{\sigma}\right)C_d(u,v). \quad (2.32)$$

## 2.5. Learning of confidence measure

From existing studies on stereo matching, main sources of mismatches are seen to be come from occlusion, flat image scenes, and texture-rich structures. Besides, global stereo minimization methods exhibit a high tendency to have multiple local minima. Sources of errors can be analyzed from a different point of view. Some useful features around the pixel of interest can be extracted to make it possible to predict whether correspondence accuracy is sufficient enough or not and use it to improve the disparity map.

Depth estimation solely depending on minimum costs can be tricky because of getting multiple minima or similar costs around the correct minimum. Recently, some studies have analyzed the cost curve and extracted some confidence measures for each pixel [2-3]. Confidence measures can be used to sort the matched pixels according to their reliability. Confidence measures proposed in the literature are usually a combination of some attributes taken from cost curve and pixel neighborhood. One may think of confidence as error prediction or a measure for clarification of results. Prediction can directly change the final result. Measures can also be used as a reliability test for each pixel. Rather than using confidence measure for prediction, it should act as a probability that each stereo method fails to deliver sufficient result. For low confidence conditions, there is not a certain algorithm to perform well. On the contrary, high confidence situations have the potential to pick up the right algorithm to work on it.

Suppose we have cost volume  $C(x, y, d)$  that has information about disparity assignments. Disparity is denoted by  $d(x, y)$  taking values in the interval of  $[d_{min}, d_{max}]$ . While building confidence features, we do not consider disparity levels outside the interval  $[d_{min}, d_{max}]$ . Each pixel has its own cost curve containing the matching costs for all disparity levels.  $c_1$  and  $c_2$  denote the first and second minimum respective values of the cost curve. In the following sections, we discuss eight confidence measures for the method proposed in this thesis.

### 2.5.1. Matching Cost

Matching cost a simple yet effective confidence measure and it is a good starting point to construct the feature set. In spite of the fact that it is the minimum cost measure among all disparities, the matching cost is useful to figure out true matching. This measure defined in (2.33) has a negative value, so the larger it is the more confidence is expected.

$$C_{MSM} = -c_1. \quad (2.33)$$

### 2.5.2. Distance from Border (DB)

This feature measures how far a current pixel is away from image top, bottom, left and right boundaries. It is straightforward to verify that pixels near borders are likely to be seen only from one camera, which is not a desired state for the stereo matching problem.

### 2.5.3. Maximum Margin (MM)

This measure evaluates the difference between the first ( $c_1$ ) and second ( $c_2$ ) minimum cost values of the selected pixel. It is most likely a true disparity for larger margin confidence. It is calculated from

$$v_{MM} = |c_2 - c_1| \quad (2.34)$$

### 2.5.4. Attainable Maximum Likelihood (AML)

It is a probability based feature for which Gaussian distribution function is obtained from the cost curve. Center of the distribution is the minimum cost  $c_1$ . Subtracting  $c_1$  from each cost values  $c(d)$ , enables us to get a better picture of all matching costs. The measure is given by

$$C_{AML} = \frac{e^{-(c_1 - c_1)^2 / 2\sigma_{AML}^2}}{\sum_d e^{-\frac{(c(d) - c_1)^2}{2\sigma_{AML}^2}}}. \quad (2.35)$$

### 2.5.5. Left-Right Consistency (LRC)

Ideally, any pixel in the left image should have correspondence in the right one. However, that is not always the case as some mismatches occur due to occlusion and noise. Binary feature presented in this section is another strong measure for catching inconsistency between image pairs. Basically,  $d_L - d_R$  is less than 1, feature value for that pixel is equal to 0, otherwise it is 1. The rationale is that the closer the disparities are, the more accurate correspondence will be. It is calculated from

$$C_{LRC}(x,y) = |d_L - d_R(x-d_L, y)|. \quad (2.36)$$

### 2.5.6. Left-Right Difference (LRD)

Similar to RLC, this feature favors when the maximum margin between  $c_1$  and  $c_2$  is large enough, yet at the same time consistent with the right image. Different from LRC, it calculates relative costs. The idea is that accurate corresponding pixels somehow should have related matching costs. The measure is defined in

$$C_{LRD}(x,y) = \frac{c_2 - c_1}{|c_1 - \min\{c_R(x-d_L, y, d_R)\}|}. \quad (2.37)$$

There are two low confidence cases: (i) margin  $c_2 - c_1$  is low with a large mismatched dominator in (2.37); (ii) a margin is large with a large dominator. Even if the small margin means the minimum cost ambiguous, the denominator part can boost the confidence value for similar matching values.

### 2.5.7. Distance from Discontinuity (DD)

For stereo algorithms, usually fail to perform well in textured regions. The possibility of mismatching in depth discontinuity is high. We define any pixel whose disparity is not equivalent to all four neighbors as discontinuous. This measure horizontal distance to that nearest discontinuous region.

### 2.5.8. Difference with Median Disparity (MED)

Pixels whose disparities are correlated with their neighbor pixel have a tendency to be a valid match. Basically, this feature uses median disparity with the pixel being centered in the 5x5 window. The measure is equal to the absolute difference between pixel's disparity and median disparity are given in (2.38)

$$v_{MED} = |d_L - d_L^{MED}|. \quad (2.38)$$

## CHAPTER 3      PROPOSED METHOD

### 3.1. Mixture of local and global methods

There exist many methods to solve a stereo problem. Aggregation based methods are mainly referred to as local stereo and the ones that apply optimization are called global stereo. For the local methods, filters are applied over the cost volume and a simple WTA is used to obtain the disparity map. Different aggregation methods and filters were explained in Chapter 2.

In general, local methods in [12], [13] give satisfactory results on Middlebury dataset even though for real time applications one method may not be compatible with all stereo pairs. For instance, the method in [13] needs a large support kernel to get reliable results. On one hand, a larger window size may be needed to evaluate pixels in flat regions, on the other hand for high-frequency texture regions one need smaller window size to avoid edge blurring. Moreover, nonlocal method in [20] with specific settings may not deliver good results for textured or flat regions.

In the following, we propose a combined stereo method that mixes various methods with different settings so that a combined approach is expected to provide more accurate results. We learn the behavior of each individual method with the help of confidence measures extracted from aggregated cost volumes on the pixelwise scale. Finally, mixing coefficients of the combined stereo approach is formulated with respect to confidence measure reliability. Recently a combined model has been proposed in [29]. It is claim in the study that one can use any but just one filter type and disparity map is calculated across multiple scales from finest to lowest resolutions. Another work proposed in [35] uses normalized cross-correlation (NCC) measure with various window sizes, and treats each NCC with different window size as a separate method. Contribution of the study is that multiple aggregations and global methods can be combined into one framework, which makes it easy to benefit from different types of stereo techniques. Additionally, in case if all methods fail to deliver accurate prediction about the true disparity, the proposed approach takes the average of all estimations.

A given filter with a specific parameter setting may have a potential to work for an image pair, but may not provide satisfactory performance for other image pairs. To

overcome this issue, we propose a mixture-of-experts' model in which a heterogeneous set of filters on the cost volume is applied and the results are adaptively combined. By adding a global filter to the pool of local filters, we get improved matching results since global methods tend to give better results especially in the occluded areas in comparison to local methods that are likely to get fast and reliable results in high texture images. Afterward, post-processing is included to minimize average matching error. The general definition of multi-method model for stereo is:

$$C(x,y,d)=\sum_{m\in M}g^m(\mathbf{v}_{x,y})\cdot C^m(x,y,d) \quad (3.1)$$

where  $C^m(x, y, d)$  is the aggregated cost value of the pixel with spatial coordinates  $\{x, y\}$ ,  $m$  represents the number of methods and  $g^m(\mathbf{v}_{x,y})$  is softmax function used to categorize methods or we call it model weighting coefficient (3.2).  $\mathbf{v}_{x,y}$  is a feature vector for pixel  $(x, y)$  computed from confidence measures (Section 2.5). The definition in (3.1) has some similarities with the conventional local cost aggregation given in (2.12). They both process over pixelwise matching costs and perform calculations in three-dimension  $\{x, y, d\}$ . Basically,  $g^m(\mathbf{v}_{x,y})$  defined in (3.2) determines how accurate method  $m$  predicts the disparity for pixel  $(x, y)$ ,

$$g^m(\mathbf{v}_{x,y})=\frac{\exp(\xi^m(\mathbf{v}_{x,y}))}{\sum_{k\in M}\exp(\xi^k(\mathbf{v}_{x,y}))} \quad (3.2)$$

where  $\xi^m$  is the model weighting coefficient.

In order to calculate the model weighting coefficient in (3.2), we train a decision classifier for each method  $m$  and coefficients vary due to attributes. Predictor  $\xi^m(\mathbf{v}_{x,y})$  defined in (3.3) determines the importance of the method  $m$  for the pixel  $(x, y)$ ;

$$\xi^m(\mathbf{v}_{x,y})=f^m(\mathbf{v}_{x,y})=\begin{cases} 1 & \text{predicted true disparity} \\ 0 & \text{predicted false disparity} \end{cases} \quad (3.3)$$

The problem arises when we decide to add a global method to the pool of  $\mathcal{M}$  local models. Local methods have a three-dimensional filtered cost volume  $C^m(x, y, d) \in R^{W \times H \times D}$ , where  $W, H, D$  are image width, height and disparity range, respectively. On the other hand, the output of global model is two-dimensional disparity map  $d_g^m(x, y) \in R^{W \times H}$ . Hence, some kind of transformation for  $d_g^m(x, y)$  is needed to fit the model in (3.1). To combine local and global methods we use model given in (3.4)

$$C'(x, y, d) = \alpha(x, y) \cdot C_g'(x, y, d) + \beta(x, y) \cdot C_l'(x, y, d) \quad (3.4)$$

where the second term is the combined local cost volume defined in (3.7) summing up the aggregated matching measures of all local methods. The first term is the combined global cost volume. It is calculated from (3.8) and it transforms disparity maps  $d_g^m(x, y)$  into three-dimensional costs  $C_g'(x, y, d)$ . Moreover, on the assumption that algorithms estimate disparities better on occluded regions, each method is not treated equally.  $\alpha(x, y)$  and  $\beta(x, y)$  constants weight global and local algorithms according to left-right consistency check. Once occluded region is found out, we can limitate the contribution of local methods in that region.  $M_{global}$  is the total number and  $\{d_{g_L}^m, d_{g_R}^m\}$  are (left/right) disparity maps used in (3.4);

$$\alpha(x, y) = \begin{cases} 1 & , \left[ \frac{1}{M_{global}} \sum_{m \in M_{global}} \text{abs} \left( d_{g_L}^m(x, y) - d_{g_R}^m(d_{g_L}^m(x, y), y) \right) \right] > 2, \\ 0 & , \text{otherwise} \end{cases} \quad (3.5)$$

$$\beta(x, y) = \begin{cases} 0 & , \left[ \frac{1}{M_{global}} \sum_{m \in M_{global}} \text{abs} \left( d_{g_L}^m(x, y) - d_{g_R}^m(d_{g_L}^m(x, y), y) \right) \right] < 2, \\ 1 & , \text{otherwise} \end{cases} \quad (3.6)$$

