ESTIMATION OF DISPARITY MAP FROM STEREO IMAGE PAIRS IN PRESENCE OF OCCLUSION

A

Thesis Submitted

in Partial Fulfilment of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

By

MALATHI. T

Department of Electronics and Electrical Engineering

Indian Institute of Technology Guwahati

Guwahati, India.

February, 2017

Certificate

This is to certify that the thesis entitled "**Estimation of Disparity Map from Stereo Image Pairs in Presence of Occlusion**", submitted by **MALATHI. T** (10610223), a research scholar in the *Department of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati*, for the award of the degree of **Doctor of Philosophy**, has been carried out by her under my supervision and guidance. The thesis has fulfilled all requirements as per the regulations of the institute and in my opinion has reached the standard needed for submission. The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

Dated:

Guwahati.

Dr. M.K. Bhuyan
Department of Electronics and Electrical Engineering
Indian Institute of Technology Guwahati
India- 781039.

This research work is dedicated to

MY FATHER

&

MY BROTHER

Acknowledgements

I would like to express my sincere thanks to all those people who made this dissertation possible.

First and foremost, I would like to express my profound respect and gratitude to my supervisor, Dr. M.K. Bhuyan, who has been the guiding force behind this work. I am greatly indebted for his invaluable advises, constant encouragement, and his valuable comments on my work. I am fortunate enough to have such an adviser who gave me the freedom to think independently and explore new ideas. His patience and support helped me to overcome many crisis situations and successfully complete this dissertation. I would particularly thank him for the patience he has shown in carefully reading and commenting on the manuscripts, and countless revisions of this dissertation. His commitments and dedication to research have been and will continue to be a constant source of inspiration for me. I have no doubts that finishing my degree in proper and timely manner was impossible without his help. I am highly privilege to have got an opportunity to work with such a wonderful person.

I would also like to thank my doctoral committee members Prof. P. K. Bora, Dr. Kannan Karthik, and Dr. Arijit Sur for their moral support, thorough evaluations, and suggestions that helped me to improve my research work. I am also thankful to the Head of the Department, other faculty members, and staffs for their kind help carried out during my academic studies. I specially thank Mr. Sanjib Das for providing various resources useful for the research work.

I am grateful to my husband Mr. K. Dakshina Murthy for his love, sacrifices, and motivation for the successful completion of this research work.

My thanks to my friend, who encouraged me to purse PhD and helped me to join in IIT Guwahati. On a personal note, I would like to thank my friend Mr. Sunil Kumar for many thought provoking discussions. I thank you for always being there for me whenever I needed help and moral support. I am thankful to you for the patience you have shown in commenting my thesis.

Thanks go out to my friends Mr. Gnana Praveen, Ms. SriRanjani, Mr. Santhosh Kumar Yadav, Mr. Parveen Malik, Mr. Gaurav Kumar Yadav, Mr. Abishek R Vahadane, Ms. Tanima Dutta, Ms. Dixcy Jaba Sheeba, Ms. Sharmila, Ms. Padam Priyal, Mr. Amit Vishwakarma, Mr. Arghya Chakravarty, Mr. Vinoth, Mr. Mathan Kumar, Mr. Vijay Krishna, and Mr. Biplab Ketan Chakraborty for motivating me. I had a great time with many friends at IIT Guwahati, including (but not limited to) Sindhujarani, Shanmuga Priya, Kohila, Sumitha Banu, Padmavathi, Vanitha, Monisha Javadi, and Umesh. I thank them for their support and encouragement.

I am grateful to my parents, sisters, and brothers whose love and encouragement made this research possible.

I am thankful to IIT Guwahati for providing the research scholarship to undertake my PhD research. Finally, I would like to thank the Almighty God for bestowing me this opportunity and showering his blessings on me to come out successful against all odds.

Malathi. T

Abstract

Stereo correspondence finds corresponding matching pixels in the stereo image pairs. The difference between the coordinates of these matching pixels gives the disparity value, which in turn can be used for finding the depth information of a scene.

In last few decades, a number of stereo matching methods have been proposed for different Computer Vision applications, such as robot navigation, 3D modelling, object detection, tracking etc. In view of this, a new Gabor feature-based stereo matching method is proposed. Our method mainly has four steps, namely matching cost computation, cost aggregation, disparity map computation using Winner-Take-All (WTA) selection, and finally disparity map refinement. In our proposed method, local features extracted by using Gabor wavelet in spatial domain are used for matching cost computation. Gabor wavelet can extract texture information from an image, which is very similar to the information perceived by a human visual system. Subsequently, Kuwahara filter is employed for cost aggregation to smooth the estimated disparity map by preserving the disparity discontinuities.

The estimation of a fine disparity map is quite challenging in presence of occlusion. For this, a novel occlusion detection method is proposed by only using a single disparity map instead of two disparity maps employed in existing methods. In our method, a linear mapping function is employed by observing the characteristics of the pixels in the reference image and their corresponding matching pixels in the target image. This mapping function is subsequently modelled by linear regression. Finally, a novel occlusion filling method is proposed to get a fine disparity map. For this, a disparity value of a neighbouring non-occluded pixel is assigned to a selected occluded pixel. The colour similarity of a selected occluded pixel with a set of neighbouring non-occluded pixels is used to select a neighbouring non-occluded pixel for filling. This colour similarity score is calculated by using the support weights of both left and right images. Experimental results demonstrate that our proposed disparity map estimation algorithm can give a fine disparity map in presence of occlusion, which is suitable for many applications.

Additionally, the accuracy of proposed Gabor wavelet features in representing an image is experimentally studied for different Gabor wavelet parameters. Also, the behaviour of the Gabor features are analyzed for radiometric variations.

Contents

List of Figures

List of Tables

List of Acronyms

AD	Absolute Difference
ANCC	Adaptive Normalized Cross Correlation
ASW	Adaptive Support Weight
CC	Cross Correlation
DSI	Disparity Space Images
GC	Graph Cut
GF	Guided Filter
GGWF	Global Gabor Wavelet Feature
GW	Gabor Wavelet
GWFV	Gabor Wavelet Feature Vector
LGWF	Local Gabor Wavelet Feature
LNDA	Left Right consistency Check (LRC) and Neighbors Disparity Assignment (NDA)
LRC	Left Right consistency Check
MGJ	Match Goodness Jumps
MSE	Mean Square Error
NCC	Normalized Cross Correlation
NDA	Neighbors Disparity Assignment
OCC	Occlusion Constraint
ORD	Ordering Constraint
PCA	Principal Component Analysis
SAD	Sum-of-Absolute-Differences
SD	Squared Difference
SLS	Segmentation-based Least Squares
SSD	Sum-of-Squared-Differences
SSI	Structural Similarity Index
UQI or QI	Universal Quality Index
WLS	Weighted Least Squares
WTA	Winner-Take-All

List of Symbols

R^2	2D projective space
R^3	3D projective space
O_w	Origin of world coordinate system
O_c	Origin of camera coordinate system
O_i	Origin of image coordinate system
P	Three-dimensional scene point
O_L	Left camera center
O_R	Right camera center
I_l	Left image
I_r	Right image
e_l	Left epipole point
e_r	Right epipole point
$\mathbf{p} = (p_1, p_2)$	Pixel in left image
$\mathbf{p}' = (p_1', p_2')$	Matching pixel of pixel $\mathbf{p}$ in right image
D	Disparity map
d_p	Disparity value of pixel $\mathbf{p}$
$\mathcal{N}_p$	Support window or neighborhood region of pixel $\mathbf{p}$
$\boldsymbol{\pi}_i$	Image plane

1

Introduction

Computer vision is a vast field in itself, encompassing a wide variety of applications which deals with the acquisition of the real world scene, processing, analyzing, and finally understanding of the captured images for a final decision. For image acquisition, a real world scene is projected onto a two-dimensional image plane. Projection of three-dimensional scenery to two-dimensional image will lose disparity/depth information of a real three-dimensional scene. Reconstruction of the original three-dimensional scene is done using disparity information. This reconstruction of a scene is required in many computer vision applications. This chapter gives an overview of the fundamental concept of image formation in a single camera-based setup. Subsequently, geometry of image formation in a stereo vision setup is described. In a stereo vision setup, depth information of a scene can be obtained by estimating a disparity map from the two stereo images. The major challenges of obtaining an accurate disparity map are also briefly discussed in this chapter.

1.1 Image Formation Models

Two-dimensional image is formed as a combination of a light source and the rays of light reflected or observed by the object present in the scene being captured. The reflected rays of light enter the camera through an aperture. An image is formed when the reflected rays strike the image plane. Mapping of a three-dimensional scene to a two-dimensional image plane is known as perspective projection. Orthographic projection is another way of obtaining images of a three-dimensional scene. In this, a set of parallel light rays perpendicular to the image plane form an image. Based on applications, images can be captured using single or multiple cameras.

1.1.1 Image formation in a single camera-based setup

The concept of two-dimensional image formation of a three-dimensional real world scene can be explained with the help of a basic camera model. This simplest imaging system comprises of pinhole camera and an image plane. This setup is similar to a human visual system. The pinhole camera lies in-between the observed world scene and the image plane. Any ray reflected from the surface of a scene passes through the pinhole and impinges on the image plane. Therefore, there is a corresponding area in the real world for each of the areas in the image. Thus, an image formation process is a linear transformation from the three-dimensional projective space $\mathcal{P}^3$ to the two-dimensional projective space $\mathcal{P}^2$. The geometry of this imaging system is shown in Figure 1.1. Here, the real world scene is projected to the image plane $\boldsymbol{\pi}_i$. The optical axis is shown by a vertical dotted line. The pinhole camera is located at the focal point $\mathbf{f}_p$, which is perpendicular to the optical axis. Focal length f is the distance from the lens to the point where the light rays converge to form the image of the observed object [8]. The concept of image formation can be explained with the help of four coordinate systems namely:

- World Euclidean coordinate system (Blue colour): point - $\mathbf{x}_w = [X_w, Y_w, Z_w]^T$, Origin - $\mathbf{O}_w$.
- Camera Euclidean coordinate system (Red colour): point - $\mathbf{x}_c = [X_c, Y_c, Z_c]^T$, Origin - $\mathbf{O}_c$. The origin of this coordinate system is positioned at the focal point $\mathbf{f}_p$.
- Image Euclidean coordinate system (Pink colour): point - $\mathbf{x}_i = [X_i, Y_i, Z_i]^T$, Origin - $\mathbf{O}_i$.
- Image affine coordinate system (Green colour): point - $\tilde{\mathbf{u}} = [u_a, v_a, w_a]^T$.

Image affine coordinate system is the sheared and rescaled representation of image Euclidean coordinate system.

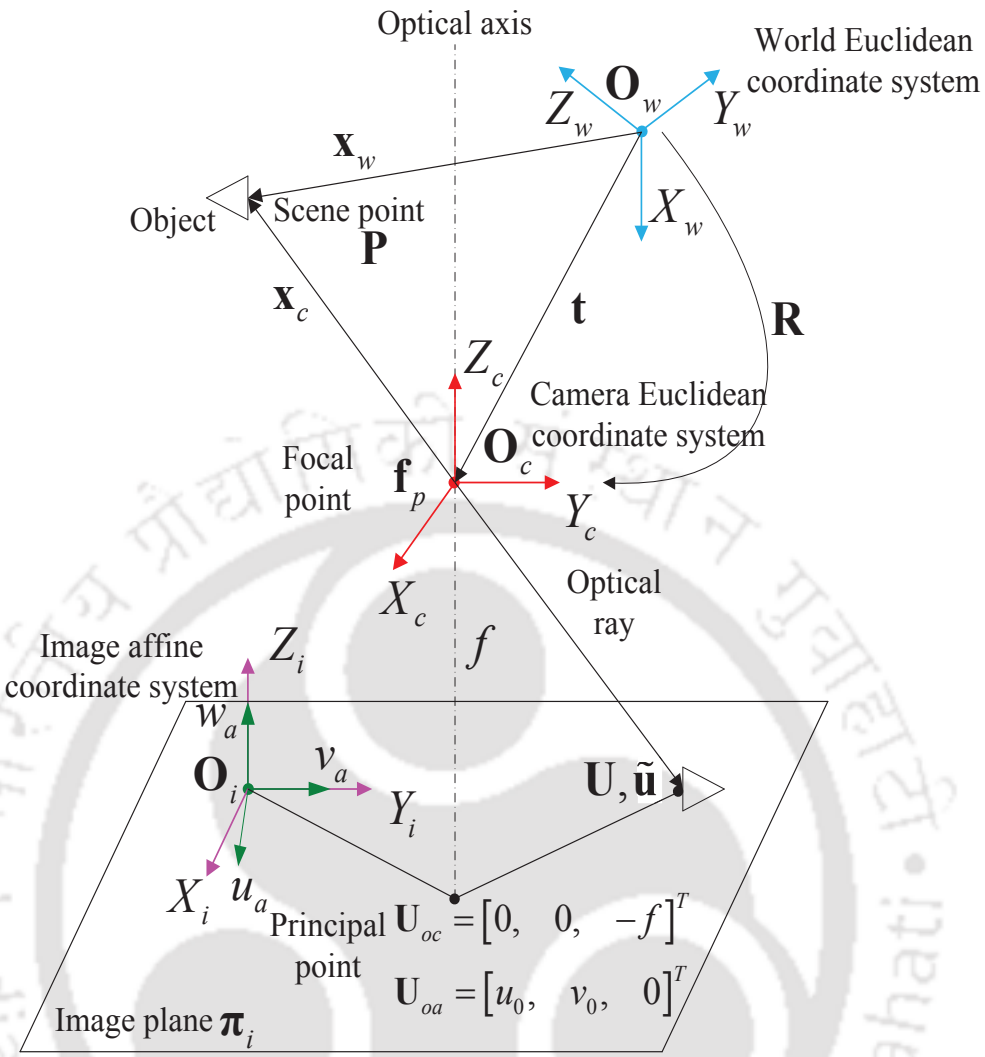


Figure 1.1: The geometry of a linear perspective camera system [1].

The projection of three-dimensional space to two-dimensional image plane can be factorized into three simpler transformations. Each transformation is a transition from one of the coordinate systems to another listed as follows:

(i) The transformation of the point $\mathbf{x}_w$ to $\mathbf{x}_c$ is performed by translating it by the vector $\mathbf{t}$, and subsequently rotating by the matrix $\mathbf{R}$, expressed as follows:

$$\mathbf{x}_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R}(\mathbf{x}_w - \mathbf{t}) \tag{1.1}$$

The rotation matrix $\mathbf{R}$ gives the rotation along the three axes X_w, Y_w, and Z_w, while vector $\mathbf{t}$

gives translation undergone by these axes. Thus, these parameters *i.e.,* rotation matrix **R** and translation vector **t** are called the camera extrinsic calibration parameters.

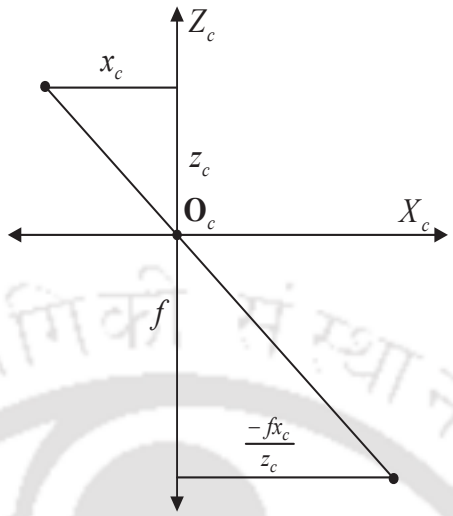


Figure 1.2: Calculation of the x-coordinate of the projected point in the image plane by using properties of similar triangles [1].

(ii) The point $\mathbf{x}_c$ projected on the image plane is represented by $\mathbf{x}_i$. The coordinates of this point can be obtained by taking into consideration the similar triangles shown in Figure 1.2, and can be written as follows:

$$\mathbf{x}_i = \begin{bmatrix} \frac{-fx_c}{z_c} & \frac{-fy_c}{z_c} & -f \end{bmatrix}^T \tag{1.2}$$

(iii) Now, we determine the projected point $\mathbf{x}_i$ in image affine coordinate system. The vertical optical axis intersects the image plane at the point $\mathbf{U}_0$. This point in image affine coordinate system is expressed as:

$$\mathbf{U}_{0a} = \begin{bmatrix} u_0 & v_0 & 0 \end{bmatrix}^T \tag{1.3}$$

The principal point $\mathbf{U}_0$ in the image plane expressed in homogenous coordinates is denoted as $\tilde{\mathbf{u}}$, which is given by the equation:

$$\tilde{\mathbf{u}} = \begin{bmatrix} U \\ V \\ W \end{bmatrix} = \begin{bmatrix} a_s & b_s & -u_0 \\ 0 & c_s & -v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{-fx_c}{z_c} \\ \frac{-fy_c}{z_c} \\ 1 \end{bmatrix} = \begin{bmatrix} -fa_s & -fb_s & -u_0 \\ 0 & -fc_s & -v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{x_c}{z_c} \\ \frac{y_c}{z_c} \\ 1 \end{bmatrix} \tag{1.4}$$

Here, a_s, b_s and c_s are the shear along the coordinate axes, and u_0 and v_0 are the affine coordi-

nates of the point $\mathbf{U}_0$ in the image plane. So,

$$\begin{aligned}\tilde{\mathbf{u}} &= \begin{bmatrix} -fa_s & -fb_s & -u_0 \\ 0 & -fc_s & -v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \\ &= \begin{bmatrix} -fa_s & -fb_s & -u_0 \\ 0 & -fc_s & -v_0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{R}(\mathbf{x}_w - \mathbf{t}) = \mathbf{KR}(\mathbf{x}_w - \mathbf{t})\end{aligned} \tag{1.5}$$

The upper triangular matrix $\mathbf{K}$ is called the intrinsic camera calibration matrix. Expressing scene point $\mathbf{P}$ in homogenous coordinates $\tilde{\mathbf{x}}_w = [\ \mathbf{x}_w \quad 1\ ]^T$, the above Equation (1.5) is written as:

$$\tilde{\mathbf{u}} = [\mathbf{KR}| - \mathbf{KRt}] \begin{bmatrix} \mathbf{x}_w \\ 1 \end{bmatrix} = \mathbf{M} \begin{bmatrix} \mathbf{x}_w \\ 1 \end{bmatrix} = \mathbf{M}\tilde{\mathbf{x}}_w \tag{1.6}$$

where $\tilde{\mathbf{x}}_w$ is the three-dimensional scene point in the homogeneous coordinate. The delimiter symbol "|" denotes that the matrix is composed of two submatrices. The leftmost 3×3 matrix gives the rotation information, while the rightmost column gives the translation details. The matrix $\mathbf{M}$ is called the projective matrix or camera matrix. In simple words, the three-dimensional point can be projected onto the image plane by multiplying this point with the camera matrix.

