Dense Disparity Estimation using Stereo Images

by

SONAM NAHAR
201021013

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
in
INFORMATION AND COMMUNICATION TECHNOLOGY
to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY

April, 2017

Declaration

I hereby declare that

i) the thesis comprises of my original work towards the degree of Doctor of Philosophy in Information and Communication Technology at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,

ii) due acknowledgment has been made in the text to all the reference material used.

Sonam Nahar

Certificate

This is to certify that the thesis work entitled DENSE DISPARITY ESTIMATION USING STEREO IMAGES has been carried out by SONAM NAHAR for the degree of Doctor of Philosophy in Information and Communication Technology at *Dhirubhai Ambani Institute of Information and Communication Technology* under my supervision.

Prof. Manjunath V. Joshi
Thesis Supervisor

Acknowledgments

First and foremost, I would like to thank the God, the Almighty, for giving me the strength, knowledge, ability, and opportunity to undertake this research study and to preserve and complete it successfully. Without his blessings and support, this achievement would not have been possible. The thesis appears in its current form due to the assistance, guidance, support and blessings of many wonderful people. I would, therefore, like to offer my sincere thanks to all of them.

I would like to express my sincere gratitude to my supervisor **Prof. Manjunath V. Joshi** for the guidance, encouragement, constant feedback, immense knowledge, motivation, and advice he has provided me throughout all the different steps of my doctoral research endeavor for the past few years. In my journey towards this degree, I have found a teacher, a friend, an inspiration, a role model and a pillar of support in him. He has given me all the freedom to pursue my research, while constantly forcing me to remain focused on achieving my goal. I will always remain indebted for his understanding and support during the times when I was really down and depressed due to my poor health and failures in research work. Working with Prof. Joshi has enriched my life not only in the academic sense. His achievements, his work ethics, and his keen eye for every important detail have been an inspiration throughout all the years I have worked with him.

I wish to acknowledge all the professors of DA-IICT who have inspired me directly or indirectly. I would like to sincerely thank the members of my thesis advisory and synopsis committees, **Prof. Suman K. Mitra**, **Prof. Hemant Patil**, **Prof. Sanjeev Gupta**, **Dr. Aditya Tatu** for their warm encouragement and enlightening suggestions at various stages. I express my sincere gratitude to Direc-

tor, **Prof. K. S. Dasgupta**, Ex. Director, **Prof. R Nagaraj**, Ex. Director, **Prof. S.C. Sahasrabudhe**, Registrar, Dean-AP, Dean-R&D, Convener-PGC for their continuous support and guidance throughout my time at DA-IICT. I would like to thank the members of administrative and technical staff, Girls Hostel Warden and Supervisor for helping me in their respective roles. A special mention of thanks to my seniors, friends, and colleagues who have created such a good atmosphere in the lab and helped me in many steps of my study. My special word of thanks goes to **Prakash Gajjar** and **Milind Padalkar** for their ever-ready helping attitude.

I would like to thank the anonymous reviewers and the editors of our publications and the examiners of my thesis for their constructive suggestions that have greatly improved the publications and the thesis, respectively. Moreover, I wish to express my gratitude to **International Association of Pattern Recognition** (IAPR) for awarding me travel grant to present our research paper in 23rd IEEE International Conference on Pattern Recognition (ICPR), 2016.

It's my fortune to gratefully acknowledge the LNM Institute of Information Technology (LNM-IIT), Jaipur where I have been serving as an Assistant Professor since last 5 years. I owe a debt of gratitude to the Director of the LNM-IIT, **Prof. S.S. Gokhale** for his continuous support, cooperation, and providing me numerous opportunities to learn and develop as a teacher. He has always encouraged and provided me the necessary facilities and resources for carrying out my Ph.D. work. I express my heart-felt gratitude to **Prof. Ravi P. Gorthi**, Dean AP, for his friendly nature, constant support and advises throughout my journey. I also express my thanks to the Head of Computer Science Department and all my colleagues in the LNM-IIT for their cooperation and support. I gratefully acknowledge the financial support received from this institute for presenting our papers in international conferences during various stages of my Ph.D.

Finally, I would like to acknowledge the people who mean world to me, my **family**. I owe my deepest gratitude towards my parents **Nirmala** and **Ummed Nahar** for their blessings, showing faith in me and allowing me to realize my own potential. Thank you for encouraging me in all of my pursuits and supporting me emotionally and financially. I salute you for the selfless love, care, pain, patience,

and sacrifice you did to shape my life. My parents' dreams for me have resulted in this achievement and without their support, determination, encouragement, loving upbringing and nurturing, I would not have been where I am today and what I am today. They will remain my strength and inspiration throughout my life. It is true that if god ever existed, he would be in the form of parents because only the parents can love and give without expecting anything in return. I dedicate this milestone to them.

I owe thanks to my life, my husband, **Rahul Bohra** for his continuous and unfailing love, utmost care, support, motivation and understanding during my pursuit of Ph.D. degree that made the completion of thesis possible. Rahul, you are indeed my best friend, soul-mate and life partner. He appreciated and celebrated my success with full joy at various stages of my Ph.D. work. Besides his busy schedules, he accompanied me for attending the international conferences. He always showed keen interest in knowing about my research progress and discussions with my supervisor. There were times during the past few years when everything seemed hopeless and I was very depressed. I can honestly say that it was his determination, positive attitude and constant encouragement that ultimately made it possible for me to see this Ph.D. work through to the end. I dedicate this achievement to him.

I express my heart-felt regards to my paternal grand parents **Chanchal** and **Ranjeet Mal Nahar** and maternal grand parents **Anoop** and **Balwant Singh Mehta** for their blessings and affection. A special and profound thanks to my elder brother **Saurabh**, sister-in-law **Garima**, younger sister **Nikita**, brother-in-law **Siddharth** and loving kids for their continuous love and support. Thank you *Bhaiya* for expressing confidence in my abilities and encouraging me to follow my dreams. Thank you *NN* for your valuable prayers and always being there for me as a friend in all aspects of my life.

My heart-felt gratitude to my parents-in-law **Poornima** and **Subhash Chand Bohra**, and family for their understanding, support, love, showing pride in me and giving me the liberty to choose what I desired. I would not have been reached the stage of Ph.D. completion without their contribution and utmost support.

Contents

Abstract

"Stereo vision" refers to the ability to infer information on the three-dimensional (3D) structure and distance/depth of a scene using two images captured from different view-points. It imitates one of the tasks performed by the human brain and the two eyes. In the stereo vision, a scene point is projected onto different locations on the two image planes (left and right cameras) and the main goal here is to find the corresponding pixels i.e., pixels resulting from the projection of the same 3D point onto the two image planes. The displacement between corresponding pixels is called "disparity", and obtaining the same at each pixel location forms a dense disparity map. However, estimation of disparities is an ill-posed problem and hence in practice is solved by formulating it as a global energy minimization problem. An energy function represents a combination of a "data term" and a "prior term" that restricts the solution space, and choosing a suitable data as well as prior models lead to accurate dense disparity estimates. In this thesis, we address this problem of dense disparity map estimation using rectified stereo images with known calibration of cameras and propose various approaches for solving it in a global energy minimization framework. We utilize "graph cuts", an efficient and fast optimization technique for minimizing our energy functions.

We first propose a method for dense disparity estimation using inhomogeneous Gaussian Markov random field (IGMRF) prior where we model the disparity map using this prior. The estimated IGMRF parameters assist us to yield a smooth solution while preserving the sharp depth discontinuities. In order to model the data term, we use the pixel-based intensity matching cost which is based on the brightness constancy assumption of the corresponding pixels. A learning based approach is used to obtain an initial disparity map which is used

in obtaining the IGMRF parameters. The dense disparity map is obtained by minimizing the energy function using graph cuts. In this case, the quality of the final solution is strongly governed by the accuracy of the IGMRF parameters.

Though, IGMRF prior captures smoothness with discontinuities, it fails to capture higher order dependencies such as sparseness in the disparity map. This motivates us to use another prior namely the prior that represents sparsity in disparities. In our next work, we combine IGMRF and sparsity priors in our energy minimization framework in order to obtain a dense disparity map. Here, the sparsity prior is defined using the learned overcomplete sparseness of disparity patches. In this work, instead of making a brightness constancy assumption, we use an intensity matching cost as a data term which is robust against outliers and insensitive to image sampling. We use two different approaches in order to obtain the sparseness of disparities. In the first method, the sparse representation is obtained by a learned overcomplete dictionary where we make use of "K-singular value decomposition" (K-SVD) algorithm. In order to better represent the sparseness, "sparse autoencoder", a non-linear model is then used. A two phase iterative approach is used to obtain the final solution. In order to achieve better performance, a good initial estimate was obtained using a classical local stereo method including a set of post-processing operations for disparity refinement.

The combination of IGMRF and sparsity priors serve as a better regularizer but the choice of an appropriate data model also plays a key role in obtaining a better disparity map. Although, the data term used earlier which was based on pixel based intensity matching is robust against outliers and insensitive to image sampling, it relies on the raw pixel values and hence the use of it may result in ambiguous and erroneous disparities in textureless areas and near depth discontinuities. Taking this into account, in our next work, we propose a method where we make use of feature matching in the energy function. Hierarchical features of given stereo image pair are learned using the "deconvolutional network", a deep learning model which is trained in an unsupervised way using a database consisting of large number of stereo images. Combining the feature matching with the intensity matching in our energy function restricts the solution space giving

us a better estimate of the disparity map. We use IGMRF as prior for regularizing the solution. In this case also, we use an iterative two phase algorithm for minimizing the energy function where the IGMRF parameters and the disparity map are refined alternatively.

Finally, we propose a method using a better constrained energy function in a global energy minimization framework. As done in the previous work, our data term has the feature as well as intensity matching. However, the prior term here is formed using IGMRF and sparsity priors. Since the sparseness of disparities can be represented better by using the sparse autoencoder, we use the same to infer the sparseness of disparities. Once again an iterative approach is used to obtain the final solution in which disparities are refined until we get the convergence.

We demonstrate the efficacy of our proposed methods by conducting extensive experiments and evaluating our results on the Middlebury stereo datasets [113]. We also compare the performance of our methods with the state of the art global dense stereo methods. The results obtained show perceptual improvements as well as quantifiable gains in terms of percentage of bad matching pixels. Our results validate the effectiveness of using appropriate data and prior terms in obtaining accurate disparities.

List of Principal Symbols and Acronyms

List of Acronyms

IGMRF Inhomogeneous Markov Random Field

K-SVD K-Singular Value Decomposition

MRF Markov Random Field

MAP Maximum a Posteriori

OMP Orthogonal Matching Pursuit

DCT Discrete Cosine Transform

2D Two Dimensional

3D Three Dimensional

AD Absolute Difference

SD Squared Difference

List of Symbols

$\mathcal{S}$	Set of Sites
$\mathcal{L}$	Set of Labels
d	Disparity Map
(x, y)	Pixel Location
$\mathcal{N}$	Neighborhood System

V	Clique Potential Function
I_L	Left Image
I_R	Right Image
$E(d)$	Energy Function
$E_D(d)$	Data Term
$E_P(d)$	Prior Term
λ	Smoothness Weight
k	Discontinuity Preservation Constant
$b^X_{(x,y)}$	IGMRF Parameter at (x,y) in X Direction
$b^Y_{(x,y)}$	IGMRF Parameter at (x,y) in Y Direction
g	Ground Truth Disparity Map
$\mathcal{B}\%$	Percentage of Bad Matching Pixels
δ	Disparity Error Tolerance
$E_{IGMRF}(d)$	IGMRF Prior Term
$E_{sparse}(d)$	Sparsity Prior Term
γ	Weight of Sparsity Prior Term
$d^{(x,y)}$	Disparity Vector at (x,y)
n	Size of $d^{(x,y)}$
$\mathcal{D}$	Overcomplete Dictionary
K	Number of Columns in $\mathcal{D}$ Or Number of Hidden Units in Autoencoder
$a^{(x,y)}$	Sparse Vector at (x,y)
t	Maximum Number of Non-Zero Values in Sparse Vector
(W,U,r,s)	Autoencoder Weights
$E_I(d)$	Intensity Matching Cost
$E_F(d)$	Feature Matching Cost
μ	Weight of Feature Matching Cost
τ^I	Truncation Threshold for Intensity Matching Cost
τ^F	Truncation Threshold for Feature Matching Cost
NL	Number of Layers in Deep Deconvolutional Network
$Z_l^{I_L}$	Features of I_L at l^{th} Layer of Deep Deconvolutional Network
$Z_l^{I_R}$	Features of I_R at l^{th} Layer of Deep Deconvolutional Network

List of Tables

List of Figures

CHAPTER 1

Introduction

Human vision gives us the ability to perceive and understand the three-dimensional (3D) world surrounding us. Computer vision aims to duplicate the effect of human vision by electronically perceiving and understanding a two-dimensional (2D) image of a 3D scene. Making computers to see the 3D world is not easy because the images acquired by image sensors (camera) are 2D and this 3D-2D transformation results in the loss of "depth" information of a scene. The distance between the viewed scene and camera is referred as the depth of a scene and is computed for each point in the scene. Computer vision algorithms are used to reconstruct a 3D scene by estimating the depth information from one or more 2D images. Depth estimation has a wide variety of applications including robotics, scene understanding and reconstruction, safe navigation, autonomous vehicles, 3D television and cinema, telepresence, 3D printing, 3D rendering and modeling, etc. Based on the application, generally two types of depth measures are obtained; "absolute" and "relative". Absolute depth is an estimate of the physical distance in units such as meters to an object from the camera. Relative depth estimates the location of objects in relation to other objects rather than in terms of physical distance.

In general, the depth estimation methods are divided into two categories: "active" and "passive". In active methods, the active range sensor projects energy (e.g., a pattern of light, sonar pulses, ultrasound) on to the viewed scene and measures the distance/depth from the reflected energy. Active range sensors use the principles of Sonars and Radars, Moire interferometry, focusing and triangulation. These methods are used to obtain an absolute depth of the viewed scene, and to

obtain the "ground truths". Though, they provide accurate depth measures, they are expensive and are used in applications where high accuracy is required. For many applications where a trade-off can be maintained between the cost and accuracy, passive methods do suffice. Passive methods perform the depth estimation using the 2D image/s of the viewed scene. In practice, the intensity (optical) images are used because acquiring and processing these images is computationally less expensive. Intensity image formed by optical sensors rely on the natural light energy and these images can be found in abundance. However, in dark, thermal images are obtained using thermographic sensors but acquisition and processing of such images are computationally expensive.

Since human beings use various visual cues to perceive depth and understand the 3D structure of the world, passive depth estimation algorithms make use of such cues to estimate the depth from the 2D images. These cues are typically grouped into two distinct categories:

- **Monocular Cues:** Monocular cues provide the absolute and relative depth information of the scene from a single image. These cues include texture variations, texture gradients, occlusion, known object sizes, light and shading, aerial perspective (haze), defocus, focus, motion parallax, etc. As an example, the texture of many objects changes as the distance of the camera from the object varies. Texture gradients that capture the distribution of the direction of edges also add as a constraint to depth estimation. For example, the distant objects may have larger variations in the line orientations and nearby objects with almost parallel lines have smaller variations in line orientations. Similarly, a grass field when viewed at different distances may have different texture gradient distributions. If two objects are known to be the same size (e.g., two trees) but their absolute size is unknown, relative size cues can provide information about the relative depth of the two objects. The property of parallel lines converging in the distance at infinity allows us to reconstruct the relative distance of two parts of an object. Aerial perspective (haze) is another depth cue and is caused by atmospheric light scattering. Due to haze, the foreground objects have high contrast and

background objects at farther depth have lower contrast and are blurred. Occlusion (also referred to as interposition) also provides a sense of depth. It happens when near surfaces overlap far surfaces. If one object partially blocks the view of another object, it is perceived as a closer object. The way that light falls on an object and reflects off its surfaces and the shadows that are cast by objects provide an effective cue to determine the shape of objects and their position in space. Defocus blur can also be used as an effective monocular cue for depth perception. As observer moves, closer objects appear to move more than farther objects. This phenomenon is called motion parallax and is used to estimate the relative depths in a scene. This effect can be clearly seen when driving in a car where nearby things pass quickly while far off objects appear stationary. If information about the direction and velocity of movement is known, motion parallax can provide absolute depth information. Humans have the ability to change the focal lengths of the eye lenses by controlling the curvature of the lens, thus helping them to focus on objects at different distances. In computer vision, the focus cue refers to the ability to estimate the distance of an object from known camera lens configuration and the sharpness of the image of an object.

- **Stereo Cues:** Stereo cues provide the depth information of the scene from two (more than two) images captured from different view-points. Stereo cues are binocular, if two cameras are used. A scene point is projected onto different locations on the two image planes (left and right cameras), depending on the distance of the scene point from the cameras. The displacement between the corresponding left and right projection of the scene point is called "disparity". The disparity varies with scene distance and is inversely proportional to the depth. It is used as a cue for depth estimation. Using the disparities and stereo vision geometry, the depth is estimated based on the principle of triangulation. Stereo cues provide very precise relative depth estimates.

Based on these cues, standard passive depth estimation methods have been proposed in the literature [125, 84]. The methods which make use of the monocular

cues are “depth from monocular image”, “shape from defocus”, “shape from focus”, “shape from shading”, etc. Other methods which incorporate the stereo cues include “depth from stereo”, “structure from motion (optical flow)”, etc. The main advantage of monocular cue based methods is that relatively low amount of operations are needed to process a single image instead of two or more. Due to perspective projection’s many to one mapping property, all points along a line pointing from the optical center towards a scene point are projected to a single image point and results in a depth information loss. The depth reconstruction of such points from monocular cues provides ambiguous results. Hence, from a single image, it is not generally possible to infer unambiguous information about the shape, location, and orientation of 3-D objects in a viewed scene. Stereo based approaches use two (or more) cameras in distinct locations to significantly reduce ambiguity. Stereo vision approaches provide precise depth measures. Most work on visual 3D reconstruction has focused on stereo vision [26]. In this thesis, we deal with the “binocular stereo vision”.

1.1 Binocular Stereo Vision

“Two are better than one; because they have a good reward for their labour”. This proverb correctly defines the process of depth perception by human visual system. The human visual system is based on two eyes and the brain where the two slightly different projections of the world are captured onto the two retinas. The displacement in the two retinal images is called disparity and the brain uses this disparity information and recovers the distance or depth. The word “stereo” comes from the Greek word "stereos" which means firm or solid. With stereo vision, we see the world as solid in three dimensions. Figure 1.1 shows an example of a stereo human vision system.

In an attempt to simulate this as a stereo vision system, two cameras are used as the eyes to capture 2D images of the physical 3D world-scene. In this case, the computer takes the role of the brain in the computational modeling, processing, and interpretation of the 2D images. Thus, the task of stereo vision is to recover

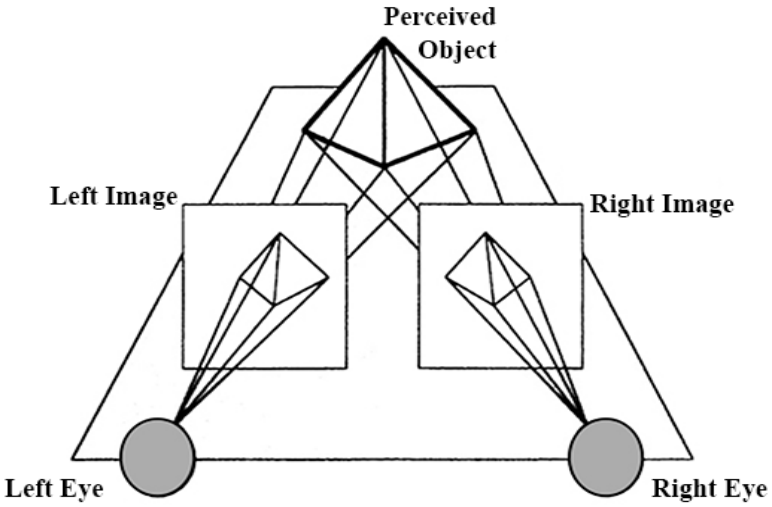

Figure 1.1: A stereo human vision system

the 3D information i.e., depth in a manner similar to the way humans perceive depth. However, depth can be estimated using more than two images of a scene taken from different view-points, referred as "multi-view stereo". Use of multiple images better constrain the solution space and results in accurate depth estimates but it also increases the cost of operations. Hence, in practice, the binocular stereo is used which considers only two images of a scene for depth estimation.

1.1.1 Applications

Stereo vision has been an active research area in the field of computer vision and has been used in different areas such as entertainment, medicine, scientific research, virtual reality, robotics, view synthesis, video coding, safe navigation in a spatial environment, etc. Few areas of application can be summarized as follows:

- **Robot Navigation:** Autonomous robot navigation in dynamic environments requires the study of the relative motion of the objects in the robot's environment with respect to the robot. Stereo vision can be used to efficiently estimate the depth to the surfaces that lie in the vicinity of the mobile robots. Depth information also allows the system to separate occluding image components, such as one chair in front of another which the robot may otherwise not be able to distinguish as a separate object.

- **Augmented Reality:** Stereo vision processing is a critical component of augmented reality systems that rely on the precise depth map estimation of a scene in order to appropriately place computer generated objects making use of real life videos.
- **Automotive applications:** The 3D perception of a car's surroundings is crucial both for driver assistance and for safety systems. An option to obtain 3D measurements of the surroundings is to use a stereo vision system.
- **Scientific Research:** Scientific applications of digital stereo vision include the extraction of information from aerial surveys, calculation of contour maps and geometry extraction for 3D building mapping.
- **Entertainment and 3D Tele-conferencing:** Entertainment is one of the major areas where stereo vision is used since there is no doubt that the presence of depth in images gives the viewer a more pleasant experience. To this end, 3D cinema, 3D TVs, and 3D video games that use stereo vision have become popular. 3D teleconferencing system based on stereo vision allows a 3D display of remote participant and maintains eye contact with multiple speakers, offering an alternative and possible improvement to traditional video conferencing and display technologies.

Now, the further sections cover the theoretical aspect of the stereo vision system. We begin with the discussion on basics of image formation, camera model and camera parameters which are further used to understand the geometry of stereo vision.

1.1.2 Camera Model and Image Formation

In computer vision, the process of image formation begins with the light rays entering the camera through an angular aperture and hitting a screen or image plane having a photosensitive device which registers light intensities. These light rays are the result of the reflection of the rays emitted by the light sources and hitting the object surfaces. A "pinhole camera" is the simplest imaging device that is suitable for many computer vision applications. The pinhole camera model captures

the geometry of perspective projection. The geometry of the pinhole camera and perspective projection is depicted in Figure 1.2.

The geometry of pinhole model consists of a 2D *image plane* π, a 3D scene

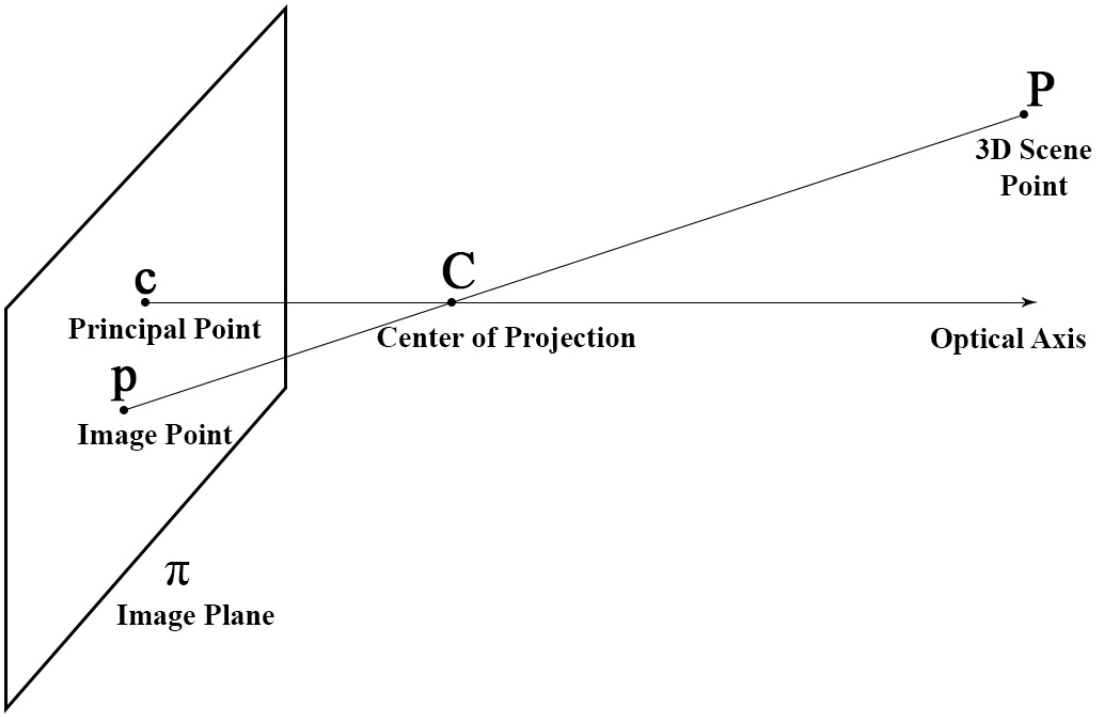

Figure 1.2: Illustration of pinhole camera model and perspective projection

point P, and the *center of projection* C. The distance between π and C is called *focal length* of the camera. The line passing through C and perpendicular to π is called the *optical axis*. The optical axis intersects the image plane at point c and is called *principal point* or *image center*. Ray of light from P passes through the pinhole camera through an infinitesimally small aperture i.e., C and the intersection of this ray with the π at p forms the image of P. Such a mapping from 3D to 2D is called perspective projection.

In order to find the mathematical relationship between a scene point and the corresponding image point, we define four different coordinate systems. The geometry of perspective projection with respect to these coordinate systems are shown in Figure 1.3.

1. The *World Coordinate System* (X_w, Y_w, Z_w) has the origin at O_w. Locations of 3D scene points are measured with respect to the world coordinate system.

2. The *Camera Coordinate System* (X_c, Y_c, Z_c) is identified by the camera and has the center of projection C as its origin O_c. The axis Z_c is aligned with the optical axis.

3. *Image Coordinate System* (X_i, Y_i) has axes aligned with the camera coordinate system, with X_i and Y_i lying in the image plane. It has principal point c as its origin.

4. *Pixel Coordinate System* (X_u, Y_u) has axes aligned with the image coordinate system but has opposite orientation. It has its origin at the top left corner of the image. The pixels in the images are represented with respect to the pixel coordinate system.

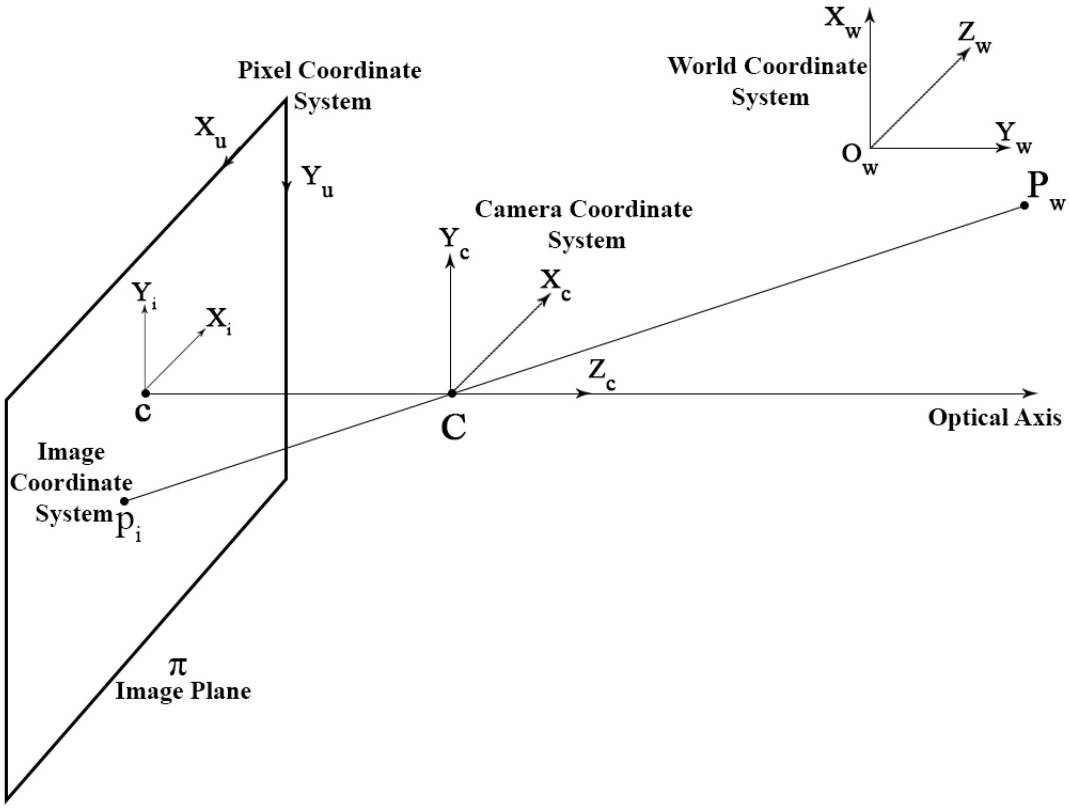

Figure 1.3: Geometry of perspective projection. The scene point P and its image point p are expressed as P_w and p_i in their world and image coordinate systems, respectively. Here, the image plane π is behind the center of projection C.

In general, the world and the camera coordinate systems are not aligned as shown in Figure 1.3. The camera coordinate system is often unknown and the common

problem is to determine the location and orientation of the camera coordinate system with respect to the known world coordinate system. The "extrinsic parameters" of the camera defines such geometric transformation.

The extrinsic parameters are defined as any set of geometric parameters that identify uniquely the transformation between the camera coordinate system and the world coordinate system. Typically, these geometric parameters include:

1. a 3D translation vector T between the relative positions of the origins of the two coordinate systems.
2. a 3×3 rotation matrix R that brings the corresponding axes of the two coordinate systems into alignment (i.e., onto each other).

Let the coordinates of a scene point P in the world and the camera coordinate systems are denoted as P_w=$[P_w^x, P_w^y, P_w^z]^T$ and P_c=$[P_c^x, P_c^y, P_c^z]^T$, respectively. We can find the relationship between P_w and P_c using the extrinsic camera parameters R and T as follows:

$$P_c = R(P_w - T). \tag{1.1}$$

Let rotation matrix $R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ and translation vector $T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$, Eq. (1.1) can be written as:

$$\begin{bmatrix} P_c^x \\ P_c^y \\ P_c^z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} P_w^x - T_x \\ P_w^y - T_y \\ P_w^z - T_z \end{bmatrix}. \tag{1.2}$$

or,

$$P_c^x = {R_1}^T(P_w - T), \tag{1.3}$$

$$P_c^y = {R_2}^T(P_w - T), \tag{1.4}$$

$$P_c^z = {R_3}^T(P_w - T), \tag{1.5}$$

where ${R_i}^T$ corresponds to the i^{th} row of the R matrix, i=1, 2, 3. The transformation described in Eq. (1.1) can be represented by a single matrix product using a homogeneous coordinate system. Let P_w is expressed as $\tilde{P}_w$=$[P_w^x, P_w^y, P_w^z, 1]^T$ in homogeneous coordinates. We define an *extrinsic camera matrix* M_{ext} of size 3×4 which consists of extrinsic parameters of the camera as:

$$M_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & -{R_1}^T T \\ r_{21} & r_{22} & r_{23} & -{R_2}^T T \\ r_{31} & r_{32} & r_{33} & -{R_3}^T T \end{bmatrix}. \tag{1.6}$$

The relation between P_w and P_c can be redefined using M_{ext} as:

$$P_c = M_{ext} \tilde{P}_w, \tag{1.7}$$

$$\begin{bmatrix} P_c^x \\ P_c^y \\ P_c^z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & -{R_1}^T T \\ r_{21} & r_{22} & r_{23} & -{R_2}^T T \\ r_{31} & r_{32} & r_{33} & -{R_3}^T T \end{bmatrix} \begin{bmatrix} P_w^x \\ P_w^y \\ P_w^z \\ 1 \end{bmatrix}. \tag{1.8}$$

The scene point P_c in camera reference frame is projected on to the image plane π as point p. The image point p can be expressed as p_i=$[p_i^x, p_i^y]^T$ in image coordinate system. The perspective projection produces an inverted image of the object/scene points. We can avoid this image inversion by assuming that the image plane π is in front of the center of projection C as shown in Figure 1.4. One can write the relationship between the scene point P_c and its corresponding image point p_i using the similar triangles as illustrated in Figure 1.4.

$$p_i^x = \frac{f P_c^x}{P_c^z}. \tag{1.9}$$

$$p_i^y = \frac{f P_c^y}{P_c^z}. \tag{1.10}$$

The above equations are referred as fundamental equations of perspective projection. Note that these equations are written in camera coordinate system and the z component of each image point is always equals to f.

Now, we need to obtain position of image point p_i in the pixel coordinate sys-

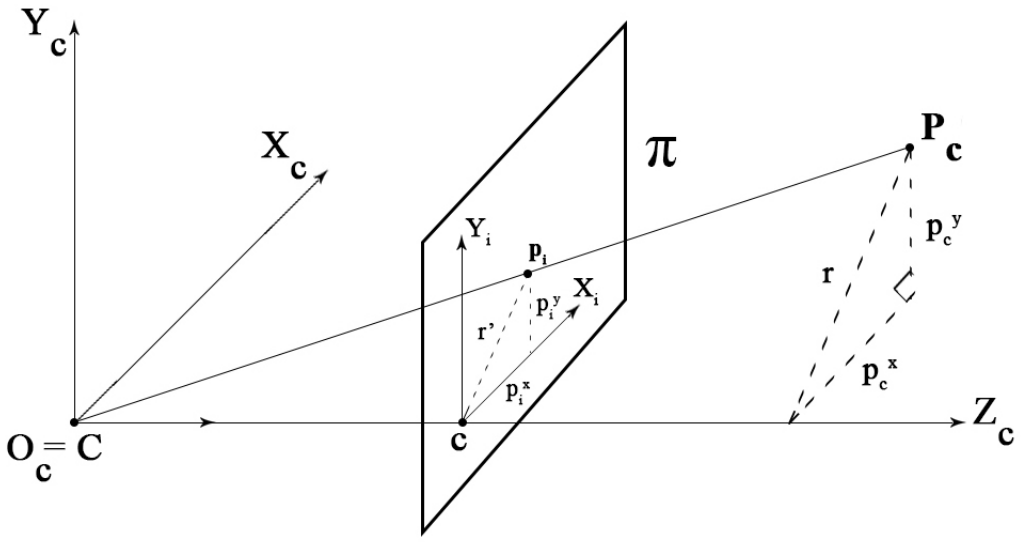

Figure 1.4: Deriving the perspective projection equations in camera coordinate system. Here, the image plane π is in front of the center of projection C.

tem, i.e. to determine the pixel coordinates that real camera actually delivers. Let us denote the coordinates of the image point p_i as p_u=$[p_u^x, p_u^y]^T$ in the pixel coordinate system. The relation between the pixel coordinates p_u and its corresponding coordinates p_i in the image plane are,

$$p_u^x = -\frac{p_i^x}{s_u^x} + c_u^x, \tag{1.11}$$

$$p_u^y = -\frac{p_i^y}{s_u^y} + c_u^y, \tag{1.12}$$

where (c_u^x, c_u^y) are the coordinates of the principal point in pixel coordinate system and (s_u^x, s_u^y) is the effective size of the pixel in the horizontal and vertical directions, respectively. Note that the sign change in Eqs. (1.11) and (1.12) is due to the fact that the X and Y axes of the image and pixel coordinate systems have

opposite orientation.