The definition in (3.7) which adds filtered cost measures to combined cost volume  $C'(x, y, d)$  is related to locally aggregated methods. Because of edge-preserving nature of local methods,  $C_l^m(x, y, d)$  contain more usefull information on high frequency texture regions. As usually aggregation is performed on local window, this type of aggregation

can capture strong edges by adjusting window size. The detailed investigation for the parameters of local methods is behind scope the of this thesis. In the simulations, we use a default settings provided by authors;

$$C'_l(x,y,d)= \sum_{m \in M_{local}} g^m(\mathbf{v}_{x,y}) \times C_l^m(x,y,d). \quad (3.7)$$

In contrast to local methods, global algorithms do most of their improvement in the optimization phase and usually skip the aggregation part. In global methods, problem of finding optimal disparity is solved through the formulation of the energy function. Energy function mostly consists of data and smoothness term. Data term keep track of overall consistency of input data. In most of the cases, it has simple non-aggregated cost volume, which doesn't deliver useful information. The smoothness term tries to maintain smoothness between the neighboring pixels assuming that pixels across the same image scene or object have similar disparities. In general, it penalizes neighboring pixels with different disparities. Since these algorithms compute disparity map on the entire image without any window, they have a tendency of wrong disparities propagating to all around the final output. For that reason, smoothness term needs specific attention and must be chosen properly. After the formulation of energy function, the next stage is to find optimal disparities that minimize the energy function. Recent research, minimization based on graph cut produces high accuracy results. Most of the high accuracy stereo algorithms are global methods, so we try to add global method to the pool of local models. A general framework for obtaining global combined cost volume is given in (3.8). Since a multi-method definition in (3.1) does calculations in three-dimensional space, we build up a cost volume using disparity map as shown in (3.9):

$$C'_g(x,y,d)= \sum_{m \in M_{global}} g^m(\mathbf{v}_{x,y}) \times C_g^m(x,y,d), \quad (3.8)$$

$$C_g^m(x,y,d)=\begin{cases} C_{\min}(x,y), & d_g^m(x,y)=d \\ C_{\text{other}}(x,y), & d_g^m(x,y)\neq d \\ C_{\text{occ}}, & d_g^m(x,y)=\text{Occ} \end{cases} \quad (3.9)$$

where  $d \in [d_{\min}, d_{\max}]$ ,  $d_g^m \in \{[d_{\min}, d_{\max}], \text{Occ}\}$ .  $C_{\min}(x,y)$ ,  $C_{\text{occ}}$ ,  $C_{\text{other}}(x,y)$  are matching cost measures required to construct the cost volume.  $d_{\max}$  is the maximum disparity value used in our implementation. Disparity level is iterated until max level is reached. Occ represents occluded pixels and will have a higher value than max disparity. The definition in (3.8) converts an output of any global method to three-dimensional cost volume. Initially, we take final disparity image and process it with three different terms as shown in (3.9). If disparity for the pixel  $(x,y)$  is equal to the current disparity level  $d_g^m(x,y) = d$ , a small  $C_{\min}(x,y)$  value is given for the combined cost in order to emphasize the importance of current disparity of pixel.  $C_{\min}(x,y)$  is average of the minimum cost values of the pixel  $(x,y)$  among local methods. Disparity map may contain some occluded pixels or region. Third term in (3.9) detects occluded pixels and gives some high  $C_{\text{occ}}$  value, which is the average of cost values in the occluded region of local methods. During the optimization phase pixels with values  $C_{\text{occ}}$  will automatically be eliminated. In the same manner second term in (3.9) intends to get rid of pixels with wrong disparities by assigning a different  $C_{\text{other}}(x,y)$  value. The reason for using different values is that they are unwanted disparities.  $C_{\min}(x,y)$ ,  $C_{\text{occ}}$ ,  $C_{\text{other}}(x,y)$  are calculated from (3.10), (3.11) and (3.12), respectively given below

$$C_{\min}(x,y)=\frac{1}{M_{\text{local}}} \sum_{m \in M_{\text{local}}} \min\{C_1^m(x,y,d_{\min}), \dots, C_1^m(x,y,d_{\max})\} \quad (3.10)$$

$$C_{\text{other}}(x,y)=\frac{1}{d_{\max}} \sum_{m \in M_{\text{local}}} \times \sum_{d \in [d_{\min}, d_{\max}]} C_1^m(x,y,d), \quad (3.11)$$

$$C_{\text{occ}} = \frac{1}{N_{\text{occ}}} \sum_{m \in M_{\text{local}}} \times \sum_{(x,y) \in R_{\text{occ}}} \times \sum_{d \in [d_{\text{min}}, d_{\text{max}}]} C_1^m(x,y,d). \quad (3.12)$$

### 3.2. Ground control points added matching cost volume

In this section, we describe the procedure of obtaining ground control points (GCPs) that modify the previously calculated combined cost volume described in section 3.1. GCPs are defined as pixels having a true disparity with high probability or high confident pixels with reliable disparities. These pixels will be used in optimization to take advantage of their decisive power to affect the neighboring pixels.

The biggest challenge is calibration of the amount of added GCPs. A small number of GCPs will have a negligible effect on overall accuracy. On the other hand, if we select too many GCPs there is a possibility that wrong disparities propagate to near pixels and deter the overall result. The aim is to maintain highest possible density of GCPs while tolerating false matches. In our proposed approach we set to 50% density of all combined cost volume. Further investigation can be done to select optimal density.

After adding GCPs, we use modified cost volume given in (3.17) as data term of the energy function. Moreover, modification is expected to have a positive impact on the overall performance. The first thing we do is to select GCPs according to the result of the classifier. Recently, widely used machine learning approaches for GCPs constructions have used some features calculated from disparity maps and used them in training. GCPs can be extracted in many different ways. They can be considered as time of flight (TOF) sensor where sensor measures the depth from the scene. Even they may be a combination of both at the same time. Before determining a set of GCPs we already have trained classifier models discussed in section 2.5. Suppose we have three trained models given in (3.13),(3.14),(3.15):

$$f^{\text{local}_1}(\mathbf{v}_{x,y}) = \begin{cases} 1, & \text{true disparity} \\ 0, & \text{false disparity} \end{cases} \quad (3.13)$$

$$f^{local_2}(\mathbf{v}_{x,y}) = \begin{cases} 1, & \text{true disparity} \\ 0, & \text{false disparity} \end{cases} \quad (3.14)$$

$$f^{global_1}(\mathbf{v}_{x,y}) = \begin{cases} 1, & \text{true disparity} \\ 0, & \text{false disparity} \end{cases} \quad (3.15)$$

where  $\mathbf{v}_{x,y}$  is a feature vector containing attributes for each pixel and  $f^{local_1}(\mathbf{v}_{x,y})$ ,  $f^{local_2}(\mathbf{v}_{x,y})$ ,  $f^{global_1}(\mathbf{v}_{x,y})$  are classifier models for local and global methods. Initially, we test three cost volumes through previously mentioned classifier models and calculate the error rate for each of them separately. Then, we take the classifier model with the least error rate and use it for obtaining GCPs. Selection of the classifier with the least error rate is given by

$$f(v_i) = \min_{\text{error}} (f^{local_1}(\mathbf{v}_{x,y}), f^{local_2}(\mathbf{v}_{x,y}), f^{global_1}(\mathbf{v}_{x,y})). \quad (3.16)$$

In the next step, we add GCPs to the combined cost volume  $\hat{C}(x, y, d)$ , constructed in (3.4). When the selected model in (3.16) predicts a pixel with a true disparity, we set a matching cost of all the other disparity levels to constant value 2, while the matching score of the corresponding pixel is left unchanged as shown in (3.17)

$$C^{GCP}(x, y, d) = \begin{cases} \hat{C}(x, y, d), & \text{if } f(\mathbf{v}_{x,y})=1 \\ 2, & \text{otherwise} \end{cases} \quad (3.17)$$

where  $\hat{C}(x, y, d)$  is a combined cost volume defined in (3.4),  $C^{GCP}(x, y, d)$  is GCPs added to the combined cost volume and  $f(\mathbf{v}_{x,y})$  is the classifier (with the least error rate) selected to obtain GCPs.

### 3.3. Energy function formulation

Most of the early stereo matching defined as pixel/disparity problems, where each pixel contains disparity information computed with a corresponded stereo algorithm. Some of the implementations of stereo vision group these pixels according to the assumption that disparities change smoothly among neighboring regions. Except for the fact that pixels in

the highly textured regions will fail to maintain smoothness and need extra attention. This kind of pixelwise disparity vision can be viewed as an energy minimization problem, which has been studied for decades in the regarded area. In recent years, there were major developments in global stereo methods, where the energy function mainly consist of data and smoothness terms. Data term evaluates the consistency of input data and smoothness term encourages the neighboring pixels to have similar disparities. Each pixel  $(x,y)$  assigned disparity from the predefined set  $m$ . An objective is to find disparity that minimizes the energy function

$$E(x,y,d)=E_d(x,y,d)+\lambda E_s(x,y,d). \quad (3.18)$$

A selection of smoothness term is an important step, as a result, plenty different kind of terms was used in literature. In stereo algorithms, most of the proposed smoothness terms can be divided into spatially regularized and segmentation based classes. The first class expects the pixels from the same scene to have similar disparities, in the same manner, a second class assume image scene with uniform color has the equivalent disparities. Majority of the present top performing stereo algorithms ensemble those two classes of terms while constructing energy functions. Even segmentation based energy functions deliver accurate depth images in less textureless images, however, they have a hard time to show the same level of performance in the image with high-frequency texture. The reason behind this lay down in the nature of segmentation tools, that are not always give the desired results in texture-rich regions. In contrast, early spatially regularized stereo methods can perform well in object boundaries, eventually, they do have the potential to over smooth object boundaries, and this type of issue is common in literature. To overcome this issue many discontinuity-preserving energy functions have been proposed that allows detecting object boundaries then set relative weights based on color and scene structure clues and our data term is

$$E_d(x,y,d)= \sum_{\text{All pixels}} C^{\text{GCP}}(x,y,d). \quad (3.19)$$

Smoothness term in (3.20) usually is a combination of spatially changing weights  $w_{(x,y,x',y')}$  and some nondecreasing potential function that evaluate the disparity difference  $V(\Delta d) = V(|d_{x,y} - d_{x',y'}|)$

$$E_s(x,y,d) = \sum_{\{x,y,x',y'\} \in N} w_{(x,y,x',y')} V(|d_{x,y} - d_{x',y'}|), \quad (3.20)$$

where  $\{x,y,x',y'\}$  is a pair of adjacent pixels, where each pixel in spatial two-dimensional coordinates.  $N$  is a 4-connected neighboring system, means vertical and horizontal nearest pixels will be used in the calculation

$$V(\Delta d) = \min(|\Delta|, 1), \quad (3.21)$$

differece  $|\Delta|$  can be squared for more sensitive differentiation.