One major disadvantage of single camera-based imaging system is limited field-of-view. One such application is object detection and tracking, which requires a large field-of-view. Furthermore, image which is acquired using single camera is the two-dimensional projection of a three-dimensional scene, and the depth information is lost in this imaging process. On the other hand, recovering three-dimensional information from the images captured by single camera is an ill-posed problem.

1.1.2 Image formation in a stereo vision setup

In this section, we will briefly discuss the fundamental concept of image formation of a three-dimensional scene when captured by two cameras at distinct viewpoints or positions. This setup has two image planes as shown in Figure 1.3. The two pinhole cameras or the camera centers are denoted by $\mathbf{O}_l$ (left) and $\mathbf{O}_r$ (right) respectively. The three-dimensional point $\mathbf{P}_w$ when viewed through the two cameras is projected at $\mathbf{p} = (p_1, p_2)$ in the left image, and at $\mathbf{p}' = (p_1', p_2')$ in the right image.

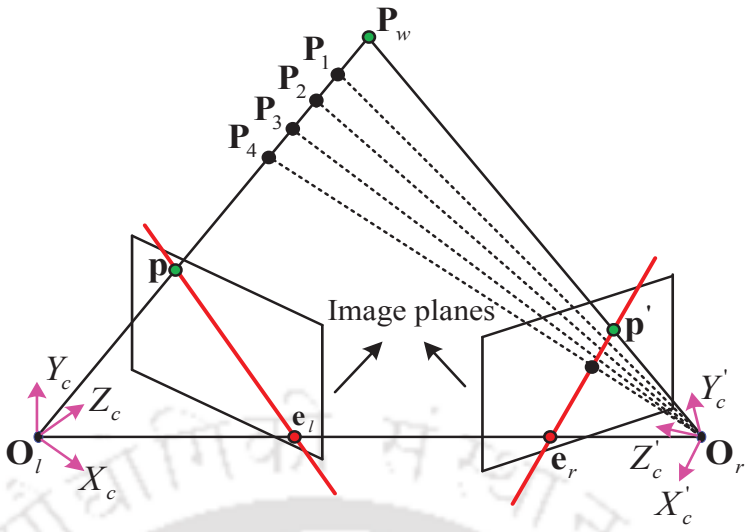


Figure 1.3: Image formation in a stereo vision setup (Epipolar geometry).

1.1.2.1 Epipolar geometry

Figure 1.3 shows an imaging configuration having two cameras. The projection of the camera centers on the other image plane are known as epipoles, which are denoted by $\mathbf{e}_l$ and $\mathbf{e}_r$ [1]. Both the epipoles and the camera centers lie on a straight line termed as the baseline line. Baseline intersects the left and right image planes at the epipoles $\mathbf{e}_l$ and $\mathbf{e}_r$ respectively. The point $\mathbf{P}_w$ and camera centers $\mathbf{O}_l$ and $\mathbf{O}_r$ form a plane, which is termed as epipolar plane. Left camera sees the line $\mathbf{O}_l\mathbf{P}_w$ as a point since it lies in the same line with the camera center. This means that the point $\mathbf{P}_w$ can lie anywhere in this line. In other words, all the points in the line $\mathbf{O}_l\mathbf{P}_w$ are projected on the same point $\mathbf{p}$ in the left image plane. On the other hand, the line $\mathbf{O}_l\mathbf{P}_w$ is seen as a line $\mathbf{e}_l\mathbf{p}^{'}$ in the right plane. Similarly, the line $\mathbf{O}_r\mathbf{P}_w$ is seen as a point $\mathbf{p}^{'}$ by the right camera, whereas this line is seen as a line $\mathbf{e}_l\mathbf{p}$ in the left image plane. These two lines $\mathbf{e}_r\mathbf{p}^{'}$ and $\mathbf{e}_l\mathbf{p}$ are termed the epipolar lines. Epipolar plane intersects the image planes at the epipolar lines. A set of epipolar lines exist in the image planes as the point $\mathbf{P}_w$ is allowed to vary over the three-dimensional scene. As the line $\mathbf{O}_r\mathbf{P}_w$ passes through the right camera center, the corresponding epipolar line must pass through the epipole $\mathbf{e}_l$ in the left image plane. Similarly, the line $\mathbf{O}_l\mathbf{P}_w$ passes through the left camera center, and hence its corresponding epipolar line must pass through the epipole $\mathbf{e}_r$ in the right image plane. With the knowledge of epipolar geometry, one can establish the correspondence between the two images with the help of a fundamental matrix $\mathbf{F}$, which is a 3×3 matrix. Mathematically, the mapping of a point $\mathbf{p}$ in one image to its corresponding matching point in the epipolar line $\mathbf{e}$ of the other image can be

written as:

$$\mathbf{Fp} = \mathbf{e} \tag{1.7}$$

The mapping of points in one image plane to their corresponding point on the epipolar lines in the other image can be performed with the help of epipolar geometry (epipolar constraint). Matching points can be easily computed if the camera positions and their corresponding image planes are known in a global coordinate system. This is called as a fully calibrated stereo setup, and the coordinates of the three-dimensional points can be computed from the coordinates of its projected two-dimensional points in the images. In the case where the two cameras in the epipolar geometry are non-parallel (*i.e.,* non-parallel optical axes), it is difficult to find the matching points for two independent image coordinates. For simplification, a simple stereo epipolar geometry is used. One way to achieve this simple geometry is to adjust both the cameras in such a way so that they become parallel. This is achieved with the help of rectification.

1.1.2.2 Rectification

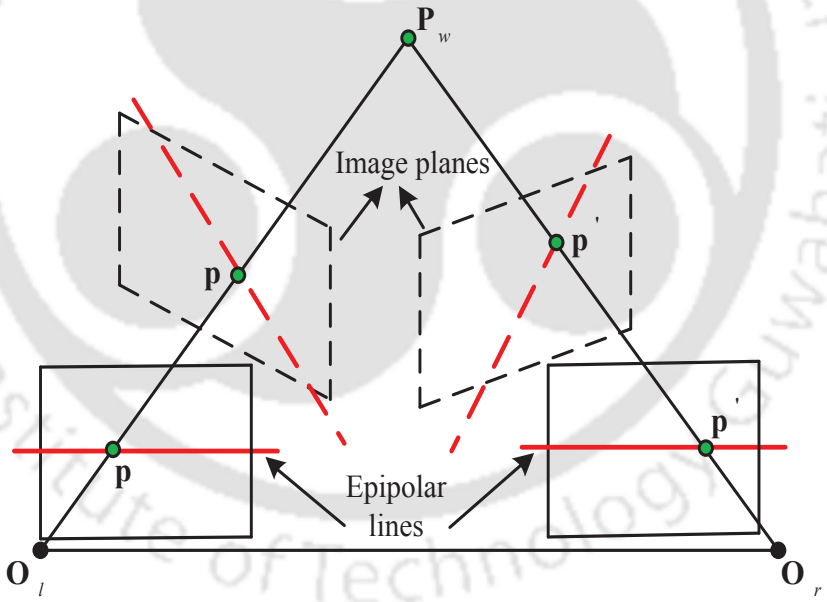


Figure 1.4: Stereo images rectification.

Rectification refers to a transformation process that reprojects both the left and right images onto a common image plane parallel to the baseline as shown in Figure 1.4. In Figure 1.4, image plane shown by dotted and bold lines refer to image planes before and after rectification respectively. In this case, the cameras are parallel, and consequently the axes are parallel to the baseline. So, the corresponding

epipolar lines would be horizontal *i.e.,* they have same y-coordinate. This stereo imaging setup is called as standard or canonical stereo setup [8]. The canonical stereo setup makes stereo matching problem much easier. This is because of the fact that the search can be done along the horizontal line in the rectified images instead of searching the entire image for finding the matching points. Rectification reduces the dimensionality of the search space for matching points from two-dimensional space to one-dimensional space. This can be done with the help of a 3×3 homography matrix $\mathbf{H}$. Mathematically, transforming the coordinates of the original image plane to a common image plane can be written as follow:

$$\begin{bmatrix} U' \\ V' \\ W' \end{bmatrix} = \mathbf{H} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{1.8}$$

Figure 1.5 shows the stereo images before and after rectification. Top row of the image shows the

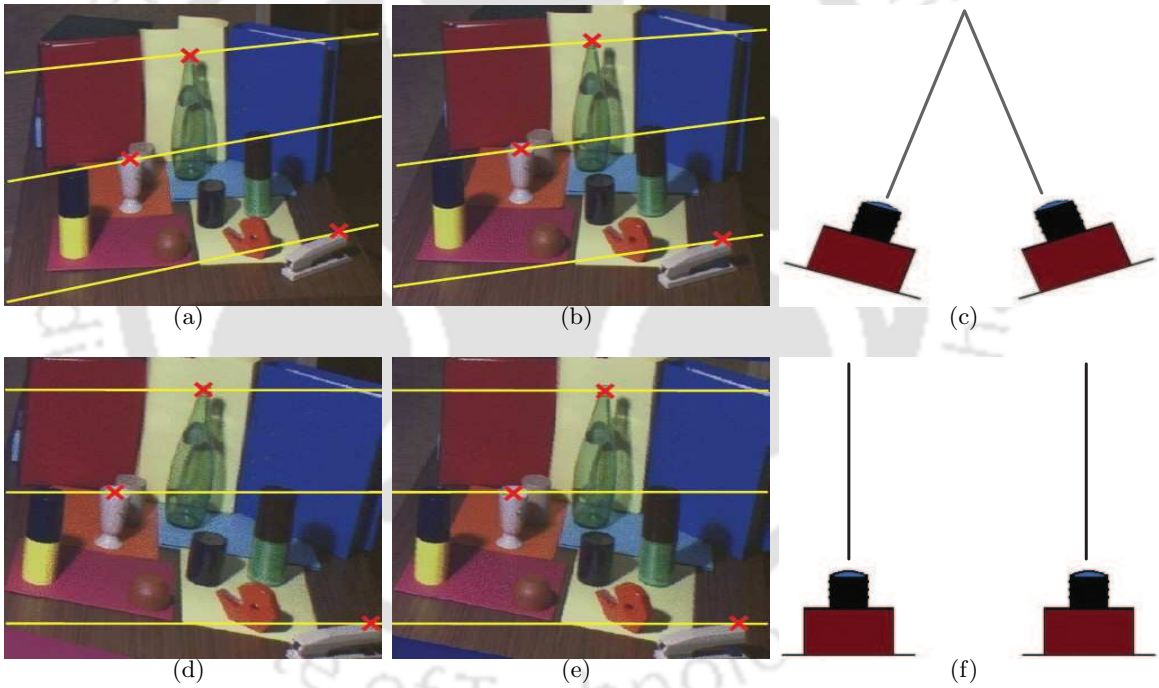


Figure 1.5: Stereo images before and after rectification. (a) Reference image before rectification; (b) Target image before rectification; (c) General stereo vision setup; (d) Reference image after rectification; (e) Target image after rectification; (f) Stereo vision setup after rectification [2].

stereo images before rectification and its corresponding camera positions. The rectified stereo images and its corresponding camera positions are shown in the bottom row.

1.1.2.3 Triangulation

Triangulation is a process of obtaining the three-dimensional scene point from its projected points in the two images planes. Let us consider the rectified stereo configuration shown in Figure 1.6. In this figure, the two parallel optical axes are separated by the baseline of length $b = 2h$. The real world point **P** is projected on the image planes at **U** and $\mathbf{U}^{'}$ respectively. The z-axis of the coordinate gives the distance from the cameras located at the point $z = 0$, whereas $x - axis$ gives the horizontal distance (y-coordinate is not shown in Figure 1.6), and the point $x = 0$ is midway between the cameras. Disparity is the horizontal distance between the points **U** and $\mathbf{U}^{'}$ (difference between their x-coordinates), which is given by:

$$d = u - u' \tag{1.9}$$

From the similar right-angled triangle **UCB** and **CPA** in Figure 1.6, we can write:

$$\begin{aligned} \frac{u}{f} &= -\frac{h+x}{z} \\ \frac{u'}{f} &= \frac{h-x}{z} \end{aligned} \tag{1.10}$$

After simplifying Equation (1.10), we can obtain the expression for z as follows:

$$z = \frac{2hf}{u' - u} = \frac{bf}{u' - u} = \frac{bf}{d} \tag{1.11}$$

Zero disparity suggests that the point **P** is at a far distance *i.e.,* infinite distance from the observer. The remaining coordinates of the considered three-dimensional point can be obtained as:

$$x = -\frac{b(u+u')}{2d}; y = \frac{bv}{d} \tag{1.12}$$

1.1.2.4 Relation between depth information and disparity value

Several practical applications require the position of the objects in the real world environment. In stereo vision, two cameras at different viewpoints are employed to acquire the images of a world scene. So, the task is to find the real world points from the given stereo image pairs. Additionally, shape and appearance of the objects can be determined from the stereo image pairs. The underlying concept behind the computation of the above information is the disparity map (matching point) computation. For the three-dimensional world point in one image, its corresponding matching points in the other

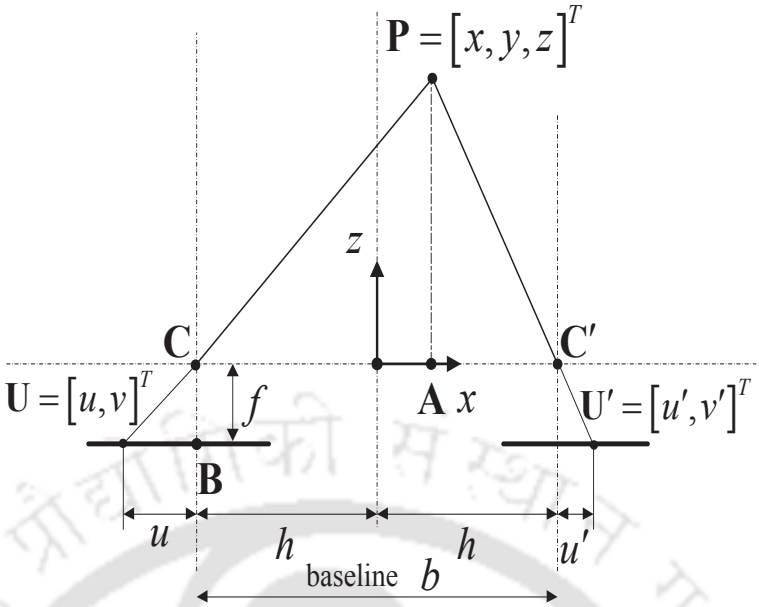


Figure 1.6: Elementary stereo geometry in the rectified configuration [1].

image can be found out. The horizontal displacement of these matching points is the disparity value for that point. Disparity values computed for all the pixels in the image give disparity map. The obtained disparity map along with the camera parameters can be used to find the depth map as described in Section 1.1.2.3. This map gives the distance of the world point from the camera.

1.1.2.5 Single camera-based depth estimation

Disparity map can also be determined for a single camera-based setup, and subsequently depth information can be roughly estimated. In single camera-based imaging system, the disparity information can be obtained from shading, textures, contours, motion, and focus/defocus.

Shape from shading extracts the shape of the objects in a three-dimensional scene from the gradual variations of the shading (intensity values) in the two-dimensional images [17]. Now for each of the pixels in an image, the light source direction and the surface shape can be determined. Surface shape can be described by two terms, namely surface normal and surface gradient. For this, we need to find solutions for system of non-linear equations having unknown variables. Hence, finding a unique solution is a difficult task.

Texture can be used as a monocular cue to recover the three-dimensional information from two-dimensional images [18]. For this, the regions having uniform texture has to be segmented from the image, and subsequently the surface normals are estimated. During segmentation, the intensity values are assumed to vary smoothly (isotropic texture). Hence, the actual shape information cannot be

estimated.