In order to link the pixel coordinates of an image point with the corresponding coordinates of the scene point in the camera coordinate system, the "intrinsic parameters" of the camera are used. The intrinsic parameters characterize the optical, geometric, and digital characteristics of the viewing camera. For a pinhole camera, the intrinsic parameters are specified as follows:

1. The perspective projection that transform the coordinates of a scene point defined in camera coordinate system to its image plane coordinates (see Eqs. (1.9) and (1.10)). Here, the parameter is focal length f.
2. The transformation between the image plane coordinates and the pixel coordinates of an image point. (see Eqs. (1.11) and (1.12)). Here, the parameters are c_u^x, c_u^y, s_u^x and s_u^y.

In short, the set of intrinsic parameters of camera are defined as the focal length f, the location of principal point in pixel coordinates (c_u^x,c_u^y), and the effective pixel size in the horizontal and vertical directions (s_u^x,s_u^y). The relation between the camera coordinates P_c of a scene point P and its corresponding image point p_u in pixel coordinates can be established by intrinsic parameters as:
Plugging Eqs. (1.9) and (1.10) in to Eqs. (1.11) and (1.12), respectively, we obtain:

$$p_u^x = -\frac{f P_c^x}{s_u^x P_c^z} + c_u^x. \tag{1.13}$$

$$p_u^y = -\frac{f P_c^y}{s_u^y P_c^z} + c_u^y. \tag{1.14}$$

The transformation described above can be represented by a single matrix product using a homogeneous coordinate system. Let p_u is expressed as $\tilde{p}_u$=$[\tilde{p}_u^x, \tilde{p}_u^y, h]^T$ in homogeneous coordinates. Here, h is a homogenization parameter and according to homogenization $p_u^x = \tilde{p}_u^x/h$ and $p_u^y = \tilde{p}_u^y/h$. We define an *intrinsic camera*

matrix, M_{int} of size 3×3 which consists of intrinsic parameters of the camera as:

$$M_{int} = \begin{bmatrix} -f/s_u^x & 0 & c_u^x \\ 0 & -f/s_u^y & c_u^y \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.15}$$

The relation between P_c and p_u can be redefined using M_{int} as:

$$\tilde{p}_u = M_{int} P_c, \tag{1.16}$$

$$\begin{bmatrix} \tilde{p}_u^x \\ \tilde{p}_u^y \\ h \end{bmatrix} = \begin{bmatrix} -f/s_u^x & 0 & c_u^x \\ 0 & -f/s_u^y & c_u^y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_c^x \\ P_c^y \\ P_c^z \end{bmatrix}. \tag{1.17}$$

Computer vision algorithms reconstructing the 3D structure of a scene or computing the position of objects in space require a direct relationship between the world coordinates of 3D points in space and pixel coordinates of their corresponding 2D image points. This is because the input for these algorithms are the digital images. With the knowledge of extrinsic and intrinsic camera parameters, one can write the relations linking directly the pixel coordinates of an image point p_u with the corresponding world coordinates P_w without explicit reference to the camera coordinates.

Plugging Eqs. (1.3), (1.4) and (1.5) into Eqs. (1.13) and (1.14), we obtain:

$$p_u^x = -\frac{f {R_1}^T (P_w - T)}{s_u^x {R_3}^T (P_w - T)} + c_u^x. \tag{1.18}$$

$$p_u^y = -\frac{f {R_2}^T (P_w - T)}{s_u^y {R_3}^T (P_w - T)} + c_u^y. \tag{1.19}$$

The transformation described in Eqs. (1.18) and (1.19) between 3D world coordinates to 2D pixel coordinates is called "projective transformation". This can be represented as a linear transformation using the product of extrinsic and intrinsic camera matrices. Considering the homogeneous representation of world and

pixel coordinates $\tilde{P}_w$ and $\tilde{p}_u$, the projective transformation is given as:

$$\tilde{p}_u = M_{int} M_{ext} \tilde{P}_w, \tag{1.20}$$

$$\begin{bmatrix} \tilde{p}_u^x \\ \tilde{p}_u^y \\ h \end{bmatrix} = \begin{bmatrix} -f/s_u^x & 0 & c_u^x \\ 0 & -f/s_u^y & c_u^y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & -{R_1}^T T \\ r_{21} & r_{22} & r_{23} & -{R_2}^T T \\ r_{31} & r_{32} & r_{33} & -{R_3}^T T \end{bmatrix} \begin{bmatrix} P_w^x \\ P_w^y \\ P_w^z \\ 1 \end{bmatrix}, \tag{1.21}$$

where

$$\tilde{p}_u^x = \frac{p_u^x}{h}; \qquad \tilde{p}_u^y = \frac{p_u^y}{h}. \tag{1.22}$$

The multiplication of M_{int} and M_{ext} matrices results in another matrix of size 3×4 referred as *projective matrix* or *camera matrix*. The problem of estimating the values in the projective matrix i.e., the estimation of intrinsic and extrinsic camera parameters is called "camera calibration". The key idea behind the calibration is to write the projection equations linking the known coordinates of a set of 3D points and their projections, and solve for the camera parameters. In order to get to know the coordinates of few 3D points, camera calibration methods rely on one or more images of a calibration pattern i.e., a 3D object of known geometry, possibly located in a known position in space. With the calibrated camera and known image points, the 3D reconstruction of a scene point can be done by solving the system of equations (1.18) and (1.19).

1.1.3 Stereo Vision Problems

As discussed in the previous section, the camera calibration and the knowledge of the coordinates of an image point allows us to determine a ray in space uniquely. If two cameras observe the same scene point, its 3D coordinates can be computed as the intersection of two such rays which is the basic principle of stereo vision. From a computational standpoint, a stereo system must solve the following three problems in order to estimate the depth map of a scene using two images:

- **Calibration of Left and Right Cameras**.
- **The Correspondence Problem:** Given the left and right views of the scene, the goal is to find corresponding pixels i.e. pixels resulting from the projection of same 3-D point on to the two image planes. The difference in position of the corresponding pixels is called disparity. Disparities of all the pixels in an image form the so-called "disparity map". In this thesis, we focus on solving the correspondence problem and propose various approaches for the same.
- **The Reconstruction Problem:** Given a number of corresponding pixels of the left and right images i.e., disparity map and possibly information on the geometry of the stereo system, the goal is to find the depth and thus construct a 3-D coordinates of the points in the scene.

1.1.4 The Geometry of Stereo Vision

The geometry of stereo vision is defined using two pinhole cameras and is shown in Figure 1.5. We consider *left* and *right* cameras with their *centers of projection* C_l and C_r, *image planes* π_l and π_r, and *focal lengths* f_l and f_r, respectively. The distance B between the center of projections C_l and C_r is the *baseline* of the stereo system. Both left and right camera identify the coordinate frames known as "left camera coordinate system" and "right camera coordinate system", respectively. We assume that both cameras have been carefully calibrated so that their intrinsic and extrinsic parameters are known with reference to fixed world coordinates.

Let P be a 3D scene point visible by the two cameras. The vectors P_l=$[P_l^x, P_l^y, P_l^z]^T$ and P_r=$[P_r^x, P_r^y, P_r^z]^T$ refer to the coordinates of P in the left and right camera coordinate systems, respectively. The coordinate systems of left and right cameras are related via a translation vector (baseline) B=$(C_r - C_l)$ and a rotation matrix R_{stereo}. Given P, the relation between P_l and P_r are given by:

$$P_r = R_{stereo}(P_l - B). \tag{1.23}$$

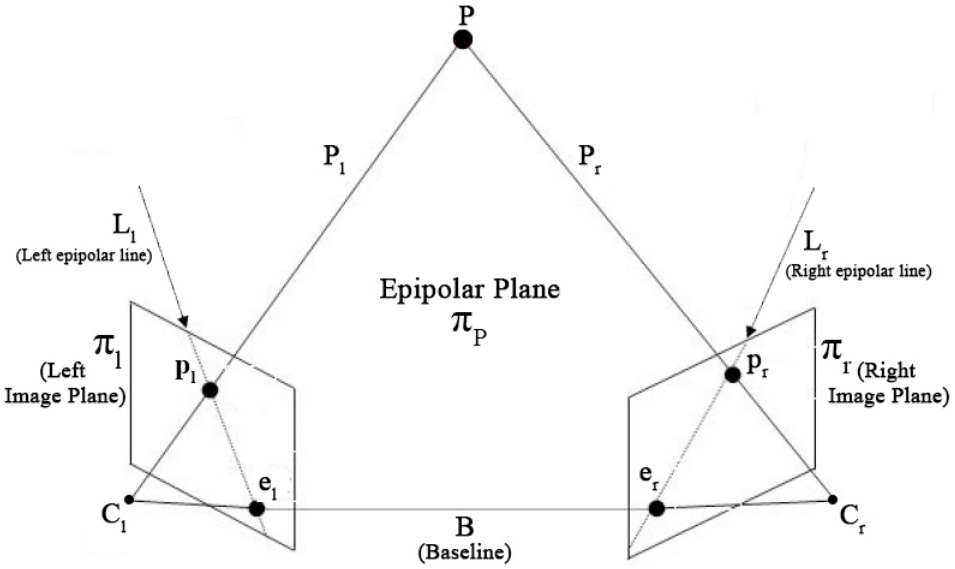

Figure 1.5: Geometry of stereo vision, epipolar geometry

The parameters B and R_{stereo} are referred as extrinsic parameters of the stereo system that describe the relative position (translation) and orientation (rotation) of the two cameras. The vectors p_l=$[p_l^x, p_l^y]^T$ and p_r=$[p_r^x, p_r^y]^T$ refer to the projections of P on to the left and right image planes π_l and π_r, respectively and are expressed in corresponding image coordinate system. The relation between P_l and p_l, or P_r and p_r is defined by the fundamental equations of perspective projection as:

$$p_l^x = \frac{f_l P_l^x}{P_l^z}; \qquad p_l^y = \frac{f_l P_l^y}{P_l^z}. \tag{1.24}$$

$$p_r^x = \frac{f_r P_r^x}{P_r^z}; \qquad p_r^y = \frac{f_r P_r^y}{P_r^z}. \tag{1.25}$$

Clearly, for all the image points we have p_l^z=f_l or p_r^z=f_r according to the image.

The geometry of stereo system is known as "epipolar geometry" that describes the relation between left and right projections (images) of 3D points in space. Given a pair of cameras, a scene point P defines a plane π_P going through P and the centers of projection of the two cameras C_l and C_r called the *epipolar plane*. The points at which the baseline B intersects the image planes π_l and π_r are called *epipoles*, and are denoted as e_l and e_r, respectively. In other words, e_l is the projec-

tion of C_r onto π_l, and e_r is the projection of C_l onto π_r.

For a point P, the epipolar plane π_P intersects left and right image planes in conjugated *epipolar lines* L_l and L_r, respectively. In other words, the left and right epipolar lines L_l and L_r are the projections of the lines C_rP and C_lP onto π_l and π_r, respectively. Therefore, all left and right epipolar lines pass through epipoles e_l and e_r, respectively. Consider the triplet P, p_l and p_r. Given p_l, P can lie anywhere on the ray from C_l through p_l. But, since the image of this ray in the right image is the epipolar line L_r through the corresponding point p_r, the correct match for p_l must lie on the epipolar line L_r. Similarly, the correct match for p_r must lie on its conjugated epipolar line L_l in the left image. This important fact is known as "epipolar constraint". The benefit of this constraint is that the search for the point corresponding to p_l need not cover the entire right image but can be restricted to the line L_r, and vice versa for the search of point corresponding to p_r.

In order to calculate the depth information or reconstruct the 3D points from a pair of images, we need to compute the epipolar geometry. The estimation of epipolar geometry determines a mapping between image point in one image and its corresponding epipolar line the other image. As seen from Figure 1.5, the three vectors P_l, $P_l - B$ and B lie in the same plane i.e., they are coplanar. Therefore, the equation of the epipolar plane through P can be written as the coplanarity condition of the vectors P_l, $P_l - B$ and B. According to the coplanarity condition, the cross product of any two vectors in a plane is perpendicular to any other vector in the same plane. Considering cross product of B and $P_l - B$ one can write,

$$(P_l - B)^T B \times P_l = 0. \tag{1.26}$$

Using the relation between P_r and P_l described in Eq. (1.23), the coplanarity condition can be rewritten as:

$$P_r{}^T R_{stereo} B \times P_l = 0. \tag{1.27}$$

Now, the cross product of vectors B and P_l can be expressed as a multiplication by a rank deficient matrix S as:

$$B \times P_l = SP_l,$$

where S represent the translation between the origins of left and right camera coordinate systems, and is defined as:

$$S = \begin{bmatrix} 0 & -B_z & B_y \\ B_z & 0 & -B_x \\ -B_y & B_x & 0 \end{bmatrix} \tag{1.28}$$

Using this fact, Eq. (1.27) becomes,

$${P_r}^T R_{stereo} S P_l = 0,$$

$${P_r}^T E P_l = 0, \tag{1.29}$$

with

$$E = R_{stereo} S. \tag{1.30}$$

The matrix E is called the *essential matrix*, and establishes a natural link between the epipolar constraint and the extrinsic parameters of the stereo system.

Here, the goal is to establish a mapping between the image points and their corresponding epipolar lines. As we know that what we actually measure from images are pixel coordinates and hence the transformation between the image coordinates and the pixel coordinates should be known. This transformation is given by the intrinsic parameters of both cameras. Let M^l_{int} and M^r_{int} be the intrinsic matrix of the left and right camera, respectively. Let p_{ul} and p_{ur} represent the pixel coordinates of p_l and p_r, respectively. Using the Eq. (1.16), the transformations between the camera coordinates and its projection in pixel coordinates for the scene point P are given as:

$$p_{ul} = M_{int}^{l} P_l; \qquad p_{ur} = M_{int}^{r} P_r. \tag{1.31}$$

M_{int}^{l} consists of the intrinsic parameters of left camera such as the focal length f_l, the location of principal point in pixel coordinates (c_l^x,c_l^y), and the effective pixel size (s_l^x,s_l^y). Similarly, M_{int}^{r} has the intrinsic parameters of right camera i.e., f_r, (c_r^x,c_r^y) and (s_r^x,s_r^y). These set of parameters are called as intrinsic parameters of a stereo system which are known by calibration. Now, using Eq. (1.31) we can write, $P_l = ({M_{int}^{l}}^{-1}) p_{ul}$ and $P_r = ({M_{int}^{r}}^{-1}) p_{ur}$. Substituting these into Eq. (1.29) we obtain,

$${p_{ur}}^{T} ({M_{int}^{r}}^{-1^{T}}) E ({M_{int}^{l}}^{-1}) p_{ul} = 0,$$

$${p_{ur}}^{T} F p_{ul} = 0, \tag{1.32}$$

where,

$$F = ({M_{int}^{r}}^{-1^{T}}) E ({M_{int}^{l}}^{-1}). \tag{1.33}$$

The matrix F is called the *fundamental matrix* and it encodes the information on the intrinsic and extrinsic parameters of the stereo. The Eq. (1.32) can be thought of as the equation of the projective epipolar line L_r in the right image that corresponds to the pixel point p_{ul} in left image or,

$$L_r = F p_{ul}. \tag{1.34}$$

We can conclude that fundamental matrix defines a mapping between the image points in pixel frame and their corresponding epipolar lines. The fundamental matrix can be computed from a number of corresponding point matches in pixel coordinate using the given left and right images only. Once the fundamental matrix is computed, one can reconstruct the epipolar geometry without any information of the intrinsic or extrinsic parameters. This indicates that the epipolar constraint as the mapping between the image points and the epipolar lines can be established with no prior knowledge of the stereo parameters.

The correspondence problem can be solved using the knowledge of fundamental matrix and epipolar geometry for the given pair of stereo images. For every pixel in the reference image, a corresponding epipolar line is determined using Eq. (1.34). Then the corresponding pixel is searched along the epipolar line in the other image by using similarity criteria. Once the corresponding pixel is found, the 3D location of the scene point is recovered by the intersection of two rays i.e., the rays passing through the left and right image projections. The determination of the intersection of two such lines generated from two images is called *triangulation*. Clearly, the determination of the scene position of an object point through triangulation depends upon matching the image location of the object point in one image to the location of the same object point in the other image.

1.2 A Simple Binocular Stereo System

In order to recover the depth of a scene, a canonical stereo system is often used because of it's simplicity. Here, a pair of cameras are arranged in such a way that baseline is parallel to the image planes, the optical axes of the cameras are parallel, the epipoles move to infinity, and the epipolar lines in the image planes are parallel. The geometric transformation that changes a general stereo configuration with non-parallel epipolar lines to the canonical ones is called "image rectification". In other words, given a pair of stereo images, rectification determines a transformation (or warping) of each image such that pairs of conjugated epipolar lines become collinear and parallel to one of the image axes, usually the horizontal one. The importance of the rectification is that the correspondence problem which involves a 2D search in general is reduced to a 1D search on a scanline identified trivially.

Figure 1.6 shows the top view of a canonical stereo system composed of left and right pinhole cameras with co-planar image planes π_l and π_r, respectively. We assume that both cameras have been calibrated. The optical axes are parallel; for this reason, the fixation point defined as the point of intersection of the optical axes lies infinitely far from the cameras. Given the left and right image points p_l

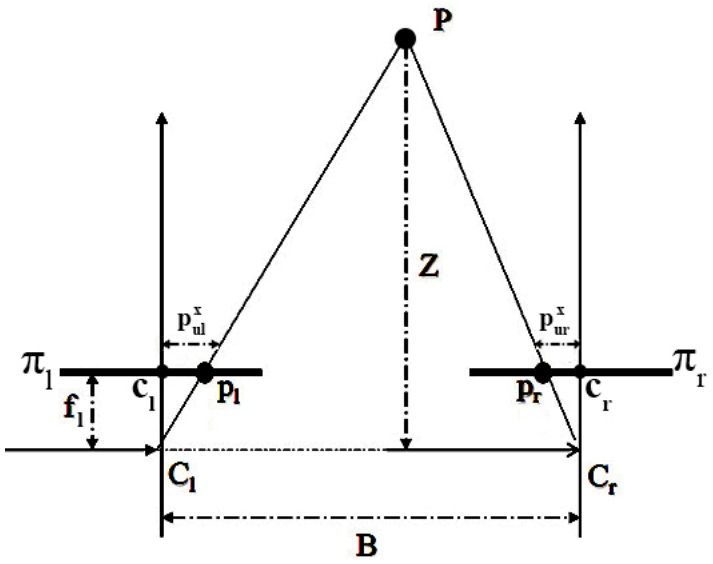

Figure 1.6: A simple stereo system. Here, the left and right image planes are co-planar and parallel to the baseline.

and p_r of the scene point P, our goal is to recover the position or depth of point P. Let Z denotes the depth of P. The depth Z is the distance between P and the baseline B i.e., distance between the camera and the scene point. Let p_{ul}=$[p_{ul}^x, p_{ul}^y]^T$ and p_{ur}=$[p_{ur}^x, p_{ur}^y]^T$ represent the pixel coordinates of p_l and p_r, respectively. The focal length of both the cameras are same i.e., f_l=f_r. Note that p_{ul} and p_{ur} are measured in the pixel coordinate frames of left and right cameras with respect to their principal points c_l=$[c_l^x, c_l^y]^T$ and c_r=$[c_r^x, c_r^y]^T$, respectively. Due to the rectification of left and right images, the corresponding epipolar lines become collinear and hence we get a displacement only between the x coordinates i.e., p_{ul}^x and p_{ur}^x of pixels p_{ul} and p_{ur}, and p_{ul}^y=p_{ur}^y. The position of any scene point P in space is determined by the triangulation and it is done by the intersection of rays defined by C_l and C_r, and p_l and p_r. From the similar triangles (p_l,P,p_r) and (C_l,P,C_r), we obtain:

$$\frac{B + p_{ul}^x - p_{ur}^x}{Z - f} = \frac{B}{Z}. \tag{1.35}$$

Solving Eq. (1.35) for Z, we obtain:

$$Z = \frac{fB}{d(x)}, \tag{1.36}$$

where $d(x)$=$p^x_{ur} - p^x_{ul}$ represents the disparity with respect to right image as a reference image. If we consider the left image as reference image then $d(x)$=$p^x_{ul} - p^x_{ur}$. If $d(x)$=0 then Z=∞ i.e., zero disparity indicates that the point is (effectively) at an infinite distance from the viewer. We can conclude from the Eq. (1.36) that depth is inversely proportional to disparity.

$$Z \propto \frac{1}{d(x)}. \tag{1.37}$$

In this thesis, we focus on solving the dense correspondence problem using the rectified pair of stereo images with known camera calibration.

1.3 Stereo Matching Constraints

One has to find the correspondence for all pixels by using their intensities or features only. The estimation of disparities is an ill-posed problem with inherent ambiguity due to the several problems described as follows:

1. **Photometric variation:** The light reflected from the scene and projected by camera depends on the position of that camera relative to the scene as well as noise and nonlinearities in the camera itself. Thus, when a camera is moved to a new position or when two cameras view a scene from two viewpoints, the intensity at the corresponding points may be different.
2. **Occlusion:** Occlusion is due to the occurrence of a depth discontinuity that causes an obstructed view of part of the scene that is observed by only one of the cameras.
3. **Repetitive texture:** When the texture is repeated, for example, bricks in a brick wall, multiple possible correspondences exist.
4. **Lack of texture:** For the untextured surfaces, it is difficult to find the corresponding point.

In order to reduce the effects of these phenomena and make the problem well-posed, several matching constraints are used. Some of these constraints follow

from the geometry of the image formation process, photometric properties of a scene, and object properties in our natural world. A list of commonly used constraints is given below:

1. **Epipolar Constraint:** Given the location of a pixel in one image, the matching location must lie on the corresponding epipolar line in other image. This constraint reduces the potential search space from 2D to 1D. The epipolar constraint never fails and it can be applied reliably once the epipolar geometry is known.
2. **Photometric Similarity Constraint:** It states that the corresponding points (matching locations) in both images should have similar color/intensity. This constraint is natural due to image-capturing conditions, and it is valid for lambertain surfaces where the appearance of the surface does not vary with view-point.
3. **Geometric Similarity Constraint:** This states that geometric characteristics of the corresponding features found in the left and right images do not differ much (e.g., length or orientation of the line segment, region or contour, etc.).
4. **Uniqueness Constraint:** For any location in one image, there should be at most one matching location in the other image.
5. **Smoothness/Continuity Constraint:** In general, the most of the scenes have the depth continuity i.e. they are smooth. This makes the disparities to vary smoothly almost everywhere over the image. This constraint fails at depth discontinuities resulting in discontinuous disparities.
6. **Ordering Constraint:** As per this constraint, if the points m and n in one image correspond to the point m' and n', respectively in other image, and if m is to the left of n then m' should also be to the left of n', and vice versa. That is corresponding points should be in the same order in both views. If there is a narrow object much closer to the camera than its background, or there is a large discontinuity in depths, the order gets changed. The ordering constraint fails at regions known as the forbidden zone. This constraint requires that the relative ordering of pixels on a scanline remain the same

between the two views, which may not be the case in scenes containing narrow foreground objects.

Different stereo matching algorithms make use of different constraints and estimate the disparity map for a given pair of stereo images. In practice, two types of disparity maps are computed; dense disparity map and sparse disparity map. In dense disparity map, the disparity is computed at each pixel location whereas, in sparse disparity map, the disparities are computed at few locations only. The sparse maps are computed by matching the sparse set of features in the left and the right images and have limited applications. The dense disparity map finds it's application in view synthesis, surface reconstruction, depth estimation and image based rendering which require disparity estimates in all image regions including those that are occluded or without texture. Hence, in this thesis, we propose various approaches for estimating the dense disparity map.

1.4 Contributions of the Thesis

In this thesis, we address the problem of dense disparity map estimation using rectified stereo images with known calibration of cameras. The dense disparity estimation is an ill-posed problem due to the presence of depth discontinuities, photometric variation, lack of texture, occlusions, etc., and hence in practice, this problem is solved in a global energy minimization framework by incorporating the regularization where a disparity map is estimated by minimizing a global energy function. In general, an energy function represents a combination of a data term and a prior term that restricts the solution space. The data term restricts a desired disparity map to be agreeable with the observed data i.e., the given stereo pair, while the prior term confines it to have a form matched with the prior knowledge about the true disparity map. Once an energy function is defined, it is minimized using an efficient optimization technique which leads to either local or global minima. However, solutions with lower energy do not always correspond to better performance. Therefore, it is more important to define a proper energy function than to search for optimization techniques in order to improve

the performance. A proper energy function is defined by the selection of an appropriate data model and the prior model. In this thesis, we propose various new approaches for solving the dense disparity estimation problem in a global energy minimization framework by defining proper energy functions, and we employ graph cuts, an efficient and fast optimization technique for minimizing our energy functions. The contributions of this thesis are summarized as under.

- Since the disparity estimation is an ill-posed problem, making use of a prior information about the nature of the true disparity map makes the ill-posed problem into better posed and leading to a better solution. In many cases, the disparities are piecewise smooth i.e., they vary smoothly except at discontinuities. This spatial smoothness among disparities can be captured by Markov random field (MRF) based models. In general, edge preserving homogeneous MRF models are used as a prior term in the global energy minimization framework. The homogeneous MRF priors consist of single or a set of global MRF parameters which may not adapt to the local structure of the disparity map and hence fail to better capture the spatial dependence among disparities. We need a prior that considers the spatial variation among disparities locally. In our first work, we propose to use an inhomogeneous Gaussian Markov random field (IGMRF) prior in an energy minimization framework to estimate the dense disparity map. The IGMRF prior captures the local variation among disparities at each pixel location using the adaptive IGMRF parameters. These parameters help us to obtain a solution which is less noisy in smooth areas and preserve the depth discontinuities in other areas. To form our energy function, we model the data term using the pixel-based intensity matching cost based on the brightness constancy assumption of corresponding pixels, and the prior term is defined using IGMRF prior. Since the true disparity map is unknown, the IGMRF parameters are computed using a close approximation of true disparity map. To this end, we also propose a learning based approach for obtaining an initial estimate of disparity map. To start the regularization process, we use the initial estimate of disparity map and compute the IGMRF parameters at ev-

ery pixel location which are then used to estimate the final disparity map by minimizing our energy function. The quality of the final solution strongly depends on the accuracy of the IGMRF parameters.

The choice of an initial estimate plays a key role in our work because the accuracy of the IGMRF parameters, and hence the quality of the final disparity map strongly depends on the quality of initial estimate. Also the use of better initial estimate accelerates the convergence while regularizing the solution. The initial estimate obtained by our learning based approach do not produce accurate disparities, and use of it affect the quality of final solution. Hence, in order to achieve better performance, in our subsequent works, we obtain a better initial estimate using a classical local stereo method including a set of post-processing operations for disparity refinement.

- Although, IGMRF prior captures the smoothness with discontinuities, it fails to capture higher order dependencies such as sparseness in the disparity map. The disparity maps can be represented in a domain in which they are sparse, and this transform domain representation can be obtained either using fixed set of basis or it can be learned using a set of training examples. We then consider the sparseness of disparities as an additional prior while regularizing the disparity map. The use of learned sparseness has proven to be more accurate when compared to the use of fixed basis while solving the ill-posed problems, and hence we prefer to learn the sparse representation of disparities in our work. Our next work thus involves both IGMRF and sparseness priors. The sparsity prior is derived using the learned overcomplete sparseness of disparity patches and captures the sparseness in the disparity map. We consider a data term using the pixel-based intensity matching cost which is robust to outliers and insensitive to image sampling. Based on this formulation, we propose two methods for dense disparity estimation. In our first method, the sparse representation of disparities are obtained by a learned overcomplete dictionary. We train our disparity dictionary using the patches of estimated disparity map of the given stereo pair via K-singular value decomposition (K-SVD) algorithm. The advantage of

our dictionary learning method is that the learned dictionary is adaptive to the disparities of the given stereo pair and we do not require the large set of ground truth maps for training. However, overcomplete dictionary model uses a linear structure, and a non-linear model can be used for the better representation of sparseness. Hence, in our second approach, we use a sparse autoencoder for learning and inferring better overcomplete sparse representation of disparities. We train our sparse autoencoder using a large set of ground truth disparity patches. The learned sparseness of disparities are further used to define the sparsity prior. In order to estimate the dense disparity map, both of our approaches start with the use of initial estimate of disparity map, and iterate and alternate between two phases until convergence. In phase one, sparseness of disparities are inferred and IGMRF parameters are computed based on the current estimate of disparity map, while in the second phase, the disparity map is refined by minimizing the energy function with other parameters fixed.

- The combination of IGMRF and sparsity priors serve as a better regularizer but the choice an appropriate data term also plays a key role in obtaining a better disparity map. Although, the data term based on intensity matching is robust against outliers, image sampling, view-point variation, etc., it relies only on the raw pixel values (intensities), and the use of such a data term may results in ambiguous and erroneous disparities in textureless areas and near depth discontinuities. The stereo images can be represented in a better way by using a feature space where they are robust, distinct and view-point invariant. This makes us to propose a method for dense disparity map estimation using feature matching as well. The use of features learned using deep learning and unsupervised learning methods have shown its superior performance than those using fixed or hand crafted features. Hence, we define a feature matching cost using the hierarchical features of given left and right stereo images, and these hierarchical features are learned using the deep deconvolutional network, a deep learning model which is trained in an unsupervised way using a database consisting of large number of stereo

images. In our energy function, we combine the learned feature matching and the intensity matching costs to form our data term. Here, our prior remains the same i,e., IGMRF based prior. Once again, we minimize our energy function by using an iterative two phase algorithm where the IGMRF parameters and the disparity map are refined alternatively.

- Finally, we propose one more approach for dense disparity estimation using a better constrained energy function. We define our data term using a combination of learned feature matching and pixel-based intensity matching costs, and the prior term is formed using the combination of IGMRF and sparsity priors. Since the sparseness of disparities are more efficiently represented by the learned sparse autoencoder than the use of learned overcomplete dictionary via K-SVD, we use sparse autoencoder for learning and inferring the sparseness of disparities. An iterative two phase algorithm is proposed to estimate the dense disparity map. We demonstrate the efficacy of our proposed methods by conducting extensive experiments and evaluating our results on the Middlebury stereo datasets [113]. We also compare the performance of our methods with the state of the art and latest global dense stereo methods.

1.5 Organization of the Thesis

In this thesis, we address the problem of dense disparity map estimation using rectified stereo images with known calibration of cameras, and propose various approaches for solving it in a global energy minimization framework. The organization of the thesis is as follows. Chapter 2 provides a review of the existing dense disparity estimation approaches proposed in the literature. In chapter 3, we discuss the labeling problem and energy minimization framework for disparity estimation. The details of the graph cuts method for optimizing the energy function is presented and a novel technique for dense disparity estimation using IGMRF as a prior is discussed. Since the IGMRF parameters are computed using the initial estimate of the disparity map, we further propose a learning based approach

for obtaining an initial estimate. We incorporate the sparsity prior in addition to the IGMRF prior in our energy function in chapter 4. We present two methods for learning the sparsity prior. In the first method, the sparse representation of disparities is learned by an overcomplete dictionary trained using K-SVD algorithm whereas, in the second method, a sparse autoencoder is used. In chapter 5, a novel approach for dense disparity estimation is proposed using learned hierarchical feature matching in an IGMRF based regularization framework. For this, a deep deconvolutional network is presented. A learned sparseness and IGMRF based priors are combined with the learned feature matching in chapter 6. We demonstrate the effectiveness of our proposed approaches by conducting various experiments on standard stereo datasets and comparison with the state of the art and the latest dense stereo methods. Finally, we summarize our work and conclude in chapter 7. We discuss the further challenges and directions for future research in chapter 8.

CHAPTER 2

Literature Review

Stereo correspondence has traditionally been and continues to be, one of the most extensively researched topics in the field of computer vision. The main goal here is to estimate the dense disparity map for a given pair of stereo images. However, estimation of disparities is an ill-posed problem with inherent ambiguities due to sensor noise, depth discontinuities, illumination and intensity variation, lack of texture, regions of repetitive textures, occlusions, etc. [26]. A variety of approaches has been proposed for the same in the literature [114, 113]. A comprehensive review of a large number of such algorithms is given in [114], and the quantitative comparison among all the state of the art and latest dense stereo methods evaluated on stereo benchmark datasets can be found on Middlebury stereo website [113].

In this chapter, we present a brief literature survey for the correspondence algorithms that estimate the dense disparity map for a pair of rectified stereo images with known camera geometry. A dense stereo correspondence algorithm generally performs (subsets of) the following four steps [114]:

1. matching cost computation,
2. cost aggregation,
3. disparity computation/optimization,
4. disparity refinement.

The actual steps taken depends on the specific algorithm. Based on these steps, we can classify dense disparity estimation algorithms in two broad classes; "local dense stereo methods" and "global dense stereo methods".