### 3.4. Energy function minimization

After the of the energy function is formulated, we minimize it using the graph cut (GC) method [36]. For the stereo problem with multiple disparities, the GC can approximately minimize the energy  $E(x,y,d)$  in (3.18) under two conditions: the potential function  $V$  has to be semi-metric and metric.  $V$  is semi-metric if for the pair of disparities  $\alpha, \beta \in D$ ,  $V$  satisfies the equations (3.22), (3.23). Moreover,  $V$  must also satisfy the triangle inequality given in (3.24) in order to be metric

$$V(\alpha, \beta) = V(\beta, \alpha) \geq 0, \quad (3.22)$$

$$V(\alpha, \beta) = 0 \Leftrightarrow \alpha = \beta, \quad (3.23)$$

$$V(\alpha, \beta) = V(\alpha, \gamma) + V(\gamma, \beta). \quad (3.24)$$

$\alpha$ -expansion algorithm in the GC optimization is shown in Figure 3.1. Steps 5 and 7 are iterated for every disparity value. The loop that begins in step 3 is successful if any disparity level in the iteration lowers the energy function. Algorithm terminates when no

---

```

1 Start with random disparity  $f$ ;
2 Set vector  $success[:] = 0$ ;
3 foreach disparity  $\alpha$  do
4   if not  $done[\alpha]$  then
5     Find a better  $\alpha$  matching  $f^*$  from  $f$  that decrease the energy;
6      $f^* \leftarrow \operatorname{argmin}_{f', \alpha\text{-expansion}} E(f')$ ;
7     if  $E(f^*) < E(f)$  then
8        $f \leftarrow f^*$ ;
9        $success[:] = 1$ ;
10    end
11    if  $success = 1$  then
12      goto step 2;
13    end
14    return  $f$ ;
15  end
16 end

```

---

**Figure 3.1.** Graph cut expansion move algorithm

improvement is found or in other words, the algorithm can not find any disparity in step 3 that has lesser energy value.

Step 5 is a crucial part of the expansion move algorithm. To implement the expansion move, a weighted graph  $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$  with two vertices or two terminals is constructed. A cut  $C \subset \mathcal{E}$  is defined as a set of edges that separates two terminals from each other and forms a new induced graph  $\mathcal{G}(C) = \langle \mathcal{V}, \mathcal{E} - C \rangle$ . The cost of  $C$  is the sum of its edge weights. GC minimization for stereo is proposed as a minimum cut problem, where the solution is the cut with the smallest cost.

Expansion move algorithm proceeds in two stages. First, it takes input partition with disparities  $d$  and  $\alpha$ , then the algorithm seeks to find the disparity that minimizes the energy  $E(x, y, d)$ .  $d'$  is within one  $\alpha$  expansion of  $d$ . Basically, the value of any pixel  $(x, y)$  with disparity  $d$  is changed to  $\alpha$  or it remains unchanged in case of non-decreasing energy.

### 3.5. Post-processing (Disparity refinement)

The last step of stereo algorithms is disparity refinement, and in most cases, it is as important as aggregation or any other steps. An interesting fact is that even after a simple aggregation using the box filter refining the result with a weighted median filter can achieve competitive results. Moreover, this refinement can be implemented in constant time to reduce computational cost. The first and the most important step of refinement is left-right consistency (LRC) checking and generally it is referred to as pre-processing. LRC catches outliers (usually in the form of occlusion, mismatches, etc.) and defines these pixels as an invalid set. If any pixel in the left image does not have exact correspondence, it is marked as an invalid or occluded pixel.

After determining the outliers with LRC, a new disparity is assigned to them. More specifically, a valid lowest neighboring (in the same scanline) pixel disparity is attached to the invalid pixel. Propagation of spatially adjacent pixel's disparity does not guarantee the most accurate result and it leads to some streak like effects on the final disparity map. To prevent this unwanted effect, a weighted median filter is applied only to invalid pixels.

#### 3.5.1. Hole filling (Densification process)

After aggregation, a raw disparity map still contains outliers and unreliable disparity drops in some regions, especially around object discontinuities. Basically, this sudden drops in disparities form a hole like structure and they can be determined by checking the nearest pixels. Hole filling cleans the holes from the disparity map. By this process, disparities of occluded pixels are changed, while non-occluded pixels keep their initial disparity values. Holes with unidentified disparities are given new disparity values based on the smallest disparity across vertical and horizontal neighboring pixels. Simple hole filling process is given by

$$d_{\text{fill}}(x,y) := \max \left\{ \begin{array}{l} d \left( x+ \min_{\substack{u \leq 0 \\ (x+u, y) \text{ is nonoccluded}}} \{u\}, y \right) \\ d \left( x+ \min_{\substack{u \geq 0 \\ (x+u, y) \text{ is nonoccluded}}} \{u\}, y \right) \end{array} \right\}, \quad (3.25)$$

### 3.5.2. Constant Time Weighted Median Filtering

Median filter cleans out the salt and pepper noise [11]. It changes pixels' disparity with a median value of two adjacent pixels. The filter computes the histogram around the pixel  $(x, y)$  as shown in (3.26);

$$h(x, y, i) = \sum_{(u,v) \in \mathcal{N}(x,y)} \delta(I_L(u, v) - i) \quad (3.26)$$

where  $\mathcal{N}(x, y)$  is the support window around the pixel  $(x, y)$  and  $I_L(u, v)$  is the intensity value of the neighbor pixel,  $i$  is a discrete intensity index between 0 and 255 for 8-bit grayscale images,  $\delta$  is equal to 1 if the argument is 0, otherwise 0. It is equal to 0.

In the unweighted median filter, each neighboring pixel contributes equally and this causes some unwanted structure changes like blurring edges. To overcome this issue, a weighted median filter has been developed, where histogram values are weighted as shown in (3.27):

$$h(x, y, i) = \sum_{(u,v) \in \mathcal{N}(x,y)} w(x, y, u, v) \cdot \delta(I_L(u, v) - i) \quad (3.27)$$

where  $w(x, y, u, v)$  is bilateral weights that are computed based on color differences with neighboring pixels.

An implementation of weighted median filter (WMF) demands high workload ( $O(N^2)$  per pixel) and this is the main obstacle for using it in real-time applications. To resolve the efficiency problem a constant time ( $O(1)$  per pixel) weighted median filtering defined below can be used

$$f(x, y, i) = \delta(I_L(x, y) - i), \quad (3.28)$$

$$h(x, y, i) = \sum_{(u,v) \in \mathcal{N}(x,y)} b(x, y, u, v) f(u, v, i), \quad (3.29)$$

where  $b(x, y, u, v)$  is a box kernel. Instead of a box filter, weights of edge aware bilateral filter given in (3.30) or image-guided weights window can lead to better results

$$w_{x,y,u,v}^{\text{BF}} := \exp\left(-\frac{\|(x,y)-(u,v)\|^2}{\sigma_s^2}\right) \cdot \exp\left(-\frac{\|I_{L,\text{filt}}(x,y)-I_{L,\text{filt}}(u,v)\|_c^2}{\sigma_c^2}\right). \quad (3.30)$$

Incorporating edge aware filter advantages of weights to median filter makes it possible to benefit from both filters. As described earlier, median filter is very useful to clean up the salt and pepper noise that is a common trouble for methods with local aggregation. At the same time, the weights  $w_{x,y,u,v}^{\text{BF}}$  are edge aware and capable to preserve edges, object boundaries, or general image structures.

### 3.5.3. Left-right consistency check

Left-right consistency captures occluded or mismatched pixels by comparing the left disparity map  $d_L(x, y)$  with the right one  $d_R(x, y)$ . The equation in (3.31) tests the reliability of matches and, if any pixel does not pass the comparison, it defined as an occluded or invalid pixel. In (3.31),  $d_L(x, y)$  and  $d_R(x, y)$  denote the left and right disparity images, respectively.

$$d_L(x,y) \neq d_R(x+d_L(x,y),y) \quad (3.31)$$

To put the equation in (3.32) to a more general framework, it can be reformulated as in (3.32):

$$|d_L(x,y)+d_R(x+d_L(x,y),y)| \geq d_{LR} \quad (3.32)$$

where a  $d_{LR} = 1$  is the truncation limit.

## CHAPTER 4 SIMULATION RESULTS AND DISCUSSION

In this section, we give parameters of the proposed approach and show simulation results. All implementations regarding the method are done in C++ (mostly OpenCV library). Additionally, the part involving machine learning was also carried out by using OpenCV's library. We used decision tree classifier of machine learning library of OpenCV 3.2.0. A tree is in classifier mode with 20 tree depth. The disparity range used for all images is 60 and evaluation metric is Bad Matching Pixels (BMP) has 5 disparity level tolerance. In addition, level matching is labeled as incorrect.

The number of methods ( $M=3$ ) is three. They are the state of the art stereo matching methods (image-guided filtering (GF), segment tree (ST) non-local aggregation and GC based global method). For more methods ( $M=6$ ), we can add additional resized aggregations of the same methods. For example, one GF results are taken with window size  $r=12$  and the other with  $r=30$ . These additions give us an opportunity to avoid the insufficiency of large window sizes in high texture scenes. If one GF method with a larger window size fails to preserve the quality in the edge structures, the other GF method with a smaller window size decreases the risk of edge blurring.