Reconstructing shape from a single line (*i.e.,* shape from contour) can be used to calculate the distance of the object *i.e.,* depth information. Contour may be line or edge in an image [19]. In practical situations, these lines are random due to the presence of noise. Interpretation of three-dimensional information based on these set of random lines is quite difficult. Another way of obtaining the three-dimensional information is from the displacement or flow field of the objects present in the consecutive image sequences [20]. From the initial image, the feature such as corner points are extracted, and these corner points are tracked in the next consecutive images. The motion trajectory estimated in this process is finally used to reconstruct their three-dimensional positions. But, it is difficult to estimate displacement for all the images of the sequence which are sampled at high rates.

Another method uses multiple images captured from the same viewpoint at different focal lengths to obtain disparity map [21]. For each of the pixels, the focal measure which gives the details of how blurry the neighbourhood of a pixel is calculated. The spatial position of the image, where this measure is maximum is also determined. This particular image position helps to link a pixel to a spatial position in order to obtain the depth map. The disadvantage of this method is that it requires large number of images for depth estimation. In addition to this, this method also requires a textured scene for depth estimation.

The above methods show some of the well-known techniques to extract three-dimensional information using monocular camera. Although the above methods extract disparity information from the images captured by single camera, they are unable to generate accurate depth information. That is why, stereo vision setup is generally used for the applications which require an accurate depth information.

1.2 Applications of Stereo Vision

Stereopsis refers to the process of depth perception using binocular vision. Some of the advantages of stereo vision setups are listed as follows:

- Two cameras of stereo vision setup provide wider field-of-view as compared to single camera-based system. Wider field-of-view is needed in many computer vision applications, like visual surveillance, autonomous vehicle [22].

- Disparity map gives information of the position of the objects in the real world scene. Small disparity value means the object is farther from the camera, whereas larger disparity value

indicates that the object is nearer to the camera. In other words, objects nearer to the camera undergoes more displacement than the objects away from the camera. This property can be used in accurate object segmentation. So, segmentation would be independent of colour, and hence the constraint of colour dissimilarity between the foreground and the background is removed [4]. Figure 1.7 shows one example of image segmentation using disparity information. In this figure, the stereo images, their corresponding disparity map, image segmented using only colour information, and the image segmented using disparity information are shown separately. The image consists of two cones having similar colour enclosed by a square box. When the image is segmented using colour information, these two objects are segmented as a single object as shown in Figure 1.7(d). On the other hand, they are segmented as two different objects by utilizing the three-dimensional information as shown in Figure 1.7(e). These two objects have different disparity values in spite of having similar colour, which can be seen in Figure 1.7(c).

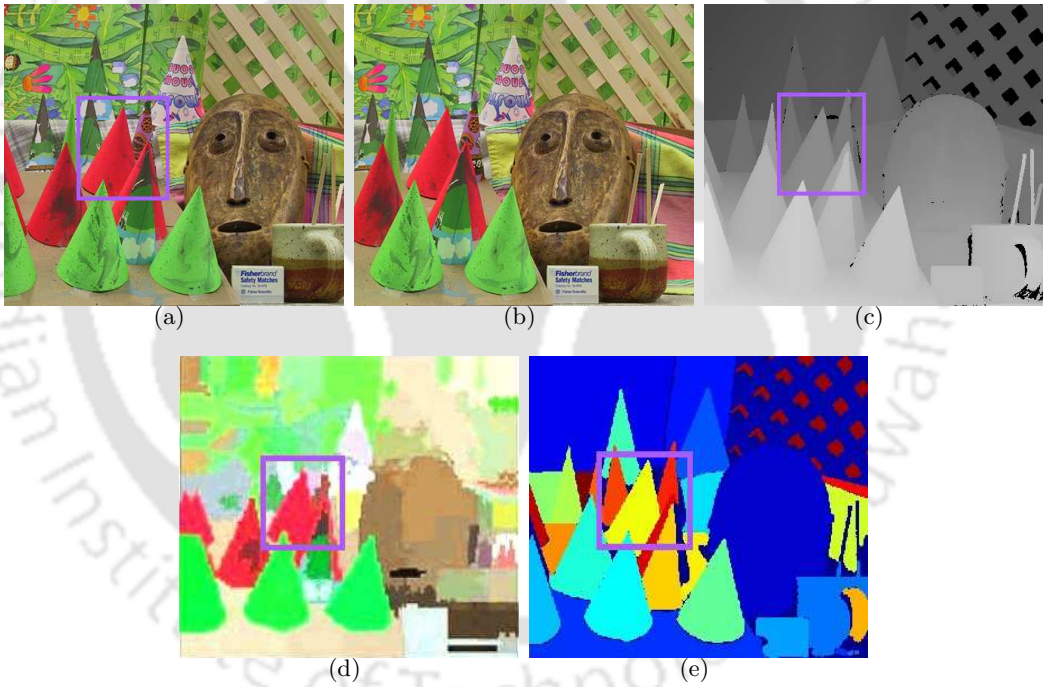


Figure 1.7: Accurate image segmentation using three-dimensional information. (a) Left image; (b) Right image; (c) Disparity map; (d) Colour-based segmentation [3]; (e) Disparity map-based segmentation [4].

- The size of an object, distance between two objects, and distance of an object from the camera can be determined in a stereo vision setup.
- The three-dimensional information obtained from stereo images enables us to understand the complex scenes in a better way. So, stereo vision can be used for the applications such as robot

navigation, objection detection, and tracking etc.

- Stereo vision provides an efficient perception of objects having curved surfaces.

So, the applications which need depth of a scene and wider field-of-view use a stereo vision setup. Some of these applications are listed as follows:

- Mobile robot navigation [23–25]
- Medical diagnosis [26–28]
- Computer graphics and virtual reality [29]
- Industrial automation [30, 31]
- Agricultural applications [32–34]

1.3 Basics of Stereo Correspondence

The primary goal of stereo vision is to find matching pixels in two stereo images. When the matching pixels are known, the difference between the coordinates of these pixels gives the disparity values. Subsequently, these disparity values can be used to find the distance of an object.

The concept of stereo correspondence is explained with the help of Figure 1.8. Figure 1.8(c) shows the target image overlapped onto the reference image. This overlapped image gives a clear visualization of the displacements of the pixels. This displacement is shown by yellow and green arrows in Figure 1.8(d). It is observed that some of the image regions have comparatively large displacements than some other regions. Objects nearer to the camera encounter more shift compared to more distant objects. This is reflected in the disparity values. Hence, objects nearer to the camera have high disparity values, while objects farther from the camera have low disparity values. This effect is seen in the ground truth disparity map as shown in Figure 1.8(e). In this figure, low gray level values correspond to low disparity values, whereas high gray level values correspond to higher disparity values. Ground truth disparity in colour map is shown in Figure 1.8(f) for better visualization. For example, Figure 1.8(a) and 1.8(b) show the corresponding matching pixels shown by cyan colour in Teddy stereo image pair. Pixel $(135, 276)$ in the reference image has corresponding matching pixel $(135, 242)$ in the target image. The disparity value is calculated as: $d = (276 - 242) = 34$.

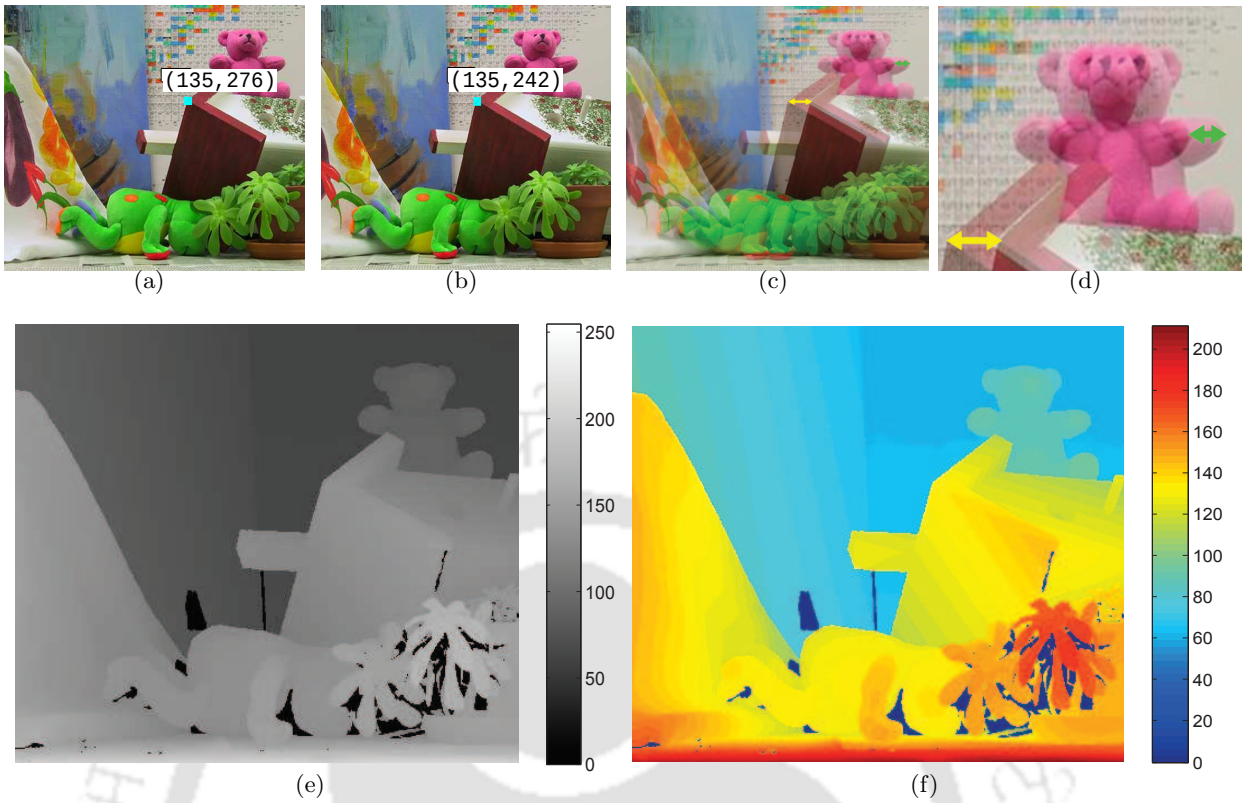


Figure 1.8: (a) Reference image; (b) Target image; (c) Overlapped stereo images; (d) A portion of the image (c); (e) Disparity map shown in gray scale; (f) Disparity map shown in colourmap.

1.4 Issues of Accurate Disparity Map Estimation

As discussed earlier, stereo matching aims to find the corresponding matching points in a stereo image pair. The matching pixels are found based on some assumptions and constraints [5]. But, finding of the exact matching pixels is quite difficult due the following factors. All these factors create ambiguity in the matching process.

1.4.1 Occlusions

Occlusion is a major obstacle in accurate disparity map estimation. The presence of a portion of a scene/object in one of the stereo images, but absence in the other image is termed as occlusion. One main reason for the presence of occlusion is due to the different field-of-view of the cameras. Occlusion may also occur due to the overlapping of different objects located at different distances from the cameras. This concept can be easily understood from Figure 1.9. In this, the presence of occlusion due to the field-of-view is highlighted by red colour boxes, while yellow colour boxes show the occluded regions due to the overlapping of the objects.

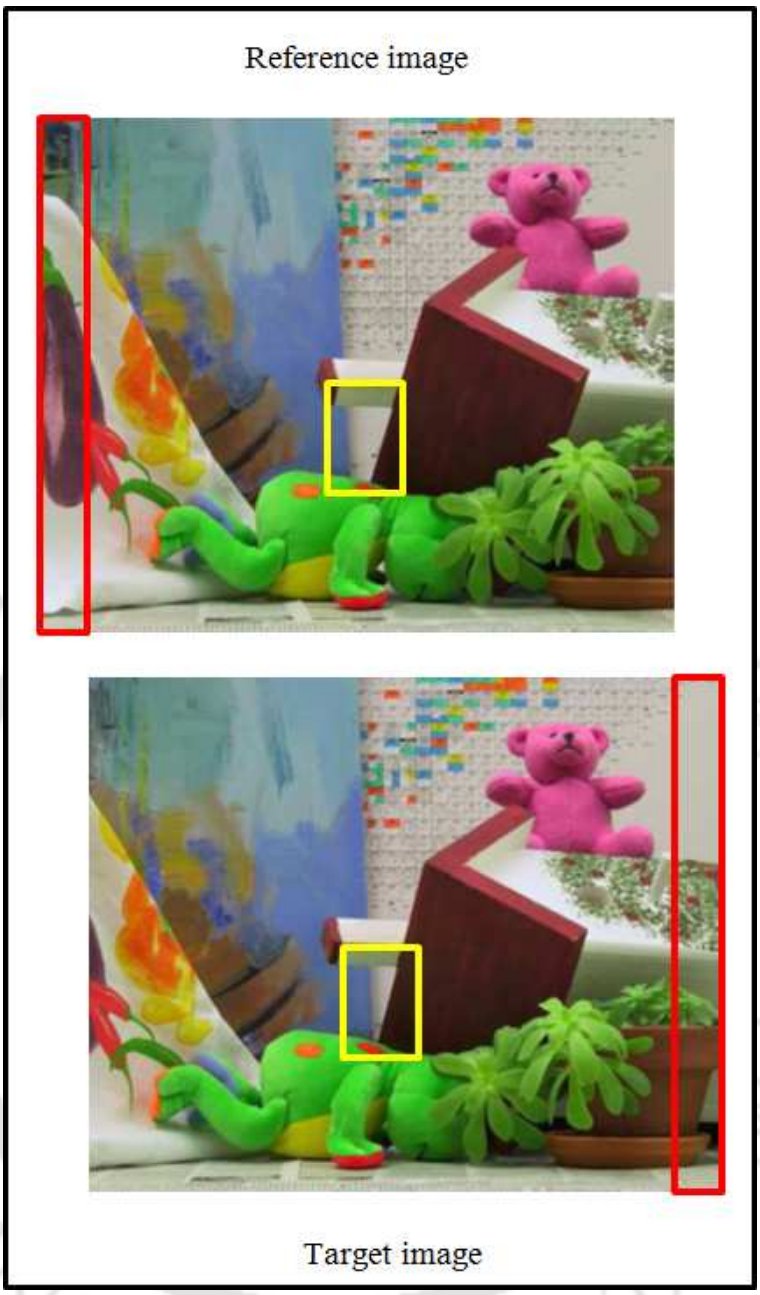


Figure 1.9: Presence of occlusion is highlighted with red and yellow colour boxes in the Teddy stereo images from the Middlebury dataset.

1.4.2 Photometric variations

Photometric invariance states that the regions around the matching points in both the images have almost similar intensity values for diffuse surfaces. Two identical cameras at different positions are employed to capture stereo image pairs. But, the optical characteristics of both the cameras may slightly differ. This leads to photometric variations in the stereo image pairs. Different viewing angles

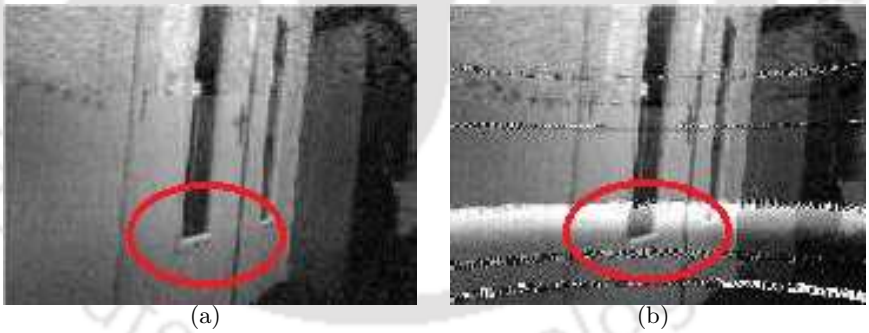
of the cameras may also cause photometric variations. Figure 1.10 shows photometric variations in a stereo image pair.

(a) (b)

Figure 1.10: Photometric variations in a stereo image pair. (a) Left image; (b) Right image [5].

1.4.3 Image sensor noise

Presence of noise in any one of the images or both the images changes the pixel intensity values, which finally leads to improper matching. Some stereo correspondence algorithms are specifically designed to handle certain types of noise. Effect of noise may be reduced by preprocessing of the images by filtering operations before matching. The effect of noise on stereo images is shown in Figure 1.11.



(a) (b)

Figure 1.11: Stereo images affected by noises. (a) Left image; (b) Right image [6].

1.4.4 Specularities and reflections

For a specular surface, the radiance leaving the surface is dependent on angle. So, the disparity map obtained from the stereo image pairs may not give actual information of the specular surface. Hence, three-dimensional information obtained from this disparity map may be completely different from the true shape of the surface. Figure 1.12 illustrates specular reflections in two stereo images.

(a) (b)

Figure 1.12: Specular surfaces in (a) Left image; (b) Right image [5].

Reflection of lights can lead to multiple occurrences of the real world points in an image. So, it is very

(a) (b)

Figure 1.13: Specular reflections in (a) Left image; (b) Right image [5].

difficult to get the corresponding matching pixels. Figure 1.13 shows the effect of reflections from the specular surfaces.

1.4.5 Foreshortening effect

(a) (b)

Figure 1.14: Foreshortening areas for two different viewpoints [5].

The appearance of an object depends on the direction of the viewpoint. Hence, an object may

appear compressed and occupies smaller area in one image as compared to the other image. In general, stereo correspondence methods assume that the areas of the objects in both the stereo images are same. But in reality, the surface area of an object would be different on account of different viewpoints. So, the exact stereo matching cannot be achieved in this scenario. Figure 1.14 shows the foreshortening areas in a stereo image pair.