2.1 Local Dense Stereo Methods

The intensity of an individual pixel does not give sufficient information since there can be many pixel locations with similar intensity in the matched image. Therefore, in local approaches, to find the disparity at a pixel location, the intensities of several neighboring pixels in a window are considered. The disparity computation at a given pixel depends only on the intensity values within a finite window. Here, an implicit smoothness assumption is made i.e., all the pixels within a window have the same disparity. The local stereo methods are also referred as area based or window based methods. Following are the steps involved in a traditional window based method:

1. Compute matching cost at every pixel location for all possible disparities.
2. Aggregate the matching cost over a window centered at every pixel location for all possible disparities.
3. Compute the disparity at every pixel location using "winner-take-all" (WTA) optimization i.e., the disparity with winning aggregated cost is selected for each pixel.
4. Refine the disparity map by applying post processing methods such as sub pixel disparity estimation, left-right consistency check, interpolation, median filtering [114], etc.

Commonly used matching costs include squared intensity differences (SD) [114, 4, 41], absolute intensity differences (AD) [95], and normalized cross correlations (NCC) [41, 107]. In the case of SD and AD, the aggregation is performed by summing/averaging the costs over a local window and the disparity with minimum cost is selected. The local algorithms that use the NCC, combine steps 1 and 2 i.e., the correlation is measured between windows centered at a pixel location and a disparity with the highest correlation is selected. Truncated matching costs are more useful and robust because they limit the influence of mismatches during aggregation [114, 14, 15]. The matching costs are mainly designed on the brightness similarity assumption of the corresponding pixels within a window. However,

there are special circumstances when corresponding pixels have different intensities due to the effects of image sampling, noise, different gains and biases of the stereo cameras, depth discontinuities, occlusion, etc. Hence, the matching costs which are insensitive to such effects are proposed and used [114, 110, 13, 47]. A better performing matching cost includes the non parametric local transforms, for example, rank and census transform [148]. These transforms rely on the relative ordering of intensities within a window, and not on the intensity values themselves, and the correlation using such transforms can tolerate a significant number of outliers and result in better disparity estimates near disparity edges.

The matching cost aggregation is the key step for the success of local stereo approaches, and the quality of cost aggregation mainly depends on the size and shape of the window. The classical window based methods use a fixed static support region, typically a squared window for cost aggregation. When the window covers a region with non-constant disparity or depth discontinuities, such methods are likely to fail, and the error in the disparity estimates grows with the window size. Hence, a central problem in these methods lies in selecting an appropriate window size. The window size must be large enough to include enough intensity variation for reliable matching, but small enough to avoid the effects of projective distortion and preserves the depth discontinuities. Hence, a window size must be selected adaptively depending on local variations of intensity and disparity. Adaptive-window methods [61, 20, 126, 127, 98] try to find an optimal support window for each pixel and result in edge preserving and less noisy disparity maps. Kanade and Okutomi [61] present a statistical method to select an adaptive window at every pixel that minimizes the uncertainty in the disparity estimates. This method is, however, highly dependent on the initial disparity estimates and is computationally expensive. Moreover, the shape of a support window is constrained to a rectangle, which is not appropriate for pixels near arbitrarily shaped depth discontinuities. On the other hand, Boykov *et al.* [20] try to choose an arbitrarily shaped connected window. They perform plausibility hypothesis testing and compute a correct window for each pixel. A useful range of window sizes and shapes are chosen in [126, 127] to explore while evaluating the

window cost, which works well for comparing windows of different sizes. However, the shapes of support windows used are not general and this method needs many user specified parameters for the window cost computation.

The aggregation can also be implemented using multiple or shiftable windows anchored at different pixel locations. Multiple-window methods [32, 114] select an optimal support window among predefined multiple windows, which are located at different positions with the same shape. For example, authors in [32] perform the aggregation with nine different windows for each pixel and retain the disparity with the smallest matching cost.

Although, the local methods using the adaptive windows or multiple windows improve the performance of disparity estimation, they have a limitation in common: the shape of a local support window is not general. In fact, finding the optimal support window with an arbitrary shape and size is very difficult. For this reason, the methods limit their search space by constraining the shape of a support window. Rectangular and constrained-shaped windows, however, may be inappropriate for pixels near arbitrarily shaped depth discontinuities. To resolve this problem, segmentation-based local methods [119, 129, 122, 83] are proposed that use segmented regions with arbitrary sizes and shapes as support windows. In these approaches, it is implicitly assumed that the disparity varies smoothly in each region. However, these methods require precise color segmentation that is very difficult when dealing with highly textured images.

Instead of finding an optimal support window, one can adjust the support weights of the pixels in a given support window [135, 144, 71, 50, 156]. In an adaptive support weight approach of [144], the support weights of the pixels in a given support window are computed using color similarity and geometric proximity, and the cost aggregation is performed using these support weights. This method obtains better results in homogeneous regions and near depth discontinuities. The cost aggregation based on adaptive support weights can also be implemented using the edge preserving smoothness filter such as bilateral filter [5, 90, 105, 81, 140, 139] and guided image filter [51]. The bilateral filter computes the weighted average of the pixels within a window with the weights depend-

ing on both the spatial and intensity difference between the central pixel and its neighbors. Bilateral filter based local methods obtain higher accuracy along depth discontinuities and lower matching ambiguity, especially within low textured regions. However, these methods are computationally expensive since a large kernel size is typically used for the sake of high disparity accuracy. To address the computational limitation of the bilateral filter, authors in [51, 7] use guided image filtering into cost aggregation whose computational complexity is independent of the kernel size. Further to this, a non-local cost aggregation method is proposed in [138] which outperforms all local cost aggregation methods such as bilateral or guided image filtering in terms of speed and accuracy. Here, the matching cost values are aggregated adaptively based on pixel similarity on a minimum spanning tree derived from the stereo image pair to preserve depth edges. The nodes of this tree are all the image pixels, and the edges are all the edges between the nearest neighboring pixels. The similarity between any two pixels is decided by their shortest distance on the tree. The advantage of this method is that it has low computational complexity and it is non-local as every node receives supports from all other nodes on the tree.

The local stereo methods are easier to implement and are computationally less expensive. These methods can easily capture accurate disparities in highly textured regions, however, they often tend to produce noisy disparities in large textureless regions or repetitive textures, blur the disparity discontinuities, and fail at occluded areas. Though, the local methods based on adaptive windows, adaptive weights and filtering improve the accuracy in these regions, but can not correct it completely. The central problem of these stereo matching methods is to determine the optimal size, shape, and weight distribution of aggregation support for each pixel. An ideal support region should be bigger in textureless regions and should be suspended at depth discontinuities. However, they assume that the disparity is same over the entire window which does not hold in reality because the disparity maps are globally smooth with sharp discontinuities. Global stereo methods overcome the limitations of local stereo methods by incorporating the global information about the disparity map such as explicit smoothness in a

global optimization framework.

2.2 Global Dense Stereo Methods

Since the dense disparity estimation is an ill-posed problem, making use of a prior or global information about the nature of the true disparity map makes the ill-posed problem into better posed, leading to a better solution. Global approaches formulate the dense disparity estimation problem in a global energy minimization framework by incorporating the prior where a disparity map is obtained by minimizing the global energy function [114]. In general, the energy function represents a combination of a data term and a prior term that restricts the solution space. The data term measures how well the disparity map to be estimated agrees with the input image pair where as the prior term measures how good it matches with the prior knowledge about the disparity map. Such algorithms typically perform the *matching cost computation*, *disparity map optimization* and *disparity refinement* steps and skip the *cost aggregation* step.

The key step of the global approaches is to define an appropriate energy function and to provide an efficient optimization method to find local or global minimum. Typically, these energy functions are non convex, and optimizing them is an NP-hard problem. Hence, a variety of optimization methods have been proposed for solving the energy minimization problem in stereo. These methods include dynamic programming [2, 9, 27, 35, 130], simulated annealing [8, 36, 87], mean field annealing [34, 96, 112], graph cuts [69, 21, 67, 68, 62, 106, 52], and belief propagation [117, 116, 143, 142, 146, 141]. Dynamic programming can find the global minimum for independent scanlines in polynomial time. Dynamic programming based approaches work by computing the minimum cost path through the matrix of all matching costs between two corresponding scanlines. However, these approaches result in streaking artifacts near region boundaries. Moreover, a semi global matching is proposed in [46, 45] which minimizes a 2D energy function by solving a large number of 1D minimization problems. Semi global matching outperforms dynamic programming and yields no streaking artifacts. Simulated an-

nealing is theoretically capable of finding the global minima of an arbitrary energy function but it requires exponential time and is extremely slow in practice. Mean field annealing approach involves the computation of partition function which is intractable. Optimization algorithms based on graph cuts [69] and belief propagation [117] are the most efficient and prominent global optimization algorithms in terms of time complexity and accuracy of the solution [120]. They guarantee that the solution so obtained either reaches the global optimum or reaches a local minima close to the global minimum with considerably less computational time complexity. Hence, many of the latest and the state of the art global stereo methods are based on these two optimization techniques [114, 113].

In general, the disparities are piecewise smooth i.e., they exhibit low variance except at discontinuities. This spatial smoothness among disparities can be captured by "Markov random field" (MRF) based models [80, 121, 36]. MRF models are used to incorporate the explicit smoothness as a prior constraint in the energy function. Many of the state of the art global stereo methods are based on the MRF formulations, for example [114, 21, 117, 68, 143, 133, 67, 94, 43, 116, 62]. These methods obtain sharp depth discontinuities at object boundaries and smooth disparities in homogeneous regions. In the standard MRF formulation, the smoothness constraint is enforced in a first or second order neighborhood.

In order to capture the smoothness in a large neighborhood and preserve the depth discontinuities, segmentation based global stereo methods have been proposed in the literature [48, 66, 16, 18, 17, 118]. These methods are based on the assumption that the scene structure can be approximated by a set of non-overlapping planes in the disparity space and that each plane is coincident with at least one homogeneous color segment in the reference image i.e., within each segment, disparities are constant, planar or vary smoothly. Here, the dense stereo matching problem is cast as an energy minimization in segment domain instead of pixel domain where the disparity plane is assigned to each segment using global optimization algorithm. Despite the fact that segmentation based approaches usually improve disparity estimates in large textureless regions, they inevitably introduce errors in rich textured areas and do not handle well the situation that the scene

contains non planar surfaces. Also, the solution here relies on the accuracy of segmentation which is itself a non trivial task.

In global methods, the data term is generally defined by the pixel-based matching cost between the intensities of corresponding pixels in the input left and right stereo images. The commonly used pixel-based matching cost include, absolute differences (AD) and squared differences (SD) measures [114] and rely on the brightness constancy assumption of corresponding pixel intensities. In order to handle the mismatches and outliers, truncated AD and SD measure have been employed [21, 117]. Due to the image sampling, discontinuities, occlusions or illumination variation, the intensity constancy assumption of corresponding pixels do not hold good in reality. Hence, in order to handle such effects, several other matching costs have been used by the global methods [114, 47]. For example, matching cost insensitive to image sampling [143, 13], rank and census transform [92], gradient based measures [94], mutual information [63, 46]. Although, global methods skip the cost aggregation step, several aggregated matching costs have been used to improve the performance of global methods near discontinuities and at occlusions. For example, in [143] color weighted correlation is used as a data term and the energy function is minimized using hierarchical belief propagation. Recently, bilateral filtering is used as a matching cost in an energy minimization framework [94].

In recent years, there has been a considerable progress in solving the stereo vision problem using machine learning methods due to the increasing availability of the ground truth data. The work of [70] learn the probability of stereo matching errors as a function of stereo images and underlying scene structure. These learned probabilities are integrated into an MRF based energy minimization framework for estimating the disparities. Since the likelihood function is dependent on the states of a large neighboring region around each pixel, a high-order MRF inference problem is solved using the simulated annealing algorithm which is extremely slow. In [155], an expectation maximization (EM) algorithm is used to iteratively estimate the disparity map and learn the MRF parameters based on the estimate in an energy minimization framework. While these methods have shown promis-

ing results, they do require some initial model whose parameters still need to be preset. In these previous works, the model is learned from the same (unlabeled) data that is to be labeled, and the parameters are adjusted in order to improve the performance. In the methods of [111] and [147], supervised learning is used for learning the model parameters by making use of the ground truth disparity maps. Scharstein and Pal [111] present a conditional random field (CRF) based model for disparity estimation where the maximum likelihood estimator for model parameters is learned using ground truth disparity maps and gradient descent method. Computing the CRF parameters, however involves the partition function which is intractable in cyclic graphs. Hence, the partition function is approximated by the model distribution which is obtained using graph cuts. However, the method has high computational complexity and this approximation can lead to poor disparity estimates. The number of CRF parameters used are also limited, affecting the solution. Similar work is proposed in [147] where authors present a CRF based model with non-parametric cost functions which is learned automatically using the structured support vector machines (SVM) with linear kernels. However, the method is also computationally expensive and use a traditional supervised learning method which requires a large set of labeled (ground truth) data.

Recently, unsupervised feature learning and deep learning approaches have achieved superior performance when compared to the traditional supervised learning methods in solving many computer vision problems [128, 74, 11, 29]. The deep learning approaches learn the hierarchical features using a large set of unlabeled data and avoid the need for feature engineering. It has also attracted the attention of stereo vision researchers in last few years. To this end, proposed approaches for dense disparity estimation based on deep learning [149, 91] have achieved better performance by obtaining a place in the top 10 dense stereo matching algorithms as per the Middlebury stereo evaluation webpage [113].

Global methods have been proved to be the most efficient methods in the stereo literature [114, 113]. They not only perform well in textured areas but also provide more precise and reliable disparity maps in untextured and repetitive textured regions, and near depth boundaries. Recently, several other global

stereo methods are proposed with better performance in terms of speed and accuracy [113]. Yamaguchi *et al.* [136] formulate dense stereo matching with a hybrid Markov random field, composed of continuous random variables for slanted 3D planes and discrete random variables for occlusion boundaries. Jung *et al.* [59] exploit the consistency criterion across real and virtual intermediate views and minimize an energy function, including the consistency term. Sinha *et al.* [115] model locally slanted planes by matching and clustering sparse local features, and solve the local plane sweep problem using semi global matching. Yamaguchi *et al.* [137] construct a slanted plane model over superpixels and optimize an energy function, composed of segment label and boundary energy terms. A mesh structure is constructed for stereo matching, and is optimized by employing a two-layer MRF in [152]. Psota *et al.* [104] perform the disparity estimation through message passing on the minimum spanning tree (MST). They represent the disparity maps as a collection of hidden states on MST, and model each MST by a hidden Markov Tree. Li *et al.* [79] generate multiple proposals on absolute and relative disparities from multi-segmentation and then these proposals are coordinated by pixel-wise competition and pair-wise collaboration within a MRF model for disparity estimation. In [65], authors apply adaptive smoothness constraint using texture and edge information of stereo images in an energy minimization framework.

Dense stereo matching is one of the most spot lighted research area in computer vision and is now quite matured problem. In our work, we propose various learning based approaches for dense disparity estimation in an energy minimization framework.

CHAPTER 3

An IGMRF based Regularization Framework for Dense Disparity Estimation

In a binocular vision system, generally the disparities are found by comparing pixel intensities or their features in the left and right images. However, estimation of disparities is an ill-posed problem and making use of a prior information about the nature of the true disparity map makes the ill-posed problem into better posed and leading to a better solution. Global approaches pose the disparity estimation problem in an energy minimization framework by incorporating prior or priors while solving. In many cases, the disparities are piecewise smooth i.e., they exhibit low variance except at discontinuities. This spatial smoothness among disparities can be captured by Markov random field (MRF) based models. It is well known that MRFs are the most general models used as priors during regularization when solving ill-posed problems [80, 36]. Hence, many of the current better performing global stereo methods are based on the MRF formulations as noted in [114, 113].

In this chapter, we propose to use an inhomogeneous Gaussian Markov random field (IGMRF) prior in an energy minimization framework to estimate the dense disparity map. The IGMRF prior is adaptive to the local structure of the disparity map and hence the use of it leads to a smooth disparity map with sharp discontinuities. The IGMRF parameters are computed using the initial estimate of the disparity map. In order to obtain the initial estimate, we propose a learning based approach. We demonstrate that the use of IGMRF prior results in better disparity map than those using edge-preserving homogeneous MRF priors proposed

in the literature. Before we start discussing the proposed approach, in the next few sections we discuss in brief about the labeling problem, MRF model, MAP-MRF estimation, energy minimization framework, and graph cuts for optimization.

3.1 The Labeling Problem

In computer vision, a labeling problem is specified in terms of a set of *sites* and a set of *labels* associated with them. The dense disparity estimation can be cast as a labeling problem where a regular $M \times N$ lattice represents the set of sites $\mathcal{S} = \{(x, y) | 1 \leq x \leq M, 1 \leq y \leq N\}$ and the disparities represent the set of labels $\mathcal{L}$. Let a disparity map $d \in \mathbb{R}^{M \times N}$ represents a labeling. The labeling problem is to assign a label to each site in the $\mathcal{S}$ i.e., the goal is to estimate d where a disparity $d(x, y) \in \mathcal{L}$ is computed at each pixel location $(x, y) \in \mathcal{S}$. In our work, we solve the dense disparity estimation problem using a rectified pair of stereo images. Use of rectified input images reduces the correspondence search to one dimensional i.e., $\forall (x, y)$, $d(x, y) \in \mathbb{R}$. In other words, we measure the disparity in the x coordinates only while the y coordinates of corresponding pixels remain same.

3.2 MRF Prior Model

In order to solve the dense disparity estimation as a labeling problem, we model the disparity map as an MRF and use a Maximum a Posteriori (MAP) estimation technique. Disparities are inversely proportional to the depth and their variations are due to various textures, sharp discontinuities as well as smooth areas within the object. Therefore, the disparity maps are context dependent i.e., there exists spatial correlation among the neighboring disparities. This dependence can be appropriately modeled by using a Markov model. Here, the relation among disparities at different pixel locations is described by Markov random fields.

The MRF provides a convenient and consistent way of modeling context dependent entities. This is achieved by characterizing the mutual influence among such entities using conditional probabilities for a given neighborhood. MRF was

first introduced in vision by Geman and Geman [36] for solving the image restoration problem and has been used widely in solving a number of problems in the field of computer vision [80]. The practical use of MRF models is largely ascribed to the equivalence between the MRF and the Gibbs random fields (GRF). In this work, we assume prior density of disparity map as an MRF. This is a valid assumption since the disparities vary smoothly except at discontinuities.

Let D be a random field over a regular $M \times N$ lattice of sites $\mathcal{S}$, and let a particular realization of D be denoted as d. D is said to be Markov random field on $\mathcal{S}$ with respect to a neighborhood system $\mathcal{N}$ if and only if the following two conditions are satisfied [80],

- $P(D = d) > 0, \forall d \in \mathbf{D}$ (positivity).
- $P(d(x,y)|d(\mathcal{S} - \{(x,y)\})$=$P(d(x,y)|d(\mathcal{N}_{(x,y)})$ (Markovianity),

where P denotes probability, $d(x,y)$ is the disparity (label) at site (x,y), $\mathcal{N}_{(x,y)}$ represents the set of sites lie in the neighborhood of (x,y), and $\mathbf{D}$ denotes the set of possible labelings. The first property is needed for technical reasons to ensure that the joint probability $P(d)$ can be uniquely determined by the local conditional densities $P(d(x,y)|d(\mathcal{N}_{(x,y)})$. The second property states that a disparity at a pixel location is dependent directly on its neighbors. This allows us to model spatial interactions among disparities.

MRF can be specified either by the joint distribution or by the local conditional distribution. However, local conditional distributions are subject to nontrivial consistency constraints, so the first approach is most commonly used. The Hammersley-Clifford theorem gives a convenient way to specify an MRF. The theorem states that D is an MRF on $\mathcal{S}$ with respect to $\mathcal{N}$ if and only if D is a Gibbs random field (GRF) on $\mathcal{S}$ with respect to $\mathcal{N}$ [80]. Based on this MRF-GRF equivalence, the MRF can be specified by the Gibbs distribution as:

$$P(D = d) = \frac{1}{\mathcal{Z}} \exp(-U(d)), \tag{3.1}$$

where $\mathcal{Z}$ is a normalizing constant called the partition function given by $\mathcal{Z} = \sum_{d \in \mathbf{D}} \exp(-U(d))$ and $U(d)$ is a prior energy function given by $U(d) = \sum_{c \in \mathcal{C}} V_c(d)$.

Here, $V_c(d)$ denotes the potential function of clique c and $\mathcal{C}$ is the set of all possible cliques. A clique c is defined as a subset of sites where each member of this subset is a neighbor of all the other members. It consists either of a single site, pair, or triple and so on. For simplicity, we consider pair wise cliques on a first-order neighborhood consisting of the four nearest neighbors for each site (x, y). Considering the pair wise interactions between pixels, the prior energy function $U(d)$ can be rewritten as:

$$U(d) = \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y')), \tag{3.2}$$

where $\mathcal{N}$ is a set of all neighboring pairs of pixels $\{(x, y), (x', y')\}$ referred as first order neighborhood system. The disparity map d modeled as MRF can now be specified as:

$$P(d) = \frac{1}{\mathcal{Z}} \exp\left(- \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y'))\right). \tag{3.3}$$

3.3 MAP-MRF Estimation

Our main goal is to find the disparity map d using a given rectified pair of stereo images, left image $I_L \in \mathbb{R}^{M\times N}$ and right image $I_R \in \mathbb{R}^{M\times N}$. In order to solve the disparity estimation as a labeling problem, we consider one of the images to be *primary* and the other one *secondary*. Let the left image I_L be chosen as primary image and the right image I_R as secondary image. Let $\mathcal{S}$ be the set of pixel locations in I_L and the task is to label each pixel in $\mathcal{S}$ with its disparity i.e., to estimate the disparity map d. Once the prior knowledge about the d is modeled and the data is known, the labeling problem of disparity estimation can be solved using the Bayes estimation i.e., the maximum a posterior (MAP) which is obtained by maximizing the posterior probability $P(d|I_L)$. We consider the primary image I_L as the data or the observation. Using Bayes rule, one can write:

$$P(d|I_L) = \frac{P(I_L|d)P(d)}{P(I_L)}, \tag{3.4}$$

where $P(d)$ is the prior probability of d, $P(I_L|d)$ is the conditional probability of the data also called as likelihood probability, and $P(I_L)$ represents the distribution of data which is constant and hence not considered while maximizing. The MAP estimate $\hat{d}$ can now be given as:

$$\hat{d} = \arg\max_{d} \quad P(d|I_L) = \arg\max_{d} \quad P(I_L|d)P(d). \tag{3.5}$$

Taking the log we can write,

$$\hat{d} = \arg\max_{d} \quad \log P(I_L|d) + \log P(d). \tag{3.6}$$

We now define an appropriate model for likelihood probability $P(I_L|d)$. Let $I_L(x,y)$ and $I_R(x+d(x,y),y)$ be the intensities at pixels (x,y) and $(x+d(x,y),y)$, respectively i.e., $I_L(x,y)$ and $I_R(x+d(x,y),y)$ are the projection of a scene point in left and right images, respectively with disparity $d(x,y)$. For a given disparity $d(x,y)$, the relationship between $I_L(x,y)$ and $I_R(x+d(x,y),y)$ can be represented as:

$$I_L(x,y) = I_R(x+d(x,y),y) + \zeta(x,y), \tag{3.7}$$

where $\zeta(x,y)$ is independent and identically distributed noise at pixel (x,y). We assume that the noise at every pixel follows a Gaussian distribution with zero mean and variance one. For a given d, the likelihood $P(I_L|d)$ can then be written as a multiplication of individual conditional probabilities $P((I_L(x,y))|d(x,y))$. One can then express,

$$P(I_L|d) = \prod_{(x,y)\in\mathcal{S}} P((I_L(x,y))|d(x,y)). \tag{3.8}$$

Using Eq. (3.7) and making use of i.i.d. condition, one can write,

$$P((I_L(x,y))|d(x,y)) = \exp\big(-(I_L(x,y) - I_R(x+d(x,y),y))^2\big). \tag{3.9}$$

Using Eqs. (3.8) and (3.9), we get,

$$P(I_L|d) = \exp\left(- \sum_{(x,y)\in\mathcal{S}} (I_L(x,y) - I_R(x + d(x,y), y))^2 \right). \tag{3.10}$$

Now, substituting the Eqs. (3.3) and (3.10) in Eq. (3.6), the MAP estimate is given by,

$$\hat{d} = \arg\max_{d} \Bigg(- \sum_{(x,y)\in\mathcal{S}} (I_L(x,y) - I_R(x + d(x,y), y))^2 - \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y')) \Bigg). \tag{3.11}$$

This is equivalent to minimizing the negative of the above function which is called as *energy function* denoted by $E(d)$ i.e.,

$$\hat{d} = \arg\min_{d} \quad E(d), \tag{3.12}$$

where $E(d) =$

$$\sum_{(x,y)\in\mathcal{S}} (I_L(x,y) - I_R(x + d(x,y), y))^2 + \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y')). \tag{3.13}$$

3.4 Energy Minimization Framework

In dense disparity estimation, we wish to compute the disparity $d(x,y)$ at every pixel location (x,y) such that pixels in I_L project to their corresponding pixels in I_R. When the disparity map is estimated using only the information about the data, it results in an ill-posed problem. Additional constraints are needed to guarantee the uniqueness of the solution to make it better-posed. This results in regularizing the solution and a better estimate of the disparity map can be obtained. The MAP labeling with a prior energy (MRF) is equivalent to regularizing the solution. Hence, a regularized disparity map is obtained by minimizing the energy

function given in Eq. (3.13).

A standard form of the energy function $E(d)$ can be written as:

$$E(d) = E_D(d) + E_P(d), \tag{3.14}$$

where the data term $E_D(d)$ measures how well the d to be estimated agrees with I_L and I_R of the scene. The prior term $E_P(d)$ measures how good it matches with the prior knowledge about the disparity map. In Eq. (3.13), the first term is considered as $E_D(d)$ and the second term as $E_P(d)$ i.e.,

$$E_D(d) = \sum_{(x,y)\in\mathcal{S}} \left(I_L(x,y) - I_R(x + d(x,y), y)\right)^2, \tag{3.15}$$

and

$$E_P(d) = \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y')). \tag{3.16}$$

The data term $E_D(d)$ defined in Eq. (3.15) assumes that the two pixels (x, y) and $(x + d(x,y), y)$ in left and right image, respectively have the correspondence if the intensities $I_L(x,y)$ and $I_R(x + d(x,y), y)$ are similar. Based on this brightness (intensity) constancy assumption, it is given as the squared difference of corresponding pixel intensities. Such data costs are referred as "intensity matching cost". Researchers have also used the intensity matching cost based on "absolute difference" as well [114].

One common constraint on the disparity map is the "smoothness" i.e., the disparity map is continously differentiable except at discontinuities. As discussed in section 3.2, this smoothness constraint is used as a prior and it is often expressed using the prior probability as MRF. Hence, considering pairwise interactions between pixels, the clique potential function in prior energy term is defined using the *smoothness prior* as a function of finite difference approximations of the first order derivative of disparity at each pixel location. In this case, the prior term $E_P(d)$ measures the extent to which the smoothness assumption is violated by d. In the stereo literature, several form of smoothness priors have been used

[114, 21, 117, 68, 143, 133, 67, 94]. Few of the smoothness priors are discussed below.

Everywhere Smooth Prior

The everywhere smooth prior encourages the disparity map which are globally smooth. To formalize this prior, one chooses $V_{\{(x,y),(x',y')\}}(d)$ in Eq. (3.16) to assign higher penalties for larger differences between neighboring disparities $d(x,y)$ and $d(x',y')$. Recall that $\{(x,y),(x',y')\} \in \mathcal{N}$ where $\mathcal{N}$ include a set all neighboring pairs of pixels $\{(x,y),(x',y')\}$. Examples of such priors are:

- Quadratic

$$E_P(d) = \lambda \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} (d(x,y) - d(x',y'))^2. \tag{3.17}$$

- Linear

$$E_P(d) = \lambda \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} |d(x,y) - d(x',y')|. \tag{3.18}$$

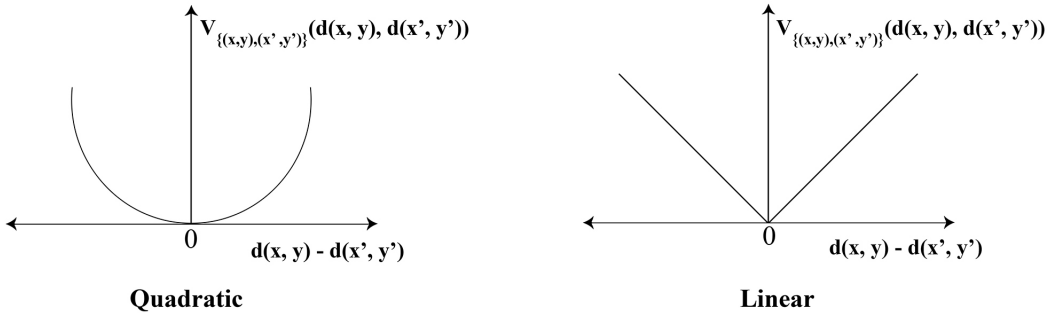

Figure 3.1: Examples of everywhere smooth prior.

Here, λ is a positive constant and represents the penalty for departure from the smoothness in d. The graphs of linear and quadratic are shown in Figure 3.1. The limitation of using such prior is that the disparity maps get oversmoothed and the discontinuities are not preserved [106].

In reality, the disparity maps are not smooth everywhere. Hence, a better model would be one that reconstructs the smooth disparities while preserving the sharp discontinuities. In order to take care of such a scenario, the discontinuity

preserving smoothness priors such as piecewise constant and piecewise smooth are used [21, 68, 117, 143].

Piecewise Constant prior

The piecewise constant prior encourages the disparity map consisting of several regions where pixels in the same region have equal disparities. It can be obtained by making $V_{\{(x,y),(x',y')\}}(d)$ zero if $d(x,y)$=$d(x',y')$ and a constant otherwise. An example of such prior corresponds to,

- Potts model

$$E_P(d) = \lambda \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} T(d(x,y) \neq d(x',y')), \tag{3.19}$$

where $T(.)$ is 1 if its argument is true and otherwise 0, and λ corresponds to smoothness penalty.

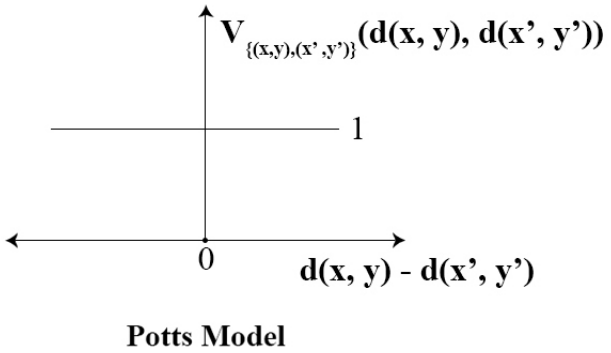

Figure 3.2: Example of piecewise constant prior.

The graph of Potts prior is shown in Figure 3.2. Such priors preserve discontinuities but are useful when the desired disparity map contains pieces of planar regions.

Piecewise Smooth Prior

Piecewise smooth prior encourages disparity map consisting of several regions where disparities in the same region vary smoothly. This prior can be constructed

by choosing such a $V_{\{(x,y),(x',y')\}}(d)$ which assigns a higher penalty for the larger difference between disparities $d(x,y)$ and $d(x',y')$ but sets a bound on the largest possible penalty. This avoids overpenalizing sharp jumps between the disparities of neighboring pixels. Examples of such discontinuity preserving smoothness functions are,

- Truncated quadratic

$$E_P(d) = \lambda \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} \min\left(k, (d(x,y) - d(x',y'))^2\right). \tag{3.20}$$

- Truncated linear

$$E_P(d) = \lambda \sum_{\{(x,y),(x',y')\}\in\mathcal{N}} \min\left(k, |d(x,y) - d(x',y')|\right). \tag{3.21}$$

  The graphs of truncated quadratic and truncated linear are shown in Figure 3.3. Here, k is a constant that sets the upper bound on the magnitude of $V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y'))$.

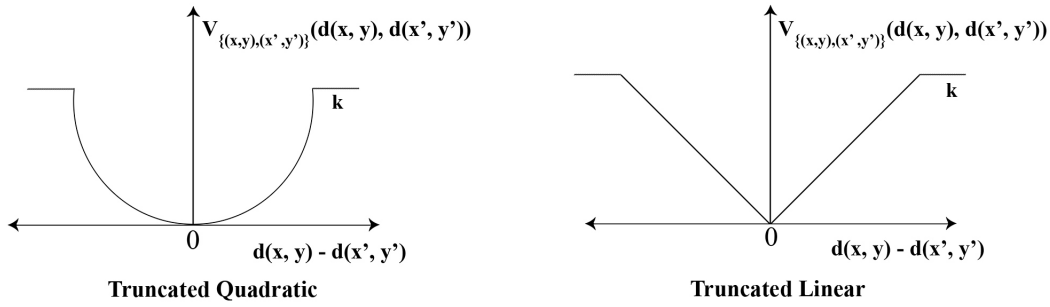

Figure 3.3: Examples of piecewise smooth prior.

These edge preserving priors contain a pair of global parameters, the smoothness penalty λ and discontinuity penalty k which are usually set by trial and error method when working on a set of images. This process is time consuming and the solution is highly sensitive to the values of these parameters. In order to overcome this problem, the parameters can either be estimated [155] or learned from a set of true disparities [111]. However, estimation or learning of parameters is computationally expensive and requires a large amount of data.

The another limitation with the use of these priors is that they are homogeneous MRF models i.e., the same set of global parameters are used at each pixel location. This assumption is not valid in practice since the variation among disparities at each pixel is different. These global parameters may not adapt to the local structure of the disparity map and hence fail to better capture the spatial dependence among disparities. We need an inhomogeneous MRF prior that considers the spatial variation among disparities locally. This motivates us to use an inhomogeneous Gaussian Markov random field (IGMRF) prior in our energy function which was first proposed in [54] for solving the satellite image deblurring problem. Use of IGMRF leads to spatially varying prior parameters thereby allow varying degrees of spatial smoothing. IGMRF can handle smooth as well as sudden changes in disparity map since it captures variation among disparities at each pixel location. IGMRF based prior model has been successfully used in solving satellite image deblurring problem [54], multiresolution fusion of satellite images [57], and super-resolution of images [33]. In this work, we propose to use an IGMRF as a prior model for disparity estimation in an energy minimization framework.