Our implementation of ST is based on the source code given in [22], for GF public code in [12] is used and for Global method (GL) the information provided by [1] is used. As all the applications of GF, ST, and GL were carried out by our own implementations, so the results given in this thesis can be slightly different from originals.

In the mixture of different methods, we used GF, ST and GC, using one method with three different settings increases the diversity. For GF, we vary the window size ( $r=3$ ,  $r=12$ , and  $r=30$ ) and constant balance value ( $\alpha = 0,5,0.8,0.1$ ). The overall results are given in Table 4.7. Performance in occluded regions is given in Table 4.6, where the proposed method produces a low error rate. In the same manner, for ST we use three different values for the dissimilarity constant ( $\sigma = 0.05, 0.1, 0.15$ ). The last method with different settings is GC (Graph Cut based stereo model) having three occlusion terms ( $K=80, 40, 10$ ). The results in Table 4.2 and Table 4.3 competitive with other top-ranked algorithms. In one way the proposed approach similar to method in [35], in

**Table 4.1.** Average percentage of BMPs of Aloe, Baby2 and Teddy, (left images) with post-processing (WMF) (r: kernel size of WMF)

<i>Method</i>	<i>Aloe</i>		<i>Baby2</i>		<i>Teddy</i>	
<i>Optimization</i>	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>
Combined (r=9)	30,76	30,23	40,68	40,92	17,90	17,94
Combined+GCP	-	30,23	-	40,92	-	26,38

**Table 4.2.** Overall error (left disparity) per method number for Baby2, Aloe and Teddy stereo pairs.

<i>Test data</i>	<i>Aloe</i>		<i>Baby2</i>		<i>Teddy</i>	
<i>Optimization</i>	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>
GF	32,50	-	28,74	-	16,00	-
ST	35,63	-	27,43	-	17,42	-
GL	38,66	-	33,08	-	18,87	-
<b><i>Combined</i></b>	<b><i>33,01</i></b>	<b><i>32,58</i></b>	<b><i>28,48</i></b>	<b><i>28,44 %</i></b>	<b><i>15,64</i></b>	<b><i>15,41</i></b>

which aggregation takes place across multiple scales from the finest to the lowest resolution.

We use the widely used Middlebury dataset to test the proposed approach. This dataset includes image pairs such as Aloe, Baby2, Midd2, Plastic, Bowling2, Teddy together with ground truth disparity maps. Different methods are compared by using the Aloe, Baby2 and Teddy.

Here, we use percentage of bad pixels (PBP) as the accuracy measure Table 4.1 and Table 4.2 show that the estimations provided by proposed method for image pairs Aloe, Baby2 and Teddy are as effective as state of the art methods. The adaptive nature of the combined aggregation gives reliable predictions in some images for which most methods are not successful.

<b>Table 4.3.</b> Average error rate (%) of 17 image pairs from Middlebury stereo dataset provided by different methods.					
<i>Method</i>	<i>GF</i>	<i>ST</i>	<i>GL</i>	<i>Combined</i>	
<i>Optimization</i>	<i>WTA</i>	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>
Raw results	28,21	25,45	30,24	26,16	26,05
With post-processing	20,16	22,02	24,64	20,29	18,69
GCP added results	-	-	-	27,12	30,34
GCP added results (with post-processing)	-	-	-	20,29	24,63

For each average results are shown in Table 4.3 that shows the average error rate over 17 stereo pairs. The key advantage of the proposed method is the ability to generate low average error results. Quantitative results show that among the four different methods the proposed approach ranks second based on the accuracy. The top average result is obtained from ST method, but its visual performance are not good. For example, Rocks 1 and Rocks 2 image pairs in Figures 4.2 and 4.3 have some sudden disparity drops in flat regions. The same remarks in are valid for Cloth1 and Baby3 image pairs in Figure 4.1. On the other hand, the global method that uses GC has an accurate prediction on the less textured region but has poor prediction for edges. The disparity map generated by the proposed adaptive approach combines positive characteristics of each method. As a result, a better visual and low error disparity map is produced. There is no significant difference between WTA and MRF optimizations in raw disparity images, but there are some minor improvements in the MRF method after post-processing.

**Table 4.4.** Error rates of GF, ST, GL and the proposed method for several stereo pairs.

<i>Method</i>	<i>GF</i>	<i>ST</i>	<i>GL</i>	<i>Combined</i>		<i>Combined (with post-processing)</i>	
				<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>
Baby3	19,70	14,97	25,70	19,60	19,59	13,98	13,74
Cloth1	8,89	9,15	10,90	8,74	8,70	2,93	2,92
Flowerpots	23,30	20,40	23,31	22,72	22,77	15,62	15,46
Lampshade2	28,18	20,62	23,58	17,75	17,45	7,29	6,65
Midd1	35,04	35,25	28,96	27,60	27,27	14,96	14,74
Monopoly	24,34	28,80	32,71	26,46	26,13	26,81	26,92
Plastic	37,90	20,59	27,45	25,23	24,72	8,24	7,04
Rocks1	11,71	11,36	16,20	12,43	12,40	7,92	7,87
Rocks2	13,09	12,23	17,57	13,28	13,22	6,02	5,96
Wood2	30,50	22,91	26,40	23,95	23,82	18,86	18,47
Art	44,22	41,80	52,16	45,03	44,99	40,29	40,59
Books	40,73	39,56	45,42	42,40	42,48	41,87	41,75
Dolls	42,90	42,09	44,43	40,81	40,71	36,62	36,33
Laundry	37,68	37,29	43,92	41,08	40,89	38,64	15,05
Moebius	19,24	18,29	23,98	19,18	19,14	13,67	13,54
Reindeer	24,69	21,64	28,03	21,54	21,35	12,23	11,62
Baby1	37,54	35,82	43,46	37,05	37,38	39,07	39,09
Average	28,21	25,45	30,24	26,16	26,05	20,29	18,69

Further enhancements can be achieved by using WMF. The corresponding results are shown in Table 4.4. Clearly, addition of post-processing slightly decreases the error rate. The results in Table 4.4 derived from the application of post-processing show that the proposed method with weighted median filtering provides the most accurate results.

**Table 4.5.** Error rates of the proposed method with GCPs are added for several stereo pairs.

<i>Method</i>	<i>Combined (GCP)</i>		<i>Combined (GCP with post-processing)</i>	
	<i>WTA</i>	<i>MRF</i>	<i>WTA</i>	<i>MRF</i>
<i>Baby3</i>	19,85	19,84	13,98	12,80
<i>Cloth1</i>	8,86	8,83	2,93	2,95
<i>Flowerpots</i>	22,99	22,96	15,62	15,80
<i>Lampshade2</i>	26,69	26,58	7,29	15,70
<i>Midd1</i>	34,00	34,11	14,96	34,85
<i>Monopoly</i>	24,72	24,54	26,81	21,69
<i>Plastic</i>	29,50	28,99	8,24	14,64
<i>Rocks1</i>	11,73	11,70	7,92	6,23
<i>Rocks2</i>	13,03	13,01	6,02	5,49
<i>Wood2</i>	24,50	24,38	18,86	18,40
<i>Art</i>	43,99	43,99	40,29	37,39
<i>Books</i>	42,76	42,86	41,87	41,13
<i>Dolls</i>	41,55	41,50	36,62	37,16
<i>Laundry</i>	37,92	93,58	38,64	93,58
<i>Moebius</i>	19,16	19,12	13,67	12,73
<i>Reindeer</i>	22,80	22,76	12,23	11,29
<i>Baby1</i>	37,12	37,08	39,07	36,88
<i>Average</i>	27,12	30,34	20,29	24,63

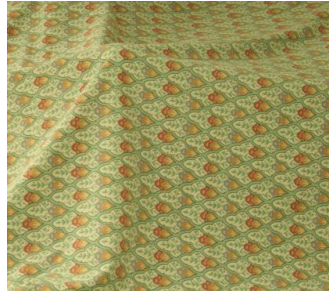
Baby3

Cloth1

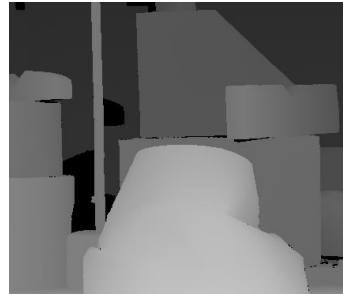
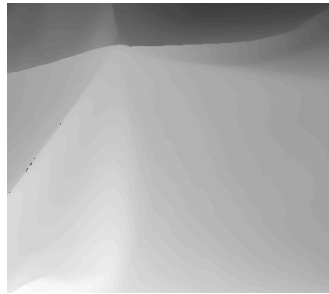
Lampshade1

Flowerpots

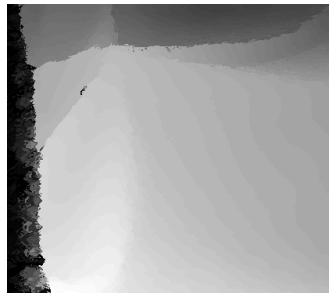
a)



b)



c)



19,70

8,89

25,03

23,30

d)