1.4.6 Perspective distortions

Perspective distortion is a geometric deformation of an object and its surrounding area due to the projection of a three-dimensional scene on a two-dimensional image plane. Perspective transformation makes an object to appear large or small as compared to its original size. Figure 1.15(a) and 1.15(b) show perspective distortion in stereo images. Figures 1.15(c) and 1.15(d) show the direction of the white marker due to the perspective projections of the left and the right cameras. So in this case also, finding of the exact matching pixels would be difficult.

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

Figure 1.15: Stereo images having perspective distortions. (a) Left image; (b) Right image; (c) Zoomed out region of image (a); (d) Zoomed out region of image (b) [5].

1.4.7 Textureless regions

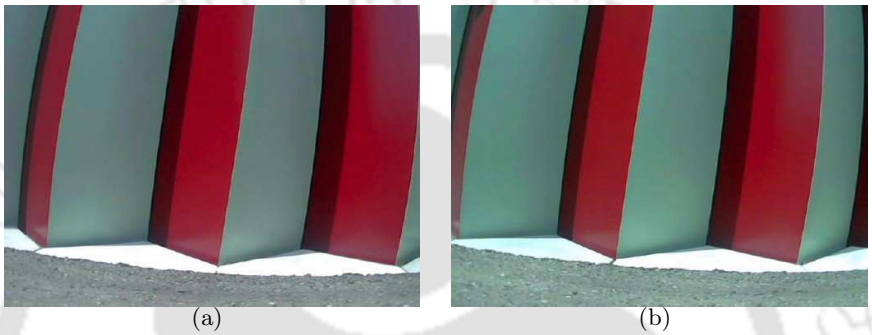
It is very difficult to find the matching pixels in the textureless image regions. Plain wall, clear sky, and too dark/bright regions are the examples of textureless regions. Figure 1.16 shows the presence of textureless regions in a stereo image pair.

1.4.8 Repetitive structures

Ambiguity in matching also occurs when the image regions have similar or repetitive pattern or structure in the horizontal direction. In Figure 1.17, there are three vertical red regions. The pixels of the first region are very much similar to the pixels of the other two regions. So, this repetitive structural patterns create ambiguity in pixel matching.

(a) (b)

Figure 1.16: Presence of textureless regions in a stereo image pair. [5].



(a) (b)

Figure 1.17: Presence of repetitive structures in a stereo image pair [5].

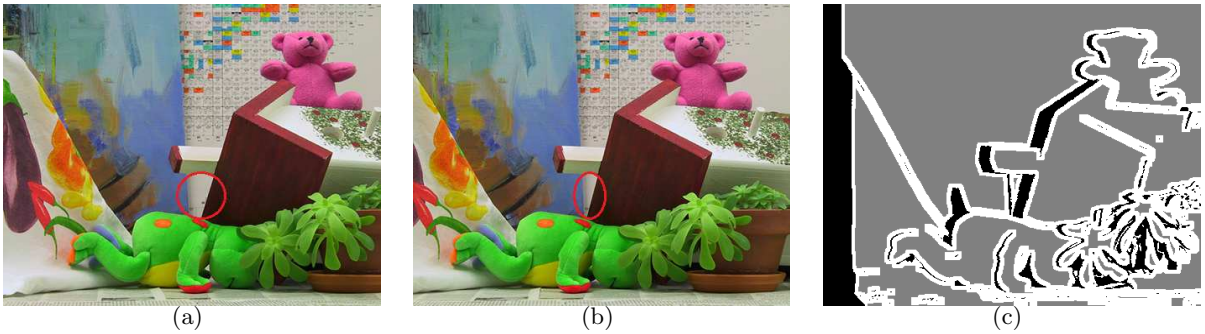
1.4.9 Discontinuity

Disparity map estimation is usually based on the assumption that the surfaces present in a scene are smooth. This assumption is valid only when a scene contains a single surface. However, this constraint is not fulfilled when multiple objects are present in the scene. Because of the discontinuities between different objects in a scene, there is an abrupt change in disparity values in the boundary regions. In this condition, it is difficult to estimate accurate disparity values at the discontinuous image regions. Figure 1.18 shows the discontinuous regions for Teddy stereo image pair.

1.5 Organization of the Thesis

This thesis contains six chapters, including present one.

Chapter 2 presents the basic concept of disparity map estimation from a stereo image pair. An overview of the existing disparity map computation methods are briefly discussed. Subsequently, occlusion detection and filling methods are also described in this chapter. Finally, motivation and objectives of our research work are finalized based on the limitations of existing disparity map estimation



(a) (b) (c)

Figure 1.18: Discontinuous regions in stereo images. (a) Left image; (b) Right image; (c) Discontinuous regions [7].

methods.

Chapter 3 describes the proposed disparity map estimation method. Gabor filter in spatial domain is used for feature extraction. These pixel-wise features are then employed to calculate the matching costs. Subsequently, matching costs of the local regions are combined using a two step filtering process for estimating the disparity map. The performance of the proposed method is evaluated for standard stereo image pairs.

Disparity map estimation in presence of occlusion is an well known research problem. The proposed method of disparity map estimation in presence of occlusion is described in **Chapter 4**. Our method can detect the occluded pixels by only using one disparity map. The weight based occlusion filling scheme is finally proposed. Our proposed occlusion filling method uses both left and right images for occlusion filling, and hence accuracy is more.

The performance of the extracted Gabor features is experimentally evaluated in **Chapter 5**. These features are analyzed for different Gabor filter parameters. The performance of Gabor features for different radiometric parameters are also examined. Experimental results presented in this chapter validate that the extracted features are quite efficient for compact representation of images for stereo matching.

Finally, we draw our conclusion in **Chapter 6** by highlighting the strengths and shortcomings of our schemes and outlining possible extensions.

2

A Review on Stereo Correspondence Methods

Stereo vision gives detailed information of a scene, and hence it has many computer vision applications. The additional details in the form of disparity information can be effectively utilized in many computer vision algorithms. Disparity map estimation is one of the techniques used to extract three-dimensional information of a scene. Occlusion is a major obstacle in finding an accurate disparity map. This chapter presents the existing state-of-the-art methods used for finding a stereo correspondence. There are two well established stereo correspondence approaches–global approach and local approach. Both of these approaches have some inherent advantages and disadvantages. Subsequently, existing literatures on occlusion detection and filling methods are highlighted. Based on the literature survey presented in this chapter, motivation and objectives of the thesis are framed.

2.1 The Basic Principle of Finding a Disparity Map

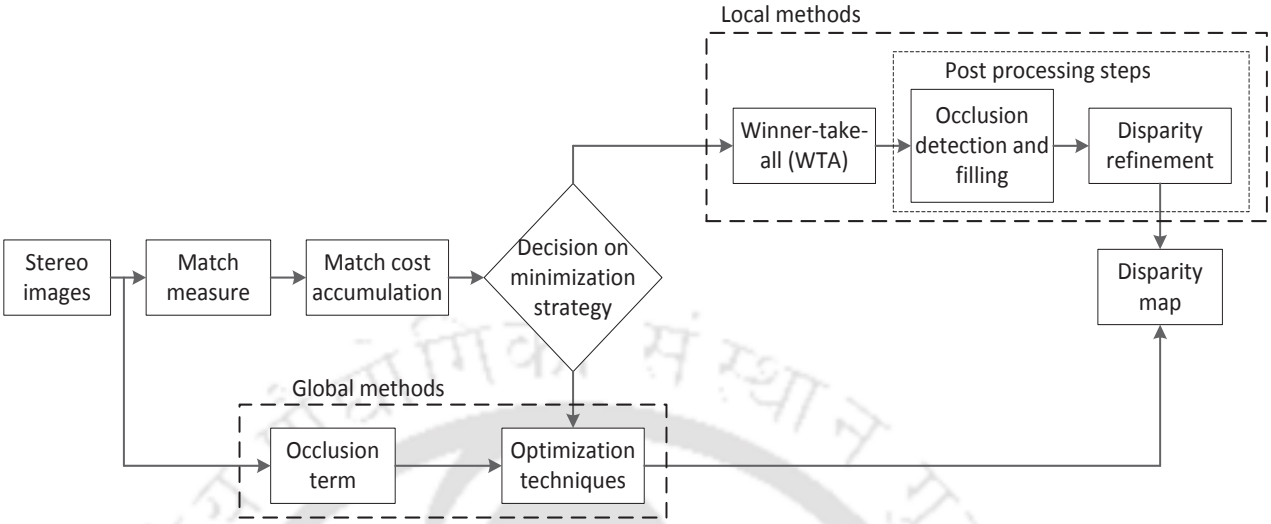


Figure 2.1: General block diagram for computing a disparity map.

Figure 2.1 shows the steps involved in the disparity map computation by both local and global methods. In the first step, similarity/dissimilarity measure between the pixels of the reference and the target images are calculated for all the possible disparity values. Subsequently, these values are accumulated. Based on the minimization strategy employed, these methods can be categorized as global or local methods. Global methods use optimization techniques to find the disparity map corresponding to the minimum energy function. Sometimes, the optimization also considers occlusion term in addition to the terms corresponding to the match measure and the accumulation. On the other hand, Winner-Take-All (WTA) strategy is employed in local methods to find the disparity value corresponding to minimum cost value. Additionally in local methods, occlusion detection/filling and disparity refinement are sequentially performed as a post-processing step for obtaining the final disparity map.

2.2 Global Algorithms

Global algorithms formulate an appropriate energy function for the entire image in order to estimate the disparity map D. Subsequently, a disparity map is obtained by minimizing this energy function using an optimization technique. The general form of this energy function is as follows:

$$E(D) = E_{data}(D) + \eta \cdot E_{smooth}(D) \tag{2.1}$$

It consists of two terms, namely data term and smoothness term. Data term penalizes the inconsistency between the reference and the candidate matching pixels. Mathematically, this can be expressed as:

$$E_{data}(D) = \sum_{\mathbf{p} \in \mathbf{I}_l} C(\mathbf{p}, d) \tag{2.2}$$

where $C(\mathbf{p}, d)$ is the pixel similarity/dissimilarity measure between the pixel $\mathbf{p} = (p_1, p_2)$ in the left image and the pixel $\mathbf{p}' = (p_1 - d, p_2)$ in the right image for disparity d. The function of the smoothness term is to spatially smooth the disparity map, while simultaneously it preserves the disparity discontinuities. This term is defined as follows:

$$E_{smooth}(D) = \sum_{\{\mathbf{a},\mathbf{b}\} \in \mathcal{N}_p} s(d_a, d_b) \tag{2.3}$$

where $\{\mathbf{a}, \mathbf{b}\}$ denotes all pairs of spatially neighbouring pixels in the neighbourhood $\mathcal{N}_p$ of pixel $\mathbf{p}$ in left image. The function $s(d_a, d_b)$ encodes the spatial smoothness assumption in the disparity values d_a and d_b corresponding to the pixels $\mathbf{a}$ and $\mathbf{b}$ respectively. In Equation (2.1), η is a user-defined parameter that decides the relative importance of data and smoothness terms. Global algorithms differ based on how the energy functions are defined and the optimization techniques used. The performance of an approximate energy minimization algorithm depends on the following two factors: (i) choice of the energy function which includes both data and smoothness terms, and (ii) optimization technique employed. A brief overview of energy function (data and smoothness terms) and optimization techniques are described below.

2.2.1 Data term

Data term gives the pixel dissimilarity measures. The simple functions employed for this task are absolute difference (AD), squared difference (SD), sum of absolute differences (SAD), sum of squared differences (SSD), and normalized cross correlation (NCC). These measures are summarized in Table 2.1. In Table 2.1, $\mathbf{p} = (p_1, p_2)$ is the pixel for which matching is calculated, and $\mathbf{q} = (q_1, q_2)$ is the pixel in the support window $\mathcal{N}_p$ of the pixel $\mathbf{p}$. Here, disparity range $d \in [0, d_{\max} - 1]$, where $d_{\max}$ is the maximum allowable disparity value. The details of the abovementioned functions are discussed in Section 2.3. The general data term used in Equation (2.1) does not consider the case of occlusion while calculating the matching cost. Hence, this affects the accuracy of the computed disparity map nearer to the object boundaries. It is quite meaningless to calculate the matching cost for the occluded pixels as these pixels are present in one image and absent in the other image. So, the cases for which

Table 2.1: Metrics used for finding the matching cost value

Similarity Measures	Mathematical Expressions
AD	$\lvert I_l(p_1,p_2) - I_r(p_1-d,p_2)\rvert$
SD	$[I_l(p_1,p_2) - I_r(p_1-d,p_2)]^2$
SAD	$\sum_{\mathbf{q}\in\mathcal{N}_p} \lvert I_l(q_1,q_2) - I_r(q_1-d,q_2)\rvert$
SSD	$\sum_{\mathbf{q}\in\mathcal{N}_p} [I_l(q_1,q_2) - I_r(q_1-d,q_2)]^2$
NCC	$\frac{\sum_{\mathbf{q}\in\mathcal{N}_p} \left(I_l(q_1,q_2)-\bar{I}_l\right)\left(I_r(q_1-d,q_2)-\bar{I}_r\right)}{\sqrt{\sum_{\mathbf{q}\in\mathcal{N}_p} \left(I_l(q_1,q_2)-\bar{I}_l\right)^2\left(I_r(q_1-d,q_2)-\bar{I}_r\right)^2}}$

the occlusion term is included in the energy function along with the data and the smoothness terms are discussed in this section. The modified data term can be expressed as follows:

$$E_{data}(D) = \sum_{\mathbf{p}\in I_l} [C(\mathbf{p},d_p)(1-O(\mathbf{p})) + \lambda_{occ} O(\mathbf{p})] \tag{2.4}$$

where

$$O(\mathbf{p}) = \begin{cases} 1, \text{ if } \mathbf{p} \text{ is occluded} \\ 0, \text{ otherwise} \end{cases} \tag{2.5}$$

In Equation (2.4), λ_{occ} is a user-defined parameter for penalizing the occluded pixels.

Kolmogorov and Zabih proposed an variant of occlusion term that imposes a penalty a_{pen} if a pixel is occluded [35]. Subsequently, visibility constraint is considered for handling occlusion [36]. Visibility term is penalized with one if this constraint is not satisfied.

Woodford *et al.* used the disparity of the left image to find the corresponding pixel in the right image by the process of warping [37]. According to this process, if two pixels of the left image correspond to the same pixel in the right image, then the pixel which is having a smaller disparity value is considered as the occluded pixel. This can be given by the following equation [38]:

$$O(\mathbf{p}) = \begin{cases} 1, \text{ if } \exists\ \mathbf{q} \in \mathbf{I}_l : p_1 - d_p = q_1 - d_q \text{ and } d_p < d_q \\ 0, \text{ otherwise} \end{cases} \tag{2.6}$$

where $\mathbf{p}$ and $\mathbf{q}$ are the two pixels in the left image with disparity values d_p and d_q respectively. Their corresponding matching points in the right image are $p_1 - d_p$ and $q_1 - d_q$ respectively. Another

variant of energy function which includes occlusion term to detect the occluded pixels is presented in [39, 40]. These approaches require disparity maps of both left and images. These methods make use of uniqueness assumption for detection of occluded pixels. This assumption is violated for horizontally slanted surfaces where many pixels in reference image may correspond to a single pixel in other image.

2.2.2 Smoothness term

The widely used smoothness terms can be expressed in terms of first and second-order functions [38]. The basic first-order function is a linear smoothness function which is given by:

$$s(d_p, d_q) = |d_p - d_q| \tag{2.7}$$

The drawback of this function is that it blurs the disparity discontinuities. Potts model is a first-order function can preserve disparity discontinuities [41, 42]. It is defined as follows:

$$s(d_p, d_q) = \begin{cases} 0, & d_p = d_q \\ 1, & \text{otherwise} \end{cases} \tag{2.8}$$

This algorithm assigns large penalties to the regions having depth discontinuities and the regions having small disparity transitions. Hence, these two regions cannot be effectively differentiated by this model. This drawback is overcome by using a truncated linear smoothness function, which can be written as follows:

$$s(d_p, d_q) = \min\left(|d_p - d_q|, \gamma_t\right) \tag{2.9}$$

where γ_t is a truncation threshold. However, first-order functions fails when the disparity transition is very small.

Second-order functions are employed to handle this problem which is defined as follows:

$$s(d_o, d_p, d_q) = |d_o - 2d_p - d_q| \tag{2.10}$$

where d_o and d_q are the disparity values of the neighbouring pixels **o** and **q** respectively. Minimization of an energy function involving a second-order smoothness term is complex as these functions give rise to higher-order cliques in a graph.