3.5 Use of Graphs cuts for Optimization

The energy functions of the form given in Eq. (3.13) are usually non-convex i.e., they have multiple local minima. Minimization of such functions is an NP-hard problem and hence computationally very expensive. In this thesis, we use the graph cuts, an efficient optimization method for energy minimization. Graph cuts [69, 21] guarantee that the solution so obtained either reaches the global optimum or reaches local minima close to the global minimum with considerably less computational time complexity. Graph cuts can be used for minimizing certain class of energy functions only i.e., the functions must satisfy the regularity condition [69, 21]. For example in Eq. (3.13), the first term is a function of a single pixel (x, y) and a function of single variable is always regular while the second term is a function of two variables, and it is regular iff $V_{\{(x,y),(x',y')\}}(d(x,y), d(x',y'))$

is either "metric" or "semi-metric". The basic approach in graph cuts is to construct a graph of the energy function to be minimized such that the minimum cut on the graph also minimizes the energy (either locally or globally). Kolmogorov and Zabih [69] present a general purpose graph construction for minimizing an energy function involving binary-valued variables only. The disparity estimation considers the energy function involving non-binary variables, i.e. real valued disparities, and this makes it NP-hard. The methods based on graph cuts can minimize an energy function with non-binary variables by repeatedly minimizing an energy function with binary variables in polynomial time.

The graph cuts algorithms are based on the *expansion* and the *swap* moves. It has been proved that running the expansion algorithm iteratively results in approximate solutions within a known factor of the global minima for an energy function with non binary labels provided that the smoothness term $V_{\{(x,y),(x',y')\}}$ is a metric. Consider a labeling d and a particular label α. A new labeling d' is defined to be an α-expansion move from d, if $d'(x,y) \neq \alpha$ implies $d'(x,y) = d(x,y)$. This means that the set of pixels assigned the label α has increased when going from d to d'. Similarly, consider a pair of labels α, β. A move from a labeling d to a new labeling d' is called an $\alpha - \beta$ swap, if $d'(x,y) = d(x,y)$ for any label not equal to α and β. This means that some pixels that were labeled α in d are now labeled as β in d', and some pixels that were labeled β in d are now labeled as α in d'. The advantage of these moves is that they allow a large number of pixels to change their labels simultaneously.

The expansion move algorithm cycles through the labels $\alpha \in \mathcal{L}$ in some order (fixed or random) and finds the lowest energy α-expansion move from the current labeling. If this expansion has lower energy than the current labeling then it becomes the current labeling. The algorithm terminates with a labeling that is a local minimum of the energy with respect to expansion moves, more precisely there is no α-expansion move for any label α, with lower energy. Similarly, the swap move algorithm finds a labeling that is a local minimum of the energy with respect to $\alpha - \beta$ moves. The key sub-problem in the expansion and swap move algorithm is to compute the lowest energy labeling within a single α-expansion and

$\alpha - \beta$ swap move of d, respectively. However, finding such a local minimum is not a trivial task since there may be an exponential number of swap or expansion moves for a given labeling d. This sub problem is solved efficiently with a simple graph cut using a graph construction.

Let G be a directed graph consisting of a set of vertices $\mathcal{V}$ and set of edges $\mathcal{E}$ with non-negative edge weights. It has two special vertices (terminals), namely the source a and the sink b. A cut C referred as the $a - b$ cut is a partition of the vertices in $\mathcal{V}$ into two disjoint sets A and B such that $a \in A$ and $b \in B$. The cost c of the cut is the sum of costs of all edges that go from A to B,

$$c(a, b) = \sum_{u \in A, v \in B, (u,v) \in \mathcal{E}} c(u, v).$$

The minimum $a - b$ cut problem is to find a cut C with the smallest cost. A minimum cut can be found in linear time by computing the maximum flow between the terminals, according to the Ford and Fulkerson theorem [19]. Note that there is a one to one correspondence between a cut $C = A, B$ and a labeling d. It is a mapping from the set of vertices $\mathcal{V} - \{a, b\}$ to $\{0, 1\}$ where $d(v) = 0$ means that $v \in A$ and $d(v) = 1$ means that $v \in B$. This is an example of a cut when d is considered as a binary valued labeling.

In order to solve the key sub problem of expansion or swap algorithm, a graph is constructed for an energy function. It is proved in [69, 21] that the minimum cut on the graph G corresponds to the lowest energy labeling within one expansion or swap move from d. One may refer [69] for details of graph construction. It is important to note that this sub problem is an energy minimization problem over binary variables even though the overall problem that the expansion or swap move algorithm is solving involves non binary variables. This is because, in expansion move each label will either keep its old value under d or acquire the new label α.

As an example, any labeling d' within a single α-expansion of the initial labeling d can be encoded by a binary labeling $f = \{f(x, y) | (x, y) \in S\}$ where $d'(x, y) = d(x, y)$ if $f(x, y) = 0$, and $d'(x, y) = \alpha$ if $f(x, y) = 1$. Since the energy

function $E(d)$ is defined over all labelings, it is also defined over labelings specified by binary labelings. Hence, the key step here is to find the minimum of $E(f)$ over all binary labelings f. The importance of energy functions of binary variables results from the fact that a cut effectively assigns one of two possible values to each vertex of the graph. So any energy minimization construction based on graph cuts relies on intermediate binary variables.

3.6 Proposed Approach

We now present our proposed approach for dense disparity estimation using IGMRF prior in an energy minimization framework.

3.6.1 IGMRF Model for Disparity

We model the disparity map d by an IGMRF prior in our energy function that adjusts the amount of regularization locally. To formalize this prior, the $V_{\{(x,y),(x',y')\}}(d)$ in Eq. (3.16) is chosen as the square of finite difference approximation to the first order derivative of disparities at each pixel. Let the prior term $E_P(d)$ defined using IGMRF is denoted as $E_{IGMRF}(d)$ in our energy function. Considering the differentiation in horizontal and vertical directions at each pixel location, one can write $E_{IGMRF}(d)$ as [54]:

$$\begin{aligned} E_{IGMRF}(d) = & \sum_{(x,y)} b^X_{(x,y)}(d(x-1,y) - d(x,y))^2 \\ & + \sum_{(x,y)} b^Y_{(x,y)}(d(x,y-1) - d(x,y))^2. \end{aligned} \tag{3.22}$$

Here, b^X and b^Y are the spatially adaptive IGMRF parameters in horizontal and vertical directions, respectively. Thus, $\{b^X_{(x,y)}, b^Y_{(x,y)}\}$ forms a 2D parameter vector of IGMRF at each pixel location (x, y) in the disparity map. A low value of b indicates the presence of an edge between two neighboring disparities. These parameters help us to obtain a solution which is less noisy in smooth areas and preserve the depth discontinuities in other areas. Now, in order to estimate IGMRF parameters, we need the true disparity map which is unknown and has to be estimated.

Therefore, an approximation of d has to be accurately determined if we want the parameters obtained from it to be significant for regularization. To start the regularization process, we use an initial estimate of disparity map obtained using a suitable approach and compute these parameters which are then used to estimate the d.

3.6.2 Learning the Initial Estimate of Disparity Map

In this section, we discuss our proposed learning based approach for obtaining an initial estimate of the disparity map. For this, we use a database of stereo images and their corresponding ground truth disparity maps. This database is then used to learn the true relationship of the spatial features of a disparity map across the scales. The advantage of our learning method is that it is a simple approach and unlike other learning based methods [111, 147] it do not need any probabilistic model. Disparities are estimated from the available data itself. For learning, we consider n_s sets of stereo images of various scenes and each set has n_v rectified views. We obtain the disparity map for each of the stereo image set using the standard multiple baseline stereo method [99]. A single level Gaussian pyramid decomposition is applied on these n_v views for each stereo set and disparities are obtained on these using the same approach. A pyramidical decomposition is used in order to better constrain the solution while learning. We now have a database consisting of n_s disparity maps estimated using the original data, n_s disparity maps corresponding to the Gaussian filtered and downsampled versions and n_s true disparity maps.

Given a test stereo image set with n_v rectified views, we first use one level Gaussian decomposition on these images. The same approach (for which the initial disparity of training set is estimated) of multiple baseline stereo is used to obtain the disparity maps for the test stereo set as well as for their downsampled versions. We divide the test disparity map into small patches of size 2×2 and estimate the final disparities for each patch separately. Similarly, all the disparity maps in the training set are also divided into small patches of size 2×2. To learn the disparity, we start with the first patch of test disparity map with the corre-

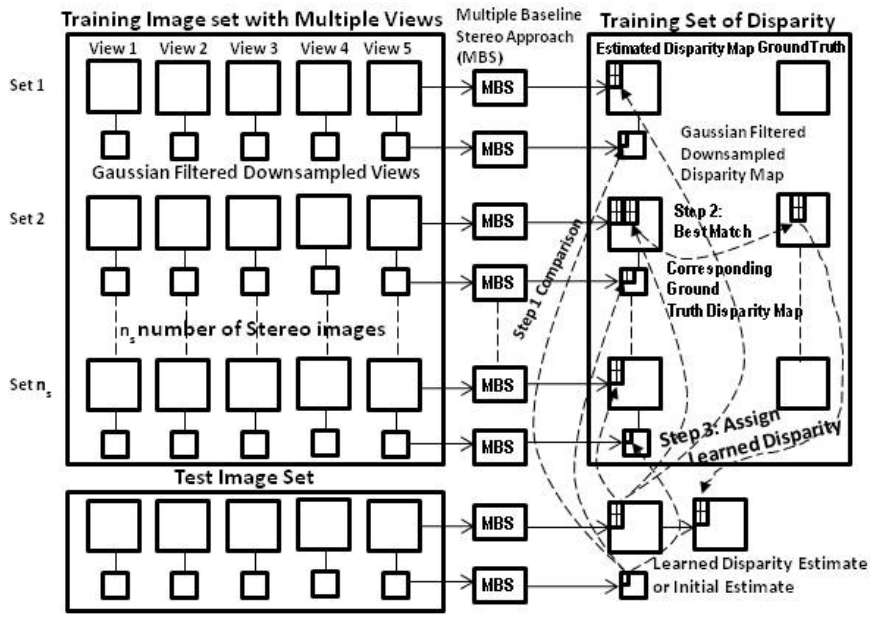

Figure 3.4: Block diagram of learning based approach for obtaining an initial estimate of the disparity map. (Here, we used $n_v = 5$ and $n_s = 60$ in our experiments).

sponding single disparity value in its downsampled version and compare these values with all the patches in training disparity maps with their corresponding single disparity value in their downsampled versions. The comparison is done using the sum of squared distance (SSD) measure. Let the patch of k^{th} disparity map in the training set gives the minimum SSD value with test disparity patch. The location of that patch is noted and the disparity patch of k^{th} true disparity map in the training set is extracted from the noted location. These true disparities are the final learned disparities for the test stereo set. The same procedure is repeated for all the patches in the test disparity map. This gives us the initial estimate for the disparity map. Our proposed learning method for obtaining the initial estimate is illustrated by the block diagram shown in Figure 3.4.

3.6.3 Estimation of IGMRF Parameters

In order to estimate the IGMRF parameters, we employ the method proposed in [54]. The maximum likelihood estimate (MLE) of true disparity map g is,

$$\hat{b}^X = \arg\max_{b^X} \quad \left(\log P(g|b^X)\right), \quad \text{and} \tag{3.23}$$

$$\hat{b}^Y = \arg\max_{b^Y} \quad \big(\log P(g|b^Y)\big). \tag{3.24}$$

The log-likelihood derivatives are,

$$\frac{\partial \log P(g|b^X)}{\partial b^X_{(x,y)}} = \mathbf{E}\big((\hat{d}(x-1,y) - \hat{d}(x,y))^2\big) - \big((g(x-1,y) - g(x,y))^2\big), \tag{3.25}$$

$$\frac{\partial \log P(g|b^Y)}{\partial b^Y_{(x,y)}} = \mathbf{E}\big((\hat{d}(x,y-1) - \hat{d}(x,y))^2\big) - \big((g(x,y-1) - g(x,y))^2\big), \tag{3.26}$$

where $\mathbf{E}$ refers to the expectation operator, $\hat{d}$ corresponds to the maximum a posteriori estimate of the disparity map and g is the true disparity map. Therefore, the estimation problem consists of solving the systems given in Eqs. (3.25) and (3.26). It can be seen that the expectation term only depends on the parameters b^X, b^Y and the other term only depends on g. This simplifies the estimation problem. It is sufficient to compute the variance of each pixel difference with respect to the prior laws $\mathbf{E}\big((g(x-1,y) - g(x,y))^2\big)$ and $\mathbf{E}\big((g(x,y-1) - g(x,y))^2\big)$. Jalobeanu *et al.* [54] propose the simplest approximation of the local variance. The variance of the gradients $(g(x-1,y) - g(x,y))^2$ and $(g(x,y-1) - g(x,y))^2$ are equal to the variance of the same gradients in the homogeneous case i.e., when all the parameters are equal to the corresponding $b^X_{(x,y)}$ and $b^Y_{(x,y)}$, respectively. Covariance matrix of the homogeneous prior distribution is diagonalized by a Fourier transform and variance can be calculated as $1/4b^X$ and $1/4b^Y$ [1]. This gives us,

$$\hat{b}^X_{(x,y)} = \frac{1}{4\big((g(x-1,y) - g(x,y))^2\big)}, \quad \text{and} \tag{3.27}$$

$$\hat{b}^Y_{(x,y)} = \frac{1}{4\big((g(x,y-1) - g(x,y))^2\big)}. \tag{3.28}$$

Since the true disparity map g is not available, we use an initial estimate d_0 of disparity map using our learning based approach discussed in previous subsection.

The parameters are obtained using,

$$\hat{b}^{X}_{(x,y)} = \frac{1}{4\big((d_0(x-1,y) - d_0(x,y))^2\big)}, \tag{3.29}$$

$$\hat{b}^{Y}_{(x,y)} = \frac{1}{4\big((d_0(x,y-1) - d_0(x,y))^2\big)}, \tag{3.30}$$

where $d_0(x,y))$ is the disparity at location (x,y) in the initial estimate. The refined estimates of the IGMRF prior parameters in X and Y directions are obtained using the following equations:

$$\hat{b}^{X}_{(x,y)} \approx \frac{1}{\max(4(d_0(x-1,y) - d_0(x,y))^2, 4)}, \quad \text{and} \tag{3.31}$$

$$\hat{b}^{Y}_{(x,y)} \approx \frac{1}{\max(4(d_0(x,y) - d_0(x,y-1))^2, 4)}. \tag{3.32}$$

As seen from the above equations, in order to avoid computational difficulty, we set an upper bound $\hat{b} = 1/4$ whenever gradient becomes zero i.e., whenever the neighboring disparities are the same. Thus, we set a minimum spatial difference of 1 for practical reasons. This avoids obtaining high regularization parameter that would slow down the optimization. It ensures that the pixels with zero disparity difference are weighted almost same as those with small disparity difference (in this case with a disparity difference of one).

3.6.4 Final Disparity Map Estimation

In this section, we present our proposed approach for the estimation of dense disparity map in an energy minimization framework. The block schematic of the proposed approach is shown in Figure 3.5. Making use of the database of stereo images and their corresponding ground truth disparity maps, we first obtain the initial estimate of the disparity map. We then model the disparity map as an IGMRF and estimate the IGMRF parameters from the initial estimate using Eqs.

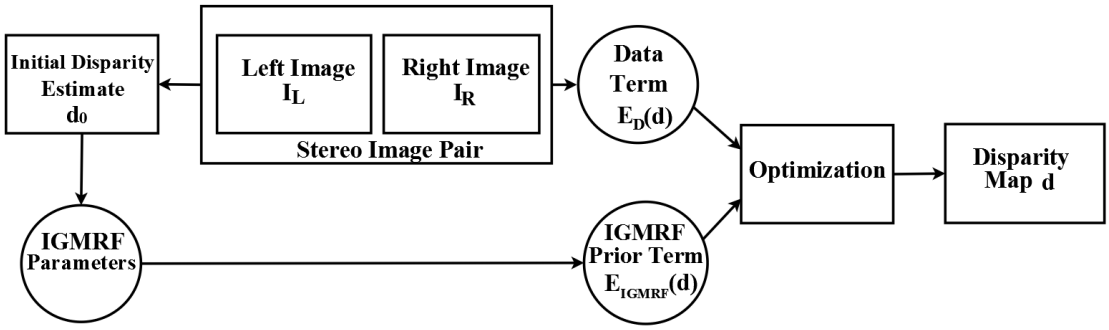

Figure 3.5: Block schematic of the proposed approach for dense disparity estimation.

(3.31) and (3.32) at every pixel location. The IGMRF model on the disparity map serves as the prior for the disparity estimation in which the prior parameters are already known. The final estimate of disparity map is obtained by using the energy minimization framework where we use the energy function of the form given in Eq. (3.14). We use the data term $E_D(d)$ as given in Eq. (3.15) and the prior term $E_P(d)$ is defined by $E_{IGMRF}(d)$ using Eq. (3.22). Using these Eqs., the final energy function to be minimized can be expressed as:

$$\begin{aligned} E(d) = & \sum_{(x,y)} (I_L(x,y) - I_R(x + d(x,y), y))^2 \\ & + \sum_{(x,y)} b^X_{(x,y)} (d(x-1,y) - d(x,y))^2 \\ & + \sum_{(x,y)} b^Y_{(x,y)} (d(x,y-1) - d(x,y))^2. \end{aligned} \tag{3.33}$$

Since the prior term is semi-metric in nature and the energy function is regular, it can be efficiently minimized using graph cuts optimization based on $\alpha - \beta$ swap move which quickly leads to the minima. The choice of an initial solution fed to the optimization process determines the speed of convergence and hence use of the initial estimate as the starting point speed-up the optimization process.

3.7 Experimental Results

In this section, we demonstrate the efficacy of the proposed method. We conducted various experiments and evaluated our results on the Middlebury stereo

Dataset [113]	Size	Disparity range
Venus	383×434	0-20
Teddy	375×450	0-64
Cones	375×450	0-64

Table 3.1: Size of stereo images and their disparity range used for experimentations

datasets [113]. Experiments were conducted on the "Venus", "Cones" and "Teddy" stereo pairs belonging to Middlebury stereo 2001 and 2003 datasets [113]. The size of each data set and disparity range are given in Table 3.1. For learning the initial estimate, we create our training database using the stereo image pairs belonging to Middlebury stereo 2005 and 2006 datasets [113] with n_s=60 and n_v=5. Note that the dataset used here for training was different from the test dataset. We mention that in all our works, we used "gray scale" stereo images for experimentations.

Figure 3.6 shows the left image, ground truth, initial estimate and the final disparity map. Here, the disparity maps are represented as brightness images. The brighter pixel indicates that the object point is nearer to the camera or at a lesser depth and hence have the higher disparity because depth and disparity are inversely proportional to each other. Similarly, darker pixel indicates the lesser disparity and higher depth from the camera. We can clearly observe that the proposed method gives pretty good disparity estimates in homogeneous as well as textured regions while preserving the sharp discontinuities at object boundaries. For example, it preserves the smooth variation of disparities in the various planar regions of "Venus" image and retains the discontinuities as well. Similarly, it better captures the disparities locally in the cones and the face of the statue well in the "Cones" image. Our method clearly preserves sharp discontinuities due to the use of inhomogeneous MRF prior. Though, the initial estimate is piecewise smooth but it shows noise in some homogeneous regions. One can see that the final disparity map shows a significant improvement in smooth regions as well as at object boundaries when compared to the initial estimate.

To perform the quantitative evaluation, we used the percentage of bad matching pixels ($\mathcal{B}$%) as the error measure with a disparity error tolerance δ. For an

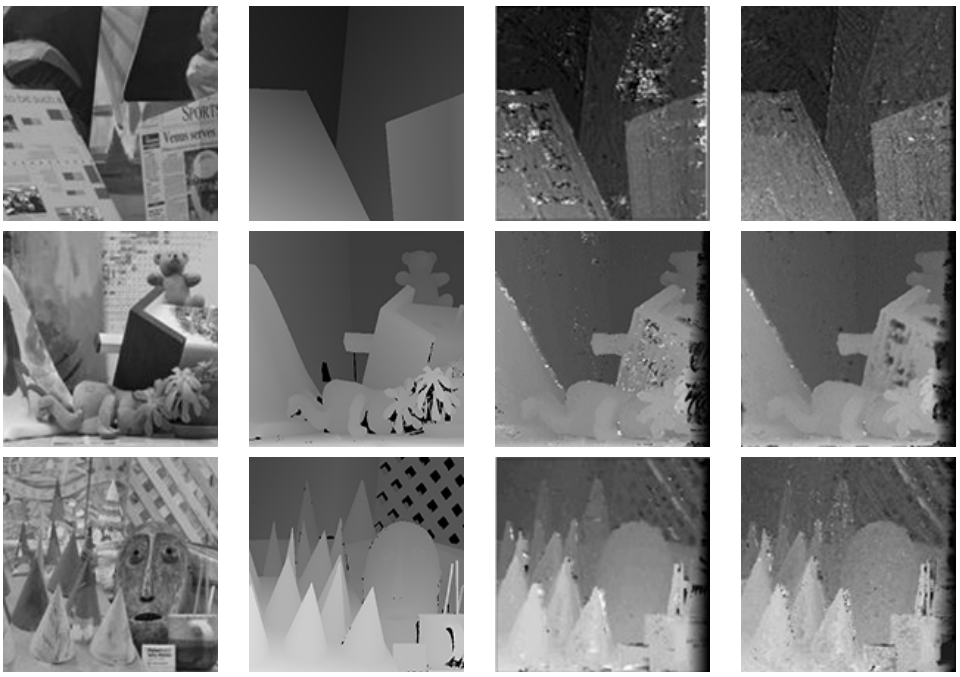

Figure 3.6: Results for the datasets of [113], "Venus" (1^{st} row), "Teddy" (2^{nd} row) and "Cones" (3^{rd} row).

Method	**Venus**	**Teddy**	**Cones**
Initial Estimate (Proposed)	4.37	26.3	21.5
GraphCuts [21]	3.44	25.0	18.0
Final Estimate (Proposed)	**3.12**	**21.3**	**17.8**

Table 3.2: Quantitative evaluation of results on the Middlebury datasets [113] in terms of % of bad matching pixels computed over the entire image with δ=1.

estimated disparity map d, the $\mathcal{B}$% is computed with respect to the ground truth disparity map g as follows [114]:

$$\mathcal{B} = \frac{1}{M * N} \sum_{(x,y)} |d(x,y) - g(x,y)| > \delta, \tag{3.34}$$

In all our works, the experiments were conducted on a computer with Core i7-3632QM, 2.20 GHz processor and 8.00 GB RAM.

The quantitative evaluation of the proposed method is shown in Table 3.2. We compare our results with those obtained using discontinuity preserving homogeneous MRF priors with energy minimization via graph cuts as proposed in [21]. For example, truncated quadratic on "Venus" data set, and truncated linear

on “Cones” and “Teddy” datasets is used. Results show that the use of IGMRF prior significantly improves the performance when compared to the use of other MRF priors. The results show the significance of learning as well as IGMRF regularization. The average time for obtaining the initial estimate was approximately 100 seconds and it was approximately 60 seconds for obtaining final disparity estimate via graph cuts optimization. The computation time for obtaining the initial estimate highly depends on the size of test image and the number of images present in the training set. One may reduce the time complexity in obtaining the initial estimate by using training images belonging to the same class as the test image and this can be done by using suitable image retrieval algorithm as a first step prior to building the database.

3.8 Conclusion

In this chapter, we have proposed a new approach for dense disparity estimation in an energy minimization framework in which an IGMRF prior was used. The model parameters were computed at each pixel location using the initial estimate of the disparity map which was obtained using a learning based approach. A database of stereo images and their corresponding ground truth disparity maps were used in learning the initial estimate. Since our energy function is non convex, we used a graph cuts based optimization technique that gives a solution close to global optimum. The experimental results showed that the disparity maps obtained using the proposed method were less noisy in homogeneous areas and preserved the textures and sharp details in other regions. Our results demonstrated that the use of IGMRF prior lead to better disparity map when compared to edge-preserving homogeneous MRF priors.

An inherent limitation of the proposed learning algorithm for obtaining initial estimate is computationally slow, and the quality of the solution depends on the computed disparities that were used while learning. In addition to this, the accuracy of the IGMRF parameters depends on the quality of the initial estimate. One can obtain better parameters by using a map close to the ground truth. The results

indicated that the initial estimate was not free from noise in homogeneous regions and contributed to bad matching pixels in the final solution which itself depends on the inhomogeneous IGMRF parameters estimated using the initial estimate. One can enhance the quality of the final results by improving the quality of initial estimate.

In the next chapter, we propose an another method for dense disparity estimation in an energy minimization framework. The approach uses a sparsity prior in addition to IGMRF prior for regularizing the solution. We use a different method for obtaining a good initial estimate for improving the accuracy of IGMRF parameters.

CHAPTER 4

Sparseness based Regularization Framework

Dense disparity estimation is an ill-posed problem and it can be efficiently solved in an energy minimization framework with the use of a proper regularization. In the previous chapter, we proposed a disparity estimation method using an inhomogeneous Gaussian MRF prior in the energy minimization framework. We demonstrated that the use of IGMRF prior results in a better disparity map than that obtained using edge-preserving homogeneous MRF priors. However, the drawback of the IGMRF prior is that it fails to capture higher order dependencies in the disparity map. To do this, one may use a triple-wise or a higher order clique potentials. One of the recently proposed approach [133] uses second order smoothness as a prior for disparity estimation. These priors capture the spatial correlatedness among disparities in a larger neighborhood, and hence perform better in untextured regions and slanted surfaces. However, these methods are computationally very expensive. We need a prior that can model the disparity characteristics in a larger neighborhood, without the need of higher order MRFs. One example of such spatial regularization is *sparsity prior*.

In general, disparity maps are made up of homogeneous regions with a limited number of discontinuities resulting in redundancy. Because of this, one can represent the disparities in a domain in which they are sparse. Here, sparsity means that there are few significant pixel locations with nonzero values. This transform domain representation can be obtained either using the fixed set of basis such as discrete Fourier transform (DFT), discrete cosine transform (DCT), discrete wavelet transform (DWT) [108, 42], or using an overcomplete dictionary [3]. Finding sparse representations of depth or disparity maps is important for

applications involving inverse problems such as depth denoising and inpainting [88], and depth map compression [75]. In reality, finding the sparse representation of a disparity map is computationally expensive, and hence a better choice would be to find the sparse representation of patches of small size individually and average the resultant sparse patches in order to get the sparse representation. The sparseness feature of the disparities can be considered as a prior knowledge, and hence it can be incorporated as a sparsity prior while regularizing. The use of sparsity information models the higher order dependencies that can be used along with the IGMRF to get a better result. This is because each of the sparseness coefficient captures the entire image characteristic, for example, for a 1D signal or an image, the computation of a Fourier coefficient that represents the frequency domain characteristic requires a complete signal/image and these coefficients are sparse. Hence, we can say that sparsity prior captures higher order dependencies in the signal/image.

In this chapter, we incorporate the sparsity prior in addition to the IGMRF prior in our energy function. The sparsity prior captures sparseness in the disparity map and is defined using the learned overcomplete sparseness of disparity patches. In this chapter, we propose two learned overcomplete sparseness models for disparity estimation. In the first method, an overcomplete dictionary is learned using K-singular value decomposition (K-SVD) algorithm. In the second method, we use a sparse autoencoder for learning a better sparse representation of disparities. In both these approaches, we use a better initial estimate obtained using a classical local stereo method [114]. We also use a data term different from the approach used in the previous chapter. Experiments are carried out to show the effectiveness of using sparsity prior in improving the accuracy of estimated disparity map.

4.1 Related Work

The first work to estimate the dense disparity map from sparseness was proposed by Hawe *et al.* [42]. Here, the authors use a compressive sensing (CS) based ap-

proach for reconstructing a disparity map from few disparity measurements. The sparseness is represented using a fixed wavelet basis and the edge points are considered as sampling locations. The method requires an initial disparity map and the sampling locations which are scene dependent. The accuracy of estimated final disparities depends on the reliable initial estimate. In [40], a dictionary is used to obtain sparse representation and the derived sparseness is used as regularizer for reconstructing a depth map. The limitation of this approach is that the dictionary is manually designed for 3D reconstruction in man-made environments.

Inverse ill-posed problems have been extensively studied for natural images [22]. However, because of differences between image and depth statistics, it is not obvious that the fixed set of basis, for example, DCT, DWT or DFT bases are the most efficient way to represent the structure of depth/disparity maps. Thus, we prefer to learn an efficient sparse representation from a large database of examples instead of using a fixed basis. Learning the sparse representation has achieved superior performance in solving various inverse problems [3]. For example, learned sparseness using the overcomplete dictionary has been successfully used as a regularizer for solving the image denoising and inpainting problems [30, 85]. This idea of learning a dictionary that yields sparse representations for a set of training images has been studied in a number of works [100, 31, 73, 72, 3]. Here, the advantage is that the representation has higher accuracy than obtained with the use of fixed basis and this can be done by adapting its columns to fit a given training data [3]. This motivates us to learn the sparse representation of disparities using an overcomplete dictionary and define a sparsity prior using the learned sparseness. Such sparsity priors complement the IGMRF prior.

Recently, Tosic *et al.* [123] propose a two-layer graphical model for inferring the disparity map. They include a sparsity prior in an existing MRF based stereo matching framework. Here, the sparse representation of disparities is inferred by a dictionary which is learned using a sparse coding technique which can cope up with non-stationary depth estimation errors. Although, it performs better when compared to discontinuity preserving homogeneous MRF prior, the solution can be improved by using inhomogeneous MRF prior. However, their method is com-

plex and computationally intensive.

4.2 Problem Formulation

Our main goal is to find the disparity map d for a given rectified pair of stereo images I_L and I_R, and as done in the previous chapter we formulate the problem as the minimization of the following energy function:

$$E(d) = E_D(d) + E_P(d). \tag{4.1}$$

In the previous chapter, for defining the data term we made an assumption that two pixels (x, y) and $(x + d(x, y), y)$ in left and right image, respectively represent corresponding pixels if their intensities are similar. Based on this, we chose the matching cost as a summation of the squared difference of corresponding pixel intensities. However, there are special circumstances when corresponding pixels may have very different intensities and this may be due to the effects of image sampling, noise, different gains and biases of the left and right camera, discontinuities, etc. Moreover, an intensity of a pixel may not represent the image of a point in the scene but of a surface patch, and two pixels that contain corresponding world points integrate light reflected from two different surface patches due to foreshortening, depth discontinuities, lens blur, image sampling, etc.

Due to the effects of image sampling, the intensities of corresponding pixels may change and the disparity may not be an integer. In this work, we derive our data term using the intensity matching cost that is insensitive to image sampling and robust to outliers proposed by Birchfield and Tomasi (BT) [13]. To do this, we first measure how well a pixel (x, y) fits into the real-valued range of disparities $(d(x, y) - \frac{1}{2}, d(x, y) + \frac{1}{2})$ by finding,

$$F^{fwd}_{(x,y)}(d(x, y)) = \min_{d(x,y)\pm\frac{1}{2}} \big(|I_L(x, y) - I_R(x + d(x, y), y)|\big). \tag{4.2}$$

Though, the intensities are known at integer locations only, linear interpolation is used to obtain the intensity at non integer pixel location. $F^{fwd}_{(x,y)}(d(x, y))$ is a

matching cost function that measures the cost of assigning the disparity $d(x,y)$ to pixel (x,y) and considers I_L as primary image and I_R as secondary image. We aim to derive a symmetric cost function such that when the reference image and the target image are switched, the form of matching cost does not change. Also, the symmetric matching cost improves the performance of global stereo methods [145]. Hence, for symmetry, we also measure,

$$F^{rev}_{(x,y)}(d(x,y)) = \min_{x\pm\frac{1}{2}}\big(|I_L(x,y) - I_R(x+d(x,y),y)|\big). \tag{4.3}$$

The final matching cost $F_{(x,y)}(d(x,y))$ is computed by,

$$F_{(x,y)}(d(x,y)) = \min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y))\}. \tag{4.4}$$

Note that we estimate the integer disparities only. In order to make our matching cost robust against outliers and occlusion, we use a truncation value τ on the cost. Hence, $F_{(x,y)}(d(x,y))$ can be rewritten as:

$$F_{(x,y)}(d(x,y)) = \min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau)\}. \tag{4.5}$$

Further using Eq. (4.5), we give our data term $E_D(d)$ which is robust against outliers and insensitive to image sampling by,

$$E_D(d) = \sum_{(x,y)}\big(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau)\}\big). \tag{4.6}$$

For finding the correspondences, we consider search from left to right as well as from right to left and hence relax the traditional ordering constraint used in disparity estimation. This results in positive as well as negative disparities.

In order to perform the regularization, we model d using its prior characteristics and form the energy term $E_P(d)$. We define $E_P(d)$ as a sum of IGMRF and sparsity priors, and it is given as:

$$E_P(d) = E_{IGMRF}(d) + \gamma E_{sparse}(d), \tag{4.7}$$

where $E_{IGMRF}(d)$ and $E_{sparse}(d)$ represent the IGMRF and sparsity prior terms, respectively. Here, γ controls the weight of the term $E_{sparse}(d)$. The IGMRF prior captures the spatial variation among disparities locally as well as it preserves sharp discontinuities while the sparsity prior captures the higher order dependencies in terms of sparseness in the disparity map. The combination of these two priors better constrains the solution. We minimize our energy function using graph cuts optimization. In general, for non-convex energy functions, graph cuts results in a local minimum that is within a known factor of the global minimum. In order to ensure global minimum, we use an iterative optimization with proper settings of parameters. At every iteration, the IGMRF parameters and sparseness are refined in order to obtain better disparity estimates (converging towards global optima). The number of iterations may vary for different stereo pairs. In the next sections, we discuss two proposed approaches for disparity estimation based on overcomplete sparseness model learned using an overcomplete dictionary and a sparse autoencoder, respectively.

4.3 Learning Sparseness using Overcomplete Dictionary

In this method, the sparsity prior is defined using the spatially varying patterns i.e., the disparity patches where each patch is encoded via a sparse representation using the learned overcomplete disparity dictionary. Here, we learn the overcomplete dictionary and sparse representation of disparities using the K-SVD algorithm.

4.3.1 Sparse Model for Disparity

The overcomplete dictionary model represents a useful framework for sparsely representing the disparities. We consider a lexicographically ordered disparity patch $d^{(x,y)} \in \mathbb{R}^n$ of size $\sqrt{n} \times \sqrt{n}$ at a pixel location (x, y) in disparity map d [1].

[1]Note that $d(x, y)$ is the disparity at location (x, y) and $d^{(x,y)}$ is the disparity patch at a location (x, y) in d.