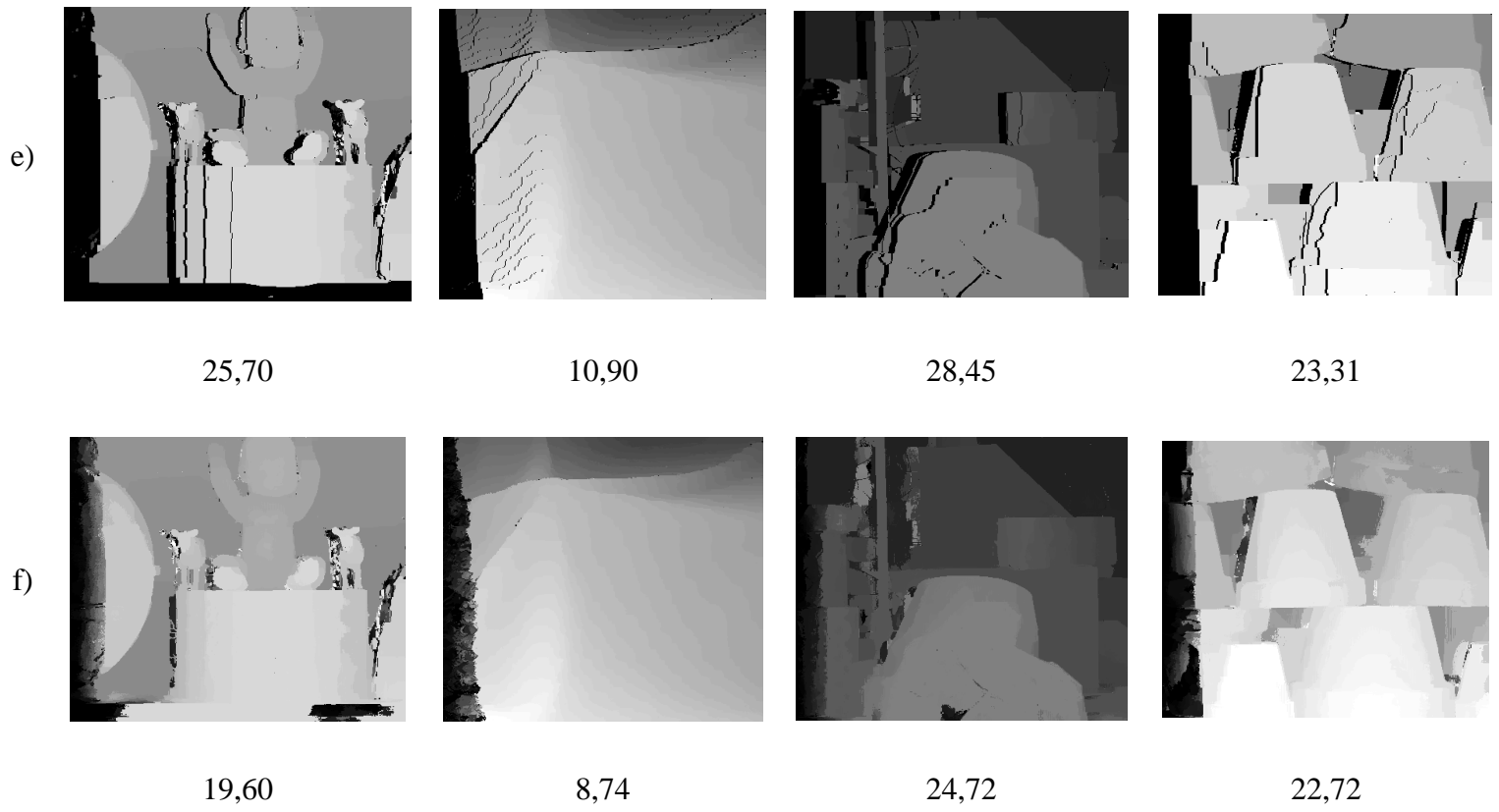


14,97

9,15

18,30

20,40



**Figure 4.1.** Results for Baby3, Cloth1, Lampshade1 and Flowerpots a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF, ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs.

Lampshade2

Monopoly

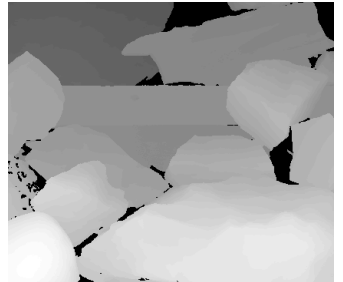
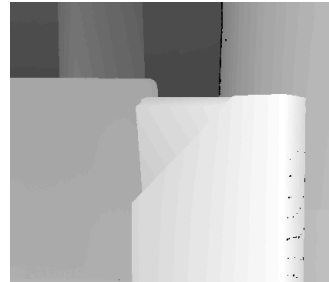
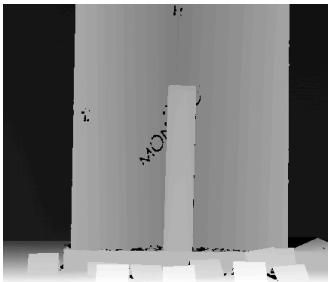
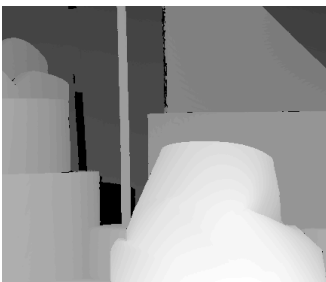
Plastic

Rocks1

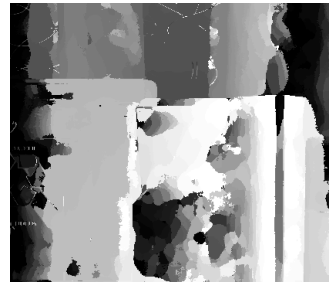
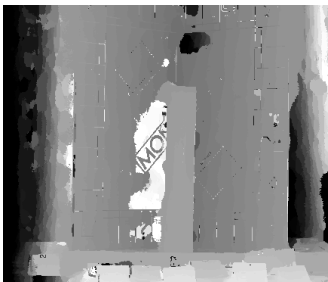
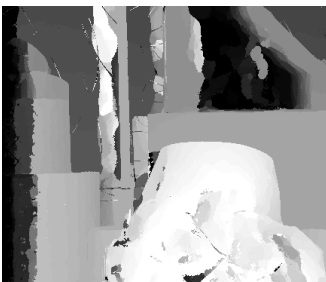
a)



b)



c)



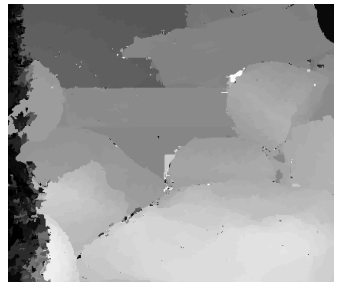
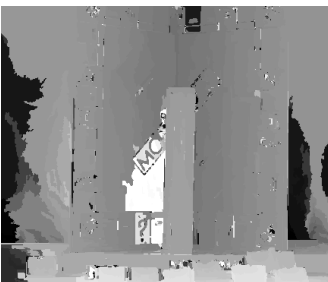
28,18

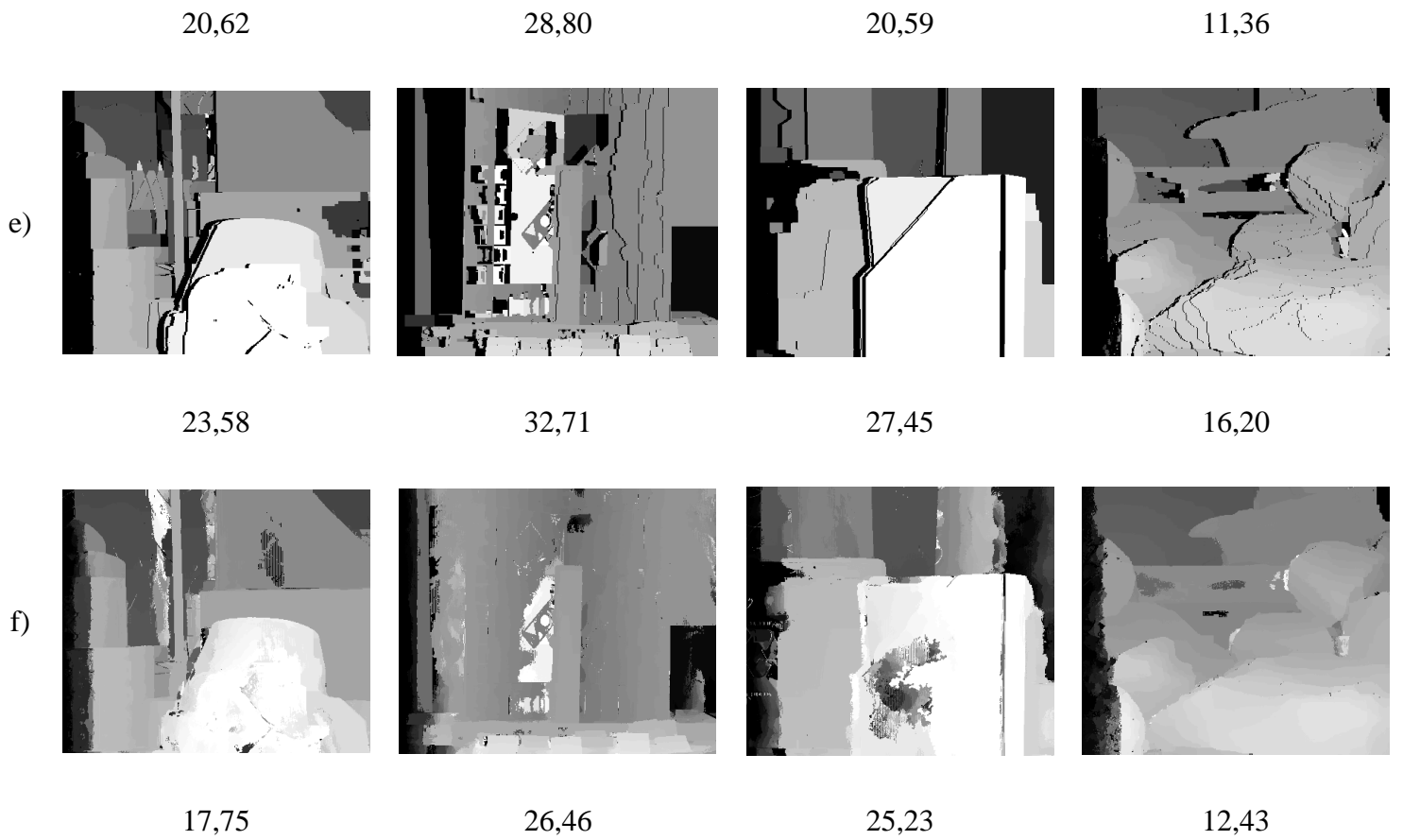
24,34

37,90

11,71

d)





**Figure 4.2.** Results for Lampshade2, Monopoly, Plastic and Rocks1 a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF, ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs.