Another smoothness function termed as edge-sensitive function is proposed in [38]. This is based on the assumption that the colour boundaries coincide with the disparity discontinuities. Then, the

smoothness term shown in Equation (2.3) can be modified as:

$$E_{smooth}(D) = \sum_{(\mathbf{j},\mathbf{k})\in\mathcal{N}_q} w(\mathbf{j},\mathbf{k}) s(d_j, d_k) \tag{2.11}$$

For example, in Equation (2.11) weight $w(\mathbf{p},\mathbf{q})$ represents colour term in bilateral filter [10]. In this case, weight is expressed as follows:

$$w(\mathbf{j},\mathbf{k}) = \exp\left(-\frac{\Delta c_{jk}}{\gamma_c}\right) \tag{2.12}$$

where Δc_{jk} represents the colour difference between the pixels $\mathbf{j}$ and $\mathbf{k}$ in the CIELab colour space, and γ_c is a user-defined parameter. In Equation (2.12), if both the pixels are of similar colour, then a higher weight value is assigned and vice versa. This approach suffers from edge shrinking bias. This problem can be overcome by considering more neighboring pixels which slightly modify the smoothness term given in Equation (2.11). This modified smoothness term is given by:

$$E_{smooth}(D) = \sum_{\mathbf{p}\in\mathbf{I}_l} \sum_{\mathbf{q}\in\mathcal{N}_p} w(\mathbf{p},\mathbf{q}) s(d_p, d_q) \tag{2.13}$$

where $\mathbf{I}_l$ represents all the pixels in the left stereo image, and $\mathcal{N}_p$ is the support region for the pixel $\mathbf{p}$. Presence of large number of edges makes optimization computationally very expensive.

To reduce the computation burden, segmentation-based algorithms are proposed [40, 43, 44]. Segmentation is performed on the reference image based on the colour cue. This process is based on the assumption that all the pixels in a segment have smooth disparity values, and the segment boundaries coincide with the disparity discontinuities. As a consequence, matching is performed segment-wise instead of pixel-wise. However, segmentation-based methods fail when the segments overlap the disparity borders.

2.2.3 Optimization

The next step of global-based disparity map estimation methods is the minimization of the energy function which can be solved by an optimization framework. As the name implies, global algorithms use the entire image in the minimization process *i.e.,* disparity value assigned to a single pixel influences the disparity values of all other pixels of the image. The main disadvantage of the global methods is that they are computationally complex. This is due to the fact that the computational complexity increases with the disparity range and the image dimension.

2.2.3.1 Dynamic programming

Dynamic programming is one of the optimization techniques which minimizes the energy function along the horizontal scanline. This method breaks down a complex problem into a set of subproblems. After the solution for each subproblem is computed, they are finally combined to obtain the solution of a complex problem. In [45], a series of prior models are formed for stereo matching. The resulting objective function is optimized using dynamic programming. The generated disparity map suffers from horizontal streaks as the matching is performed along the horizontal scanline only. To overcome this, a heuristic method is proposed which incorporates vertical smoothness in the optimization technique [45]. Furthermore, this step introduces additional streaks in the vertical direction. To overcome the problem of horizontal scanline optimization, vertical consistency is introduced by applying dynamic programming on a pixel-tree structure [46]. Pixels form the nodes and the most important edges in a four-connected neighborhood are retained in the tree structure. This tree structure gives a good approximation of the two-dimensional grid. This algorithm is fast, and hence suitable for real-time implementation. The reduction in the computational complexity is achieved by choosing an appropriate tree structure. The tree structure only contains some selected edges. But this reduction in edges affects the quality of the computed disparity map.

Gong and Yang proposed an orthogonal reliability dynamic programming algorithm for disparity map computation [47, 48]. Inter-scanline consistency is accomplished by applying the reliability dynamic programming process along both the horizontal and vertical scanlines.

In order to reduce horizontal streaks, Chang *et al.* proposed a one-dimensional optimization technique which uses output from Winner-Take-All as a prior information for dynamic optimization [49].

2.2.3.2 Graph cut

Boykov *et al.* proposed two different algorithms that allows large moves in order to obtain the local minimization of the multidimensional energy function using iterative graph cuts [41]. The allowed moves are named as $\alpha - \beta$ swap and $\alpha-$expansion. In the $\alpha - \beta$ swap algorithm, a large number of pixels with a label α are changed to label β, while another set of pixels with label β are changed to label α. In the $\alpha-$expansion algorithm, a group of pixels having different labels are assigned a label of α. These moves are performed until moves are found to get a minimum energy value. Among these two algorithms, $\alpha-$expansion performs better than $\alpha - \beta$ swap but requires linear time for execution.

A new approach termed as fusion move is proposed, which is used in binary graph cut for opti-

mization of continuous disparity values [50, 51]. Graph cut is applied to the image in the bit-level representation. It is applied in a sequential order from the most significant bit to the least significant bit, which leads to different numbers of solutions. These solutions are merged in an optimal way to obtain the final disparity map. That is why this algorithm is termed as fused move algorithm. As opposed to linear complexity, this approach has logarithmic complexity. When the graph contains non-submodular edges then obtaining the optimal fusion move turns out to be an NP-complete problem.

Zureiki *et al.* proposed a reduced graph cut algorithm for disparity map computation [52]. A simplified graph is constructed by only considering some potential values selected from the disparity range for each of the pixels. This reduction increases the coarseness of the computed disparity map [53].

A hierarchical bilateral disparity structure algorithm is presented in [53]. This hierarchical approach divides the disparity range within the stereo images into lower level bilateral disparity structures. Disparity values are classified as a foreground disparity set and a background disparity set within this bilateral structure. The disparity map specific to this bilateral structure is computed by minimizing the energy function using a graph cut method.

2.2.3.3 Belief propagation

Belief propagation is an iterative optimization process which works on the principle of sending messages to its four-connected neighbors in a graph [54]. Each message is a vector whose dimension is equal to the maximum disparity range. A message from pixel **p** to pixel **q** encodes the belief of **p** about **q** *i.e.,* the probability of **q** at a particular disparity value. New messages are computed and updated at each iteration. This finally leads to a belief vector that minimizes the energy function. This method is very slow for practical implementation.

To reduce memory requirements, Yu *et al.* applied a compression technique to efficiently represent the message [55]. This reduces the memory requirement by 12%, but the compression and decompression techniques makes the algorithm slow. A novel Envelope Point Transform method using principal component analysis (PCA) is proposed for compressing the messages. The extra belief propagation pass required to achieve this task further makes it slower than the conventional belief propagation technique.

Wang *et al.* classified pixels as reliable and unreliable using bi-labelling process [56]. Consequently, this reduces the search range of reliable pixels. The disparity values of reliable pixels are then propagated to the unreliable pixels to obtain an accurate disparity map. This method is slow as compared to the conventional belief propagation as the classification of the reliable pixels and also the joint

bilateral filter add extra computational burden.

Yang *et al.* proposed constant space belief propagation, and it is applied in a hierarchical manner [57]. This approach is independent of the number of labels used, and hence the performance increases with the number of disparity labels. The limitation of this method is that it performs poorly nearer to the object boundaries. The above method sacrifices accuracy for fulfilling the computational requirements.

Global algorithms minimize an energy function to produce a disparity map. These methods perform this task in an iterative manner, and hence they are computationally expensive. However, the energy function fails to accurately represent stereo matching. Additionally, energy function is not effectively minimized by the existing optimization techniques. That is why some of the well-established methods use local methods to find a stereo correspondence.

2.3 Local Algorithms

Local algorithms only consider a small neighboring region around a pixel of interest while computing stereo correspondence. They assume that the corresponding pixels in the left and the right images almost have similar colour (photo-consistency assumption), and lie in the same horizontal scanline, which is an epipolar constraint. The widely used pixel-based matching functions include absolute difference (AD) and squared difference (SD). AD evaluates absolute difference of the selected quantity between the reference and the target matching pixel given by:

$$C(\mathbf{p}, d) = |I_l(p_1, p_2) - I_r(p_1 - d, p_2)| \tag{2.14}$$

where I_l, I_r are the reference and the target images respectively, (p_1, p_2) is the pixel coordinate, and d is the disparity value in the range of $[0, d_{max} - 1]$. Here, d_{max} represents the maximum number of disparity values. The selected quantity for pixel matching may be intensity or colour value of the pixel and/or any other features corresponding to the pixel under consideration.

SD is the squared difference of the selected quantity between the reference and the target candidate pixels. Mathematically, this can be written as follows:

$$C(\mathbf{p}, d) = [I_l(p_1, p_2) - I_r(p_1 - d, p_2)]^2 \tag{2.15}$$

These functions create more ambiguities in the output disparity map. To overcome this drawback, smoothness assumption is considered. This assumption tells that the spatial neighboring pixels have

the same disparity value as the center pixel. Hence for a pixel in the reference image, the sum of the matching costs of all the pixels in a small neighborhood region is considered as the matching cost value. SAD, SSD, truncated SAD/SSD and NCC are the most widely used block-based cost functions [58, 59].

SAD is defined as the sum of absolute differences of all the pixels in a window region. Mathematically, this can be written as:

$$C(\mathbf{p}, d) = \sum_{\mathbf{q} \in \mathcal{N}_p} |I_l(q_1, q_2) - I_r(q_1 - d, q_2)| \tag{2.16}$$

In order to enhance the robustness of the matching cost against outliers, truncated SAD is used. It is the minimum of SAD and a truncation parameter γ_t, is defined as follows:

$$C(\mathbf{p}, d) = \min \left\{ \sum_{\mathbf{q} \in \mathcal{N}_p} |I_l(q_1, q_2) - I_r(q_1 - d, q_2)|, \gamma_t \right\} \tag{2.17}$$

Similarly, SSD is defined as the sum of squared differences of the matching costs of all the pixels in a window region, which is given as follows:

$$C(\mathbf{p}, d) = \sum_{\mathbf{q} \in \mathcal{N}_p} [I_l(q_1, q_2) - I_r(q_1 - d, q_2)]^2 \tag{2.18}$$

NCC measures the cross correlation between the pixels corresponding to the window regions of the reference and the target images. It is expressed as:

$$C(\mathbf{p}, d) = \frac{\sum\limits_{\mathbf{q} \in \mathcal{N}_p} \left(I_l(q_1, q_2) - \bar{I}_l\right) \left(I_r(q_1 - d, q_2) - \bar{I}_r\right)}{\sqrt{\sum\limits_{\mathbf{q} \in \mathcal{N}_p} \left(I_l(q_1, q_2) - \bar{I}_l\right)^2 \left(I_r(q_1 - d, q_2) - \bar{I}_r\right)^2}} \tag{2.19}$$

where $\bar{I}_l$ and $\bar{I}_r$ are the mean values of all the pixels of the support window in left and right images respectively. Higher NCC indicates a better match.

Rank and census transforms are also used to find a match measure [60]. Both rank and census transforms depend on the relative ordering of the pixels rather than on the intensity values itself. Rank is termed as the number of pixels having intensity values less than the center pixel. Matching cost is the absolute difference between the rank transforms of the reference and the candidate matching pixels. After applying rank transform to the stereo image, any of the function mentioned above is

applied to find the cost. It is represented as follows:

$$
\begin{aligned}
C(\mathbf{p}, d) &= |rank\left(I_l(p_1, p_2)\right) - rank\left(I_r(p_1 - d, p_2)\right)| \\
rank\left(I_l(\mathbf{p})\right) &= \sum_{(i,j)\in\mathcal{N}_p} I'(i,j) \\
I'(i,j) &= \begin{cases} 0, \; I(i,j) \geqslant I(p_1,p_2) \\ 1, \; I(i,j) < I(p_1,p_2) \end{cases}
\end{aligned}
\tag{2.20}
$$

In census transform, the pixels in the window region are converted to census bit strings, and subsequently Hamming distance is used to compare the bit strings [60]. Census transform is described as shown below:

$$
\begin{aligned}
C(\mathbf{p}, d) &= \sum_{\mathbf{q}\in\mathcal{N}_p} Ham\left(cen\left(I_l(q_1, q_2)\right), cen\left(I_l(q_1 - d, q_2)\right)\right) \\
cen\left(I_l(\mathbf{q})\right) &= \bigotimes_{(i,j)\in\mathcal{N}_p} I'(i,j) \\
I'(i,j) &= \begin{cases} 0, \; I(i,j) \geqslant I(p_1,p_2) \\ 1, \; I(i,j) < I(p_1,p_2) \end{cases}
\end{aligned}
\tag{2.21}
$$

where $Ham(\cdot,\cdot)$ denotes the hamming distance, $cen(\cdot)$ represents the census transform, and the symbol $\otimes$ denotes concatenation operation. Figure 2.2 shows an example employing census transform for

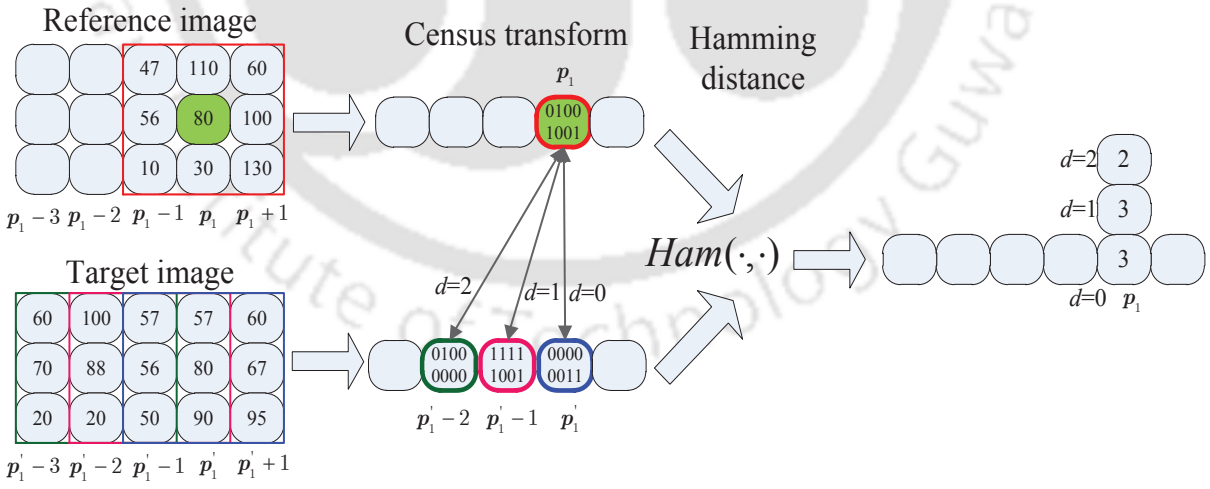


Figure 2.2: Pictorial illustration of census transform.

matching cost computation. Many variations of census transform are proposed because of its robust performance. Humenberger *et al.* proposed another variant of census transform termed as sparse

census transform to reduce computational cost [61]. In this approach, pixels only in the even rows and columns are considered for computing census transform. This approach reduces the number of computation required to obtain the sparse census transform, while at the same time it maintains the same accuracy of stereo matching as that of census transform. In addition to this, a new way of using census transform to obtain the matching cost is proposed in [62]. In this, the cost function is a combination of census transform applied to both the intensity as well as the gradient images. SAD is also merged with the abovementioned matching cost function to improve the accuracy. In both census and sparse census transforms, only two combinations occur while comparing the neighboring pixels with the center pixel. Hence, only one bit is used for each pixel. Modified census transform used the intensity value of center pixel and the mean intensity value of all the pixels in a window region for comparison with the neighboring pixels [63]. In contrast to both census and sparse census transform, it has four combinations during comparison, hence two bits are allocated for each of the pixels. In order to reduce the redundant calculations while performing correlation measure after census transform, a new generalized census transform is proposed in [64]. Generalized census transform combines edges around the center of census window. This generalized census transform is robust to image noise. Inspired by the use of adaptive support weight in cost aggregation, Perri *et al.* proposed adaptive census transform which employed support weights in the computation of census transform [65]. If the intensity value of a neighbouring pixel is less than the intensity value of the center pixel, then negative weight is assigned to that neighboring pixel, else positive weight is assigned. These weights are calculated based on the intensity dissimilarity of the neighboring pixels with the center pixel. Subsequently, the matching cost is obtained by performing SAD on the weighted census vectors. For accurate matching cost computation, a novel three-moded cross census transform is proposed in [66]. In this approach, census transform is performed with a cross-square shaped window. A noise buffer threshold is added to the center pixel during the comparison to make it more robust to noise. Also, colour and gradient distance of the center pixel are used to obtain the matching cost.

2.3.1 Problem with different window sizes

The abovementioned methods compute the matching cost of a reference pixel over a small neighborhood region. It is assumed that all the pixels in the support region have the same disparity value. A small support window preserves the depth discontinuities, but it fails to capture large texture variation. Specifically, small window fails for textureless regions, repetitive patterns, and the areas only with horizontal texture patterns. Hence small window creates matching ambiguities [38]. On the other hand, a large window fails to preserve the object borders. This is due to the fact that a larger

window may enclose two adjacent regions having different disparity values. Therefore, selection of an appropriate window of a particular shape and size is one of the crucial task for local-based disparity map estimation methods.

Many algorithms have been proposed to solve the abovementioned window size problem. These algorithms in general can be summarized as follows:

- Choosing an appropriate window from multiple windows
- Adapting the window size based on the local image characteristics
- Combining the matching cost across different window sizes (multi-resolution approach)
- Performing cost aggregation

2.3.1.1 Choosing an appropriate window

To overcome the support window size problem, algorithms to choose an appropriate window size are proposed in [67–70]. In these approaches, the matching cost is calculated for different window sizes, and the window which gives more correlation value is finally retained. The performance of these methods are better as compared to the methods which use a fixed window. However, these methods have low performance for regions having less texture. This is mainly because of the fact that all the possible resolutions and shapes of the applied windows may not be unique for all the pixels of an image.