Using an overcomplete dictionary $\mathcal{D} \in \mathbb{R}^{n \times K}$ that consists of K columns $\{\mathcal{D}_j\}_{j=1}^{K}$ such that $n < K$, a disparity patch $d^{(x,y)}$ can be represented as a linear combination of these columns as:

$$d^{(x,y)} = \mathcal{D}a^{(x,y)}. \tag{4.8}$$

The vector $a^{(x,y)} \in \mathbb{R}^K$ has the representation of $d^{(x,y)}$. The condition $n < K$ is called as *overcompleteness*. If $n < K$ and $\mathcal{D}$ has a full rank, an infinite number of solutions exists for the system in Eq. (4.8) and hence to obtain a unique solution, a regularization term is added that encourages sparsity in the solution $a^{(x,y)}$. Thus, the sparse solution is obtained by solving the following minimization problem,

$$\min_{a^{(x,y)}} \left|\left| d^{(x,y)} - \mathcal{D}a^{(x,y)} \right|\right|_2^2 \quad \text{s.t.} \quad \left|\left| a^{(x,y)} \right|\right|_0 \leq t, \tag{4.9}$$

where t is the maximum number of non-zero entries in sparse vector $a^{(x,y)}$. Here, the sparsity is enforced by the $||.||_0$ i.e., the l_0 norm. The sparse solution to Eq. (4.9) is called "sparse coding" where we assume that the dictionary $\mathcal{D}$ is known and fixed. The exact determination of sparsest solution is NP-hard and hence an approximate solution is sought [28]. In past two decades, several approximation algorithms have been proposed. The most popular and effective sparse coding algorithms are, matching pursuit [86], orthogonal matching pursuit [124, 101], basis pursuit [24] and focal under determined system solver [38]. A detailed description of these methods can be found in [3].

The choice of a dictionary plays a crucial role in obtaining the sparse solution, and the best way is to learn it from the training set of examples [3]. The advantage of using a learned overcomplete dictionary is that the representation would be sparser than obtained with the use of fixed basis or predefined dictionaries, and this is done by adapting its columns to fit a given training data. Overcomplete dictionaries designed using K-SVD algorithm [3] have been successfully used in many applications such as image denoising [30] and restoration [85]. This algorithm is flexible and can work in conjunction with any pursuit algorithm. Due to the simplicity and effectiveness of K-SVD algorithm, we propose to use it for learning our overcomplete dictionary and inferring the sparse representation of

disparities.

The dictionary can be learned either from a large set of ground truths or from the available data via K-SVD. For example, the authors in [30] consider two options for training the dictionary while solving the image denoising problem: (1) training the dictionary on a corpus of patches taken from a high quality set of natural images, or (2) training using patches from the available data i.e, corrupted image itself. The idea of learning the dictionary using corrupted patches is natural because the K-SVD has noise rejection capability [3], and we choose this idea of dictionary training. We train our dictionary using the patches of estimated disparity map of the given stereo pair. Since the true disparity map is unknown and has to be estimated, we use an initial estimate of the disparity map. The advantage of our dictionary learning method is that the learned dictionary is adaptive to the disparities of the given stereo pair and we do not require the large set of ground truth maps for training. We now present the K-SVD algorithm for learning the overcomplete dictionary of disparity patches.

4.3.2 The K-SVD Algorithm

We consider a training set $\mathcal{G}$ of overlapping disparity patches with each patch $d^{(x,y)} \in \mathbb{R}^n$ extracted at location (x, y) in the disparity map $d \in \mathbb{R}^{M \times N}$. Let the number of patches in training set $\mathcal{G}$ be n_d. Given a training set, we seek an optimal overcomplete dictionary $\mathcal{D}$ that leads to the sparse representation for each member $d^{(x,y)}$ in this set. Both the $\mathcal{D}$ and $a^{(x,y)}$ can be obtained by formulating the problem as,

$$\begin{aligned} \underset{\mathcal{D}, a^{(x,y)}}{\arg\min} \quad & \sum_{(x,y)} \left|\left| d^{(x,y)} - \mathcal{D} a^{(x,y)} \right|\right|_2^2, \\ \text{subject to} \quad & \forall (x, y) \in \mathcal{S}, \left|\left| a^{(x,y)} \right|\right|_0 \leq t. \end{aligned} \tag{4.10}$$

Here, $a^{(x,y)} \in \mathbb{R}^K$ represents the sparse vector of disparity patch $d^{(x,y)} \in \mathcal{G}$ and $\mathcal{S}$ is the set of sites or pixels. K-SVD is an iterative method that alternates between sparse coding of the training patches based on the current dictionary and a process of updating the dictionary columns to better fit the data. The update

of the columns of dictionary is combined with an update of the sparse representations, thereby accelerating convergence [3]. The algorithm can be described by the following steps,

1. Set the initial dictionary $\mathcal{D}^{(0)} \in \mathbb{R}^{n \times K}$ with l_2 normalized columns. Start with iteration number $J = 1$.
2. Repeat until convergence (stopping rule):
   (a) **Sparse Coding Stage:** Using a pursuit algorithm to compute the sparse representation $a^{(x,y)}$ for each disparity patch $d^{(x,y)} \in \mathcal{G}$ by approximating the solution of,

$$\forall (x,y) \in \mathcal{S} \quad \min_{a^{(x,y)}} \left|\left| d^{(x,y)} - \mathcal{D}a^{(x,y)} \right|\right|_2^2 \quad \text{s.t.} \quad \left|\left| a^{(x,y)} \right|\right|_0 \leq t. \tag{4.11}$$

   (b) **Dictionary Update Stage:** For each column $k = 1, 2, \ldots, K$ in $\mathcal{D}^{(J-1)}$, update it as follows:
   - Consider a matrix $\mathcal{A}$ with its column vectors as the sparse representation $a^{(x,y)}$. Let the k^{th} column of $\mathcal{D}$ is denoted as $\mathcal{D}_k$ and the sparse coefficients of each patch correspond to $\mathcal{D}_k$ is the k^{th} row in $\mathcal{A}$ denoted as a_T^k. With this, the penalty term (l_2 norm) in Eq. (4.10) can also be expressed as,

$$\begin{aligned} \sum_{(x,y)} \left|\left| d^{(x,y)} - \mathcal{D}a^{(x,y)} \right|\right|_2^2 &= ||\mathcal{G} - \mathcal{D}\mathcal{A}||_F^2 \\ &= \left|\left| \mathcal{G} - \sum_{j=1}^{K} \mathcal{D}_j a_T^j \right|\right|_F^2 \\ &= \left|\left| \left(\mathcal{G} - \sum_{j \neq k} \mathcal{D}_j a_T^j\right) - \mathcal{D}_k a_T^k \right|\right|_F^2 \\ &= \left|\left| E_k - \mathcal{D}_k a_T^k \right|\right|_F^2. \end{aligned} \tag{4.12}$$

     Here, $||.||_F$ corresponds to Frobenius norm. It can be clearly observed that in Eq. (4.12), the multiplication $\mathcal{D}A$ is decomposed into sum of K number of rank-1 matrices. Among those, $K-1$ terms are

fixed and the error matrix E_k stands for the error for all the patches in $\mathcal{G}$ when the k^{th} column of $\mathcal{D}$ is removed. The error E_k is given by,

$$E_k = \mathcal{G} - \sum_{j \neq k} \mathcal{D}_j a_T^j. \tag{4.13}$$

- Define w_k as the group of indices pointing to patches in $\mathcal{G}$ that use the dictionary column $\mathcal{D}_k$ i.e., where $a_T^k(i)$ is nonzero. Thus,

$$w_k = \{i | 1 \leq i \leq n_d, a_T^k(i) \neq 0\}. \tag{4.14}$$

- Now, restrict E_k by choosing only the columns corresponding to w_k and obtain E_k^R corresponding to this. Similarly, restrict a_T^k by discarding zero entries, resulting in a row vector a_R^k of length $|w_k|$. Based on this, the penalty term given in Eq. (4.12) can now be written as:

$$\left|\left| E_k^R - \mathcal{D}_k a_R^k \right|\right|_F^2. \tag{4.15}$$

  The minimization of Eq. (4.15) w.r.t. $\mathcal{D}_k$ and a_R^k can be efficiently done via singular value decomposition (SVD). The SVD finds the closest rank-1 matrix that approximates E_k^R by decomposing $E_k^R = X\Delta V^T$ where X, Δ and V correspond to orthonormal matrix containing eigen vectors of $E_k^R {E_k^R}^T$, rectangular diagonal matrix containing singular values of E_k^R, and orthonormal matrix containing eigen vectors of ${E_k^R}^T E_k^R$, respectively. The updated solution for the dictionary column $\mathcal{D}_k$ is chosen as the first column of X and the updated sparse coefficient vector a_R^k is chosen as the first column of V multiplied by $\Delta(1,1)$ i.e., the first element of matrix Δ.

(c) Set J=J+1.

The K-SVD algorithm efficiently learns dictionary as well as the sparse representation of each patch in the training set simultaneously. In our approach, the dictionary is refined iteratively in order to obtain a better d.

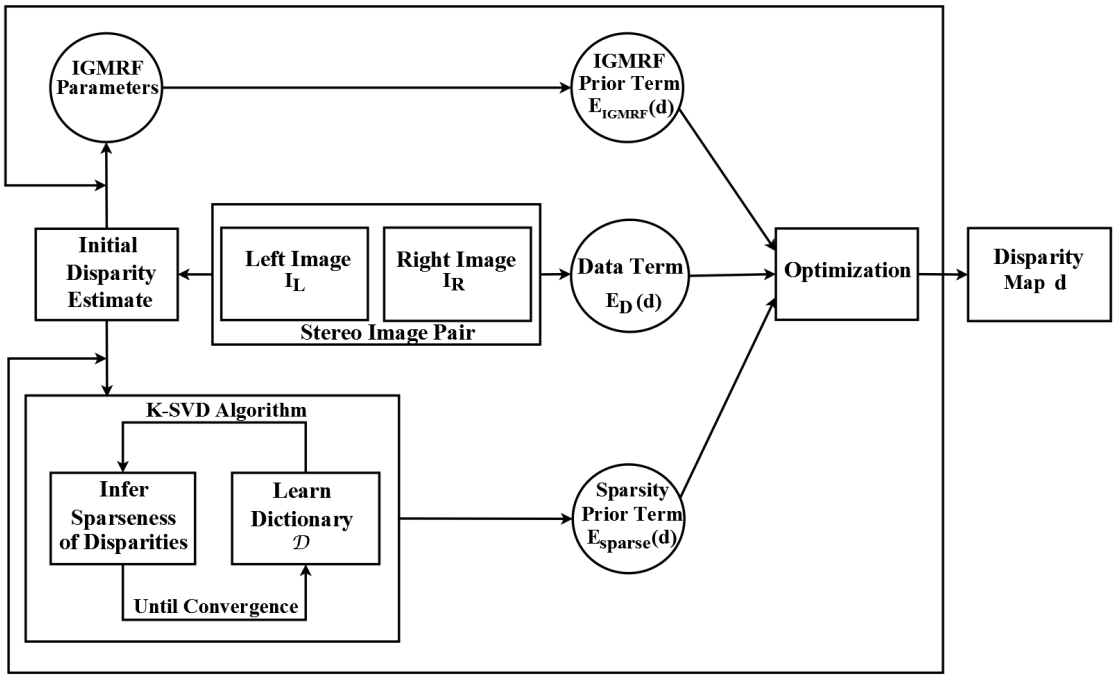

Figure 4.1: Block schematic of the proposed approach for dense disparity estimation. Here, the sparseness is learned using the overcomplete dictionary, and the algorithm starts with the use of an initial estimate and iterates until convergence.

4.3.3 Defining Sparsity Prior $E_{sparse}(d)$

The learned dictionary and the sparse representations of disparity patches are used in arriving at sparsity prior term $E_{sparse}(d)$. The sparsity prior is defined using the disparity patches extracted at each location (x, y) in d where each patch is encoded via a sparse representation using the learned overcomplete disparity dictionary. For a learned $\mathcal{D}$ and $a^{(x,y)}$ obtained at every pixel (x, y), $E_{sparse}(d)$ is then given by,

$$E_{sparse}(d) = \sum_{(x,y)} \left|\left| d^{(x,y)} - \mathcal{D}a^{(x,y)} \right|\right|_2^2. \tag{4.16}$$

$E_{sparse}(d)$ measures how well each disparity patch at location (x, y) in d agrees with its sparse representations. It represents the reconstruction error over all disparity patches in d.

4.3.4 Dense Disparity Estimation

We now estimate the dense disparity map based on the formulation discussed in section 4.2, and the block schematic for the same is shown in Figure 4.1. In the chapter 3, we proposed a learning based approach for obtaining an initial estimate of the disparity map. An inherent limitation of this learning algorithm is that it is computationally slow and the quality of the solution depends on the raw disparity estimates obtained using a multiple baseline approach given in [99]. The accuracy of IGMRF parameters depends on the quality of the initial disparity estimates. Our results in chapter 3 showed that the initial estimate was noisy in homogeneous regions and had bad match disparities, which in turn affected the quality of the final results. In this chapter, we use a different approach for obtaining an initial estimate in order to obtain a better disparity map. A classical local stereo method [114] is used in which the absolute differences of corresponding pixel intensities (AD) with truncation, aggregated over a fixed window is employed as a matching cost. In order to reduce computation time, we optimize this cost by graph cuts instead of the classic "winner take all" (WTA) optimization. Post-processing operations such as left-right consistency check, interpolation, and median filtering [114] are applied in order to obtain a better initial estimate that gives faster convergence while regularizing.

Using Eqs. (4.6), (3.22) and (4.16) defined for $E_D(d)$, $E_{IGMRF}(d)$ and $E_{sparse}(d)$ terms, respectively our final energy function is given by,

$$E(d) = \sum_{(x,y)} \left(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau)\}\right) + \sum_{(x,y)} b^{X}_{(x,y)}(d(x-1,y)-d(x,y))^2 + \sum_{(x,y)} b^{Y}_{(x,y)}(d(x,y-1)-d(x,y))^2 + \gamma \sum_{(x,y)} \left|\left| d^{(x,y)} - \mathcal{D}a^{(x,y)} \right|\right|_2^2. \quad (4.17)$$

In order to minimize this, we start with the initial estimate of disparity map, and iterate and alternate between the following two phases until convergence:

Phase 1: With d being fixed, learn the dictionary $\mathcal{D}$ and obtain sparse vectors $a^{(x,y)}$ at every pixel (x, y) by solving the optimization problem given in Eq. (4.10) using K-SVD algorithm. With the current d, compute the IGMRF parameters $b^X_{(x,y)}$ and $b^Y_{(x,y)}$ at every pixel location using Eqs. (3.31) and (3.32).

Phase 2: With $\mathcal{D}$, $a^{(x,y)}$, $b^X_{(x,y)}$, and $b^Y_{(x,y)}$ fixed as obtained in phase 1, minimize the Eq. (4.17) for d using the graph cuts method.

4.4 Learning Sparseness using Sparse Autoencoder

We now propose another method to estimate disparity in which we use sparseness learned using autoencoder instead of K-SVD. In our formulation, the sparsity prior is defined using the spatially varying patterns i.e., the disparity patches where each patch is encoded via an overcomplete sparse representation using the learned weights of a sparse autoencoder. We first train our sparse autoencoder using a large set of ground truth disparity patches and then infer the sparseness of disparities using its learned weights. We demonstrate that the sparse representation of disparities using sparse autoencoder is more effective than the use of overcomplete dictionaries learned using K-SVD, and results in an improved disparity map.

4.4.1 Motivation

Finding efficient sparse representation of natural images and depth maps using the learned overcomplete dictionaries have been extensively studied since last decade, and have shown excellent performance as priors in regularizing the ill-posed problems [3, 30, 85, 123]. However, a practical problem with sparse coding and dictionary learning techniques, for example, K-SVD algorithm is that they are computationally expensive. This is because the dictionaries are learned by iteratively recovering sparse vectors and updating one column of the dictionary at a time [3]. Though, dictionary learning methods perform well in practice, they use a linear system. Recent research suggests that non-linear, neural networks can

achieve superior performance while learning an efficient representation of images [11]. Examples of such networks are, sparse autoencoder [103, 78], belief network [44, 12], convolutional neural network [53]. These networks learn the hierarchical features of images at hidden layers including the sparse representation of images. Among these methods, we are motivated to use sparse autoencoder for learning the efficient overcomplete sparse representation of disparities. A sparse autoencoder performs well in solving vision problems since it has a structure similar to human visual cortex [77, 78, 89]. It represents an unsupervised feature learning algorithm and has been successfully used in regularizing the solution, for example, in denoising [128] and inpainting [134] giving superior performance when compared to the use of K-SVD based dictionary learning. Sparse autoencoders can be easily generalized to represent complex models. In our work, we consider a simple structure of sparse autoencoder comprising of a single hidden layer.

In our previous approach that was based on K-SVD, the learning of dictionary was carried out at every iteration based on the current estimate of the disparity map, and it added to the computational complexity of the algorithm. In this approach, we train our sparse autoencoder using a set of ground truth disparity patches. Since the autoencoder is trained in an offline way, it does not add to the computational complexity.

4.4.2 Sparse Model for Disparity

An autoencoder is an artificial feed forward neural network which sets the desired output same as the input and has one hidden layer. It comprises of an encoder that maps an input vector to a hidden representation and a decoder that maps this hidden representation back to a reconstructed input.

Consider a lexicographically ordered disparity patch $d^{(x,y)} \in \mathbb{R}^n$ at a pixel location (x, y) in disparity map d. Let the input to an autoencoder be a disparity patch $d^{(x,y)}$, its corresponding representation at hidden layer be $a^{(x,y)} \in \mathbb{R}^K$, and the reconstructed output be $\tilde{d}^{(x,y)} \in \mathbb{R}^n$. The input and the hidden layers have one extra unit referred as bias unit. With this, the input, hidden and output layers have $n+1$ input, $K+1$ hidden, and n output units, respectively. The autoencoder has

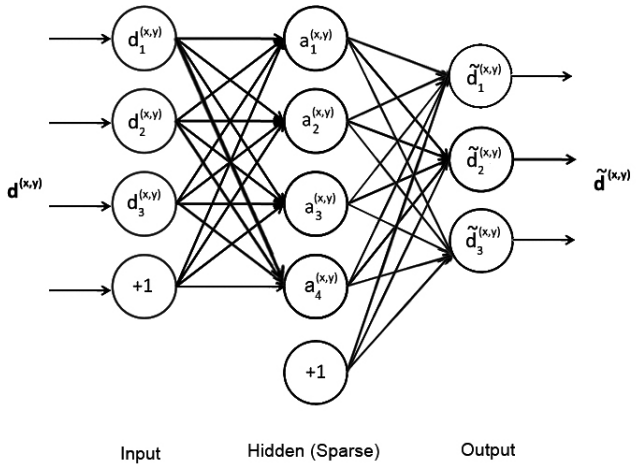

Figure 4.2: An example of an autoencoder with $n = 3$ and $K = 4$. Here, +1 represents a bias unit.

the weights (W, U, r, s) where $W \in \mathbb{R}^{n \times K}$ represents the encoder weight matrix between the input and the hidden layer, $U \in \mathbb{R}^{K \times n}$ is the decoder weight matrix between the hidden and the output layer, and $r \in \mathbb{R}^K$ and $s \in \mathbb{R}^n$ are the bias weight vectors for hidden and output layer, respectively. An element $W(i, j)$ in W denotes the weight associated with the connection between units i and j in the input and the hidden layers, respectively. Similarly, any $U(i, j)$ in U denotes the weight associated with the connection between units i and j of hidden and output layers, respectively. The element $r(i)$ denotes the weight associated with the bias unit in the input layer and unit i in the hidden layer. Similarly, the element $s(i)$ denotes the weight associated with the bias unit in the hidden and unit i in the output layer. An example of an autoencoder is shown in Figure 4.2.

For a fixed set of weights (W, U, r, s), the $a^{(x,y)}$ and $\tilde{d}^{(x,y)}$ can be computed by forward propagation as:

$$a^{(x,y)} = f(W^T d^{(x,y)} + r), \tag{4.18}$$

$$\tilde{d}^{(x,y)} = f(U^T a^{(x,y)} + s), \tag{4.19}$$

where f is an activation function and it is applied element wise on its input as vector. We choose f to be a "sigmoid" function. For any value $z \in \mathbb{R}$, it is defined

as,

$$f(z) = \frac{1}{1 + e^{-z}}.$$

The activation value $f(.)$ ranges between 0 and 1, increasing monotonically. As it maps a very large domain of inputs to a small range of outputs, it is often referred as a "squashing function". The advantage of choosing sigmoid is that it is a non-linear and a differentiable function.

The autoencoder learns an approximation to the identity function. Learning an identity function is a trivial task but by placing constraints on the hidden layer, one can discover interesting structure about the input data. An autoencoder is called as sparse autoencoder when the sparsity constraint is imposed on its hidden layer. It learns an overcomplete sparse representation of input when the number of hidden units K is greater than the number of input units n i.e., $K > n$. Let $a_j^{(x,y)}$ be the activation value of hidden unit j. A sparsity constraint on the activation of hidden units (neurons) is imposed by forcing them to be inactive most of the time. A unit is active when its activation value is close to one and inactive when it is close to zero. Let us define ρ as a global sparsity parameter for all hidden units, typically a small value close to zero. Further, let $\hat{\rho}_j$ be the average activation of hidden unit j (averaged over a training set of disparity patches). Then the sparsity constraint for each j^{th} hidden unit is enforced by a penalty term which penalizes $\hat{\rho}_j$ deviating significantly from ρ as:

$$\sum_{j=1}^{K} KL(\rho||\hat{\rho}_j) = \sum_{j=1}^{K} \rho \log \frac{\rho}{\hat{\rho}_j} + (1-\rho) \log \frac{1-\rho}{1-\hat{\rho}_j}, \tag{4.20}$$

where $KL(\rho||\hat{\rho}_j)$ is the Kullback Leilbler (KL) divergence between two Bernaulli random variables with mean ρ and mean $\hat{\rho}_j$, respectively. KL-divergence is a standard function for measuring how different two distributions are. $KL(\rho||\hat{\rho}_j) = 0$ if $\hat{\rho}_j = \rho$, and otherwise it increases monotonically as $\hat{\rho}_j$ diverges from ρ.

4.4.3 Training the Sparse Autoencoder

Consider a training set $\mathcal{G}_d$=$\{d^{(1)}, d^{(2)}, \ldots, d^{(n_g)}\}$ consisting of large number of disparity patches with each patch $d^{(i)} \in \mathbb{R}^n$. One can extract these disparity patches from the available ground truth disparity maps, and let the number of training patches are n_g. Using the training set $\mathcal{G}_d$, we can train the sparse autoencoder to learn the weights (W, U, r, s). To do this, first the following objective function is formed:

$$\begin{aligned}\frac{1}{n_g}\sum_{i=1}^{n_g}\Big(\frac{1}{2}||d^{(i)} - f(U^T(f(W^T d^{(i)} + r)) + s)||_2^2 \\ + \frac{\eta}{2}\Big(\sum_{i=1}^{n}\sum_{j=1}^{K}(W_{ij})^2 + \sum_{i=1}^{K}\sum_{j=1}^{n}(U_{ij})^2\Big) \\ + \beta\sum_{j=1}^{K} KL(\rho||\hat{\rho}_j). \end{aligned} \tag{4.21}$$

Here, the first term represents the average reconstruction error over all training inputs and is formed using Eqs. (4.18) and (4.19). The second term is a regularization term on the weights to prevent the overfitting by making them smaller in magnitude, and η controls the relative importance of this term. β controls the weightage of the third term which corresponds to sparsity penalty term given by Eq. (4.20). Note that the $\hat{\rho}_j$ in sparsity term is a function of W and r because it is the average activation of hidden unit j, and the activation of a hidden unit depends on the weights W and r i.e.,

$$\hat{\rho}_j = \frac{1}{n_g}\sum_{i=1}^{n_g}\big[a_j^{(i)}(d^{(i)})\big],$$

where $a_j^{(i)}(d^{(i)})$ denotes the activation of a hidden unit j when the sparse autoencoder is given i^{th} input i.e., $d^{(i)}$. In order to train the sparse autoencoder, the objective function defined in Eq. (4.21) is minimized w.r.t. (W, U, r, s) using well known back propagation algorithm [93].

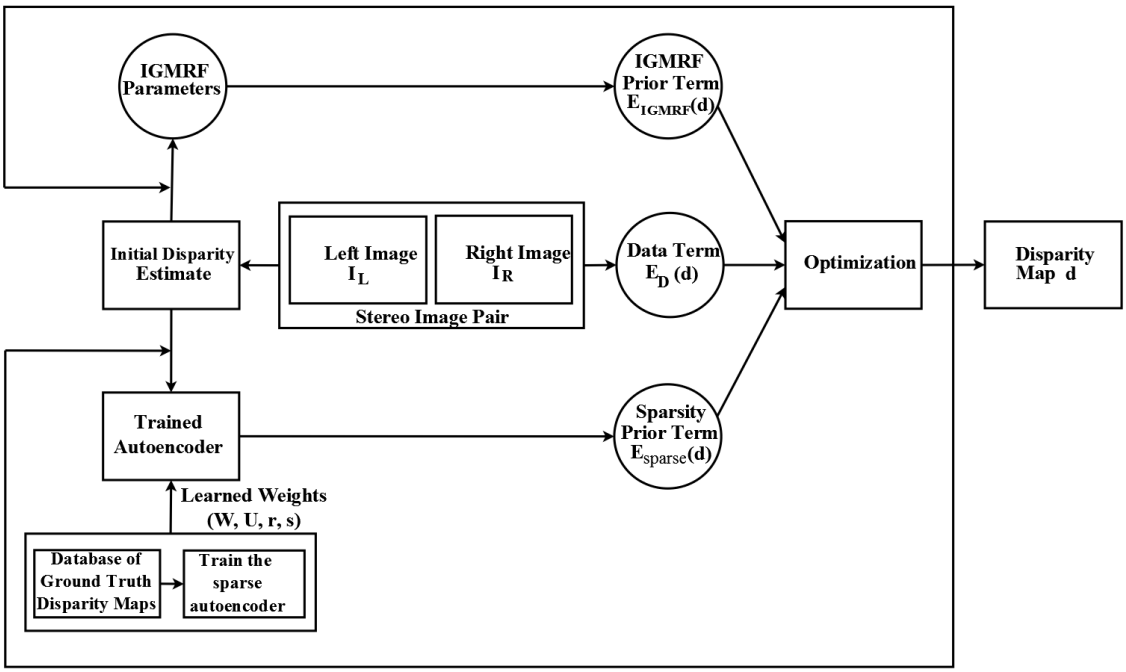

Figure 4.3: Block schematic of the proposed approach for dense disparity estimation. Here, the sparseness is learned using the sparse autoencoder, and the algorithm starts with the use of an initial estimate and iterates until convergence. Note that except *K-SVD block* all other blocks are the same as in Figure 4.1.

4.4.4 Defining Sparsity Prior $E_{sparse}(d)$

The sparsity prior is defined using the disparity patches extracted at each location (x, y) in d where each patch is encoded via a sparse representation inferred using the learned weights of autoencoder. Once the sparse autoencoder is trained, d can be modeled by the sparsity prior $E_{sparse}(d)$ as follows:

$$E_{sparse}(d) = \sum_{(x,y)} \left|\left| d^{(x,y)} - f(U^T a^{(x,y)} + s) \right|\right|_2^2. \tag{4.22}$$

For the input disparity patch $d^{(x,y)}$, its corresponding sparse representation $a^{(x,y)}$ is inferred from the trained sparse autoencoder using the forward propagation given in Eq. (4.18).

4.4.5 Dense Disparity Estimation

We now estimate the dense disparity map based on the formulation discussed in section 4.2. The block diagram of the proposed method is shown in Figure 4.3.

Using Eqs. (4.6), (3.22) and (4.22) defined for $E_D(d)$, $E_{IGMRF}(d)$ and $E_{sparse}(d)$ terms, respectively our final energy function is given by,

$$E(d) = \sum_{(x,y)} \left(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau)\}\right) + \sum_{(x,y)} b^{X}_{(x,y)}(d(x-1,y)-d(x,y))^2 + \sum_{(x,y)} b^{Y}_{(x,y)}(d(x,y-1)-d(x,y))^2 + \gamma \sum_{(x,y)} \left|\left| d^{(x,y)} - f(U^T a^{(x,y)} + s)\right|\right|_2^2. \quad (4.23)$$

Our algorithm proceeds with the use of an initial estimate of disparity map, and iterates and alternates between two phases until convergence. We use the same method of obtaining initial estimate of disparity map as discussed in section 4.3.4. The proposed algorithm can be described in following steps:

1. **Input:** Stereo image pair I_L and I_R, and a set of ground truth disparity patches $\mathcal{G}_d$=$\{d^{(1)}, d^{(2)}, \ldots, d^{(n_g)}\}$.
2. **Sparse autoencoder training:** Train the sparse autoencoder using $\mathcal{G}_d$ by minimizing Eq. (4.21), and obtain weights (W, U, r, s).
3. **Initialization:** Obtain an initial disparity map d_0 and initialize $d = d_0$.
4. Repeat until convergence,
   (a) **Phase 1**: With d being fixed, infer the sparse vector $a^{(x,y)}$ for each disparity patch $d^{(x,y)}$ in d using trained sparse autoencoder (Eq. (4.18)). Compute IGMRF parameters $b^{X}_{(x,y)}$ and $b^{Y}_{(x,y)}$ at each pixel location using Eqs. (3.31) and (3.32).
   (b) **Phase 2**: With $a^{(x,y)}$, $b^{X}_{(x,y)}$, and $b^{Y}_{(x,y)}$ fixed as obtained in phase 1, minimize the Eq. (4.23) for d using graph cuts [69].

4.5 Experimental Results

In this section, we demonstrate the efficacy of our proposed approaches. In order to estimate the dense disparity maps, we conducted various experiments and evaluated our results on the Middlebury stereo datasets [113].

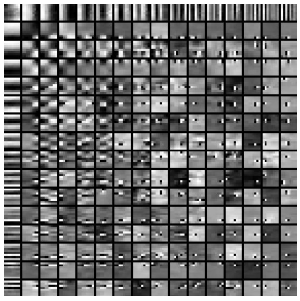

Figure 4.4: Learned overcomplete dictionary for "Cones" image. Here, each column of dictionary has a size of 64×1, and is displayed by a 8×8 block in the figure.

4.5.1 Parameter Settings

We first give the parameters used for learning our overcomplete dictionary using K-SVD algorithm. Our dictionary was trained using the overlapping disparity patches of the estimated disparity map (for example, when testing for "Cones" stereo image of size 375×450, we use $n_d = 163024$ disparity patches). The size of the dictionary $\mathcal{D}$ was chosen as 64×256 with the size of a patch and dimension of a sparse vector as $8 \times 8 = 64$ and 256, respectively. The K-SVD algorithm used orthogonal matching pursuit (OMP) algorithm for sparse coding with $t = 16$ and converged within approximately 30 iterations. As an example, the learned overcomplete disparity dictionary for "Cones" stereo image is shown in Figure 4.4. One can observe that the dictionary learns Gabor like filters and hence can capture edge like features of disparity patches.

In our second proposed approach, a sparse autoencoder is used for modeling the sparseness in disparity map. We trained the sparse autoencoder using a set of $n_g = 5 \times 10^5$ true disparity patches extracted from the ground truth disparity maps of Middlebury 2005 and 2006 stereo datasets [113]. Note that the training set used here is different from the one used for testing. Here, also the size of each disparity patch set to 8×8 i.e., $n = 64$ and the dimension of sparse vector to $K = 256$. The parameters in Eq. (4.21) were empirically chosen as: $\eta = 10^{-4}$, $\beta = 0.1$ and $\rho = 0.01$. With these parameter settings, the sparse autoencoder was trained to obtain the weights (W, U, r, s). The learned weights W are shown in Figure 4.5. We see that the different hidden units have learned to detect edges at different

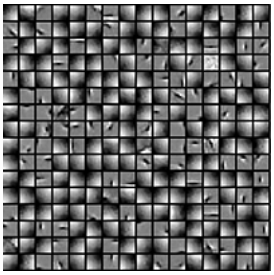

Figure 4.5: Learned weights W between the input and the hidden layer in the trained sparse autoencoder. Here, each square block is of size 8×8 which shows the weights between a hidden unit and each input unit. Note that there are 256 hidden and 64 input units.

positions and orientations in the disparity patches.

Both of our algorithms for disparity map estimation were initialized with the initial estimate and converged with in $5 - 10$ iterations for all the stereo pairs used in our experiments. The parameter γ was initially set to 10^{-4} and exponentially increased at each iteration from 10^{-4} to 10^{-1}. We used the same parameters for all the experiments and this demonstrates the robustness of our methods. Note that in our first approach, the dictionary was learned using the estimated disparity map. Since the dictionary was refined at each iteration based on the current estimate of the disparity map, it added to the computational complexity. In the second approach, the sparse autoencoder was trained from the set of ground truths. Since it was an offline operation, it did not add to the computational complexity of the algorithms. The average run time of our first approach based on dictionary learning was 300 seconds, and was 130 seconds for the second one.

4.5.2 Quantitative Comparison

In order to perform the quantitative evaluation, we used the percentage of bad matching pixels ($\mathcal{B}\%$) as the error measure with a disparity error tolerance δ. The performance of the proposed approaches was tested under different scenarios. We first estimated the disparity map using only the IGMRF prior. We then combined IGMRF and sparsity priors but the sparseness of disparity was obtained using fixed DCT dictionary. Finally, we tested the two methods in which the sparseness was represented by the learned overcomplete dictionary and by trained sparse autoencoder, respectively. In these four experiments, we used the initial estimate

Method	Venus	Teddy	Cones
Initial estimate	3.47	19.65	16.43
IGMRF based prior	1.78	18.1	14.42
IGMRF+DCT based prior	1.61	16.9	13.1
IGMRF+KSVD based prior **(Proposed)**	**1.43**	**12.5**	**11.42**
IGMRF+autoencoder based prior **(Proposed)**	**0.22**	**10.7**	**9.64**

Table 4.1: Evaluation results on the Middlebury datasets [113] in terms of % of bad matching pixels computed over the entire image with δ=1. Comparisons are made among different cases: (1^{st} row): Initial Estimate. (2^{nd} row): Using IGMRF prior only. (3^{rd} row): Using IGMRF and sparsity prior with DCT dictionary. (4^{th} row): Using IGMRF and sparsity prior learned using overcomplete dictionary via K-SVD (Proposed). (5^{th} row): Using IGMRF and sparsity prior learned using sparse autoencoder (Proposed).

which was obtained from the method discussed in section 4.3.4. The Table 4.1 summarizes the disparity estimation results under these experimented cases.