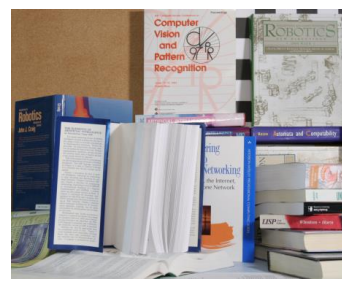
Rocks2

Wood2

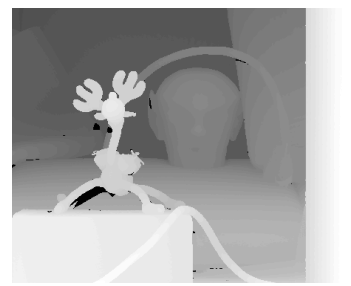
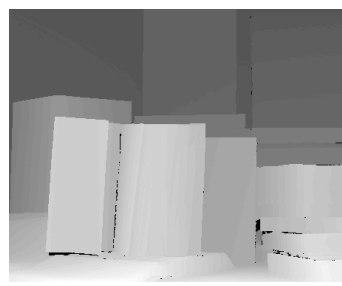
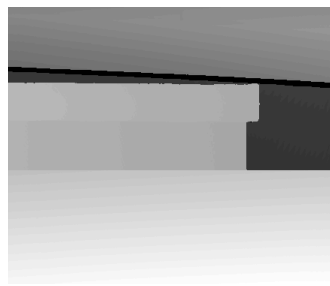
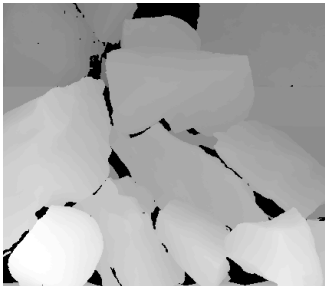
Books

Reindeer

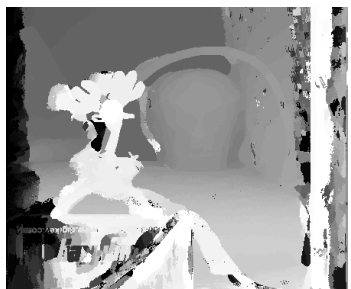
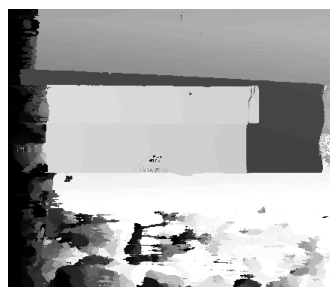
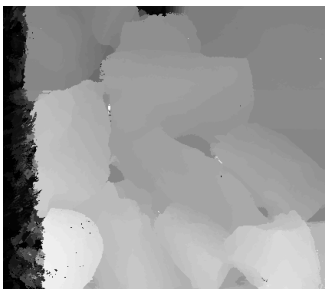
a)



b)



c)



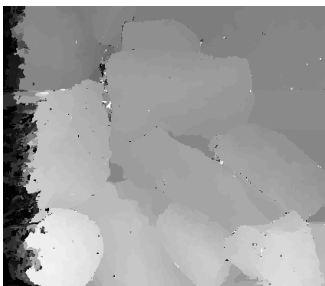
13,09

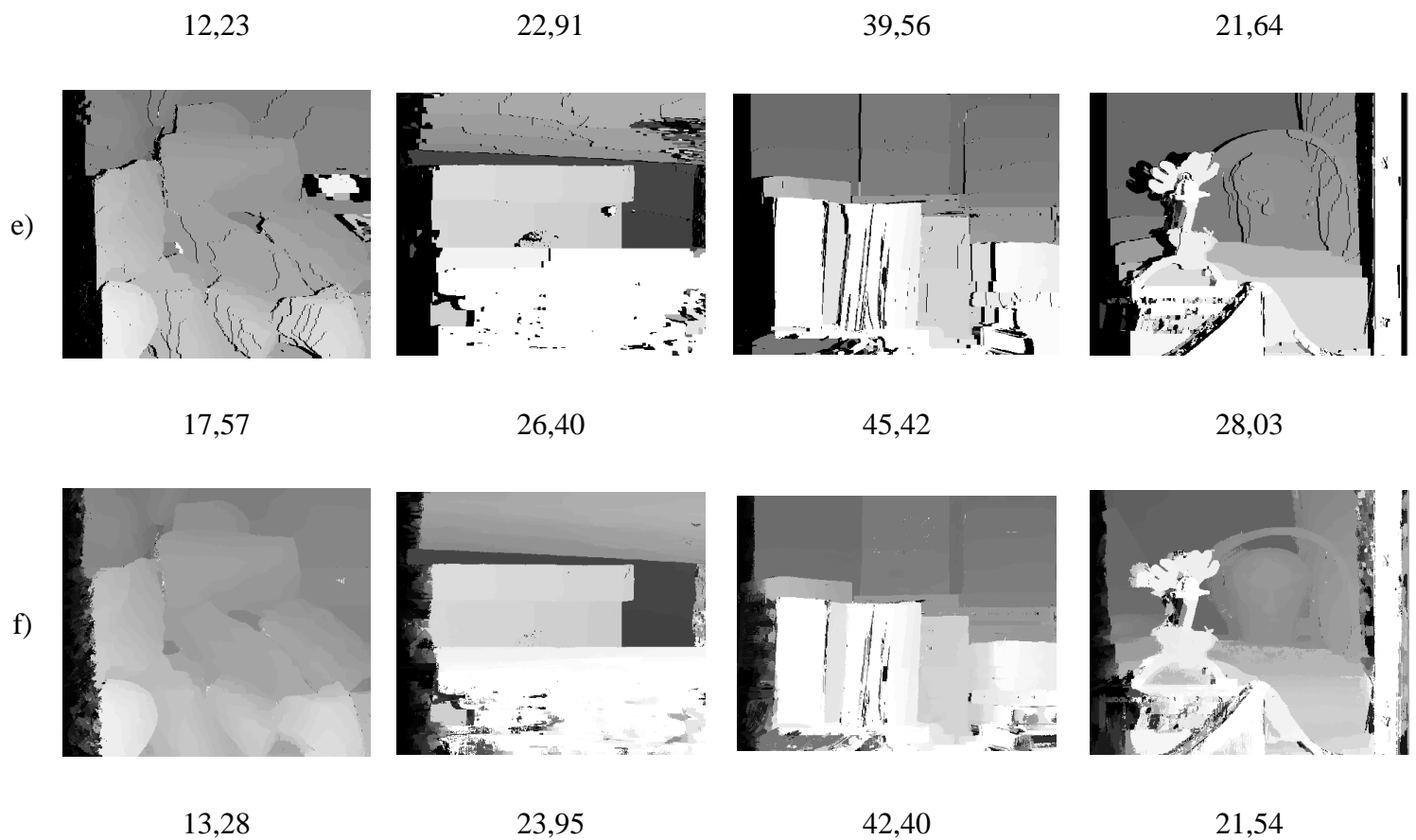
30,50

40,73

24,69

d)



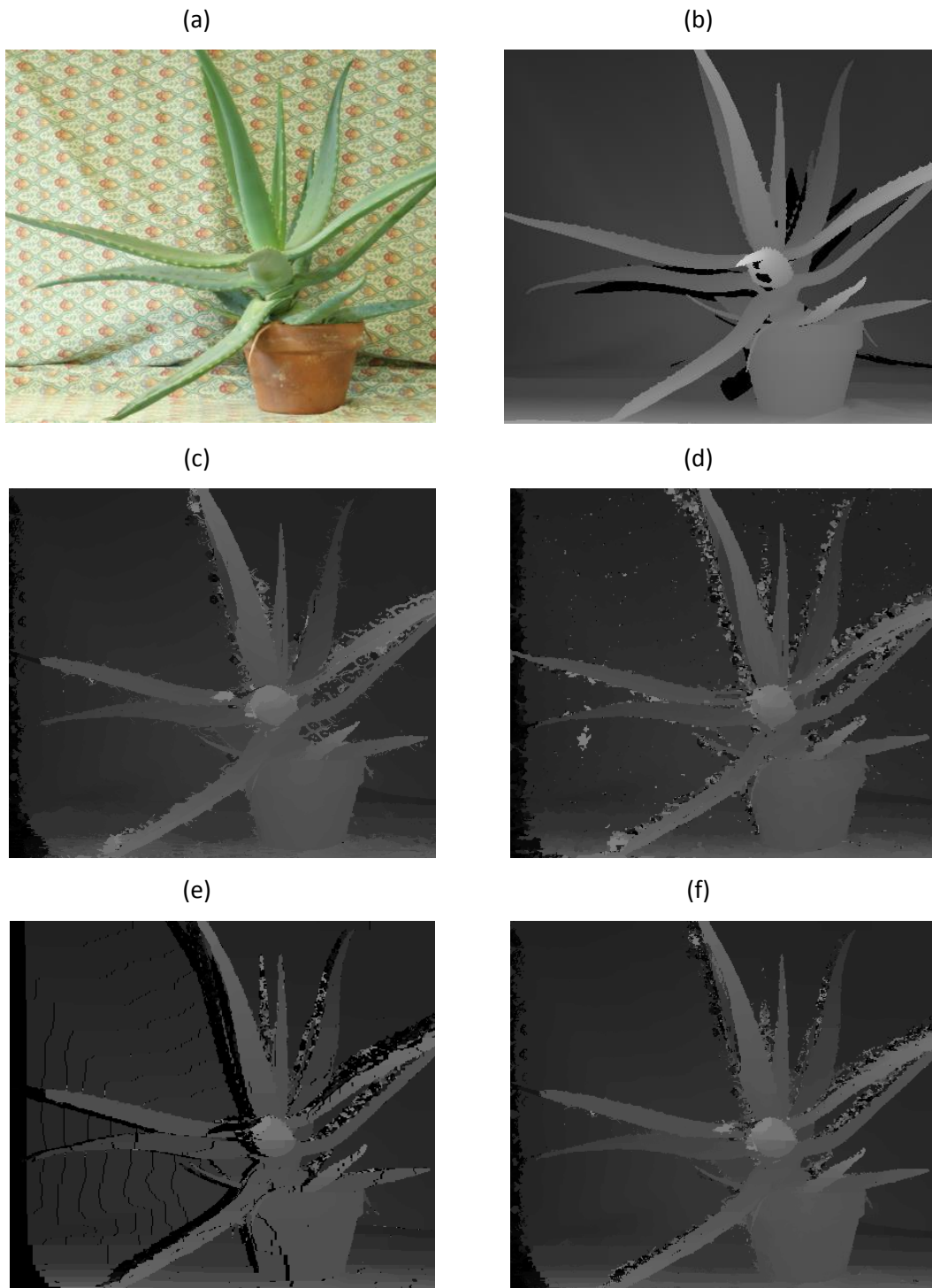


**Figure 4.3.** Results for Rocks2, Woods2, Books and Reindeer a) Left image, b) True disparity maps, (c) through (f) disparity maps obtained by GF, ST, GC and the proposed method respectively. Numbers below the images denote the percentage of BMPs.