Hirschmuller *et al.* proposed multiple window-based method to preserve ambiguities at the depth borders [71]. In this method, a combination of three different windows is used. In addition to the center window, few best neighboring windows are selected based on the correlation value. This is done to find the final matching cost. Three different ways are described to select the best neighboring windows which are considered for final matching cost calculation. As a post processing step, disparity map accuracy at the object borders is improved by applying border correction method.

In [72], a new algorithm for handling object boundary problem is proposed. Initially the boundary and non-boundary regions are detected, and suitable methods are used for each of these regions. At the boundary regions, correlation values are calculated by using different windows with different center pixels. Finally, the best window with more correlation value is selected. On the other hand, a fixed window is applied for the pixels of the non-boundary regions.

Jeon *et al.* used eight different sized windows to preserve object boundaries [73]. These windows are designed in such a way that they can detect horizontal, vertical, corner as well as diagonal edges.

Initially the window size is fixed, and the final matching cost is the minimum value obtained from all these windows. Subsequently, the left-to-right consistency is checked. The algorithm stops when the consistency check is passed. Otherwise, the window size is expanded, and the same procedure is repeated.

Adhyapak *et al.* proposed an algorithm to select an appropriate window size from a set of windows using a reliability test [74]. To avoid computational load, this algorithm is performed in an iterative manner for the large disparity range.

In [75], Veksler proposed an algorithm to select a compact window shape for getting a reasonable window size. The algorithm termed as minimum ratio cycle chooses an appropriate window shape from many compact windows using an efficient optimization method. Despite its better performance, this method has the problem of efficient real-time implementation.

A new and fast variable window approach is proposed in [76]. In this approach, a window size is selected based on the variance of the pixels. In this method, matching cost is computed using the integral images to improve the computational time.

2.3.1.2 Adaptive window size and multi-resolution approaches

An approach to adaptively vary the window size and shape is presented in [77]. This approach determines the window size in an iterative manner. A minimum window size is initiated, and the match uncertainty is calculated. Then, the window is expanded in each of the directions separately by one pixel to calculate the matching uncertainty. The gradual expansion of the window is prohibited only when the matching uncertainty value crosses the uncertainty value corresponding to the minimum window. This procedure is iterated until all the directions are prohibited. But, this method is sensitive to initial disparity map.

Boykov *et al.* proposed a variable window method [78]. Plausibility hypothesis testing is performed to obtain an arbitrarily shaped window for each of the pixels. The performance of this method is relatively better for the regions having depth discontinuities. This approach may increase the size of the object due to the inclusion of nearby low textured regions [71].

A multi-resolution stereo algorithm is proposed in [79, 80]. In [79], matching cost is calculated for each of the levels separately, and then the final matching cost is estimated by summing up the matching costs obtained from different levels. This method produces a smooth disparity map by reducing the effect of noise. But, the accuracy of the obtained disparity map at the object borders is comparatively less.

Yang *et al.* proposed an adaptive support window which is a two-pass algorithm [81]. In the first

pass, cost from four neighboring pixels are added. In the second pass, another four different pixels are selected, and the minimum two cost values are added to the cost value obtained in the previous step. In all, this algorithm uses six different support windows which are having different shape configurations. A new disparity map computation method that uses two different sized windows of which one is larger and another one is smaller is proposed in [82]. In the first step, larger window is employed to compute the initial disparity map. Although this step produces good results at low textured region, it blurs the regions near the disparity discontinuities. Smaller window is applied to all the pixels which lies in this region to obtain the disparity values. In this method, increasing the window size increases the computational load.

2.3.1.3 Cost aggregation

As explained earlier, the fixed window affects the smoothness in computed disparity values in low textured regions, and it also blurs the disparity discontinuities. So, cost aggregation is generally used to perform the smoothness operation while preserving the discontinuous regions. It aims to select a best set of pixels, and their cost values are accumulated. This step reduces the error at low textured regions and preserves the edges *i.e.,* depth discontinuities. With the invent of cost aggregation methodologies, there is a significant breakthrough of the earlier proposed local stereo correspondence methods. Consequently, the performance of the local algorithms are now comparable to the global algorithms.

The concept of cost aggregation is introduced by Yoon and Kweon [10]. The weight for each of the pixels in a support region is computed. The weight is a combination of the colour and spatial distances of the neighboring pixels in that support region with respect to the center pixel. This is equivalent to the process of weight computation in bilateral filtering. The cost aggregation method proposed in [10] is very similar to the filtering of the disparity space images (DSI) with joint bilateral filter [83]. Mathematically, this can be expressed as follows:

$$C_{agg}(\mathbf{p}, d) = \frac{\sum\limits_{\mathbf{q}\in\mathcal{N}_p}\sum\limits_{\mathbf{q}'\in\mathcal{N}_{p'}} w_l(\mathbf{p},\mathbf{q}) w_r(\mathbf{p}',\mathbf{q}') C(\mathbf{q}, d)}{\sum\limits_{\mathbf{q}\in\mathcal{N}_p}\sum\limits_{\mathbf{q}'\in\mathcal{N}_{p'}} w_l(\mathbf{p},\mathbf{q}) w_r(\mathbf{p}',\mathbf{q}')} \tag{2.22}$$

where C_{agg} is the aggregated cost, $\mathbf{p}'$ and $\mathbf{q}'$ are the candidate matching pixels in the target image corresponding to the pixels $\mathbf{p}$ and $\mathbf{q}$ in the reference image, w_l denotes the weight between the pixels $\mathbf{p}$ and $\mathbf{q}$ of a support window in the left image, and w_r denotes the weight between the pixels $\mathbf{p}'$ and

$\mathbf{q}'$ of a support window in the right image. The weights are calculated as follows:

$$w(\mathbf{p},\mathbf{q}) = \exp\left\{-\left(\frac{\Delta c_{pq}}{\gamma_c} + \frac{\Delta g_{pq}}{\gamma_p}\right)\right\} \tag{2.23}$$

where Δg_{pq} and Δc_{pq} represents the spatial distance and colour difference between the pixels $\mathbf{p}$ and $\mathbf{q}$ in the CIELab colour space respectively. Also, γ_c and γ_p are two constant parameters. The disadvantage of this method is that the computational complexity increases quadratically with the increase of window size. Additionally, this method requires a large window size to handle non-textured regions. An integral histogram is used as an approximation of the joint bilateral filter [84–86]. The bilateral filter weight is a combination of geometric and range weights. Geometric weight is based on the spatial distance between the pixels, while range weight is based on the intensity difference. The bilateral filter depends on the histogram of the difference image which is independent of the window size. In this case, it is assumed that the geometric weights are same for all the pixels. Richardt *et al.* presented a stereo matching algorithm which includes the reformulation of the adaptive support weights algorithm [87]. In this method, filtering is done by employing bilateral grid. To preserve the edges in both the stereo images, bilateral grid is extended to include both the input images while performing cost aggregation. The poor performance of this method at the object boundaries is due to the bilateral grid which is designed using only gray scale images. Hence, colour having similar gray values are difficult to differentiate. Including colour information in the grid increases the memory requirement, and hence only two colour channels are involved. This is still slower, and produces inferior results compared to the gray scale approach. Mattoccia *et al.* proposed a cost aggregation algorithm which is also based on the joint bilateral filter [88]. The support weights are calculated according to a spatial and a range filter. Spatial weights are calculated based on the spatial distance between the center and neighboring pixels. The range weights are calculated by dividing the support window into blocks. Each block is assigned a single weight based on the colour distance between the center pixel and the mean value of all the pixels in the block. The abovementioned methods sacrifice quality in order to achieve higher computational speed.

Inspired by the performance of bilateral filter for cost aggregation, Hosni *et al.* performed cost aggregation using a guided filter (GF) [11]. In this method, weights of the support window are computed as follows:

$$w(\mathbf{p},\mathbf{q}) = \frac{1}{|\mathcal{N}_p|^2}\sum_{\mathbf{q}\in\mathcal{N}_p}\left[1 + \left(\mathbf{I}_p - \boldsymbol{\mu}_p\right)^T(\boldsymbol{\Sigma}_p + \varepsilon\mathbf{U})^{-1}\left(\mathbf{I}_q - \boldsymbol{\mu}_p\right)\right] \tag{2.24}$$

where $\boldsymbol{\mu}_p$ and $\boldsymbol{\Sigma}_p$ are the mean vector and the covariance matrix of all the pixels in the window $\mathcal{N}_p$,

$\mathbf{U}$ is a 3×3 identity matrix. $|\mathcal{N}_p|$ is the number of pixels in the window $\mathcal{N}_p$, and ε is a user-defined smoothness parameter. Yang *et al.* accomplished cost volume filtering by employing a full image-based guided filter [89,90]. The support weights are calculated by a weight propagation scheme. This scheme uses four-connected grid based on the guided image. The weight propagation starts from the source pixel, and travels in the horizontal direction towards the target pixel. This propagation is continued until the propagation path encounters the column where the target pixel is located. Then the path is propagated in the vertical direction until it reaches the target pixel. Inspired by the abovementioned method, Huang *et al.* proposed a new eight-connected weight propagation scheme [91]. The propagation path starts from the source pixel either in horizontal or vertical direction until it reaches the pixel which is diagonal to the target pixel. Then the path is proceeded in the diagonal direction until it reaches the target pixel. The computational complexity of the guided filter depends on the size of the image and the number of disparity values used.

Gerrits and Bekaert proposed a colour segmentation-based cost aggregation method [92]. This method is based on the assumption that the depth discontinuities coincide with the object boundaries. In this method, the reference image is segmented using a mean-shift algorithm, and the support weights are computed on the basis of the pixel locations. Pixels which belong to the same segment as that of the center pixel are assigned one as a weight, while pixels lying outside the segment are given a weight value of zero. Tombari *et al.* also performs cost aggregation based on image segmentation. But, both the stereo images are considered for computing the support weights [93]. In this method, information of a pixel connectivity and shape of a segment is considered by using both the stereo images rather than using the information of colour and spatial distances. For this, pixels which occupy the same segment as that of the center pixel are assigned a weight value of one, while pixels lying outside the segment are assigned dynamic weights based on the colour distance. In [94], an efficient cost aggregation method using a colour segmentation method is presented. This method consists of two terms, one for segmentation, and another term is for weight correction. Segmentation term ensures that the shape of the support window is adapted according to the image local characteristics. The correction term adds additional weights to those pixels which are spatially near to the pixel under consideration, but it does not satisfy the segmentation assumption. Segmentation assumption tells that the abrupt change in the disparity values generally occurs at the object boundaries. Muninder *et al.* proposed pixel-based disparity map computation for the pixels of a segmented image region [95]. A set of plane label is generated with the help of initial disparity map and the segmented image regions. The cost for assigning a plane to each of the pixels is computed. Then this cost is aggregated by a spanning tree-based approach, and a label of a plane is assigned to each of the pixels. Segmentation-based

methods fail when the image segments overlap at the disparity discontinuities.

2.3.2 Problem of finding a disparity map for varying illumination

The general stereo correspondence methods are based on the assumption that the corresponding matching pixels have similar colour, which is termed as colour consistency or photometric assumption. In practical scenarios, this assumption does not always hold. This assumption is not fulfilled when there is a radiometric variation in the scene. This assumption is violated when the scene illumination, camera exposure, and its settings undergo a change. The performance of some of the matching functions discussed in Section 2.3 degrades for the abovementioned radiometric changes. To overcome this problem, some new matching functions are proposed. They are

- Gabor phase-based approach
- Adaptive normalized cross correlation (ANCC)
- Mutual information-based matching

These methods are briefly discussed in the following subsections.

2.3.2.1 Gabor phase-based stereo correspondence

Sanger used phase difference of the Gabor filter for disparity map computation [96]. This is due to the fact that the neurophysiological research on visual cortex of mammalian brains suggests that Gabor filter response has an analogous characteristics as that of the response shown by the cortical neurons. Hence, the neuron response can be modeled by a Gabor filter. One-dimensional Gabor filter can be expressed as follows:

$$G\left(x-x_0\right)=\exp\left\{-\frac{1}{2}\left(\frac{x-x_0}{\sigma}\right)^2\right\}\exp\left\{i\omega_0\left(x-x_0\right)\right\} \tag{2.25}$$

Here, x_0 is the spatial location of the filter, ω_0 is the frequency of the harmonic component, and σ is the spatial half-bandwidth of the filter. In this method, the left and right images are convolved with the above Gabor filter. Then phase difference is obtained from their respective convolved responses. It is represented as:

$$\Delta\phi=\phi_l-\phi_r \tag{2.26}$$

where ϕ_l and ϕ_r are the phase angles of Gabor filter responses for left and right images respectively. Now, disparity value d_ω at a particular scale ω is given by:

$$d_\omega = \Delta x_\omega = \frac{\Delta\phi}{\omega_0} \tag{2.27}$$

To span the entire spatial frequency spectrum, Gabor filter at different spatial scales are employed. The disparity map d_ω at different spatial scales ω are combined by a simple weighted averaging operation. The ambiguities in a phase-based disparity map estimation method occurs due to the presence of singularities in phase information. This leads to phase instability.

Fleet *et al.* found that magnitude of Gabor filter becomes unstable with respect to change in scale, and hence they are not suitable for stereo matching [97]. Phase information is stable for scale perturbation except for some regions which result in inaccurate disparity information. The primary cause of this instability is due to the existence of singularities in the phase information. Singularities in phase information corresponds to zero value in the Gabor magnitude information. The inaccuracy in disparity value can be avoided by detecting these singularities and their neighborhood. The neighborhood above and below singularities are detected by implying constraint on phase information at a particular scale. Similarly, the neighborhood to the left and right of singularity are detected by implying constraint on Gabor magnitude at a particular scale.

In [98], phase information is obtained by convolving the input images with the Gabor filter at different scales. The response of the Gabor filter corresponds to the central frequency of the filter. Since Gabor filter is a bandpass filter, it can measure responses from any frequency within this range. But a single Gabor filter is not sufficient enough to cover the whole frequency range of an image. To overcome this problem, image derivatives are used in the disparity map estimation method. Each image gradient is convolved with Gabor filter, and a disparity map is obtained. These disparity maps from different image derivatives are combined to obtain a final disparity map.

In [9], input images are convolved with a Gabor filter, and a disparity map is obtained using the phase difference. To avoid inaccuracy in the disparity map, phase singularities are detected using the constraints proposed in [97]. In addition to this, the phase wraparound problem is handled by finding the measured phase difference which is close to the ideal phase difference. In summary, it is seen that the Gabor phase-based methods have inherent disadvantage of phase singularities.

2.3.2.2 Adaptive normalized cross correlation

A variant of NCC which is termed as adaptive NCC is proposed in [99]. In this approach, weighted NCC is computed for each channel separately in both RGB and CIELab colour spaces. For this, weight is computed based on the colour and spatial distances from the center pixel, which is a bilateral filter. Subsequently, these two weighted NCCs are merged using a relative weight factor δ which is shown below:

$$C(\mathbf{p},d) = 1 - \left[\delta \sum_{\xi_1} \frac{ANCC_{\xi_1}(\mathbf{p})}{3} + (1-\delta)\sum_{\xi_2} \frac{ANCC_{\xi_2}(\mathbf{p})}{3}\right] \tag{2.28}$$

where

$$ANCC(\mathbf{p}) = \frac{\sum\limits_{\mathbf{q}\in\mathcal{N}_p}\sum\limits_{\mathbf{q}'\in\mathcal{N}_{p'}} w_l(\mathbf{p},\mathbf{q})w_r(\mathbf{p}',\mathbf{q}')\left(I_l(\mathbf{q})-\bar{I}_l\right)\left(I_r(\mathbf{q}')-\bar{I}_r\right)}{\sqrt{\sum\limits_{\mathbf{q}\in\mathcal{N}_p}\left|w_l(\mathbf{p},\mathbf{q})\left(I_l(\mathbf{q})-\bar{I}_l\right)\right|^2}\sqrt{\sum\limits_{\mathbf{q}'\in\mathcal{N}_{p'}}\left|w_r(\mathbf{p}',\mathbf{q}')\left(I_r(\mathbf{q}')-\bar{I}_r\right)\right|^2}} \tag{2.29}$$

Here, $\xi_1 \in \{\log(Chrom\_R), \log(Chrom\_G), \log(Chrom\_B)\}$, $\xi_2 \in \{R,G,B\}$, w_l and w_r are the weights for the window regions in the left and right images respectively.

2.3.2.3 Mutual information-based matching

Mutual information is used for stereo matching under radiometric changes. In [100], mutual information is used in window-based approach. But in this approach, a larger window is required to calculate joint probability distribution. As mentioned earlier, a larger window blurs the disparity boundaries. Kim *et al.* used pixel-based mutual information for matching in a graph cut approach [101]. Later, mutual information is employed in a hierarchical approach for computing stereo correspondence [102]. Here, it is shown that the accuracy of the disparity map produced by hierarchical approach is similar as that of the disparity map obtained from the iterative approach. Mutual information-based cost calculation fails to handle local radiometric changes [99].

2.4 Occlusion Detection and Filling

Occlusion occurs in an image when a portion of an object is hidden by another object in the scene. Due to this, some portions of an object is present in one image, while absent in another image in a stereo image pair. It happens due to the fact that a three-dimensional scene is viewed by two cameras from different viewpoints. In this scenario, it is difficult to find the matching pixels in the stereo image pairs by the methods described in the previous section. To obtain an accurate disparity map, firstly occluded pixels need to be identified, and subsequently appropriate disparity values have to

be assigned to the detected pixels. This process is called occlusion detection and filling for finding a stereo correspondence.