The results in Table 4.1 show that the performance using the proposed methods are better when compared to other experimented cases. We can see that incorporation of sparsity prior in addition to IGMRF prior significantly improves the performance or in other words, using the combination of sparsity with IGMRF prior always performs better than the use of IGMRF prior alone. This is expected because IGMRF and sparsity priors together capture the disparity characteristics in different ways, and their combination serves as a better regularizer. It can be clearly observed that the use of learned sparseness gives better results than the use of fixed DCT dictionary. This shows the effectiveness of the learned sparseness over the sparseness represented by fixed bases. Finally, the results in Table 4.1 indicate that method based on sparse autoencoder gives superior performance when compared to that using the K-SVD based learning. This is because the sparseness is better captured by the learned weights of autoencoder. Overall, the results show the effectiveness of the learned sparseness using the overcomplete dictionary and sparse autoencoder with the combination of IGMRF for accurate disparity estimation.

We also tested the performance of our first method where we learned the dictionary using the set of ground truth disparity patches rather than using the estimated disparities from a given stereo pair. We observed that the results were

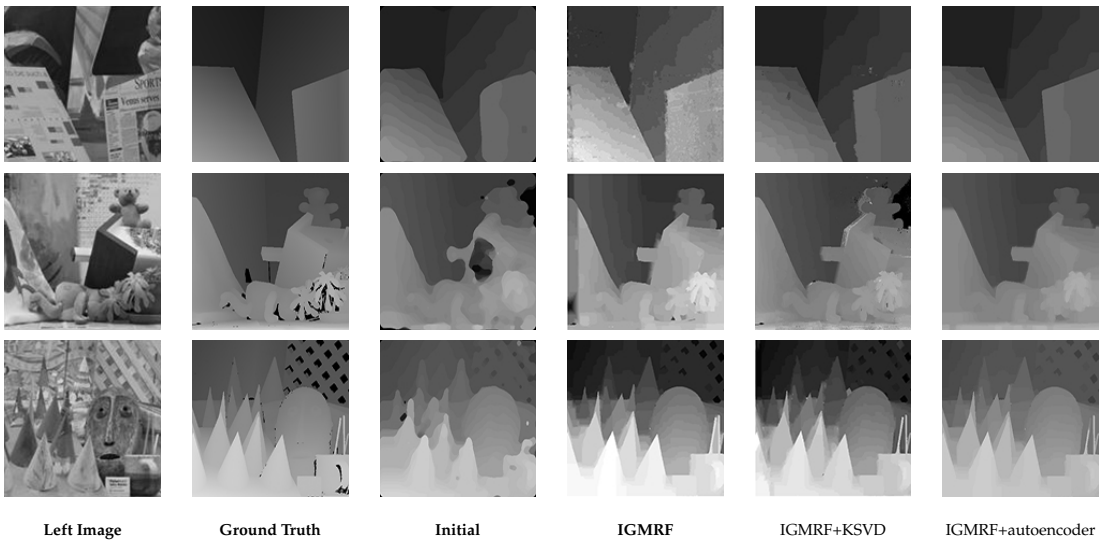

Figure 4.6: Disparity maps estimated for the datasets of [113], "Venus" (1^{st} row), "Teddy" (2^{nd} row) and "Cones" (3^{rd} row). (1^{st} column): Left Image. (2^{nd} column): Ground Truth. Results for different experimented cases, (3^{rd} column): Initial Estimate, (4^{th} column): Using IGMRF prior only, (5^{th} column): Using IGMRF and sparsity prior learned using overcomplete dictionary via K-SVD (Proposed), and (6^{th} column): Using IGMRF and sparsity prior learned using sparse autoencoder (Proposed).

not better and hence we considered the dictionary learned using the given stereo pair. Contrary to this, in our second method, we used a set of ground truths for training the sparse autoencoder because the sparse autoencoder works efficiently for a large amount of training data.

Note that we do not include the results where the sparsity prior $E_{sparse}(d)$ is used alone as a prior term in our energy function. We observed through our experiments that if $E_P(d)$ is formed using $E_{sparse}(d)$ only then the solution converges towards the initial estimate and no improvement is found. Hence, in order to obtain better solution, we combine the $E_{sparse}(d)$ with $E_{IGMRF}(d)$ to form our prior term.

4.5.3 Qualitative Analysis

Figure 4.6 shows the disparity maps obtained using different experimented cases. One can see that the final disparity maps estimated by both the proposed methods are piecewise smooth and visually plausible and show improvement in the quality over the initial estimate and the disparity maps computed using IGMRF prior

Method	Venus	Teddy	Cones
GraphCuts [21]	3.44	25.0	18.0
MultiGC [68]	3.13	17.6	11.8
SecondOrderMRF [133]	0.49	15.4	10.8
BeliefProp [143]	0.45	8.30	8.78
Mumford [10]	0.76	14.3	9.91
LearnedSparse [123]	-	8.14	11.98
GroundPoints [131]	0.53	11.5	9.49
LearnedCRF [111]	1.3	11.1	10.8
CompressSensing [42]	0.68	13.30	9.79
TwostepGlobal [94]	1.49	9.40	7.66
ConsistencyPrior [59]	0.61	12.4	9.35
IGMRF+KSVD based prior (Proposed)	1.43	12.5	11.42
IGMRF+autoencoder based prior (Proposed)	0.22	10.7	9.64

Table 4.2: Comparison with the state of the art global dense stereo methods evaluated on the Middlebury stereo 2001 and 2003 datasets [113] in terms of % of bad matching pixels computed over the entire image with δ=1.

only. Results show the effectiveness of learned overcomplete dictionary as well as learned sparse autoencoder used for capturing the disparity characteristics in a larger neighborhood. We can clearly observe that the proposed method based on sparse autoencoder results in better disparity estimates in homogeneous/textureless regions as well as in textured regions while preserving the sharp discontinuities at object boundaries when compared to the method based on K-SVD dictionary learning.

4.5.4 Comparison with the State of the Art Methods

In order to validate the results of both the proposed methods, we compare them with the state of the art global dense disparity estimation methods in terms of percentage of bad matching pixels ($\mathcal{B}$%) as shown in Table 4.2. We see that our methods perform significantly better when compared to the state of the art methods based on edge preserving smoothness priors [21, 68, 111]. Our method based on sparse autoencoder gives superior performance when compared to that using second order smoothness prior [133]. This shows the effectiveness of the learned sparsity prior which captures the disparity characteristics in a larger neighborhood without the need of computationally expensive higher order cliques used in

Method	Adiron	Jadep	Motor	PianoL	Pipes	Playrm	Playt	Recyc	Shelvs	Vintage
AdaptSmooth [65]	14.3	24.5	9.82	31.3	9.80	28.9	22.4	17.8	45.4	31.8
MeshStereo [152]	20.6	35.3	20.6	37.9	23.4	39.5	34.1	25.9	53.0	35.6
HiddenMarkov [104]	26.1	33.6	24.4	44.0	19.3	41.1	50.4	27.5	59.5	51.4
MultiDisparity [79]	25.0	35.9	30.4	41.7	29.0	42.7	47.8	31.3	54.5	43.6
PlaneSweep [115]	29.9	34.7	12.3	59.6	15.8	41.4	33.4	33.6	51.5	45.8
TwostepGlobal [94]	55.9	73.3	52.3	74.7	50.9	72.3	66.2	52.4	71.0	76.6
IGMRFAutoencoder (Proposed)	62.4	67.5	60.2	69.1	34.9	71.2	79.9	65.9	79.9	60.1

Table 4.3: Quantitative evaluation on Middlebury stereo 2014 datasets [113], and comparison with current better performing global dense stereo methods. Evaluation is in terms of % of bad matching pixels in non-occluded regions with δ=1.

Markov random fields (MRF). It can be clearly seen from Table 4.2 that our method based on sparse autoencoder works well for all the three datasets and gives the least bad matching error for the "Venus" stereo pair. These results reflect the effectiveness of using IGMRF and learned sparseness for disparity estimation.

Before we end this section on experimental results, we discuss the experiments on the recently released Middlebury stereo 2014 datasets. As seen from Table 4.2, our method based on sparse autoencoder performs superior than the method that uses K-SVD. Hence, here we compare the performance of our sparse autoencoder based method with few of the latest global dense stereo methods discussed in chapter 2. Table 4.3 shows the performance evaluation and the comparisons in terms of $\mathcal{B}$% using few of these datasets. Results indicate that the performance of our method is comparable to the latest global stereo methods. We see that our approach is not the best because the accuracy of our method is sensitive to the parameters of the model, and one can enhance the results by carefully choosing the parameters.

4.6 Conclusion

In this chapter, we have proposed the use of IGMRF and overcomplete sparsity priors for dense disparity estimation. The IGMRF prior captures the spatial variation among disparities locally as well as it preserves sharp discontinuities while the sparsity prior accounts for redundancy in disparities, and captures the sparse-

ness in the disparity map. The combination of these two priors better constrains the solution. In our first method, the sparse representation of disparities is obtained by using an overcomplete dictionary which is learned using the K-SVD algorithm. Though, this kind of learning is adaptive and do not require a large set of ground truths while training, it is a linear model. Our second approach based on a sparse autoencoder is non-linear and better captures the sparseness. Both the approaches are iterative and use alternate minimization until convergence. Our method based on K-SVD is computationally expensive since the dictionary is also refined in the two-phase algorithm. We have shown that the combination of sparsity and IGMRF priors always perform better than the use of IGMRF prior alone. We have also demonstrated the effectiveness of learning the sparseness instead of using fixed basis. Our methods show the superior performance over the state of the art and comparable preformance with the latest methods global stereo methods.

In the next chapter, we propose a different energy minimization framework for disparity estimation where we introduce feature matching as an additional matching cost in the data term.

CHAPTER 5

Use of Hierarchical Feature Matching

We solve the dense disparity estimation problem in an energy minimization framework. However, solutions with lower energy do not always correspond to better performance [120]. It is important to define a proper energy function in order to obtain a better solution. To arrive at the energy function, a set of constraints are incorporated that assigns minimum energy to the solution satisfying these constraints. For example, an energy function which is a combination of data and prior terms is commonly used for disparity estimation. In chapter 3, we used an energy function with IGMRF prior, and in chapter 4, a combination of IGMRF and sparsity priors was incorporated in the energy function. In these chapters, we focused on arriving at a proper energy function by using suitable priors. The choice of appropriate data constraint also plays an important role. Hence in this chapter, we propose to use a new data term keeping the prior as IGMRF.

5.1 Motivation and Related Work

In an energy function, data term measures how well the disparity map to be estimated agrees with the observation i.e., left and right images of a scene. The data term is generally defined by using the pixel based matching cost between the intensities of corresponding pixels in the left and right images. It is built on the brightness constancy assumption. In chapter 3, we used the data term as a sum of squared intensity differences at corresponding pixels. Due to the effects of image sampling, noise, depth discontinuities, occlusion, view-point and illumination variation, etc., corresponding pixels may have different intensities, and

hence various robust matching costs are used to reduce such effects [114, 110, 47, 21, 117, 143, 68]. In chapter 4, we considered a data term that is insensitive to image sampling using the technique proposed in [13]. In addition, robust window based matching costs have been commonly used in stereo methods. A pixel based matching cost can be extended to window based matching cost by integrating the pixel based costs within a certain neighborhood, and such costs include: normalized cross correlation [114], rank and census transform [92], bilateral filtering [94], etc. However, these costs assume that every pixel in a patch has the same disparity and such an assumption does not hold in practice. To overcome this limitation, Jung *et al.* [58] propose to use a data term based on higher order likelihood model of stereo images. This method results in robust estimates around disparity discontinuities but it is computationally very expensive.

Although, the conventional pixel or window based matching costs are robust against the outliers, occlusion, discontinuities, view-point variation, etc., they rely only on the raw pixel values (intensities). Therefore, the use of data term defined using intensity matching results in ambiguous estimates. One can represent stereo images in a better way by using a feature space where they are robust, distinct and view-point invariant [39, 6, 56]. The basic features, for example, edges, gradients, corners, segments, and learned features are commonly used for matching the stereo images. Since these features are extracted at few locations, the methods based on feature matching yield sparse disparity maps [56, 23, 132, 25, 102]. The dense map can be obtained by simply interpolating the sparse disparity map. However, interpolation is not a good choice because it results in inaccurate disparities. In our work, we use feature matching to obtain the dense disparity map by using it in the data term in a global framework.

Hong et al. [48] and Klaus et al. [66] use non-overlapping segments of the stereo images as features and cast the dense stereo matching problem as an energy minimization in segment domain instead of pixel domain where the disparity plane is assigned to each segment via graph cuts or belief propagation optimization technique. Their approaches assume that the disparities in a segment vary smoothly which is not true in practice due to the depth discontinuities. Also,

the solution here relies on the accuracy of segmentation which is a non-trivial task. Hirschmuller [46] uses the mutual information (MI) based feature matching in an MRF framework for estimating the dense disparities. Wang and Zhang [76] obtain dense correspondences from sparse matches using the propagation and seed growing methods. The accuracy in such approaches depends on the initial sparse disparity estimates. However, matching based on basic features still results in ambiguities, especially in textureless areas. Hence to reduce the ambiguities, one needs to use more descriptive features. Hand crafted features of stereo images are designed and then embedded in an MRF model for disparity estimation in [109]. Recently, Liu *et al.* [82] propose a scale invariant feature transform (SIFT) flow algorithm for finding the dense correspondences by matching the pixel-wise SIFT descriptors while preserving spatial discontinuities using MRF regularization. Similarly, a deformable spatial pyramid model is proposed in a regularization framework using multiple SIFT features by Kim *et al.* [64]. The drawback of these approaches is that designing such features is computationally expensive, time consuming, and requires domain knowledge of the data.

Recently, there has been a considerable progress in feature learning using machine learning methods in order to estimate disparity [111, 147]. Feature learning is attractive as it exploits the availability of large amount of data and avoids the need of feature engineering. Zhang and Shen [153] propose unsupervised feature learning for dense stereo matching. They learn the features from a large number of image patches using K-SVD dictionary learning approach. The drawback here is that the features are learned using image patches and not the entire image i.e., global contextual constraint is not considered while learning the features. Also, higher level features are not learned, instead they are estimated using a simple max pooling operation from the layer beneath. Here, the higher layer correspondence matches are used to initialize the matching at lower layer and hence the accuracy depends on the higher layer matches only. Currently, deep learning approaches have been developed [11] and achieved excellent performance in solving many computer vision problems such as image classification [74], image restoration [128], super-resolution [29], object detection [37], semantic segmen-

tation [97], visual tracking [49], action recognition [55], etc. Deep learning is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using deep architectures composed of multiple linear/non-linear transformations [11]. Deep learning is more than a traditional supervised learning where label information is readily available in training. That is, rather than focusing on feature engineering which is often labor-intensive and varies from one task to another, deep learning methods are focused on end-to-end learning based on raw data. To accomplish end-to-end optimization starting with raw features and ending in labels, layered structures are used for example, neural networks. At each layer different intermediate representation (abstraction) of raw input are learned. Many deep learning algorithms are framed as unsupervised learning problems. Because of this, these algorithms can make use of the unlabeled data that supervised algorithms cannot. Unlabeled data is abundant, making this an important benefit of these algorithms. Hence using deep learning, one can obtain hierarchical features of input images from the unlabeled data. Few such approaches include: deep autoencoders [78], deep belief networks [44, 12], deep convolutional network [60] and deep deconvolutional network [150, 151], etc.

Deep learning field has also attracted the attention of stereo vision researchers in recent years. Zbontar and Lecun [149] use the deep convolutional neural network for learning similarity measure on small patches of left and right images. They train the network in a supervised manner by constructing a binary classification dataset with examples of similar and dissimilar pair of patches. Based on the learned similarity measure, the disparity map is estimated using the traditional stereo methods, for example, cross based aggregation [154] and semi global matching [46] methods. Here, the learning is done on small size patches instead of entire image i.e., global contextual constraint is not taken into account while learning the similarity measure. The method does not provide a single framework for dense disparity estimation though it improves the results of state of the art stereo methods. In [91], a convolutional neural network is used to directly predict the disparity map from a pair of stereo images. Given a large dataset

consisting of stereo image pairs as input and the corresponding ground truths as output, the network is trained end to end for learning the disparities for the input stereo image pair. In their architecture, features of left and right images are learned separately at lower level and then at higher level these feature maps are matched or compared using the correlation layer. The layers on top of the correlation layer learn predicting disparities from matches. In other words, their method learns the image feature representations and also learn to match them at different locations in the two images.

In this chapter, we propose yet another approach for dense disparity estimation in a global energy minimization framework. Inspired by the recent development in deep learning methods, we propose to use a feature matching cost which is defined using the learned hierarchical features of given left and right stereo images. The hierarchical features are learned using the deep deconvolutional network [150] which is trained in an unsupervised way using a database consisting of large number of stereo images. Since the intensity values are available at every pixel location, we combine our learned feature matching cost with the pixel based intensity matching cost and estimate the dense disparity map. The combination of these two matching costs form the data term in our energy function, better constrains the solution. As a regularization prior, we use IGMRF in our energy function that captures the smoothness as well as preserves the sharp discontinuities in the disparity map. As in the previous chapter, an iterative two phase algorithm is proposed to estimate the dense disparity map. IGMRF prior parameters are computed in phase one keeping the disparity map fixed, and in phase two the disparity map is refined by minimizing the energy function using graph cuts by fixing IGMRF parameters. We demonstrate the effectiveness of the use of learned feature matching on the disparity estimation by conducting experiments on the standard datasets.

5.2 Problem Formulation

As done in the earlier chapters, the disparity map d is obtained by minimizing the energy function:

$$E(d) = E_D(d) + E_P(d). \tag{5.1}$$

In this work, the data term $E_D(d)$ consists of both the intensity matching $E_I(d)$ and the feature matching costs $E_F(d)$ i.e.,

$$E_D(d) = E_I(d) + \mu E_F(d), \tag{5.2}$$

where μ controls the weight of $E_F(d)$. For a given d, the terms $E_I(d)$ and $E_F(d)$ measure the dissimilarity among corresponding pixel intensities and corresponding features in I_L and I_R, respectively. Using the Eqs. (4.2), (4.3) and (4.6), $E_I(d)$ can be expressed as,

$$E_I(d) = \sum_{(x,y)} \big(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y), \tau^I)\}\big). \tag{5.3}$$

where τ^I is the truncation threshold which is used to make $E_I(d)$ robust against outliers and occlusion. For defining the feature matching cost $E_F(d)$, we extract the features of stereo image pair at multiple layers of deep deconvolutional network, and the same is discussed in next section. We consider the search from left to right as well as from right to left for finding the disparities, as done in chapter 4. In order to perform the regularization, we use the same prior as used in chapter 3 i.e., we model d as IGMRF and form $E_P(d)$. The final energy function is minimized using graph cuts optimization. An iterative optimization is carried out and at every iteration the IGMRF parameters are refined in order to obtain a better estimate of the disparity map.

5.3 Deep Deconvolutional Network for Extracting Hierarchical Features

Deconvolutional network [150] is an unsupervised feature learning model that is based on the convolutional decomposition of images under sparsity constraint and generates sparse, overcomplete features. Stacking such network in a hierarchy results in a deep deconvolutional network. In our approach, we train a deep deconvolutional network using a large set of stereo images, and learn both the filters and the features as done in an image deconvolution problem. The learned filters capture the image information at different layers in the form of low-level edges, mid-level edge junctions, and high-level object parts. The deep deconvolutional network is quite different from the deep convolutional neural networks (CNN). The Deep CNN [60] is a bottom-up approach where an input image is subjected to multiple layers of convolutions, non-linearities, and subsampling whereas deep deconvolutional network is a top down approach. Here, an input image is generated by a sum over convolutions of the feature maps with learned filters. Unlike deep CNN, the deep deconvolutional network does not spatially pool features at successive layers and hence preserves the mid-level cues emerging from the data such as edge intersections, parallelism, and symmetry. Contrary to deep autoencoders [78] and deep belief networks [44], they scale well to the entire image and hence learn the features for the full input image instead of small size patches. This makes them consider global contextual constraint while learning. In chapter 4, we used K-SVD and sparse autoencoder for learning the sparseness of disparities. Both these approaches consider the disparity patches and cannot scale well to entire disparity map. Hence, here we use the deep deconvolutional network which considers the entire stereo image for feature learning.

5.3.1 Training the Deep Deconvolutional Network

We first consider a deconvolutional network having a single layer. To train this network, a training set consisting of a large number of stereo images $\mathcal{I}=\{I^1, \ldots, I^{n_s}\}$

are used where each image I^i is considered as an input to the network. Here, n_s is the number of images in the training set $\mathcal{I}$ and we consider only left images of different scenes for training the network. Note that one may use right stereo images as well. Let P_1 be the number of $2D$ feature maps in the first layer. Considering the input at layer 0, we can write each image $I^i \in \mathcal{I}$ as composed of P_0 channels $\{I^i_1, \ldots, I^i_{P_0}\}$. For example, if we consider a color image, we have P_0=3. Each channel c of input image I^i can be represented as a linear sum of P_1 feature maps s^i_p convolved with filters $f_{p,c}$ i.e.,

$$\sum_{p=1}^{P_1} s^i_p \oplus f_{p,c} = I^i_c, \tag{5.4}$$

where $\oplus$ represents the 2D convolution operator. In our work, we have used gray scale stereo images only and hence P_0=1. If I^i is an $M \times N$ stereo image and the filters are of size $H \times H$ then each feature map has a size of $(M + H - 1) \times (N + H - 1)$. We see that Eq. (5.4) represents an under-determined system since both the features and filters are unknown and hence to obtain a unique solution to both the features and filters, a regularization term is also added that encourages sparsity in the latent feature maps. This gives us an overall cost function for training a single layer deconvolutional network with input training set $\mathcal{I}$ as:

$$C_1(\mathcal{I}) = \sum_{i=1}^{n_s} \left[\frac{\alpha}{2} \sum_{c=1}^{P_0} \left|\left| \sum_{p=1}^{P_1} s^i_p \oplus f_{p,c} - I^i_c \right|\right|_2^2 + \sum_{p=1}^{P_1} \left|\left| s^i_p \right|\right|_1 \right]. \tag{5.5}$$

Here, $\left|\left| s^i_p \right|\right|_1$ is the l_1-norm on the vectorized version of s^i_p that encourages sparsity in s^i_p. The relative weighting of the reconstruction error of each I^i and the sparsity of their feature maps s^i_p is controlled by the parameter α. The network is learned by minimizing $C_1(\mathcal{I})$ with respect to s^i_p's and $f_{p,c}$'s when the input to network is $\mathcal{I}$. Note that the set of filters $f_{p,c}$ are the parameters of the network, common to all images in the training set while each image has its own set of feature maps s^i_p.

The single layer network described above can be stacked to form a deep deconvolutional network consisting of multiple layers. Let the deep network is formed by NL number of layers, $(l = 1 \ldots NL)$. This hierarchy is achieved by considering

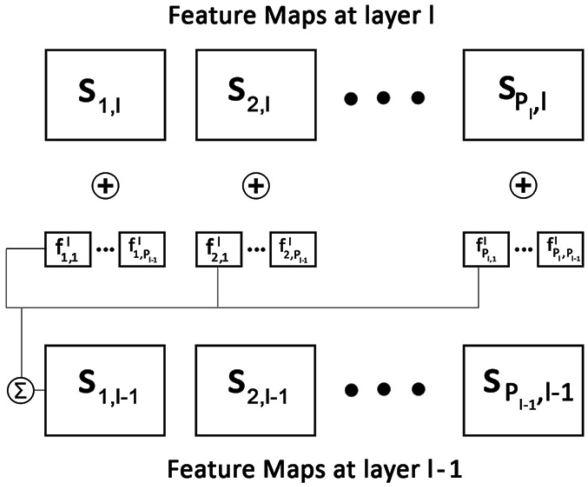

Figure 5.1: A deep deconvolutional network illustrating learning of l^{th} layer.

the feature maps of layer $l-1$ as the input to layer l, $l > 0$. In this case, the P_0 channels of an input image are considered as feature maps at layer 0. Let P_{l-1} and P_l are the number of feature maps at layer $l-1$ and l, respectively. The cost function for training the l^{th} layer of a deep deconvolutional network can be written as a generalization of Eq. (5.5) as:

$$C_l(\mathcal{I}) = \sum_{i=1}^{n_s} \left[\frac{\alpha}{2} \sum_{c=1}^{P_{l-1}} \left|\left| \sum_{p=1}^{P_l} g_{p,c}^l (s_{p,l}^i \oplus f_{p,c}^l) - s_{c,l-1}^i \right|\right|_2^2 + \sum_{p=1}^{P_l} \left|\left| s_{p,l}^i \right|\right|_1 \right], \tag{5.6}$$

where $s_{c,l-1}^i$ and $s_{p,l}^i$ are the feature maps of image I^i at layer $l-1$ and l, respectively and thus it shows that layer l has as its input coming from P_{l-1} channels. $f_{p,c}^l$ are the filters at layer l and $g_{p,c}^l$ are the elements of a fixed binary matrix that determine the connectivity between the feature maps at successive layers i.e., whether $s_{p,l}^i$ is connected to $s_{c,l-1}^i$ or not. For $l = 1$, we assume that $g_{p,c}^l$ is always 1 but for $l > 1$, we make this connectivity as sparse. Since $P_l > 1$, the model learns overcomplete sparse feature maps. This structure is illustrated in Figure 5.1.

A deep deconvolutional network having NL number of layers ($l = 1 \ldots NL$) is trained upwards in a layer wise manner starting with the first layer ($l = 1$) where the inputs at layer l=0 are the training images $\mathcal{I}$. Each layer l is trained in order to learn a set of filters $f_{p,c}^l$ which is shared across all images in $\mathcal{I}$ and infer the set of feature maps $s_{p,l}^i$ of each image I^i in $\mathcal{I}$. To learn the filters, we alternately minimize $C_l(\mathcal{I})$ w.r.t. the filters and feature maps by keeping one of them constant

while minimizing the other.

5.3.2 Feature Encoding

Our goal here is to extract the hierarchical features of the given left I_L and the right I_R stereo images using the trained deep deconvolutional network. The network described above is top-down in nature i.e., given the latent feature maps and the filters, one can synthesize an image. But unlike the deep autoencoder [103] or deep belief networks [44], there is no mechanism (for example, encoder) for directly generating the feature maps from the given input apart from minimizing the cost function C_l in Eq. (5.6). Hence, once the network is learned/trained, we apply the given I_L and I_R separately as input to the trained deep deconvolutional network with the fixed set of learned filters and infer the feature maps $s_{p,l}^{I_L}$ and $s_{p,l}^{I_R}$ of I_L and I_R at layer l, respectively by minimizing the cost functions $C_l(I_L)$ and $C_l(I_R)$, respectively. For I_L, we stack its P_l number of inferred feature maps $s_{p,l}^{I_L}$ and obtain a single feature map $Z_l^{I_L}$ thereby we get a feature vector of size $P_l \times 1$ at every pixel. Similarly for I_R, we obtain a feature map $Z_l^{I_R}$ thereby we get a feature vector of size $P_l \times 1$. These two vectors represent l^{th} layer features of I_L and I_R, respectively at pixel (x, y).

5.3.3 Deriving Feature Matching Cost $E_F(d)$

Once the features of I_L and I_R are obtained at each layer of the deep deconvolutional network, we arrive at our learned feature matching cost $E_F(d)$ as follows:

$$E_F(d) = \sum_{l=1}^{NL} \sum_{(x,y)} \min(|Z_l^{I_L}(x,y) - Z_l^{I_R}(x + d(x,y), y)|, \tau^F). \tag{5.7}$$

At each pixel location (x, y) having disparity $d(x, y)$, Eq. (5.7) measures the absolute distance between the feature vector $Z_l^{I_L}(x, y)$ and the corresponding matched feature $Z_l^{I_R}(x + d(x, y), y)$. Similar to intensity matching cost, τ^F represents the truncation threshold.

Note that in our problem formulation, we do not use the feature matching cost alone to define the data term. As the deconvolutional network learns the sparse

features i.e., it results in the significant features at few locations in the image. Now, if one uses only the feature matching cost as a data term then at those pixel locations where the features are not significant, it results in ambiguous disparity estimates. Since the intensity values are available at every pixel location, we define our data term using a combination of intensity and feature matching costs. The combination of intensity and learned feature matching better constrains the solution space and hence results in unambiguous and dense disparities.

5.4 Dense Disparity Estimation

Our proposed method for obtaining the dense disparity map can be explained using a block schematic, shown in Figure 5.2. As discussed in section 5.2, in order to obtain the dense disparity map, we minimize the energy function given in Eq. (5.1). Our data term given in Eq. (5.2) consists of intensity and feature matching costs that are given in Eqs. (5.3) and (5.7), respectively. Similarly, our prior term uses IGMRF as given by Eq. (3.22). The final energy function to be minimized is then given by,

$$\begin{aligned} E(d) = & \sum_{(x,y)} \left(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau^I)\}\right) \\ & + \mu \sum_{l=1}^{NL} \sum_{(x,y)} \min\left(|Z^{I_L}_l(x,y) - Z^{I_R}_l(x+d(x,y),y)|, \tau^F\right) \\ + \sum_{(x,y)} b^X_{(x,y)} & (d(x-1,y) - d(x,y))^2 + \sum_{(x,y)} b^Y_{(x,y)} (d(x,y-1) - d(x,y))^2. \end{aligned} \tag{5.8}$$

Note that although we do not consider the occlusions explicitly, they are handled by clipping the matching costs using thresholds $\{\tau^I, \tau^F\}$. Our algorithm proceeds with the use of an initial estimate of disparity map, and iterates and alternates between the following two phases until convergence. As already discussed in Chapter 4, the two phase minimization is done as follows:

Phase 1:With d being fixed, compute IGMRF parameters $b^X_{(x,y)}$ and $b^Y_{(x,y)}$ using Eqs. (3.31) and (3.32), at each pixel location.

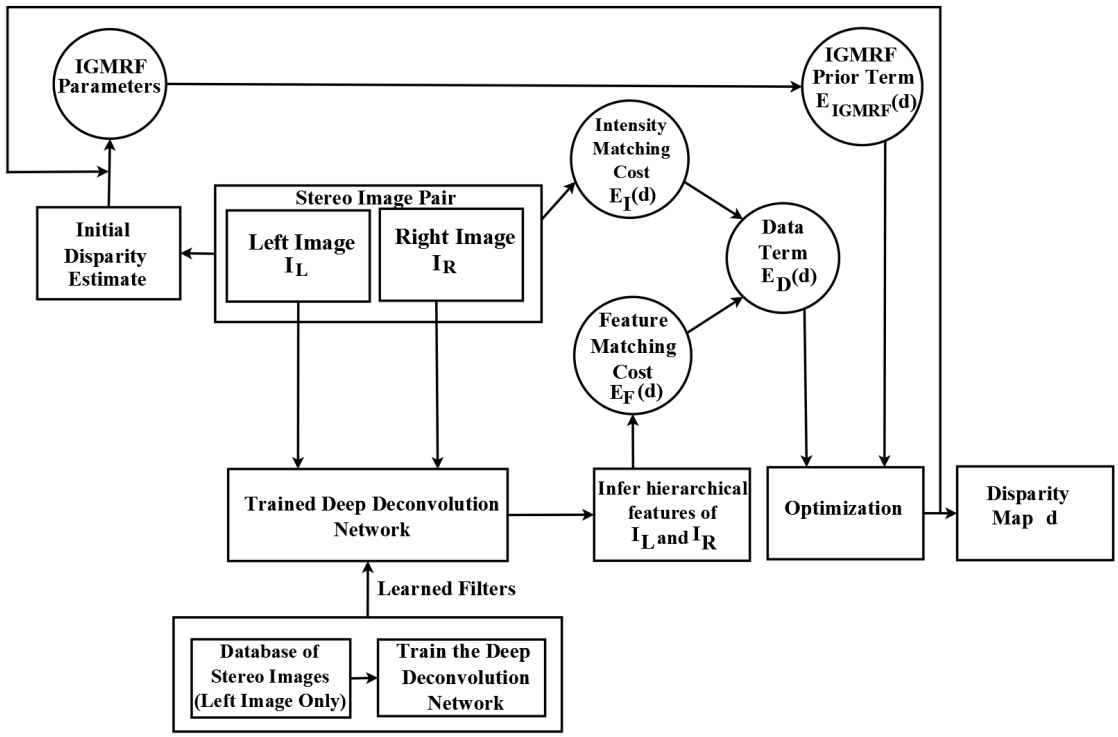

Figure 5.2: Block schematic of the proposed approach for dense disparity estimation. Here, the algorithm starts with the use of an initial estimate and iterates until convergence. Note that the given I_L and I_R are applied separately as input to the trained deep deconvolutional network in order to obtain the hierarchical features.

Phase 2: With $\{b^X_{(x,y)}, b^Y_{(x,y)}\}$ fixed as obtained in phase 1, minimize the Eq. (5.8) for d using graph cuts optimization based on α-β swap moves [69].

5.5 Experimentations

In this section, we demonstrate the efficacy of our proposed method by conducting various experiments and also testing our results on the Middlebury stereo datasets [113].

5.5.1 Parameter Settings

We first provide the details of various parameters used in training the deep deconvolutional network. In our experiments, a 2-layer deep deconvolutional network was trained over n_s=60 left stereo images downloaded from the Middlebury 2005

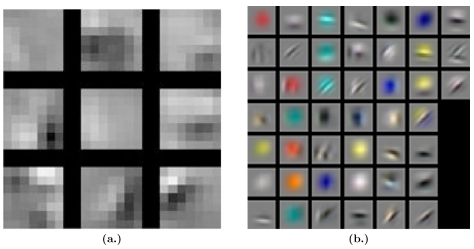

Figure 5.3: Filters learned at first and second layers of the deep deconvolutional network. (a.) Filters learned at first layer (9). (b.) Filters learned at second layer (81) where 36 filters in pair are shown in color and remaining 9 filters are shown as gray scale.

and 2006 datasets [113]. Considering NL=2, we set the number of feature maps at layer 1 and layer 2 as P_1=9 and P_2=45, respectively. We experimented on gray scale stereo images only and hence P_0=1. In our network, the feature maps at layer 1 were fully connected to the input. In order to reduce the computations, 36 feature maps in layer 2 were connected to a pair of maps in layer 1 and remaining 9 were singly connected. In this way, we have 9 and $36 * 2 + 9 = 81$ filters at layer 1 and 2, respectively. The parameter α in Eq. (5.6) was set to unity and the filters of size 7×7 were learned. These parameters were manually set as per the experimental settings done in [150] except that we used gray scale stereo images for training. With these parameter settings, our deep deconvolutional network was then trained to obtain a set of filters. The learned filters at the first and the second layers are shown in Figure 5.3 where the first layer learns Gabor like filters, and the learned filters in the second layer lead to mid-level features such as center-surround corners, T and angle-junctions, and curves.