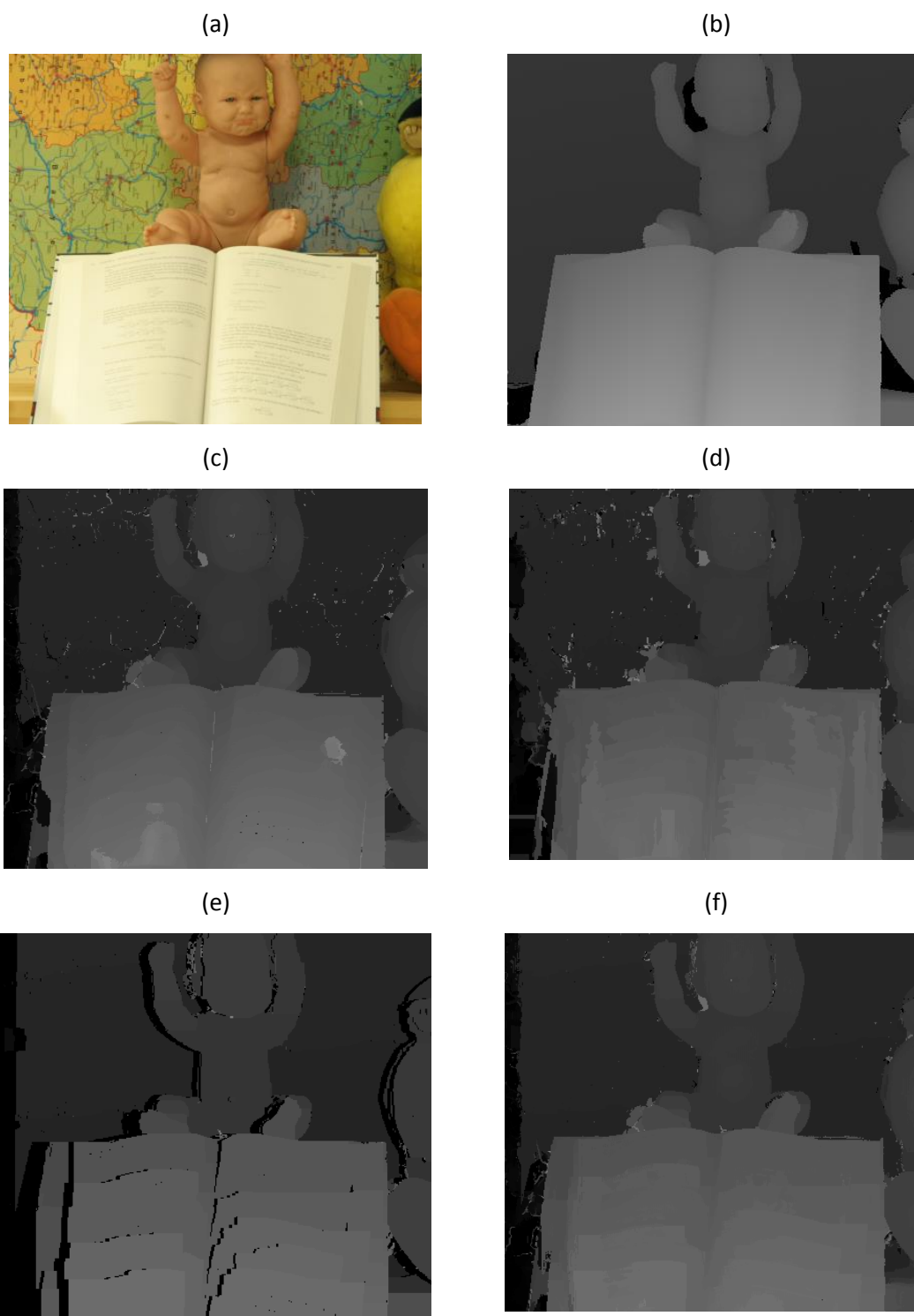
At this point, we analyze the quantitative results. Moreover, the post-processing technique based on the median filter is integrated to improve remaining artifacts after the aggregation step. A weighted median filter is applied to invalid pixels in order to avoid the efficiency problem since computation of WMF on all pixels requires a lot of time. Percentage of BMPs is determined by using the ground truth disparity maps. Figure 4.4 shows results of combined aggregation and metrics where clearly this method has accurate disparity images alongside with GF and ST. The stereo image pairs from Middlebury dataset generally have two types of image structures: flat regions with lower varieties in depth and textured regions containing abundant edges. For flat structure case, ST method has potentially better handling of low textured regions because of its non-locality. Disparity map for Baby2 image pair supports this observation. On the other hand, GF produces accurate estimation for highly textured image scenes due to its edge

preserving characteristics. Disparity map for Aloe image pair is a good example supporting this fact. Baby2 image contains both image structures (flat, textured). GF and GL fail to deliver satisfying results. However, the proposed method manages to get a satisfying matching error and disparity map.

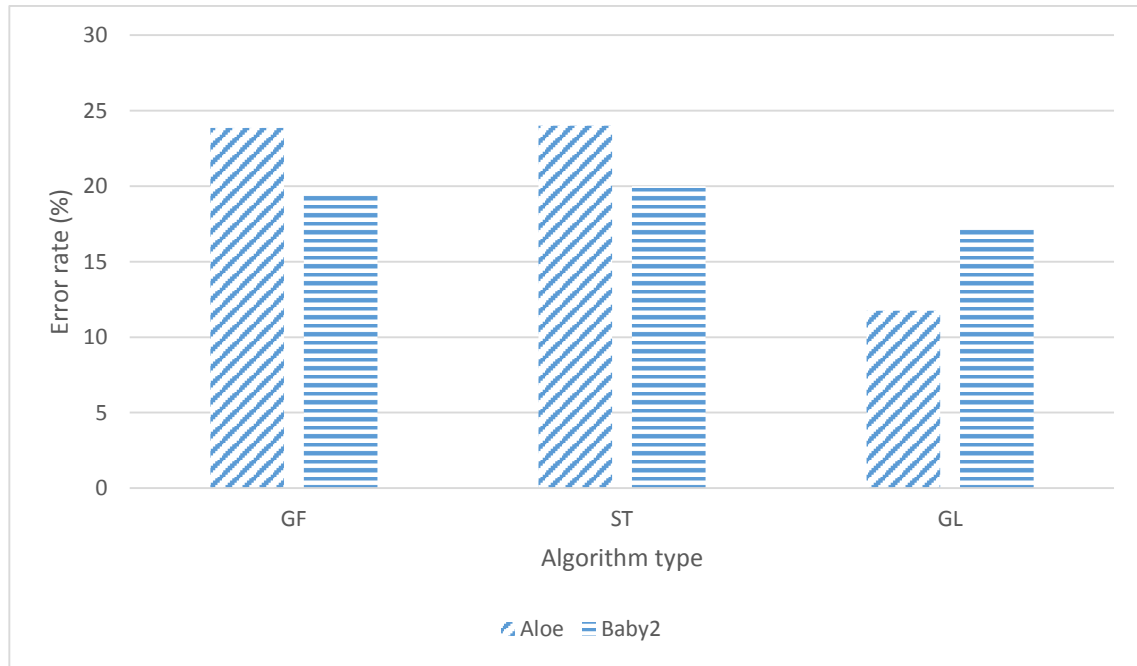
We evaluate different methods closely in Figures 4.4 and 4.5 that show disparity maps obtained by GF, GL, ST, and the proposed methods together with their matching error rates. The proposed approach gives a satisfying result even though other methods provide poor disparity maps. The weak estimation of each models is diminished by using a combined model with mixing weights. It is obvious from Figure 4.5 that GF and ST fail in flat regions while GL has more accurate depth in those areas. The reason behind this is that the global methods treat the stereo problem as piecewise smoothing. Again the GL method discriminates occluded region (marked in black color) better than the other two methods. Although GL copes well with textureless regions, it has difficulties in the edges. To overcome this issue GF is added to the set of local methods because it has edge preserving property avoiding edge blurring. There is a chance of getting good estimations with GF in flat regions, but this usually costs in the degradation of quality in textured areas.



**Figure 4.4.** Disparity maps obtained by different methods for Aloe image pair (a) left image, (b) the true disparity map, (c) through (f) disparity maps obtained by GF (32,50), ST (35,63), GL (38,66) and the proposed (32,58) methods, respectively. Numbers in brackets are percentage of BMPs.



**Figure 4.5.** Disparity maps obtained by different methods for Baby2 image pair (a) left image, (b) the true disparity map, (c) through (f) disparity maps obtained by GF (28,74), ST (27,43), GL (33,08) and the proposed (28,44) methods, respectively. Numbers in brackets are percentage of BMPs.



**Figure 4.6.** Average classification error per method for Aloe and Baby2 image pairs.

The overall performance of the proposed method depends on decision tree accuracy or the capacity to predict the correct disparity assignment. Figure 4.6 presents the prediction accuracy for different methods for Baby2 and Aloe image pairs. The GL method has the highest prediction accuracy, the GF and ST have similar higher error rates. If the classifier provides high confidence estimates, the proposed method will be robust, otherwise, increasing in APBP is inevitable. Consequently, a better and more accurate classifier will improve the disparity map of the proposed method. We limited ourselves to built-in machine learning models, but there exists many other options to construct a tree or random forest classifier with a larger tree depth. Due to software restriction, we could use only 20 tree depth yielding pleasing results.

**Table 4.6.** Error rates (left disparity) for Aloe, Baby2 and Teddy image pairs in occluded regions without post-processing.

<i>Method</i>	<i>Aloe</i>		<i>Baby2</i>		<i>Teddy</i>	
	<i>Occ.</i>	<i>Nonocc.</i>	<i>Occ.</i>	<i>Nonocc.</i>	<i>Occ.</i>	<i>Nonocc.</i>
<b><i>Combined</i></b>	<b><i>11,93</i></b>	<b><i>21,08</i></b>	<b><i>11,02</i></b>	<b><i>17,46</i></b>	<b><i>15,31</i></b>	<b><i>0,33</i></b>
GF	11,49	21,01	11,84	16,90	15,71	0,29
ST	13,40	22,23	10,42	17,01	16,48	0,94
GL	16,50	22,16	13,67	19,41	18,48	0,39

Finally, Table 4.6 and Table 4.7 show the error rates in occluded area for all methods. In Table 4.7 the same method with different parameters are considered as separate models. The proposed method has an accurate estimations for all pairs. It ranks second among all the methods in occluded regions. Note that GL method has a very poor result for Teddy image pairs. While the proposed method leads to higher accuracy.