2.4.1 Occlusion detection

Spoerri and Ullman presented a histogram-based occlusion detection method [103–105]. In this method, if a pixel is occluded then its local neighboring pixels have two different disparity values. One disparity value corresponds to the occluding region, while the other disparity value corresponds to the occluded region. Hence, the histogram of this local region will have two peaks. In contrary to this, a local region from a single object produces only one mode or peak in the histogram. Therefore, the ratio of the second highest peak to the first highest peak gives the bimodal information. Analytically, it is given by:

$$\text{Bimodality in the histogram} = \frac{M_2}{M_1} \tag{2.30}$$

where M_1 and M_2 denote the first and second highest peak in the histogram. If the bimodality value is close to one, then it indicates that the pixel is from the occluded region.

Another occlusion detection method based on the correlation value is proposed in [106]. In this method, if a pixel is from a non-occluded region then the correlation value will be relatively high, while it is low for an occluded pixel. This is called match goodness jumps. A particular pixel is considered as an occluded pixel when the neighboring pixels have relatively high matching score with respect to the pixel under consideration.

In another occlusion detection scheme, pixel ordering (ORD) constraint is considered [45, 70]. Ordering constraint ensures that the order of a set of pixels in the reference image is maintained by the corresponding matching pixels in the other image. Violating this constraint indicates that a particular pixel is occluded.

Occlusion (OCC) constraint is used for occlusion detection [67]. Occlusion constraint states that disparity map is continuous over the entire image except near the object boundaries. There is a sudden change in the disparity value of the occluded surface with respect to the background. This sudden change in the right disparity map corresponds to an occlusion region in the left disparity map and vice versa.

Left right check (LRC) is another popular method to detect occluded pixels based on left right consistency verification. [107, 108]. In this method, if the disparity values of a pair of matching pixels are different, then the pixel in the reference image is considered as the occluded pixel. The x-coordinate p_1^{est} of estimated left pixel $\mathbf{p}^{est}$ corresponding to x-coordinate p_1' of pixel $\mathbf{p}'$ in the right image is given

by:

$$p_1^{est} = p_1' + d_{p'} \tag{2.31}$$

where $d_{p'}$ is the calculated horizontal disparity value of pixel $\mathbf{p'}$ with right image as reference. LRC detects a pixel as occluded, if the following condition is not satisfied:

$$p_1' - (p_1^{est} - d_{p^{est}}) = 0 \tag{2.32}$$

where $d_{p^{est}}$ is the disparity value of pixel $\mathbf{p}^{est}$ computed by considering left image as reference. It is to be mentioned that LRC is the most widely used occlusion detection algorithm for last few decades [10, 11, 41, 109]. LRC detects both occluded pixels as well as pixels having wrong disparity values.

Bimodality and match goodness jumps can detect the border occluded regions while LRC, ORD, and OCC algorithms can detect the entire half-occluded regions [110]. Though the abovementioned methods are able to detect the occluded pixels, these methods also detect many correctly matched pixels as the occluded pixels. Hence, these methods produce more false positives for some of the cases.

2.4.2 Occlusion filling

In stereo vision, a portion of an object is only present in any one of the stereo images and occluded in the other image. For the pixels present in these regions, finding the corresponding matching pixels is a difficult task. To tackle this problem, these pixels are explicitly detected and subsequently assigned disparity values. The task of assigning appropriate disparity value to an occluded pixel is termed as occlusion filling.

The basic and simple occlusion filling algorithm considers the disparity value of the closest horizontally left non-occluded neighborhood pixel with respect to the occluded pixel under consideration. It assigns the disparity value of this selected neighborhood pixel to the occluded pixel [111]. Another approach assigns minimum disparity value of the closest left d_l and right d_r horizontal non-occluded pixels [112]. This representation is expressed as follows:

$$D(\mathbf{p}) = \min\{d_l, d_r\} \tag{2.33}$$

where $\mathbf{p}$ is the occluded pixel to be filled up. This approach is similar to the previous approach as the disparity value of the closest horizontal left non-occluded pixel always have minimum disparity value among the disparity values of both left and right horizontal non-occluded pixels. However, these

two approaches introduce horizontal streaks as only the horizontal neighboring pixels are considered in this process. Weighted median filter is subsequently used to smooth these streaks. The weights of the median filter are computed based on the geodesic distance between the two pixels [112]. So, the drawback of this method is that it requires tuning of several manual parameters, and hence difficult to design optimal median filter.

Oh and Kuo performed occlusion filling by assigning the disparity value of the neighboring non-occluded pixels based on colour similarity [14]. This approach fails to address the order (left-to-right or right-to-left) in which occlusion filling is done. In neighbor disparity assignment (NDA) algorithm, border pixels are filled in right-to-left direction, while non-border pixels are filled in left-to-right direction. An occluded pixel is assigned the disparity value of the closest horizontal non-occluded pixel. Based on the order of filling, this closest pixel may be present in the left or right side of the occluded pixel [14].

Huq *et al.* proposed an occlusion filling algorithm which is based on the assumption that the occluded region along with its neighbors forms a planar surface [14]. Planarity is enforced by a linear model, and this algorithm uses least square approach to estimate the parameters of the linear model. It considers all the non-occluded neighbors or filled occluded pixels as the control points in weighted least square (WLS) approach. Some of these control points may belong to foreground objects and hence, their influence are suppressed by assigning the appropriate weights. These weights are calculated based on the colour similarity of the pixels. In this case also, occlusion filling is performed in the same order as that used in NDA algorithm. Another occlusion filling algorithm termed as segmentation-based least squares (SLS) is also proposed in [14]. In this algorithm, control points are segmented from the neighboring pixels based on the colour, visibility, and disparity gradient constraints. SLS performs better than WLS, but this method fails when the segments overlap the disparity borders.

2.5 Summary

In this chapter, existing stereo correspondence methods are briefly discussed by highlighting their advantages and disadvantages. These methods can be broadly classified into two categories: global-based and local-based approaches. In the global-based method, disparity value assigned to a particular pixel indirectly influences the disparity values of every other pixels. The reason behind the suboptimal performance of the global methods is due to the fact that the energy function fails to represent the stereo problem accurately. Also, it is difficult to find a suitable optimization strategy to minimize the energy function. Szeliski *et al.* presented a comparison of the energy minimization techniques [113].

It is shown that local energy minima can be obtained by employing the best performing optimization technique. This local minimum is actually close to the exact solution. Apparently, the degradation of the disparity map accuracy is due to the selection of a weak energy function. Additionally, this study shows that the energy of the ground truth image is higher than that of the minimum energy obtained by an optimization technique [38,114]. Hence, the selected energy function may not efficiently represent a particular stereo correspondence problem. Therefore, it is concluded that the performance of global algorithms are only influenced by the formulation of a energy function [115].

On the other hand, local methods give a dense disparity map and they are computationally efficient. The performance of local methods are comparable to that of the global methods after the incorporation of cost aggregation step in stereo correspondence methods. Local methods mainly suffer from two problems. One problem is due to the ambiguity in finding the matching pixels in the stereo image pairs, and the second problem arises on account of finding the correspondence near the object boundaries. Ambiguity in the disparity map mainly occurs due to the fact that the features employed in finding the matching pixels fails to discriminate between the matching pixel from its neighboring pixels. This can be reduced by selecting of appropriate features for matching. This consideration can significantly improve the accuracy of the computed disparity map. Cost aggregation step helps in estimating a relatively accurate and smooth disparity map around the object boundaries. But, the cost aggregation methods increase the computation burden.

For accurate disparity map estimation, occlusion regions are to be detected, and the suitable disparity values has to be assigned to the occluded pixels. Occlusion can be detected either implicitly or explicitly. In the first case, occluded pixels are detected along with stereo correspondence. In the second case, occluded pixels are detected after initial matching. Global algorithms detect occluded pixels in an iterative manner, while local methods perform occlusion detection and filling as a post processing step. In local methods, the next step after occlusion detection is to fill the occluded pixels with the satisfactory disparity values. As discussed in the previous section, Gabor filter can also be employed for finding the stereo correspondence. But, the Gabor filter has an inherent disadvantage of phase instability.

Based on the literature survey and its summary, the motivation and objectives of the thesis are framed in the sections to follow.

2.6 Motivation of the Thesis

Human vision system captures two different images of a scene with the help of slightly different views of the eyes. These two images are then processed by the human brain to visualize the depth information. Stereo vision which mimics human vision system is used in many computer vision applications such as augmented reality, robotic applications, human computer interaction, etc. Stereo vision consists of two cameras of same characteristics with different viewing angles separated by some distance. Disparity information is estimated by finding the corresponding matching pixels in the stereo images. Depth map can be obtained from the disparity information along with the camera parameters. Estimating disparity map is a difficult task due to the presence of textureless regions, occlusion etc. Hence, numerous research has been done on this topic. From the literature survey discussed in the previous sections, estimation of disparity map suffers from the drawbacks which are listed below:

(i) Existing stereo matching methods which use sum-of-absolute-differences, rank transform for finding the corresponding pixels in the stereo images cannot effectively distinguish a matching pixel from its neighbouring pixels in the target image [116]. So, a feature which can discriminate a particular pixel from its neighbourhood pixels is required. This is quite essential to find the correct matching pixels in the process of stereo matching.

(ii) One of the major problem in disparity map estimation is the selection of an appropriate window. Inappropriate window size may blur the discontinuities in the computed disparity map. These discontinuities can be preserved by employing suitable cost aggregation methods. The existing cost aggregation methods compute an accurate disparity map at the expense of computational complexity [10, 85, 87]. Therefore, a simple cost aggregation method which can preserve the edges of an image is required for disparity map estimation.

(iii) Existing occlusion detection algorithms perform occlusion detection by cross checking of the disparity values of both the stereo images [109]. This increases the computational burden as it requires disparity map computation for both left and right images. In addition to this, the existing methods introduce more false positives during the detection process.

(iv) The most widely used occlusion filling methods produce horizontal streaks in the disparity map [111, 112]. In existing occlusion filling algorithms, neighbouring non-occluded pixels are selected based on the colour information obtained from one of the images of the stereo image pairs [14]. This approach fails when the pixel to be filled and the neighbouring occluded pixel have similar

colour characteristics. This occurs when the occluded pixel is wrongly classified as non-occluded by the occlusion detection methods.

This motivates us to carry forward the research in this direction.

2.7 Objective of the Thesis

The goal of this research work is to estimate a fine disparity map by selecting an appropriate feature. The selected feature should provide almost similar information as visualized by the human vision system. It is also important to analyze the characteristics of the proposed feature by considering its associated parameters. The possibility of incorporation of simple edge preserving filter for cost aggregation is another objective of this research. The thesis also aims at developing occlusion detection algorithm by employing only one disparity map instead of two. This can be accomplished by analyzing the characteristics of pixels in the reference image and their corresponding matching pixels in the target image. Occlusion filling is the next step after occlusion detection. Occlusion filling will be done by utilizing both the stereo images to nullify the errors caused by colour similarity between an occluded pixel and the neighbouring non-occluded pixels.

3

Disparity Map Estimation using Spatial Domain Local Gabor Wavelet

The stereo matching problem takes two images captured by nearby cameras and attempts to recover disparity information. Most of the existing stereo matching algorithms cannot perfectly estimate disparity values at the discontinuous and texture-less regions in the images. This estimation problem is even more difficult in presence of occlusion. In the last few decades, a number of stereo matching methods are proposed to overcome some of these problems. In the same line of thought, a new feature-based stereo matching method is proposed, which consists of four basic steps – feature-based stereo correspondence, two-pass cost aggregation, disparity computation using Winner-Take-All (WTA) selection, and finally the disparity refinement. In our proposed method, local features extracted from Gabor wavelet in spatial domain are employed for matching cost computation, and subsequently cost aggregation step is implemented by combined use of Kuwahara and median filters. Experimental results on Middlebury benchmark database shows that the proposed method outperforms many existing stereo matching methods. This chapter gives an overview of general framework of stereo correspondence algorithms, and the proposed stereo correspondence method.

3.1 Introduction

The goal of stereo matching is to find the correspondences between the left and the right images captured by two cameras. Stereo correspondence problem can be simplified by using rectified stereo images taken at different viewpoints. The following subsections give an overview of the constraints used for finding stereo correspondence, and general framework of stereo correspondence algorithms.

3.1.1 Stereo matching constraints and assumptions

The ambiguity that arises during stereo correspondence problem can be reduced by using some of the constraints and assumptions. Some of these constraints depend on the geometry of the imaging system, photometric properties of the scene, and also on the disparity values. The most commonly used constraints and assumptions are listed as follows [1, 8]:

- **Epipolar constraint:** This constraint states that the matching point of a pixel in the left image lies in the corresponding epipolar line in the right image. This reduces the matching point search from two-dimensional space to one-dimensional space.

- **Uniqueness constraint:** This constraint claims that there exist at most one matching pixel in the right image corresponding to each pixel in the left image. Opaque objects satisfy this constraints, whereas transparent objects violate this. This is because of the fact that many points in a three-dimensional space project onto the same point in an image plane. Figure 3.1 shows a scenario where uniqueness constraint fails due to the above many to one mapping. Here, A, B, and C are three different points in a scene that are seen as a single point $\tilde{A}$ by the left camera, while these points are seen as three different points by the right camera.

- **Photometric compatibility constraint:** The intensity values of the pixels in a region in the left image and its corresponding matching region in the right image only slightly differ in intensity values. This slight difference in intensity values is due to the different camera positions from where the images are captured. Let us consider two regions U_l and U_r in left and right images respectively. Then the center pixels of these regions are matching pixels if and only if the following condition holds:

$$\exists_{\tau} \left| \sum_{\mathbf{p} \in \mathbf{U}_l} I_l(p_1, p_2) - \sum_{\mathbf{p}' \in \mathbf{U}_r} I_r(p_1', p_2') \right| < \tau \tag{3.1}$$

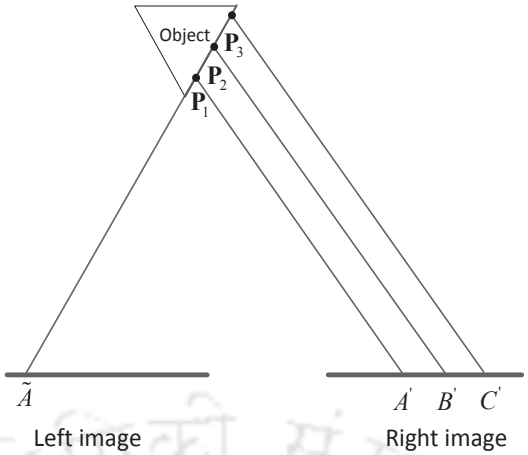


Figure 3.1: An example where uniqueness constraint fails.

where I_l and I_r are the left and right images respectively, $\exists$ denotes there exists, and τ is a user-defined threshold.

- **Geometric similarity constraint:** This constraint is based on the geometric characteristics such as length or orientation of a line segment, contours or regions of the matching pixels in left and right images. In details,

  (i) A segment S_l in the left image with spatial orientation θ_l corresponds to the segment S_r in the right image with spatial orientation θ_r if the following condition holds:

  $$|\theta_l - \theta_r| < \tau \tag{3.2}$$

  (ii) A segment S_l of length l_l in the left image corresponds to the segment S_r of length l_r in the right image if the following condition holds:

  $$|l_l - l_r| < \tau \tag{3.3}$$

  where τ is a threshold.

- **Ordering constraint:** Ordering constraint says that for regions having similar depth, the order of the pixels in the left image and the order of their matching pixels in the right image are same. Figure 3.2 shows two cases, one that satisfies ordering constraint and other does not. In Figure 3.2(a), ordering constraint is fulfilled as the points A, B, and C, and their corresponding matching points A', B', and C' follow the same spatial order. In Figure 3.2(b), this constraint

fails as the order of the points (A and B) in the left image is different from the order of the corresponding matching points (A' and B') in the right image.

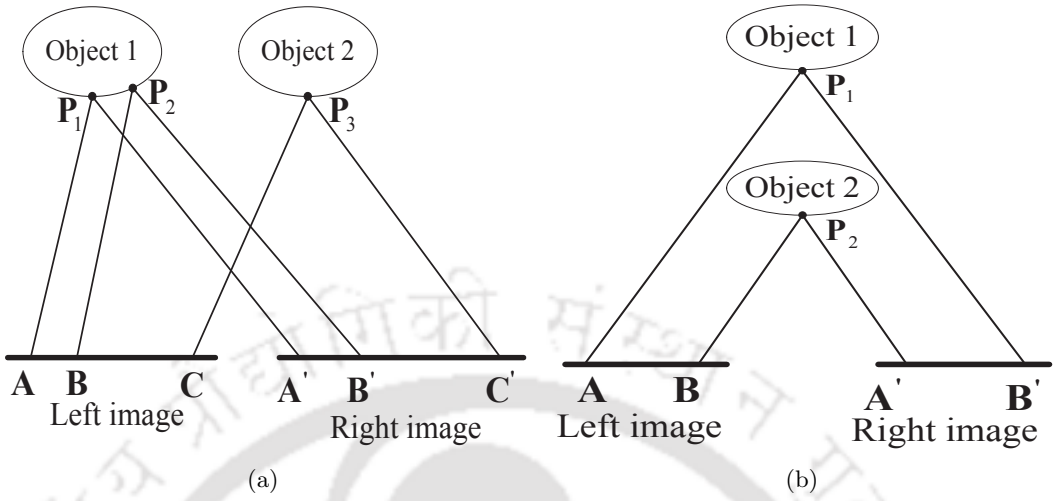


Figure 3.2: Illustration of ordering constraint in two scenarios.