In order to estimate the dense disparity map, we experimented on the Middlebury stereo 2001 and 2003 datasets [113] which are different from the training datasets used earlier. While minimizing Eq. (5.8), the data cost thresholds $\{\tau^I, \tau^F\}$ and the parameter μ were chosen manually for best performance. Our proposed algorithm for disparity map estimation was initialized with the initial estimate of disparity map and converged with in $5 - 10$ iterations for all the stereo pairs used in our experiments. We used the same parameters for all the experiments, and this demonstrates the robustness of our methods. The average run time of our algorithm was 150 seconds. Note that the training of deep deconvolutional network

Method	Venus	Teddy	Cones
Initial Estimate	3.47	19.65	16.43
AD	1.90	16.49	12.14
$E_I(d)$	0.95	15.67	11.89
$E_I(d)$+gradient	0.89	14.9	11.32
Proposed	**0.40**	**11.41**	**9.98**

Table 5.1: Performance evaluation in terms of % of bad matching pixels computed over the entire image with δ=1. Here, the optimization of energy function is carried out using different data terms $E_D(d)$ with IGMRF as prior term $E_P(d)$. (1^{st} row): Initial Estimate. (2^{nd} row): Using $E_D(d)$ as absolute differences (AD) between corresponding pixel intensities. (3^{rd} row): Using $E_D(d)$ as $E_I(d)$. (4^{th} row): Using $E_D(d)$ as $E_I(d)$+gradient matching. (5^{th} row): Proposed method where $E_D(d)$ is $E_I(d)$+$E_F(d)$.

is an offline operation and hence do not add to the computational complexity.

5.5.2 Quantitative Evaluation

In order to perform the quantitative evaluation, we used the percentage of bad matching pixels ($\mathcal{B}$%) as the error measure with a disparity error tolerance δ. The performance was tested under different scenarios. We first estimated the disparity map by considering the energy function with the use of truncated absolute differences of corresponding pixel intensities (AD) [114] as data term. We then considered the energy function with the robust data term which is insensitive to image sampling as given in Eq. (4.6). Note that this data term represents our intensity matching cost $E_I(d)$ (see Eq. (5.3)). In the third scenario, we considered the energy function with the data term derived using the combination of our $E_I(d)$ and gradient based feature matching cost which is given as the truncated absolute distances between the corresponding gradient features in the stereo image pair. In these three experimented scenarios, we used the IGMRF prior for regularization. Finally, we estimated the disparity map using our proposed method. For all these experimented cases, we used the same initial estimate and minimize the energy function using the two phase iterative algorithm. The quantitative results of these experiments are summarized in Table 5.1.

These results indicate that the performance of our proposed method is su-

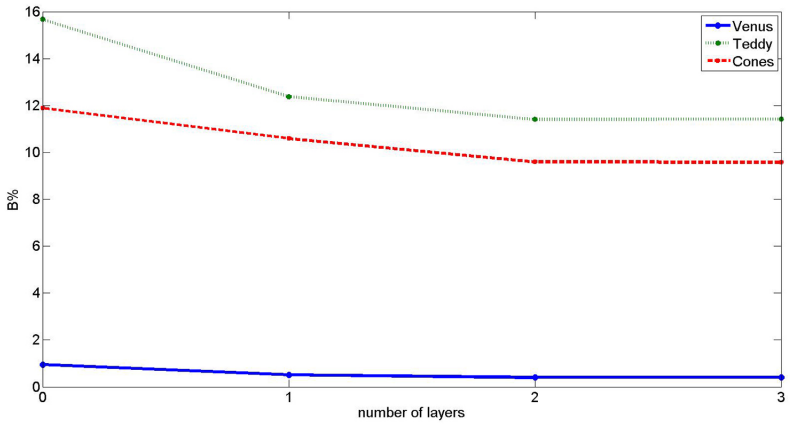

Figure 5.4: Results in terms of % of bad matching pixels by varying the number of layers NL in deconvolutional network. Here, NL=0 means $E_F(d)$ is not been used in optimization of Eq. (5.8).

perior when compared to the other experimented cases i.e., the performance is significantly improved when we use the data term which is a combination of intensity $E_I(d)$ and feature matching $E_F(d)$ costs, and outperforms those which use traditional data terms based on intensity matching. One can clearly observe that we get better results when $E_F(d)$ is added to $E_I(d)$ in comparison to the use of $E_I(d)$ only. This clearly shows the effectiveness of the learned feature matching in our approach. The efficacy of hierarchical feature learning using deep deconvolutional network is shown in the results. The characteristics of the stereo images are better represented by the learned hierarchical features, and matching of these features in addition to matching of raw intensities would better constrain the solution, and hence results in accurate disparity estimate at each pixel location. The results also show that the use of learned hierarchical features gives better disparities when compared to the use of basic gradient features.

We now demonstrate the performance of our proposed approach by varying the number of layers in the deep deconvolutional network. We first obtained the disparity map when $E_F(d)$ was derived using the learned features extracted only at the first layer of deep deconvolutional network. Next, the results were obtained using the learned features at both the first and the second layers. In other words, we considered NL=1 and NL=2 in Eq. (5.8) for these two cases. Note that the other terms in Eq. (5.8) remain same. Figure 5.4 shows that the performance improves when we use learned features of both the layers. This shows the effectiveness of

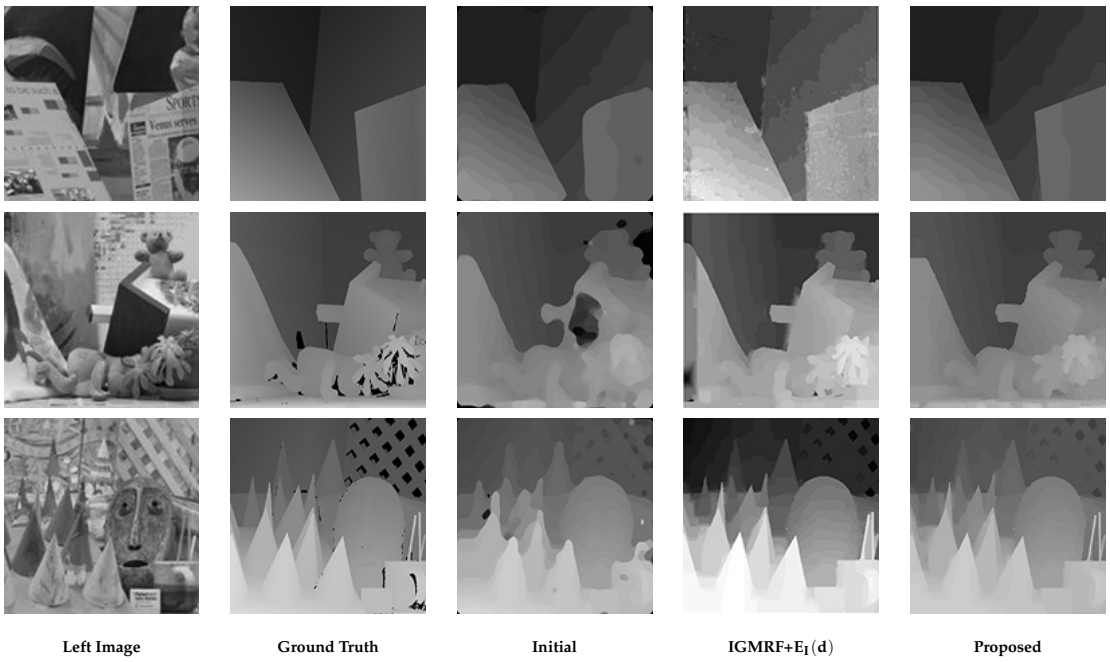

Figure 5.5: Disparity maps estimated for the datasets of [113], "Venus" (1^{st} row), "Teddy" (2^{nd} row) and "Cones" (3^{rd} row). (1^{st} column): Left Image. (2^{nd} column): Ground Truth. Results for different experimented cases, (3^{rd} column): Initial estimate, (4^{th} column): Using $E_D(d)$ as $E_I(d)$, and (5^{th} column): Proposed.

stereo matching using hierarchical features. We also experimented with the use of three layers but we did not find significant improvement when the number of layers NL is greater than 2 (see Figure 5.4). Based on these observations, we trained only 2-layer deep deconvolutional network in our work. This shows the effectiveness of the use of deep learning with limited number of layers.

5.5.3 Qualitative Analysis

In order to show the visual quality of our results, we now display the computed disparity maps. Each row of Figure 5.5 displays the left image of the stereo pair, ground truth, initial estimate obtained, disparity map obtained using IGMRF prior with the data term using only the intensity matching cost $E_I(d)$, and the final disparity map obtained using our proposed method. One can see that the final disparity maps obtained using the proposed method are piecewise smooth and visually plausible, and show improvement in the quality over the initial estimate as well as that computed using only $E_I(d)$ as the data term. Looking at the results, one can see that the accuracy of estimated disparities is better in the planar

Method	Venus	Teddy	Cones
GraphCuts [21]	3.44	25.0	18.0
MultiGC [68]	3.13	17.6	11.8
BeliefProp [143]	0.45	8.30	8.78
SegmentBeliefProp [66]	0.21	7.06	7.92
FeatureExpansion [76]	0.45	12.6	10.1
SemiGlobal [46]	1.57	12.2	9.75
LearnedCRF [111]	1.3	11.1	10.8
Proposed	0.40	11.41	9.98

Table 5.2: Comparison with the state of the art global dense stereo methods evaluated on the Middlebury stereo 2001 and 2003 datasets [113] in terms of % of bad matching pixels computed over the entire image with δ=1.

and textureless regions for the proposed approach. These results show the effectiveness of learned hierarchical feature matching in addition to intensity matching for disparity estimation. Use of learned filters to capture sparse features in an unsupervised way makes our method to better handle the outliers and hence results in accurate disparity maps.

5.5.4 Comparison with the State of the Art Methods

We now compare our results with the state of the art global dense stereo methods. Once again the comparisons are shown in terms of percentage of bad matching pixels ($\mathcal{B}$%), and the same are shown in Table 5.2. Here, we do not show the comparison with few of the global stereo methods which are based on hand crafted and learned features [82, 64, 109, 153, 91] since their results are not available using the Middlebury datasets. The results in Table 5.2 show that our method achieves better performance and is comparable to the other state of the art global stereo methods. Our method gives superior performance except those proposed in [143] and [66]. This is because these methods handle the occlusions explicitly and use belief propagation for minimization of their energy functions.

In order to compare the performance of our method with the latest best performing global stereo methods, we experimented on the recently released Middlebury stereo 2014 datasets [113]. These comparisons are shown in Table 5.3. We can observe that our results are comparable to the methods listed in Table 5.3.

Method	Adiron	Jadep	Motor	PianoL	Pipes	Playrm	Playt	Recyc	Shelvs	Vintage
AdaptSmooth [65]	14.3	24.5	9.82	31.3	9.80	28.9	22.4	17.8	45.4	31.8
DeepCNN [149]	11.6	24.5	9.66	30.0	8.84	30.2	25.9	16.1	42.7	41.2
MeshStereo [152]	20.6	35.3	20.6	37.9	23.4	39.5	34.1	25.9	53.0	35.6
HiddenMarkov [104]	26.1	33.6	24.4	44.0	19.3	41.1	50.4	27.5	59.5	51.4
MultiDisparity [79]	25.0	35.9	30.4	41.7	29.0	42.7	47.8	31.3	54.5	43.6
PlaneSweep [115]	29.9	34.7	12.3	59.6	15.8	41.4	33.4	33.6	51.5	45.8
TwostepGlobal [94]	55.9	73.3	52.3	74.7	50.9	72.3	66.2	52.4	71.0	76.6
Proposed	63.7	68.6	60.4	71.9	37.1	71.7	80.7	66.7	81.9	61.8

Table 5.3: Quantitative evaluation on Middlebury stereo 2014 datasets [113] and comparison with current better performing global dense stereo methods. Evaluation is in terms of % of bad matching pixels in non-occluded regions with δ=1.

Though, our results are not better than the deep learning based method [149], one can improve the results by using better hierarchical features learned via an efficient deep learning method.

5.6 Conclusion

In this chapter, we have proposed a dense disparity map estimation approach using learned feature matching in a regularization framework. A feature matching cost derived using the learned hierarchical features from the given left and the right stereo images were used. We combined this with the pixel-based intensity matching cost to form our data term in our energy function. The matching by combining both these would better constrain the solution and hence results in robust and accurate disparity map. Inspired by the recent development and success of the deep learning methods, we used the deep deconvolutional network for learning the hierarchical features of stereo images. The deep deconvolutional network is trained using a large set of stereo images in an unsupervised way which in turn results in a diverse set of filters. These learned filters capture the image characteristics at different levels in the form of low, mid and high-level features. In order to perform the regularization, we used the IGMRF prior in our energy function that captures the smoothness as well as preserves sharp discontinuities in the disparity map. The energy function was minimized using graph cuts in an iterative two phase algorithm where the IGMRF parameters and the disparity

map were refined alternatively.

Experimentations on the Middlebury datasets validate the performance of our proposed approach. Our results demonstrate the effectiveness of the use of learned feature matching, and the results significantly improved when the learned hierarchical features are used for matching when compared to the use of basic features. Similarly, the improvement was observed when the feature matching is combined with intensity matching. The power of deep learning in stereo matching is demonstrated in our results. The performance of the proposed method was comparable to many of the state of the art and the latest global methods.

In the next chapter, we propose a different energy minimization framework for disparity estimation where we use feature matching in an IGMRF and sparsity based regularization framework.

CHAPTER 6

Feature Matching in an IGMRF and Sparseness based Regularization Framework

The dense disparity estimation is a difficult problem due to depth discontinuities, photometric variation, lack of texture, repetitive texture, occlusions, etc. To handle these issues, the problem is solved using a data model that relates the disparity with the acquired left and right images and a prior model based on theoretical deduction about the disparity. An energy minimization framework is then used to get a solution. However, the quality of solution depends on the assumed models. In chapter 3, we used an energy function with IGMRF prior, and in chapter 4, a combination of IGMRF and sparsity priors was incorporated in the energy function. In these chapters, we focused on arriving at a proper energy function by using suitable priors. In Chapter 5, we used an energy function consisting of pixel intensity as well as feature matching costs as data term with IGMRF prior. We derived our feature matching cost from the learned hierarchical features of given left and right stereo images, and these hierarchical features were learned using the deep deconvolutional network.

In this chapter, we propose a better energy function to be used in an energy minimization framework. We use the same data term as derived in chapter 5 i.e., the combination of intensity and learned feature matching. In order to perform the regularization, in this chapter, we use sparseness as prior in addition to the IGMRF prior. A sparse autoencoder is used for learning and inferring the sparse representation of disparities. Use of additional prior should better constrain the solution and hence we expect a better estimate of the disparity map. As done

in earlier chapters, an iterative two phase algorithm is proposed to estimate the dense disparity map. Experiments on the Middlebury stereo datasets demonstrate that our proposed method leads to better disparity maps than the other proposed methods discussed in previous chapters, and also the performance is comparable to the state of the art global stereo methods.

6.1 Proposed Method

The proposed technique of dense disparity estimation is illustrated by the block diagram shown in Figure 6.1. The problem is once again formulated in an energy minimization framework where the energy function consists of sum of data term $E_D(d)$ and prior term $E_P(d)$. Our data term remains the same as Eq. (5.2) as used in chapter 5 i.e.,

$$E_D(d) = E_I(d) + \mu E_F(d), \tag{6.1}$$

We use the pixel based symmetric measure for $E_I(d)$ as given in Eq. (5.3) and the feature matching term $E_F(d)$ derived using the learned and hierarchical features of stereo image pair as given in Eq. (5.7). We form the prior term $E_P(d)$ as a sum of IGMRF and sparsity priors and it is given by,

$$E_P(d) = E_{IGMRF}(d) + \gamma E_{sparse}(d). \tag{6.2}$$

The formation of $E_{IGMRF}(d)$ is given in Eq. (3.22). In chapter 4, we demonstrated that the sparse representation of disparities learned using sparse autoencoder is more effective than the use of overcomplete dictionaries trained via K-SVD, and using it significantly improves the performance of disparity estimation. Therefore, here we use the sparse autoencoder for learning and inferring the sparseness of disparities, and the formation of $E_{sparse}(d)$ is given by Eq. (4.22).

Using Eqs. (5.3), (5.7), (3.22) and (4.22) defined for $E_I(d)$, $E_F(d)$, $E_{IGMRF}(d)$

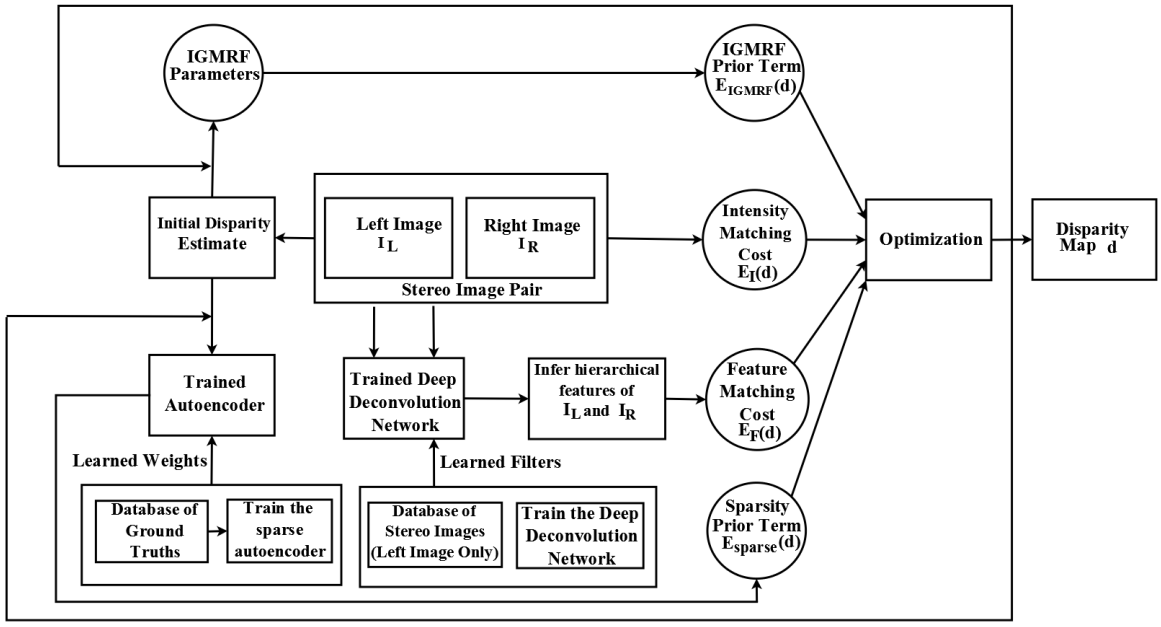

Figure 6.1: Block schematic of the proposed approach for dense disparity estimation. Here, the algorithm starts with the use of an initial estimate and iterates until convergence.

and $E_{sparse}(d)$ terms, respectively, our final energy function is given by,

$$\begin{aligned} E(d) = & \sum_{(x,y)} \left(\min\{F^{fwd}_{(x,y)}(d(x,y)), F^{rev}_{(x,y)}(d(x,y),\tau^I)\}\right) \\ & + \mu \sum_{l=1}^{NL} \sum_{(x,y)} \min\left(|Z_l^{I_L}(x,y) - Z_l^{I_R}(x+d(x,y),y)|, \tau^F\right) \\ & + \sum_{(x,y)} b^X_{(x,y)} (d(x-1,y) - d(x,y))^2 + \sum_{(x,y)} b^Y_{(x,y)} (d(x,y-1) - d(x,y))^2 \\ & + \gamma \sum_{(x,y)} \left|\left| d^{(x,y)} - f(U^T a^{(x,y)} + s) \right|\right|_2^2. \end{aligned} \tag{6.3}$$

The above energy function is semi-metric and hence we minimize it using the $\alpha - \beta$ swap move based graph cuts optimization. We do not consider the occlusions explicitly but they are handled by clipping the matching costs using thresholds $\{\tau^I, \tau^F\}$ that prevent the outliers from disturbing the estimation. For finding the correspondences, we consider search from left to right as well as from right to left and hence relax the traditional ordering constraint used in disparity estimation.

We propose an iterative two phase algorithm. It proceeds with the use of an initial estimate of disparity map, and iterates and alternates between two phases

Algorithm 1: Proposed Algorithm

Input: Stereo image pair I_L and I_R, a set of ground truth disparity patches $\mathcal{G}_d$=$\{d^{(1)}, d^{(2)}, \ldots, d^{(n_g)}\}$, and a set of stereo images $\mathcal{I}$=$\{I^1, I^2 \ldots, I^{n_s}\}$.

1 Train a deep deconvolutional network consisting of NL number of layers, by minimizing Eq. (5.6) for each layer l and learn a set of filters;

2 Infer the hierarchical features $Z_l^{I_L}$ and $Z_l^{I_R}$ of I_L and I_R, respectively ($l = 1 \ldots NL$) using the method discussed in section 5.3.2 ;

3 Train the sparse autoencoder using $\mathcal{G}_d$ by minimizing Eq. (4.21) and obtain weights (W, U, r, s);

4 Obtain an initial disparity map d_0 and set $d = d_0$;

5 **do**

6 **Phase 1**: With d being fixed, infer the sparse vector $a^{(x,y)}$ for every disparity patch $d^{(x,y)}$ in d using trained sparse autoencoder (Eq. (4.18)). Compute IGMRF parameters $b^X_{(x,y)}$ and $b^Y_{(x,y)}$ at every pixel location using Eqs. (3.31) and (3.32);

7 **Phase 2**: With $a^{(x,y)}$, $b^X_{(x,y)}$, and $b^Y_{(x,y)}$ fixed at every (x, y), as obtained in phase 1, minimize the Eq. (6.3) for d using graph cuts [69].;

8 **while** *convergence*;

until convergence as given in Algorithm 1. We use the same method of obtaining an initial estimate of disparity map as used by our methods proposed in chapters 4 and 5. However, any other suitable disparity estimation method can also be used in obtaining the initial estimate.

6.2 Experimentations

In this section, we demonstrate the efficacy of the proposed method by conducting various experiments and evaluating our results on the Middlebury stereo benchmark images [113]. In our experiments, a 2-layer deep deconvolutional network was trained using n_s=60 left stereo images, and a sparse autoencoder was trained using a set of $n_g = 5 \times 10^5$ true disparity patches extracted from the ground truth disparity maps obtained from the Middlebury 2005 and 2006 datasets [113]. The details of the parameters used for training the sparse autoencoder and the deep deconvolutional network are given in the experimental sections of chapter 4 and 5, respectively. In order to estimate the dense disparity map, we experimented on the Middlebury stereo 2001 and 2003 datasets [113] which are different from the

training datasets used earlier.

While minimizing Eq. (6.3), the data cost thresholds $\{\tau^I, \tau^F\}$ and the parameter μ were chosen on a trial and error basis until a better solution was obtained. The parameter γ was initially set to 10^{-4} and increased exponentially at every iteration from 10^{-4} to 10^{-1}. We used the same parameters for all the experiments and this demonstrates the robustness of our method. Our algorithm was initialized with the initial estimate of disparity map and it converged within $5-10$ iterations for all the stereo pairs used in our experiments. The average run time of our algorithm was 200 seconds. Note that the training of deep deconvolutional network and the autoencoder were an offline operation and hence they did not add to the computational complexity.

6.2.1 Quantitative Comparison

In order to perform the quantitative evaluation, we used the percentage of bad matching pixels ($\mathcal{B}\%$) as the error measure with a disparity error tolerance δ. We compare the performance of our proposed method with our other methods proposed in previous chapters. To perform a fair comparison, we used two phase iterative algorithm for all our methods, and also use the same initial estimate obtained from the method discussed in chapter 4. The quantitative comparisons are summarized in Table 6.1. We also tested our proposed approach for the case where $E_{sparse}(d)$ in Eq. (6.2) was learned using the K-SVD algorithm. We observed that the results using this case were not better than the one obtained using sparse autoencoder, and hence we do not include these results in Table 6.1.

The results in Table 6.1 show that the performance of the method in this chapter is best among all our earlier approaches. Our results demonstrate the effectiveness of the stereo matching using learned hierarchical features and the regularization using IGMRF and sparsity priors. The combination of feature and intensity matching combining the IGMRF and the sparsity priors would better constrain the solution and resulting in accurate dense disparity map.

Method	Venus	Teddy	Cones
Initial estimate	3.47	19.65	16.43
IGMRF based prior (chapter 3)	1.78	18.1	14.42
IGMRF-KSVD based prior (chapter 4)	1.43	12.5	11.42
IGMRF-Autoencoder (chapter 4)	0.22	10.7	9.64
Intensity+Feature Match (chapter 5)	0.40	11.41	9.98
Proposed	**0.20**	**9.76**	**8.46**

Table 6.1: Evaluation results on the Middlebury datasets [113] in terms of % of bad matching pixels computed over the entire image with δ=1. Comparisons include different cases: (1^{st} row): Initial Estimate. (2^{nd} row): Using IGMRF prior. (3^{rd} row): Using IGMRF and sparsity prior learned using K-SVD dictionary. (4^{th} row): Using IGMRF and sparsity prior learned using sparse autoencoder. (5^{th} row): Using intensity and learned feature matching. (6^{th} row): Proposed Method.

6.2.2 Qualitative Analysis

We have till now discussed the quantitative performance where we use true disparity map as a reference to find the bad matching pixels. We now show the perceptual quality assessment of the proposed method experimented using Middlebury stereo 2001 and 2003 datasets [113]. In Figure 6.2, we show the estimated disparity maps using the stereo images of these datasets. We also show the error maps associated with the disparity maps in the last column of Figure 6.2. The error maps show the regions where the estimated disparities differ from the ground truth (black and gray regions correspond to errors in occluded and non-occluded regions, respectively and white indicates no error). We can see that the proposed method has higher accuracy in discontinuous as well as non-occluded regions. This is because the IGMRF prior preserves the discontinuities and the sparsity prior learns the edge-like sparse features in disparity map and the use of these two with the learned feature and intensity matching results in accurate disparities. As can be seen from Figure 6.2, our method not only preserves geometrical details near depth discontinuities but performs better in textureless and homogeneous regions as well. We mention here that although we do not consider occlusions in our problem formulation, our method works well in these regions as well. Performance improvement in occluded regions is due to the presence of data term truncation thresholds τ^I and τ^F.

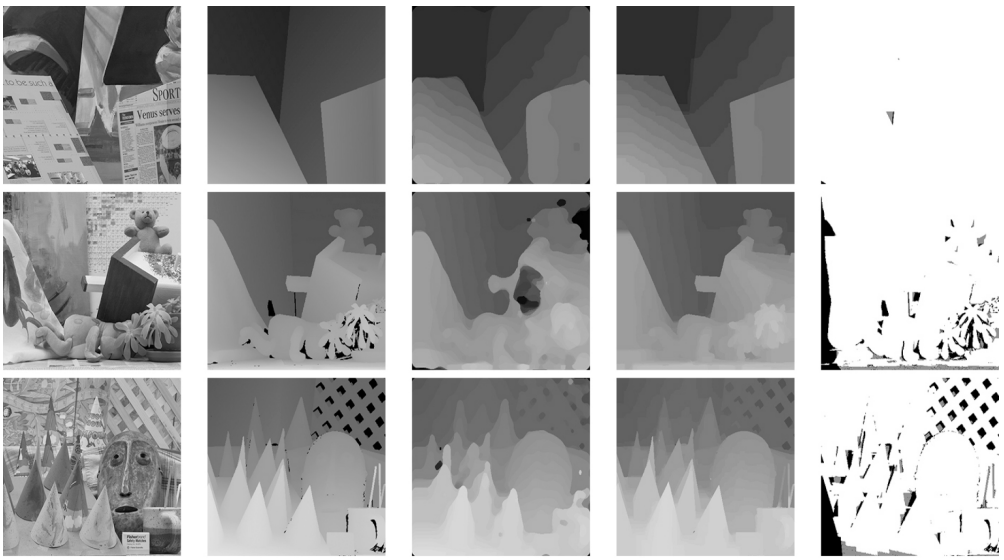

Figure 6.2: Experimental results for the Middlebury stereo 2001 and 2003 datasets [113], "Venus" (1^{st} row), "Teddy" (2^{nd} row) and "Cones" (3^{rd} row). The left image I_L and the ground truth disparity map are shown in first and second columns, respectively. The third column shows the initial disparity map used in optimizing the energy function given in Eq. (6.3). The final disparity and the error maps estimated using the proposed method are shown in the fourth and the fifth columns, respectively.

6.2.3 Comparison with the State of the Art Methods

In order to validate the results of our method, we compare it with the state of the art global dense stereo methods. The compared approaches include edge preserving first order MRF prior [21, 68, 143], higher order MRF [133], learned regularization parameters [111], sparsity regularization [42, 123], learned data cost [147], higher order likelihood [58], feature matching [66, 76, 46], Mumford Shah regularization [10], consistency prior [59], bilateral filtering [94] and ground control points [131]. The Table 6.2 shows the results and comparison done in terms of bad matching pixels ($\mathcal{B}$%) computed over the entire image as well as in the non occluded regions. As seen from the Table 6.2, the performance of our method is best among all the other methods, when measured in non-occluded regions. It also gives the least number of bad matching pixels over the entire image as well as in the non-occluded regions for the "Venus" stereo pair. Although, our method does not handle the occlusions explicitly, it gives better or comparable performance measured when compared to the other state of the art global stereo methods.

Method	Venus		Teddy		Cones	
	all	*nonocc*	*all*	*nonocc*	*all*	*nonocc*
SegmentBeliefProp [66]	0.21	0.10	7.06	4.22	7.92	2.48
BeliefProp [143]	0.45	0.13	8.30	3.53	8.78	2.90
GroundPoints [131]	0.53	0.16	11.5	6.44	9.49	3.59
FeatureExpansion [76]	0.45	0.27	12.6	7.42	10.1	4.09
SemiGlobal [46]	1.57	1.00	12.2	6.02	9.75	3.06
SecondOrderMRF [133]	0.49	0.24	15.4	10.9	10.8	5.42
CompressSensing [42]	0.68	0.31	13.30	7.88	9.79	3.97
MultiGC [68]	3.13	2.79	17.6	12.0	11.8	4.89
Mumford [10]	0.76	0.28	14.3	9.34	9.91	4.14
GraphCuts [21]	3.44	1.79	25.0	16.5	18.0	7.70
LearnedCRF [111]	1.3	-	11.1	-	10.8	-
LearnedSparse [123]	-	-	8.14	-	11.98	-
HighLikelhood [58]	-	0.22	-	5.62	-	2.40
LearnedSVM [147]	-	-	-	8.15	-	3.77
TwostepGlobal [94]	1.49	0.88	9.40	6.24	7.66	2.77
ConsistencyPrior [59]	0.61	0.34	12.4	7.67	9.35	3.33
Proposed	0.20	**0.10**	9.76	**3.44**	8.46	**2.36**

Table 6.2: Quantitative evaluation on Middlebury stereo 2001 and 2003 datasets [113] and comparison with the state of the art global dense stereo methods in terms of % of bad matching pixels over the entire image as well as in non-occluded regions. Here, δ=1 and '-' indicates the result not reported.

6.2.4 Experiments on the Latest Middlebury Datasets

We now discuss the experiments on the recently released Middlebury stereo 2014 (version 3) datasets that consists of 15 training and 15 test stereo pairs. Here, we submitted the estimated disparity maps online to the server available on Middlebury website [113] which in turn returned the evaluation of our method in terms of bad matching pixels and also the comparison with the latest stereo methods. Since the test datasets do not have ground truth, evaluation is done by submitting the estimated disparity maps only. We mention here that one cannot adjust the parameters for test datasets because the submission can be done only once. Our results were ranked at 53 and 60, respectively for training and test sets. Although, the proposed method is not ranked among the top, the results indicate that it is comparable to the other latest stereo methods.

Since our method corresponds to a global approach based on regularization

Method	Adiron	Jadep	Motor	PianoL	Pipes	Playrm	Playt	Recyc	Shelvs	Vintage
AdaptSmooth [65]	14.3	24.5	9.82	31.3	9.80	28.9	22.4	17.8	45.4	31.8
DeepCNN [149]	11.6	24.5	9.66	30.0	8.84	30.2	25.9	16.1	42.7	41.2
MeshStereo [152]	20.6	35.3	20.6	37.9	23.4	39.5	34.1	25.9	53.0	35.6
HiddenMarkov [104]	26.1	33.6	24.4	44.0	19.3	41.1	50.4	27.5	59.5	51.4
MultiDisparity [79]	25.0	35.9	30.4	41.7	29.0	42.7	47.8	31.3	54.5	43.6
PlaneSweep [115]	29.9	34.7	12.3	59.6	15.8	41.4	33.4	33.6	51.5	45.8
TwostepGlobal [94]	55.9	73.3	52.3	74.7	50.9	72.3	66.2	52.4	71.0	76.6
IGMRFAutoencoder (chapter 4)	62.4	67.5	60.2	69.1	34.9	71.2	79.9	65.9	79.9	60.1
LearnedFeatureMatch (chapter 5)	63.7	68.6	60.4	71.9	37.1	71.7	80.7	66.7	81.9	61.8
Proposed	60.1	66.2	60.0	70.8	32.6	66.8	75.3	63.2	77.2	56.8

Table 6.3: Quantitative evaluation on Middlebury stereo 2014 datasets [113], and comparison with the current better performing global dense stereo methods and our methods proposed in previous chapters. Evaluation is in terms of % of bad matching pixels in non occluded regions with δ=1.

and deep learning, we show the performance comparison in terms of quantitative measures with the few of the best performing latest global and learning based dense stereo approaches. From Table 6.3, one can see that the performance of our method is comparable to the latest stereo methods. We see that our approach is not the best. This is because the accuracy of our method is sensitive to the parameters of the model and the choice of an initial estimate. Also, it does not handle the occlusions explicitly which is still a great challenge in stereo research. However, one may carefully choose the parameters and make a proper choice of the initial estimate to improve the accuracy. However, this results in increased time complexity.