<b>Table 4.7.</b> Error rates (left disparity) for Teddy and Baby2 image pairs in occluded regions without post-processing.				
<i>Method</i>	<i>Teddy</i>		<i>Baby2</i>	
	<i>Occ.</i>	<i>Nonocc.</i>	<i>Occ.</i>	<i>Nonocc.</i>
GF (r=3)	18,06	3,25	16,02	15,87
GF (r=12)	12,17	2,97	12,32	16,53
GF (r=30)	10,78	4,04	11,21	17,86
<b><i>Combined (GF)</i></b>	<b><i>13,91</i></b>	<b><i>2,99</i></b>	<b><i>13,17</i></b>	<b><i>15,72</i></b>
ST ( $\sigma = 0,05$ )	14,41	3,01	17,88	9,55
ST ( $\sigma = 0,1$ )	14,41	3,01	32,10	38,17
ST ( $\sigma = 0,15$ )	25,81	51,16	32,10	38,17
<b><i>Combined (ST)</i></b>	<b><i>14,48</i></b>	<b><i>3,41</i></b>	<b><i>28,16</i></b>	<b><i>10,21</i></b>
GL (K=20)	15,43	4,41	31,53	1,30
GL (K=40)	13,72	3,56	31,83	1,40
GL (K=80)	13,02	5,34	33,89	1,47
<b><i>Combined (GL)</i></b>	<b><i>13,18</i></b>	<b><i>4,40</i></b>	<b><i>31,71</i></b>	<b><i>1,30</i></b>

## CHAPTER 5 CONCLUSIONS AND FUTURE WORKS

Theoretically, estimating depth image using two pairs of perfectly calibrated cameras can be done by triangulation. Matching measures are assumed to be noise or ambiguity free. In practice, however, there will be some interference like reflection, the inconsistency of intensities or simply wrong correspondence calculations. In stereo vision literature, there exist many studies addressing previously mentioned problems. Especially edge preserving methods have attracted more attention because of their ability to avoid edge blurring. Edge-aware and image-guided aggregation is one of the top performing stereo methods for which the runtime is totally independent of the window size. Despite the high-quality depth map of GF, it produces poor results in flat image regions. We proposed a method based on GC that performs better in the flat image scenes and also has special occlusion handling energy term. Benefiting from each method separately inspired us to work on this topic. In other words, individual stereo methods were ensembled giving a combined stereo approach.

The results suggest that adaptive (combined) form of solving a stereo vision problem ensures the lowest or a low error rate compared with the state of art algorithms like ST, GF, or GL. We used model mixing weights to obtain the likelihood of each method. Existing adaptive approaches rely on local filter weights and filtering is implemented over three-dimensional cost volume. However, there are plenty of top performing global methods that do not use any filtering technique. Instead, they formulate an energy function and minimize it. We developed an idea allowing the addition of a global model to the pool of local models. Our study provides a new framework for integrating local and global methods.

For a future work, we plan to focus on efficiency of implemetations so that the proposed method can be applied in real time for practical applications..

## REFERENCES

1. Kolmogorov, V., Monasse, P., Pauline, T., (2014). Kolmogorov and Zabih's Graph Cuts Stereo Matching Algorithm. *Image Processing On Line*, Vol. 4, pp. 220-251.
2. Hu, X., Mordohai, P., (2012). A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 34 (No. 11), pp. 2121-2133.
3. Spyropoulos, A., Komodakis, N., Mordohai, P., (2014). Learning to Detect Ground Control Points for Improving the Accuracy of Stereo Matching. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1621-1628.
4. Park, M. G., Yoon, K. J., (2015). Leveraging Stereo Matching with Learning-based Confidence Measures. *Computer Vision and Pattern Recognition (CVPR) 2015 IEEE Conference*, pp. 101-109.
5. Breiman, L., (2001). Random Forests. *Machine Learning*, Vol. 45 (No. 1), pp. 5-32.
6. Choi, O., Chang, H. S., (2016). Yet Another Cost Aggregation Over Models. *IEEE Transactions on Image Processing*, Vol. 25 (No. 11), pp. 5397-5410.
7. Zhao, J., Yu, S., Cai, H., (2006). Local-global stereo matching algorithm. *Aircraft Engineering and Aerospace Technology: An International Journal*, Vol. 78 (No. 4), pp. 289-292.
8. Shi, C., Wang, G., Yin, X., Pei, X., Lin, X., (2015). High-Accuracy Stereo Matching Based on Adaptive Ground Control Points. *IEEE Transactions on Image Processing*, Vol. 24 (No. 4), pp. 1412-1423.
9. Wang, L., Yang, R., (2011). Global stereo matching leveraged by sparse ground control points. *IEEE (CVPR)*, pp. 3033-3040.
10. Sun, X., Mei, X., Jiao, S., Zhou, M., Wang, H., (2011). Stereo Matching with Reliable Disparity Propagation. *2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pp. 132-139.

11. Ma, Z., He, K., Wei, Y., Sun, J., Wu, E., (2013). Constant time weighted median filtering for stereo matching and beyond. *IEEE (CVPR) 2015*, pp. 1621-1628.
12. Hosni, A., Rhemann, C., Bleyer, M., Rother, C., Gelautz, M., (2013). Fast Cost-Volume Filtering for Visual Correspondence and Beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35 (No. 2), pp. 504-511.
13. Yoon, K.J., Kweon, I.S., (2006). Adaptive Support-Weight Approach for Correspondence Search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28 (No. 4), pp. 650-656.
14. Zhang, K., Lu, J., Lafruit, G., (2009). Cross-based local stereo matching using orthogonal integral images. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 19 (No. 7), pp. 1073 - 1079.
15. Zhan, Y., Gu, Y., Huang, K., Zhang, C., Hu, K., (2016). Accurate Image-Guided Stereo Matching With Efficient Matching Cost and Disparity Refinement. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 26 (No. 9), pp. 1632-1645.
16. Kordelas, G. A., Alexiadis, D. S., Daras, P., Izquierdo, E., (2016). Content-based guided image filtering, weighted semi-global optimization. *IEEE Transactions on Multimedia*, Vol. 18 (No. 2), pp. 155-170.
17. Li, L., Zhang, C., Yan, H., (2011). Stereo Matching Algorithm Based on a Generalized Bilateral Filter Model. *Journal of software*, Vol. 6 (No. 10), pp. 1906-1913.
18. Zhu, S., Wang, Z., Zhang, X., Li, Y., (2016). Edge-preserving guided filtering based cost aggregation for stereo matching. *Journal of Visual Communication and Image Representation*, Vol. 39, pp. 107-119.
19. Yang, Q., Ji, P., Li, S., Yao, S., Zhang, M., (2014). Fast stereo matching using adaptive guided filtering. *Image and Vision Computing*, Vol. 32 (No. 3), pp. 202-211.
20. Yang, Q., (2015). Stereo matching using tree filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 37 (No. 4), pp. 834-846.

21. Chen, D., Ardabilian, M., Wang, X., Chen, L., (2013). An improved nonlocal cost aggregation method for stereo matching based on color and boundary cue. 2013 IEEE International Conference on Multimedia and Expo (ICME).
22. Mei, X., Sun, X., Dong, W., Wang, H., Zhang, X., (2013). Segment-Tree Based Cost Aggregation for Stereo Matching. IEEE Conference on Computer Vision and Pattern Recognition.
23. Xiang, X., Zhang, M., Li, G., He, Y., Pan, Z., (2012). Real-time stereo matching based on fast belief propagation. *Machine Vision and Applications*, Vol. 23 (No. 6), pp. 1219-1227.
24. Yang, Q., Wang, L., Yang, R., Stewenius, H., Nister, D., (2009). Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31 (No. 3), pp. 492-504.
25. Wang, H.Q., Wu, M., Zhang, Y. B., Zhang, L., (2013). Effective stereo matching using reliable points based graph cut. 2013 Visual Communications and Image Processing (VCIP).
26. Wang, Z. F., Zheng, Z. G., (2008). A region based stereo matching algorithm using cooperative optimization. 2008 IEEE Conference on Computer Vision and Pattern Recognition.
27. Huan, M., Wang, K., Zuo, W., Li, Z., (2011). Template Based Stereo Matching Using Graph-cut. 2011 First International Conference on Instrumentation, Measurement, Computer, Communication and Control.
28. Daolei, W., Lim, K. B., (2011). Obtaining depth map from segment-based stereo matching using graph cuts. *Journal of Visual Communication and Image Representation*, Vol. 22 (No. 4), pp. 325-331.
29. Zhang, K., Fang, Y., Min, D., Sun, L., Yang, S., Yan, S., Tian, Q., (2014). Cross-scale cost aggregation for stereo matching. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1590-1597.

30. Mozerov, M., Weijer, J. V. d. W., (2015). Accurate Stereo Matching by Two-Step Energy Minimization. *IEEE Transactions on Image Processing*, Vol. 24 (No. 3), pp. 1153-1163.
31. Xue, H., Cai, D., (2016). Stereo Matching by Joint Energy Minimization. *arXiv.org* > cs > arXiv:1601.03890.
32. Hartley, R., Zisserman, A., (2004). *Multiple view geometry*. s.l. : Cambridge University Press, Cambridge, UK.
33. Birchfield, S., (1998). A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20 (No. 4), pp. 401-406.
34. Hirschmuller, H., Scharstein, D., (2008). Evaluation of Stereo Matching Costs on Images with Radiometric Differences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31 (No. 9), pp. 1582 - 1599.
35. Zhang, K., Fang, Y., Min, D., Sun, L., Yang, S., Yan, S., (2017). Cross-scale cost aggregation for stereo matching. *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 27 (No. 5), pp. 965 - 976.
36. Boykov, Y., Veksler, O., Zabih, R., (2001). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol 23 (No. 11), pp. 1222 - 1239.

## **ÖZGEÇMİŞ**

**MAKSAT YLYASOV**

Phone: +90-534-871-6487

maksat2192008@gmail.com

### **EDUCATION:**

**B.A, Electronics Engineering, Uludag University, Bursa, Turkey 2014**

### **RESEARCH EXPERIENCE:**

**Undergraduate Project, Uludag University, 2013-2014**

Design of general filter framework.

Analysing of White Gaussian noise in mobile communication channel.

Design of Flope filter.

### **GRANTS AND FELLOWSHIPS:**

Turkiye undergraduate scholarships (Uludag University 2008-2014)

### **RELEVANT SKILLS**

- Programming ability in C/C++, Java, Matlab, Delphi, C#, Assembly.
- Fluent in English, Russian, and Turkish.