6.3 Conclusion

In this chapter, we have proposed the use of learned feature and intensity matching for dense disparity estimation in an IGMRF and sparsity regularization framework. Our data term was formed using the combination of pixel based intensity and learned feature matching costs. Our regularization term included IGMRF as well as the sparsity prior. Once again, an iterative two phase algorithm was used to estimate the dense disparity map. The results on the Middlebury stereo

datasets have demonstrated that the proposed method in this chapter leads to better disparity maps than the other methods proposed in previous chapters. These results indicate that the performance is superior compared to other the state of the art methods when measured in non-occluded regions and comparable when measured over the entire disparity map. The comparable performance is also seen with the latest global stereo methods. Our results show the effectiveness of using the intensity and feature matching along with the combination of IGMRF and sparsity prior in the energy function.

CHAPTER 7

Conclusions and Future Research Work

7.1 Conclusions

In this thesis, we have addressed the problem of dense disparity estimation using a pair of rectified stereo images with known camera calibration. In general, disparities are obtained by comparing pixel intensities or their features in the left and right images. However, estimation of disparities is an ill-posed problem, and this problem is formulated and solved using a global energy minimization framework by incorporating regularization. An energy function represents a combination of data term and prior terms that restricts the solution space. The data term ensures that the disparity map is in consent with the observed data i.e., the given stereo pair while the prior term confines it to have a form matched with the advance knowledge about the true disparity map. It is important to design a proper energy function that embeds the constraints. In addition, it is necessary to find a good minimization technique that leads to global optimal solution. Selection of the appropriate data and prior models help us to obtain an accurate dense disparity map. In this thesis, we have proposed various approaches for disparity map estimation using the energy minimization framework. We have employed graph cuts, an efficient and fast optimization technique to minimize our energy functions.

We began by proposing a dense disparity estimation method using inhomogeneous Gaussian Markov random field (IGMRF) as a prior in chapter 3. The IGMRF prior captures the spatial dissimilarity among disparities at every pixel location while preserving the sharp discontinuities. In practice, edge preserv-

ing homogeneous MRF prior models have been used in global stereo methods in which a limited number of global parameters capture the spatial variations. These parameters are usually either set by trial and error or estimated when working on a set of images. Although, these homogenous priors result in piecewise smooth solution, they regularize disparities by the same set of limited number of parameters. This assumption does not hold good in reality because the variation among disparities at each pixel is different and hence such methods fail to better capture the spatial dependency among disparities. This motivated us to use an IGMRF prior that captures the variations at each pixel location using the adaptive IGMRF parameters. To form our energy function, we defined the data term using the pixel-based intensity matching cost based on the brightness constancy assumption and the prior term was defined using IGMRF prior. The IGMRF parameters were computed using the initial estimate of disparity map that was obtained using a learning based method. To start with, we used the initial estimate of disparity map to compute the IGMRF parameters at every pixel location which were then used to estimate the final disparity map by minimizing our energy function. Our experimental results showed that the use of IGMRF prior leads to accurate disparity map when compared to those using edge-preserving homogeneous MRF priors. Disparity maps obtained using the proposed method were less noisy in homogeneous areas and preserved the textures and sharp details in other regions. However, the limitation of this approach was that the quality of the final solution strongly dependent on the accuracy of the IGMRF parameters. We observed that the initial estimate obtained using our learning based method resulted in noisy disparities in homogeneous regions and near the edges that affected the performance. In order to take care of the same in our subsequent approaches, we have used a classical local stereo method with disparity refinement techniques for obtaining an initial disparity map.

Although, IGMRF prior captures the smoothness as well as discontinuities, it fails to capture higher order dependencies in the disparity map. One of the characteristics of natural scenes is that there exists significant amount of redundancy in disparity map. Due to this, the disparity maps are generally sparse in

a transform domain which can be obtained either by using a fixed set of basis or can be learned using a set of training examples. We considered the sparseness of disparities as a prior knowledge and used it for regularization. Instead of using fixed bases, we learned an efficient sparse representation of disparities. We then proposed the use of IGMRF and sparsity priors in our approach for dense disparity estimation in chapter 4. The sparsity prior is defined using the learned overcomplete sparseness of disparity patches which captures the higher order dependencies in the disparity map. The combination of IGMRF and sparsity priors better constrains the solution. In this case, we considered a data term using the pixel-based intensity matching cost which is robust to outliers and insensitive to image sampling. Based on this, we proposed two methods for dense disparity estimation. In our first method, sparse representation of disparities were obtained by a learned overcomplete dictionary where we used training via K-SVD algorithm. The advantage of our K-SVD based dictionary learning method is that the learned dictionary is adaptive to the disparities of the given stereo pair and hence avoids the use of ground truth disparities while training. However, overcomplete dictionary model uses a linear structure. Hence, in our next approach, we used a non linear model, sparse autoencoder to infer better sparse representation of disparities. We trained our sparse autoencoder using a large number of ground truth disparity patches. In order to estimate the dense disparity map, we started with the use of an initial estimate of disparity map and iterated and alternated between two phases until convergence. In phase one, sparseness of disparities were inferred and IGMRF parameters were computed based on the current estimate of disparity map while in the second phase, the disparity map was refined by minimizing the energy function keeping the other parameters fixed. Our experimental results demonstrated the effectiveness of both the proposed approaches showing better performance with the use of a combination of sparsity and IGMRF priors. Our results also verify the effectiveness of using the learned sparseness using overcomplete dictionary and the sparse autoencoder when compared to the use of fixed basis. The results obtained using the method based on sparse autoencoder have shown a significant improvement when compared to the other method

based on K-SVD. The two methods have also shown the superior and comparable performance when compared to many of the state of the art and the latest global stereo methods, respectively.

The combination of IGMRF and sparsity priors serve as a better regularizer but the choice of an appropriate data model also plays a key role in obtaining a better disparity map. Although, the data term used in chapter 4 is robust against outliers, image sampling, view-point variation, etc., it rely on the raw pixel values (intensities) only and it's use may result in ambiguous and erroneous disparities in the textureless areas and near the depth discontinuities. The stereo images can be represented in a better way by using a feature space where they are robust, distinct and invariant to view point. Motivated by the significant progress in solving the disparity estimation problem using the machine learning, our next work used the self learned features which are obtained in a different way when compared to hand crafted features. Using this, in chapter 5 we proposed a method using feature matching where we derived a feature matching cost using the learned hierarchical features of the given left and right stereo images and combined it with the pixel-based intensity matching cost to form our data term. Self learned hierarchical features were obtained using the deep deconvolutional network, a deep learning model which was trained in an unsupervised way using a database consisting of large number of stereo images. An IGMRF prior was used in regularizing the solution. Once again, an iterative two phase algorithm was used to obtain the final solution. Our experimental results demonstrated that the combination of feature and intensity matching and the use of this IGMRF prior in the energy function better constrain the solution, and this results in robust and accurate disparity maps. We observed that the results were significantly improved when compared to the use of basic and hand crafted features. Improved performance was also seen when the feature matching was combined with intensity matching. The power of deep learning i.e., the use of hierarchical features in stereo matching were clearly seen in our results. Performance of the proposed method was comparable to many of the state of the art and the latest global stereo methods.

Finally, in chapter 6, we proposed a method using a better constrained energy

function. We used our data term as a combination of learned feature matching and pixel-based intensity matching costs, and form the prior term using the combination of IGMRF and sparsity priors. Since the use of sparse autoencoder has shown superior performance in representing the sparseness of disparities when compared to the use of learned K-SVD dictionary, we used sparse autoencoder for learning the sparse representation of disparities. An iterative two phase algorithm was proposed to estimate the dense disparity map. The experimental results on the Middlebury stereo datasets have demonstrated that this method leads to better disparity maps when compared to our other proposed methods. The comparison with the state of the art methods also showed that the performance of our method is superior when measured in non-occluded regions and comparable when measured over the entire disparity map. The comparable performance of our method was also seen with the latest global stereo methods. All these results reflect the effectiveness of the combination of intensity and feature matching and the combination of IGMRF and sparsity prior resulting in better and accurate disparities. Since the influence of the data term and the prior term in getting better results is controlled by the set of model parameters, we can conclude that every term in the energy function has equal significance in achieving a better solution.

7.2 Future Research Work

Dense disparity estimation is an ill-posed problem. A solution to this problem requires information available from the stereo image pair and the prior knowledge about the true disparity map. It is a challenging task to compute the accurate disparities at occluded points, near depth discontinuities, and in textureless regions. Therefore, several of these issues are to be appropriately addressed in order to get better disparity estimates. In this section, we discuss the future research works in this direction.

- In our work, we have estimated the disparities using the left and right rectified views of a scene. In order to better constrain the solution, one can

use multiple rectified views/images of a scene in our proposed energy minimization framework for disparity estimation.

- The disparities are estimated using various stereo matching constraints, and in our methods, we have employed few of the constraints such as epipolar constraint, photometric similarity constraint, geometric/feature similarity constraint, smoothness constraint, and sparsity constraint. However, one can incorporate the "uniqueness constraint" as well in the energy function in order to obtain unique and unambiguous matches. According to the uniqueness constraint, for any location in one image, there should be at most one matching location in the other image, and vice versa.
- Occlusions are a major challenge for the accurate computation of visual correspondence. Occluded pixels are visible in only one image, so there is no corresponding pixel in the other image. In our work, we have not handled the occlusions explicitly. Most of the dense stereo approaches check for the occluded regions as a post processing operation. However, the occluded regions can be explicitly modeled along with the disparity map by using an *occlusion term* in the energy function. An occlusion term imposes a penalty for the pixels which are occluded.
- Our proposed methods are flexible to work for the color images also. One can use color images for improving the estimation accuracy since color images provide more information.
- Most of the disparity estimation techniques have high computational complexity. This limits the use of the algorithms for real-time applications. Since our proposed methods are based on iterative process, the time complexity increases. In order to make our algorithms suitable for the real-time application, one can use the *graphics processing unit* (GPU) for running the algorithms. One can also use *extreme learning machines* for reducing the time complexity.
- In our work, we have proposed the disparity estimation approaches for rectified stereo images. However, the rectified images are obtained after ap-

plying a set of transformation on the stereo images captured from different view-points, resulting in the information loss. Hence, one can consider the real stereo images as input for disparity estimation. In this case, the disparity is estimated in x as well as y directions. However, estimation of this in a global energy minimization framework is computationally expensive.

- Recently, many of the computer vision problems are addressed effectively by using machine learning approaches. Hence, one can explore the use of deep learning based techniques in order to estimate the disparities in a global framework. Here, the mapping between the stereo image pair (left appended with right image) and the disparity can be learned using a set of stereo images and their corresponding true disparity maps. Once the network is trained, we may apply the given test stereo image pair as input to the network and infer the disparity map.
- Given left and right views of a scene, one may use single image depth estimation techniques to obtain two separate depth maps; one for each image and then obtain the disparity maps d_l and d_r for left and right images, respectively using the known camera calibration. These disparity maps are closed to the disparity map obtained using the stereo pair. These two slightly different disparity maps can then be used in a minimization framework by adding the following constraint to get the final disparity map d:

$$\sum_{(x,y)} (d_l(x,y) - d(x,y))^2 + (d_r(x,y) - d(x,y))^2.$$

- The proposed approaches for disparity estimation using stereo image pair can be extended to stereo video disparity estimation. This results in a video disparity or 3D disparity map which can be further used in the applications such as, 3D TV, 3D cinema, or telepresence.

References

[1] L. F. A. Jalobeanu and J. Zerubia. Estimation of adaptive parameters for satellite image deconvolution. In *Pattern Recognition, International Conference on*, volume 3, pages 318–321, 2000.

[2] F. Aaron and S. Stephen. Large occlusion stereo. *Computer Vision, International Journal of*, 33(3):181–200, September 1999.

[3] M. Aharon, M. Elad, and A. Bruckstein. K -SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *Signal Processing, IEEE Transactions on*, 54(11):4311–4322, November 2006.

[4] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *Computer Vision, International Journal of*, 2(3):283–310, 1989.

[5] A. Ansar, A. Castano, and L. H. Matthies. Enhanced real-time stereo using bilateral filtering. In *3D Data Processing, Visualization and Transmission, International Symposium on*, pages 455–462, 2004.

[6] N. Ayache and B. Faverjon. Efficient registration of stereo images by matching graph descriptions of edge segments. *Computer Vision, International Journal of*, 1(2):107–131, June 1987.

[7] E. T. Baek and Y. S. Ho. Cost aggregation with guided image filter and superpixel for stereo matching. In *Annual Summit and Conference, Asia-Pacific Signal and Information Processing Association*, pages 1–4, December 2016.

[8] S. Barnard. Stochastic stereo matching over scale. *Computer Vision, International Journal of*, 3(1):17–32, May 1989.

[9] Belhumeur and N. Peter. A bayesian approach to binocular steropsis. *Computer Vision, International Journal of*, 19(3):237–260, August 1996.

[10] R. Ben-Ari and N. Sochen. Stereo matching with mumford-shah regularization and occlusion handling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(11):2071–2084, November 2010.

[11] Y. Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, January 2009.

[12] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle. Greedy layer-wise training of deep networks. In *Advances in Neural Information Processing Systems 19*, pages 153–160. 2007.

[13] S. Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(4):401–406, April 1998.

[14] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Computer Vision, International Conference on*, pages 231–236, May 1993.

[15] M. J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *Computer Vision, International Journal of*, 19(1):57–91, 1996.

[16] M. Bleyer and M. Gelautz. A layered stereo algorithm using image segmentation and global visibility constraints. In *Image Processing, IEEE International Conference on*, volume 5, pages 2997–3000, October 2004.

[17] M. Bleyer, C. Rother, and P. Kohli. Surface stereo with soft segmentation. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 1570–1577, June 2010.

[18] M. Bleyer, C. Rother, P. Kohli, D. Scharstein, and S. Sinha. Object stereo - joint stereo matching and object segmentation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 3081–3088, June 2011.

[19] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, September 2004.

[20] Y. Boykov, O. Veksler, and R. Zabih. A variable window approach to early vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(12):1283–1294, December 1998.

[21] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, November 2001.

[22] S. G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *Image Processing, IEEE Transactions on*, 9(9):1532–1546, September 2000.

[23] D. Chen, M. Ardabilian, and L. Chen. Depth edge based trilateral filter method for stereo matching. In *Image Processing, IEEE International Conference on*, pages 2280–2284, September 2015.

[24] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *Scientific Computing, Society for Industrial and Applied Mathematics Journal on*, 43(1):129–159, January 2001.

[25] C. Cigla. Recursive edge-aware filters for stereo matching. In *Computer Vision and Pattern Recognition Workshops, IEEE Conference on*, pages 27–34, June 2015.

[26] S. D. Cochran and G. Medioni. 3-D surface description from binocular stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(10):981–994, October 1992.

[27] I. J. Cox, S. L. Hingorani, S. B. Rao, and B. M. Maggs. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, May 1996.

[28] G. Davis, S. Mallat, and M. Avellaneda. Adaptive greedy approximations. *Constructive Approximation*, 13(1):57–98, 1997.

[29] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 38(2):295–307, February 2016.

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

[31] K. Engan, S. O. Aase, and J. H. Husoy. Method of optimal directions for frame design. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, volume 5, pages 2443–2446, 1999.

[32] A. Fusiello, V. Roberto, and E. Trucco. Efficient stereo with multiple windowing. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 858–863, June 1997.

[33] P. P. Gajjar and M. V. Joshi. New learning based super-resolution: Use of DWT and IGMRF prior. *Image Processing, IEEE Transactions on*, 19(5):1201–1213, May 2010.

[34] D. Geiger and F. Girosi. Parallel and deterministic algorithms from MRFs: surface reconstruction. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(5):401–412, May 1991.

[35] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. *Computer Vision, International Journal of*, 14(3):211–226, 1995.

[36] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 6(6):721 –741, November 1984.

[37] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 580–587, June 2014.

[38] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. *Signal Processing, IEEE Transactions on*, 45(3):600–616, March 1997.

[39] W. E. L. Grimson. Computational experiments with a feature based stereo algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 7(1):17–34, January 1985.

[40] C. Hane, C. Zach, B. Zeisl, and M. Pollefeys. A patch prior for dense 3D reconstruction in man-made environments. In *3D Imaging, Modeling, Processing, Visualization Transmission, IEEE International Conference on*, pages 563–570, Oct 2012.

[41] M. Hannah. *Computer Matching of Areas in Stereo Images*. Ph.D. thesis, Stanford University, 1974.

[42] S. Hawe, M. Kleinsteuber, and K. Diepold. Dense disparity maps from sparse disparity measurements. In *Computer Vision, IEEE International Conference on*, pages 2126–2133, November 2011.

[43] P. Heise, S. Klose, B. Jensen, and A. Knoll. PM-Huber: Patchmatch with Huber regularization for stereo matching. In *Computer Vision, IEEE International Conference on*, pages 2360–2367, December 2013.

[44] G. E. Hinton and S. Osindero. A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554, July 2006.

[45] H. Hirschmuller. Stereo vision in structured environments by consistent semi-global matching. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 2, pages 2386–2393, 2006.

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

[47] H. Hirschmuller and D. Scharstein. Evaluation of cost functions for stereo matching. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, June 2007.

[48] L. Hong and G. Chen. Segment-based stereo matching using graph cuts. In *Computer Vision and Pattern Recognition, IEEE Conference on*, volume 1, pages I–74–I–81, June 2004.

[49] S. Hong, T. You, S. Kwak, and B. Han. Online tracking by learning discriminative saliency map with convolutional neural network. In *Machine Learning, International Conference on*, pages 597–606, 2015.

[50] A. Hosni, M. Bleyer, M. Gelautz, and C. Rhemann. Local stereo matching using geodesic support weights. In *Image Processing, IEEE International Conference on*, pages 2093–2096, November 2009.

[51] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(2):504–511, February 2013.

[52] H. Ishikawa and D. Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *Computer Vision, European Conference*, pages 232–248, 1998.

[53] V. Jain and S. Sebastian. Natural image denoising with convolutional networks. In *Advances in Neural Information Processing Systems 21*, pages 769–776. 2009.

[54] A. Jalobeanu, L. Blanc-Feraud, and J. Zerubia. An adaptive Gaussian model for satellite image deblurring. *Image Processing, IEEE Transactions on*, 13(4):613–621, April 2004.

[55] S. Ji, W. Xu, M. Yang, and K. Yu. 3D convolutional neural networks for human action recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):221–231, January 2013.

[56] D. G. Jones and J. Malik. Computational framework for determining stereo correspondence from a set of linear spatial filters. In *Computer Vision, European Conference on*, volume 588, pages 395–410, 1992.

[57] M. V. Joshi and A. Jalobeanu. MAP estimation for multiresolution fusion in remotely sensed images using an IGMRF prior model. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(3):1245–1255, March 2010.

[58] H. Y. Jung, K. M. Lee, and S. U. Lee. Stereo reconstruction using high order likelihood. In *Computer Vision, International Conference on*, pages 1211–1218, November 2011.

[59] I. L. Jung, T. Y. Chung, J. Y. Sim, and C. S. Kim. Consistent stereo matching under varying radiometric conditions. *Multimedia, IEEE Transactions on*, 15(1):56–69, January 2013.

[60] M. R. K. Jarrett, K. Kavukcuoglu and Y. LeCun. What is the best multi-stage architecture for object recognition? In *Computer Vision, IEEE International Conference on*, pages 2146–2153, 2009.

[61] T. Kanade and M. Okutomi. A stereo matching algorithm with an adaptive window: Theory and experiment. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(9):920–932, September 1994.

[62] S. B. Kang, R. Szeliski, and J. Chai. Handling occlusions in dense multi-view stereo. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 1, pages I–103–I–110, April 2001.

[63] J. Kim, V. Kolmogorov, and R. Zabih. Visual correspondence using energy minimization and mutual information. In *Computer Vision, IEEE International Conference on*, pages 1033–1040, October 2003.

[64] J. Kim, C. Liu, F. Sha, and K. Grauman. Deformable spatial pyramid matching for fast dense correspondences. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 2307–2314, June 2013.

[65] K. R. Kim and C. S. Kim. Adaptive smoothness constraints for efficient stereo matching using texture and edge information. In *Image Processing, IEEE International Conference on*, pages 3429–3433, September 2016.

[66] A. Klaus, M. Sormann, and K. Karner. Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In *Pattern Recognition, IEEE International Conference on*, volume 3, pages 15–18, August 2006.

[67] V. Kolmogorov and R. Zabih. Computing visual correspondence with occlusions using graph cuts. In *Computer Vision, IEEE International Conference on*, volume 2, pages 508–515, July 2001.

[68] V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In *Computer Vision, European Conference on*, pages 82–96, May 2002.

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

[70] D. Kong and H. Tao. A method for learning matching errors for stereo computation. In *British Machine Vision Conference*, pages 11.1–11.10, 2004.

[71] J. Kowalczuk, E. Psota, and L. C. Perez. Real-time stereo matching on CUDA using an iterative refinement method for adaptive support-weight correspondences. *Circuits System Video Technology, IEEE Transactions on*, 23(1):94–104, 2013.

[72] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. Lee, and T. J. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural Computing*, 15(2):349–396, February 2003.

[73] K. Kreutz-Delgado and B. D. Rao. FOCUSS - based dictionary learning algorithms, 2000.

[74] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012.

[75] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang. Multiview imaging and 3DTV. *IEEE Signal Processing Magazine*, 24(6):10–21, November 2007.

[76] Z. L. L. Wang and Z. Zhang. Feature based stereo matching using two-step expansion. *Mathematical Problems in Engineering*, 2014, December 2014.

[77] Y. LeCun. Learning invariant feature hierarchies. In *Computer Vision, European Conference on*, pages 496–505, October 2012.

[78] H. Lee, C. Ekanadham, and A. Y. Ng. Sparse deep belief net model for visual area V2. In *Neural Information Processing Systems*, pages 873–880, 2007.

[79] A. Li, D. Chen, Y. Liu, and Z. Yuan. Coordinating multiple disparity proposals for stereo computation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 4022–4030, June 2016.

[80] S. Z. Li. *Markov random field modeling in computer vision*. Springer-Verlag, 1995.

[81] Y. Lin, N. Lu, X. Lou, F. Zou, Y. Yao, and Z. Du. Matching cost filtering for dense stereo correspondence. *Mathematical Problems in Engineering*, 2013, 2013.

[82] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):978–994, May 2011.

[83] T. Liu, P. Zhang, and L. Luo. Dense stereo correspondence with contrast context histogram, segmentation-based two-pass aggregation and occlusion handling. In *Advances in Image and Video Technology, Pacific Rim Symposium*, pages 449–461, 2009.

[84] V. H. M. Sonka and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering, 2007.

[85] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *Image Processing, IEEE Transactions on*, 17(1):53–69, January 2008.

[86] S. G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on*, 41(12):3397–3415, December 1993.

[87] J. Marroquin, S. Mitter, and T. Poggio. Probabilistic solution of ill-posed problems in computational vision. *Journal of the American Statistical Association*, 82:76–89, 1987.

[88] E. Martinian, A. Behrens, J. Xin, and A. Vetro. View synthesis for multiview video compression. In *Picture Coding Symposium*, 2006.

[89] J. Masci, U. Meier, D. Cirecsan, and J. Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *Artificial Neural Networks, International Conference on*, pages 52–59, 2011.

[90] S. Mattoccia, S. Giardino, and A. Gambini. Accurate and efficient cost aggregation strategy for stereo correspondence based on approximated joint bilateral filtering. In *Computer Vision, Asian Conference on*, volume 5995, pages 371–380, 2009.

[91] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 4040–4048, June 2016.

[92] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang. On building an accurate stereo matching system on graphics hardware. In *Computer Vision Workshops, IEEE International Conference on*, pages 467–474, November 2011.

[93] T. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[94] M. G. Mozerov and J. V. Weijer. Accurate stereo matching by two-step energy minimization. *Image Processing, IEEE Transactions on*, 24(3):1153–1163, March 2015.

[95] K. Muhlmann, D. Maier, J. Hesser, and R. Manner. Calculating dense disparity maps from color stereo images, an efficient implementation. *Computer Vision, International Journal of*, 47(1-3):79–88, April 2002.

[96] G. Mussardo. *Statistical Field Theory: An Introduction to Exactly Solved Models in Statistical Physics*. Oxford Graduate Texts. OUP Oxford, 2009.

[97] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *Computer Vision, IEEE International Conference on*, pages 1520–1528, December 2015.

[98] M. Okutomi and T. Kanade. A locally adaptive window for signal matching. *Computer Vision, International Journal of*, 7(2):143–162, 1992.

[99] M. Okutomi and T. Kanade. A multiple-baseline stereo. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15:353–363, April 1993.

[100] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37(23):3311–3325, 1997.

[101] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, Asilomar Conference on*, pages 1–3, 1993.

[102] D. Pena and A. Sutherland. Non-parametric image transforms for sparse disparity maps. In *Machine Vision Applications, IAPR International Conference on*, pages 291–294, May 2015.

[103] C. Poultney, S. Chopra, and Y. Lecun. Efficient learning of sparse representations with an energy-based model. In *Advances in Neural Information Processing Systems*, 2006.

[104] E. T. Psota, J. Kowalczuk, M. Mittek, and L. C. Perez. MAP disparity estimation using hidden markov trees. In *Computer Vision, IEEE International Conference on*, pages 2219–2227, December 2015.

[105] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson. Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid. In *Computer Vision, European Conference on*, volume 6313, pages 510–523, September 2010.

[106] S. Roy. Stereo without epipolar lines: A maximum-flow formulation. *Computer Vision, International Journal of*, 34(2):147–161, August 1999.

[107] T. W. Ryan, R. T. Gray, and B. R. Hunt. Prediction of correlation errors in stereo-pair images. *Optical Engineering*, 19(3):193312, 1980.

[108] M. Sarkis and K. Diepold. Depth map compression via compressed sensing. In *Image Processing, IEEE International Conference on*, pages 737–740, November 2009.

[109] A. Saxena, S. H. Chung, and A. Y. Ng. 3-D depth reconstruction from a single still image. *Computer Vision, International Journal of*, 76:53–69, January 2008.

[110] D. Scharstein. Matching images by comparing their gradient fields. In *Pattern Recognition, IEEE International Conference on*, volume 1, pages 572–575, October 1994.

[111] D. Scharstein and C. Pal. Learning conditional random fields for stereo. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, June 2007.

[112] D. Scharstein and R. Szeliski. Stereo matching with non-linear diffusion. *Computer Vision, International Journal of*, 28:155–174, 1998.

[113] D. Scharstein, R. Szeliski, and R. Zabih. Middlebury stereo. http://vision.middlebury.edu/stereo.

[114] D. Scharstein, R. Szeliski, and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Computer Vision, International Journal of*, 47(1):7–42, April 2002.

[115] S. N. Sinha, D. Scharstein, and R. Szeliski. Efficient high-resolution stereo matching using local plane sweeps. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1582–1589, June 2014.

[116] J. Sun, Y. Li, S. B. Kang, and H. Y. Shum. Symmetric stereo matching for occlusion handling. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 2, pages 399–406, June 2005.

[117] J. Sun, N. N. Zheng, and H. Y. Shum. Stereo matching using belief propagation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(7):787–800, July 2003.

[118] Y. Taguchi, B. Wilburn, and C. L. Zitnick. Stereo reconstruction with mixed pixels using adaptive over-segmentation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, June 2008.

[119] H. Tao, H. S. Sawhney, and R. Kumar. A global matching framework for stereo computation. In *Computer Vision, IEEE International Conference on*, volume 1, pages 532–539, 2001.

[120] M. F. Tappen and W. T. Freeman. Comparison of graph cuts with belief propagation for stereo using identical MRF parameters. In *Computer Vision, IEEE International Conference on*, volume 2, pages 900–906, October 2003.

[121] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 8(4):413–424, July 1986.

[122] F. Tombari, S. Mattoccia, and L. Stefano. Segmentation-based adaptive support for accurate stereo correspondence. In *Advances in Image and Video Technology, Pacific Rim Symposium*, pages 427–438, 2007.

[123] I. Tosic, B. Olshausen, and B. Culpepper. Learning sparse representations of depth. *Selected Topics in Signal Processing, IEEE Journal of*, 5(5):941–952, September 2011.

[124] J. A. Tropp. Greed is good: algorithmic results for sparse approximation. *Information Theory, IEEE Transactions on*, 50(10):2231–2242, October 2004.

[125] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.

[126] O. Veksler. Stereo correspondence with compact windows via minimum ratio cycle. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(12):1654–1660, December 2002.

[127] O. Veksler. Fast variable window for stereo correspondence using integral images. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 1, pages 556–561, June 2003.

[128] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research*, 11:3371–3408, December 2010.

[129] L. Wang, S. Kang, and H. Shum. Cooperative segmentation and stereo using perspective space search. In *Computer Vision, Asian Conference on*, volume 1, pages 366–371, 2004.

[130] L. Wang, M. Liao, M. Gong, R. Yang, and D. Nister. High-quality real-time stereo using adaptive cost aggregation and dynamic programming. In *3D Data Processing, Visualization, and Transmission, International Symposium on*, pages 798–805, June 2006.

[131] L. Wang and R. Yang. Global stereo matching leveraged by sparse ground control points. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 3033–3040, June 2011.

[132] J. Witt and U. Weltin. Sparse stereo by edge-based search using dynamic programming. In *Pattern Recognition, International Conference on*, pages 3631–3635, November 2012.

[133] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon. Global stereo reconstruction under second order smoothness priors. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, June 2008.

[134] J. Xie, L. Xu, and E. Chen. Image denoising and inpainting with deep neural networks. In *Advances in Neural Information Processing Systems 25*, pages 350–358. 2012.

[135] Y. Xu, D. Wang, T. Feng, and H. Shum. Stereo computation using radial adaptive windows. In *Object recognition supported by user interaction for service robots*, volume 3, pages 595–598, 2002.

[136] K. Yamaguchi, T. Hazan, D. McAllester, and R. Urtasun. Continuous markov random fields for robust stereo estimation. In *Computer Vision, European Conference on*, pages 45–58, October 2012.

[137] K. Yamaguchi, D. McAllester, and R. Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *Computer Vision, European Conference on*, pages 756–771, September 2014.

[138] Q. Yang. A non-local cost aggregation method for stereo matching. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1402–1409, June 2012.

[139] Q. Yang. Recursive bilateral filtering. In *Computer Vision, European Conference on*, pages 399–413, 2012.

[140] Q. Yang. Hardware-efficient bilateral filtering for stereo matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 36(5):1026–1032, May 2014.

[141] Q. Yang, L. Wang, and N. Ahuja. A constant-space belief propagation algorithm for stereo matching. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 1458–1465, June 2010.

[142] Q. Yang, L. Wang, and R. Yang. Real-time global stereo matching using hierarchical belief propagation. In *British Machine Vision Conference*, pages 101.1–101.10, 2006.

[143] Q. Yang, L. Wang, R. Yang, H. Stewenius, and D. Nister. Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(3):492–504, March 2009.

[144] K. Yoon and I. S. Kweon. Adaptive support-weight approach for correspondence search. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(4):650–656, April 2006.

[145] K. J. Yoon and I. S. Kweon. Stereo matching with symmetric cost functions. In *Computer Vision and Pattern Recognition, IEEE Conference on*, volume 2, pages 2371–2377, June 2006.

[146] T. Yu, R. S. Lin, B. Super, and B. Tang. Efficient message representations for belief propagation. In *Computer Vision, IEEE International Conference on*, pages 1–8, October 2007.

[147] L. Yunpeng and D. Huttenlocher. Learning for stereo vision using the structured support vector machine. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1–8, June 2008.

[148] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *Computer Vision, European Conference on*, pages 151–158, 1994.

[149] J. Zbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1592–1599, June 2015.

[150] M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus. Deconvolutional networks. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 2528–2535, June 2010.

[151] M. D. Zeiler, G. W. Taylor, and R. Fergus. Adaptive deconvolutional networks for mid and high level feature learning. In *Computer Vision, IEEE International Conference on*, pages 2018–2025, November 2011.

[152] C. Zhang, Z. Li, Y. Cheng, R. Cai, H. Chao, and Y. Rui. Meshstereo: A global stereo model with mesh alignment regularization for view interpolation. In *Computer Vision, IEEE International Conference on*, pages 2057–2065, December 2015.

[153] C. Zhang, C. Shen, and T. Shen. Unsupervised feature learning for dense correspondences across scenes. *Computer Vision, International Journal of*, 116(1):90–107, 2016.

[154] K. Zhang, J. Lu, and G. Lafruit. Cross-based local stereo matching using orthogonal integral images. *Circuits and Systems for Video Technology, IEEE Transactions on*, 19(7):1073–1079, July 2009.

[155] L. Zhang and S. M. Seitz. Parameter estimation for MRF stereo. In *Computer Vision and Pattern Recognition, IEEE Conference on*, volume 2, pages 288–295, June 2005.

[156] G. Zheng, S. Xianyu, L. Yuankun, and Z. Qican. Local stereo matching with adaptive support-weight, rank transform and disparity calibration. *Pattern Recognition Letters*, 29(9):1230–1235, 2008.

List of Publications

- **Journal:**

  1. S. Nahar and M. V. Joshi, "A Learned Sparseness and IGMRF based Regularization Framework for Dense Disparity Estimation using Unsupervised Feature Learning", in *IPSJ Transactions on Computer Vision and Applications*, 9(1):1-15, February 2017.

- **Conferences:**

  1. S. Nahar and M. V. Joshi, "Dense Disparity Estimation Based on Feature Matching and IGMRF Regularization", in *23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, December, 4-8, 2016, pp. 3804-3809.
  2. S. Nahar and M. V. Joshi, "A Regularization Framework for Stereo Matching using IGMRF Prior and Sparseness Learned from Autoencoder", in *23rd IEEE International Conference on Image Processing (ICIP)*, Phoenix, Arizona, USA, September, 25-28, 2016, pp. 3434-3438.
  3. S. Nahar and M. V. Joshi, "A Learned Over Complete Sparseness and IGMRF based Regularization Framework for Dense Disparity Estimation", in *3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, Kuala-Lumpur, Malaysia, November, 3-6, 2015, pp. 306-310.
  4. S. Nahar and M. V. Joshi, "A Learning Based Approach for Dense Stereo Matching with IGMRF Prior", in *4th National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*, Jodhpur, India, December, 18-21, 2013, pp. 1-4.