- **Disparity continuity constraint:** This constraint states that there is an abrupt change in disparity values at the object boundaries, whereas the disparity values do not change significantly for smooth regions. Suppose $\mathbf{p}$ and $\mathbf{q}$ are two points in a reference image, where $\mathbf{q}$ is a neighbourhood point of $\mathbf{p}$. These two points correspond to points $\mathbf{p}'$ and $\mathbf{q}'$ in a target image with disparity values d_p and d_q respectively. This constraint says that the absolute difference of disparity values of these points should be small. This can be mathematically expressed as:

$$\exists_{\tau} |d_p - d_q| < \tau \tag{3.4}$$

where τ is a threshold. This constraint is not satisfied at the image boundaries.

- **Disparity limit constraint:** It imposes a global limit on the maximum allowable disparity value between the stereo images. This is based on the psychovisual experiments which say that the human visual system can only fuse the stereo images if the disparity values do not exceed a limit. Mathematically,

$$\exists_{\tau} |d_p| < \tau, \ \forall \mathbf{p} \in \mathbf{I}_l \tag{3.5}$$

where d_p is the disparity value between the pixel $\mathbf{p}$ and pixel $\mathbf{p}'$ in the left and right images respectively, and τ is a threshold.

- **Disparity gradient limit constraint:** Disparity gradient Γ is defined as the ratio of the

difference in the disparity values between two pair of corresponding points to their cyclopean distance. Mathematically, it can be expressed as follows:

$$\Gamma(\mathbf{A},\mathbf{B}) = \frac{d_p - d_q}{G(\mathbf{A},\mathbf{B})} \tag{3.6}$$

where $\mathbf{A} = \{\mathbf{p}, \mathbf{p}'\}$ and $\mathbf{B} = \{\mathbf{q}, \mathbf{q}'\}$ represents two corresponding matching points, d_p is the disparity value between the elements of $\mathbf{A}$, d_q is the disparity value between the elements of $\mathbf{B}$. $G(\mathbf{A},\mathbf{B})$ is a cyclopean distance between pair of points $\mathbf{A}$ and $\mathbf{B}$ and it can be expressed as:

$$G(\mathbf{A},\mathbf{B}) = \left| \frac{\mathbf{p}+\mathbf{p}'}{2}, \frac{\mathbf{q}+\mathbf{q}'}{2} \right| \tag{3.7}$$

$G(\mathbf{A},\mathbf{B})$ gives the length of the distance between middle points of the segments $\mathbf{p}$ and $\mathbf{p}'$ as well as $\mathbf{q}$ and $\mathbf{q}'$ respectively. This concept is illustrated in Figure 3.3. Now, the disparity gradient limit can be formulated as follows:

$$\exists_{\tau} |\Gamma(\mathbf{A},\mathbf{B})| < \tau \tag{3.8}$$

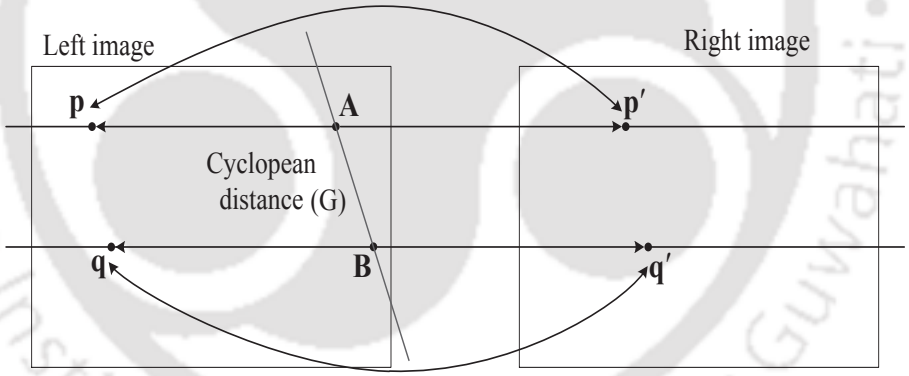


Figure 3.3: Cyclopean distance [8].

3.1.2 General steps of disparity map computation

In general, most of the existing stereo matching methods comprise the following four basic steps:

(i) Matching cost computation;

(ii) Cost aggregation;

(iii) Disparity computation/optimization; and

(iv) Disparity map refinement.

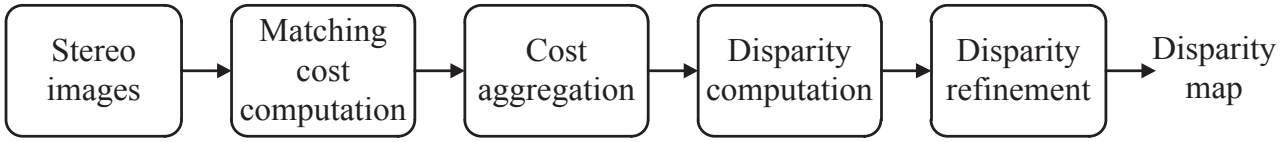


Figure 3.4: General steps of stereo correspondence methods.

Figure 3.4 shows the flowchart of disparity map estimation methods. All these steps are discussed in details as follows.

3.1.2.1 Matching cost computation

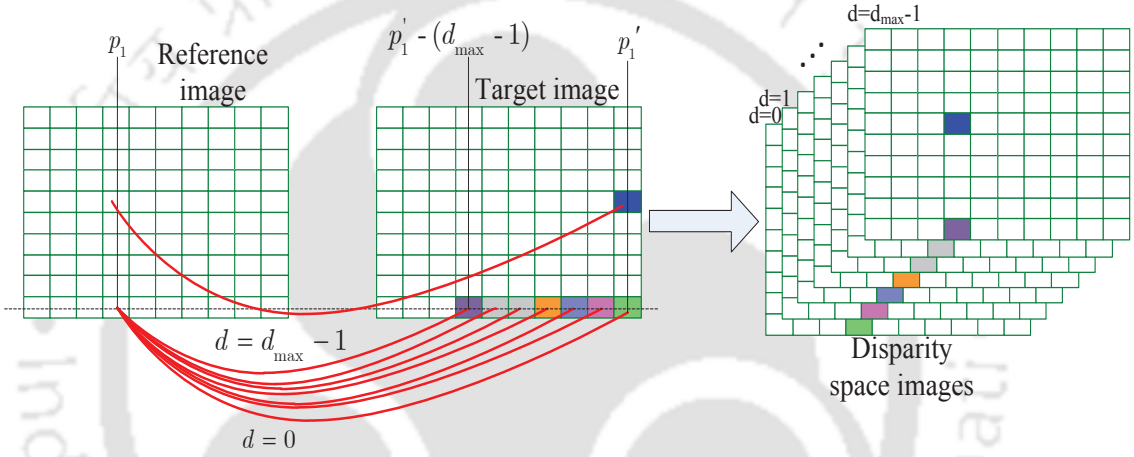


Figure 3.5: Matching cost computation.

Matching cost defines the closeness of a reference pixel to a candidate matching pixel *i.e.,* how much a pixel under consideration is similar or dissimilar with a candidate matching pixel. To find the disparity value, a pixel in a reference image is compared with a set of candidate matching pixels in a target image. The candidate matching pixels are selected on the basis of a range of disparity values. Usually features are used to find a matching. Features may be either intensity values themselves or some other information extracted from the intensity values. Matching can be performed either by pixel level or block level. In pixel matching, only feature of a particular pixel is used for computing cost, while the neighbouring pixels in addition to a particular pixel are also used for finding the matching cost in block-based matching. A survey of the cost functions used for matching was discussed in Section 2.3 of Chapter 2.

Figure 3.5 shows how matching cost is computed from a stereo image pair. For different disparity values, matching costs between pixels in reference and target images are computed. The higher the

cost value, the lower is the similarity between the pixels and vice-versa. For example, suppose the matching cost for pixel $\mathbf{p}$ in the reference image needs to be computed, then this pixel is compared with pixels $\mathbf{p}'$ of the target image for disparity range, $d = 0$ to $d_{max} - 1$, where d_{max} is the maximum disparity value. Suppose absolute difference is considered for performing matching then it can be expressed as follows:

$$C(\mathbf{p}, d) = |I_l(p_1, p_2) - I_r(p_1 - d, p_2)| \tag{3.9}$$

where I_l and I_r are the left and right images respectively, and d is the disparity value within the range $[0, d_{max} - 1]$. This computation gives an array of values for each of the pixels in the reference image for different disparity values. Hence, computing it for the entire image results in a three-dimensional volume of size $M \times N \times d_{max}$, where $M \times N$ is the size of the image, and d_{max} is the maximum allowable disparity range. This three-dimensional volume is called the disparity space image (DSI).

In Figure 3.5, pixel $\mathbf{p}$ in the left image is compared with a set of candidate matching pixels $\mathbf{p}'$ in the right image for different values of d. This corresponds to pixels shown by different colours in Figure 3.5. For example, if $d = 0$, pixel $\mathbf{p}$ in the left image is compared with the pixel $\mathbf{p}'$ in the right image (shown by green colour), which lies in the same horizontal line as the left pixel $\mathbf{p}$. Subsequently for $d = 0$, comparison is done for all the pixels in the reference image which in turn gives an image having the pixel values corresponding to the dissimilarity costs. Similar images are obtained for different disparity values. These images when stacked together form a three-dimensional matching cost volume termed as the disparity space image.

3.1.2.2 Cost aggregation

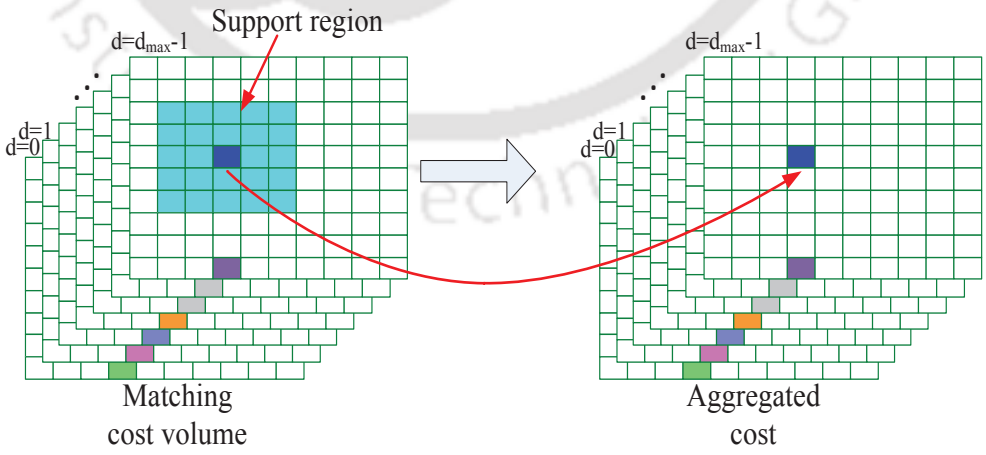


Figure 3.6: Cost aggregation.

The matching costs are combined over a local support region to preserve depth discontinuities. This step is also important to overcome edge flattening effects of local stereo correspondence methods. It is based on the assumption that all the pixels in the support regions have the same disparity value. This process is termed as cost aggregation. This step is essential to estimate an accurate disparity map near the discontinuous regions. Mathematically, a simple cost aggregation step is given by:

$$C_{agg}(\mathbf{p}, d) = \sum_{\mathbf{q} \in \mathcal{N}_p} C(\mathbf{q}, d) \tag{3.10}$$

where $\mathcal{N}_p$ is the support region of pixel $\mathbf{p}$. This can be illustrated as shown in Figure 3.6. Let us consider a pixel $\mathbf{p}$ shown by dark blue colour for $d = d_{max} - 1$. The cyan coloured pixels surrounding the pixel $\mathbf{p}$ are its neighbouring pixels *i.e.,* support region. Cost aggregation for pixel $\mathbf{p}$ is performed by combining the cost values of all the pixels in the support region.

In addition to this, cost aggregation can be performed by filtering the images in the disparity space image using edge preserving filters such as bilateral and guided filters. This step is similar to weighted average of the cost values over a small image region. Cost aggregation makes the performance of the local methods comparable to that of the global methods.

3.1.2.3 Disparity computation/optimization

The disparity map is obtained by determining the disparity d_p of all the pixels $\mathbf{p}$ in the reference image. In local stereo correspondence methods, disparity computation is accomplished by taking the index of the minimum value of the aggregated cost for a particular pixel (Winner-Take-All approach). Mathematically, the disparity d_p of a pixel $\mathbf{p}$ is given by:

$$d_p = \arg\min_{d} C_{agg}(\mathbf{p}, d) \tag{3.11}$$

where $C_{agg}(\mathbf{p}, d)$ is the aggregated matching cost of a pixel $\mathbf{p}$ for a disparity value d.

This concept is illustrated in Figure 3.7. The aggregated cost for a particular pixel $\mathbf{p}$ is shown (right side of Figure 3.7) as a graph plotted between disparity values and the associated costs. The disparity value corresponding to the minimum aggregated cost value is the final estimated disparity value corresponding to the pixel $\mathbf{p}$. Different colours namely green, pink, blue, red, and violet in the histogram correspond to different disparity values such as $d = 0, 1, 2, 3$ and $d_{max} - 1$ respectively for pixel $\mathbf{p}$ in the aggregated cost volume, while gray colour corresponds to the intermediate disparity values.

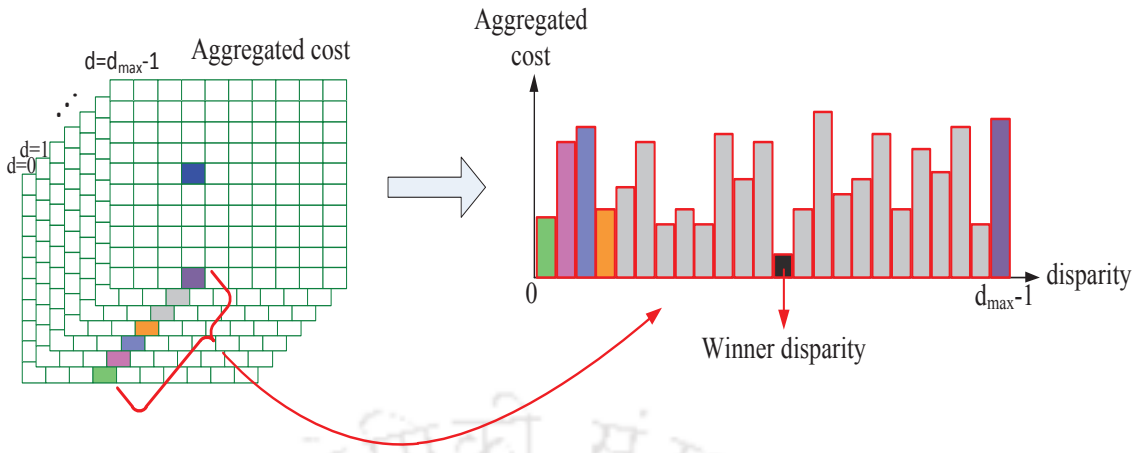


Figure 3.7: Disparity computation.

Global algorithms perform this step by minimizing a global cost function comprising data and smoothness terms. Minimization is performed using the optimization techniques. In global algorithms, the disparity value of a particular pixel is influenced by the disparity values of the neighbouring pixels.

3.1.2.4 Disparity map refinement

This is the post processing step which is done to obtain an accurate disparity map. It includes occluded regions detection, filling of appropriate disparity values to these detected regions, and finally performing refinement of the obtained disparity map.

Based on the abovementioned general steps of a stereo correspondence algorithm, we proposed a local stereo matching algorithm. The details of our method are discussed in the following section.

3.1.3 A brief overview of existing stereo correspondence algorithms

An excellent survey of the stereo matching methods and the taxonomy used for performance evaluation of the stereo correspondence algorithms is presented in [58]. Also, the Middlebury website provides benchmark datasets for the stereo matching, and comparative evaluation of many recent methods [117].

Stereo matching using non-parametric transform was also proposed by some researchers, in which a local non-parametric rank transform is applied to the stereo input images before performing the stereo correspondence [60]. Rank transform about a point on a fixed window is the number of pixels in the window having intensity values less than the center pixel. After applying the rank transform, matching is performed by SAD. An alternative to the above method is proposed in [60, 118], which incorporates

census transform for stereo matching. The limitation of both these methods is that the local measures depend on the intensity value of the center pixel. In addition to this, Histograms of Oriented Gradient (HOG) descriptors are also used for stereo matching [119]. A rectangular region around the pixel of interest is divided into smaller subregions, and HOG is computed for all the subregions. The concatenation of all HOG represents the feature of the considered pixel. The disadvantage of this method is that the size of the feature vector depends on the size of the rectangular and small subregions. Also, the size of the feature vector depends on the number of bins of the histogram. Hence, the size of feature vector is large.

Utilization of the adaptive support weights significantly improves the performance of the local stereo matching algorithms. Yoon and Kweon proposed a stereo matching method by using adaptive support weights (ASW) [10]. In this method, weights are assigned to the pixels within the correlation window, which are inversely proportional to the spatial distance and the colour dissimilarity from the center pixel. The support window must be sufficiently large for better accuracy. But, a larger window severally affects the algorithm in real time implementation.

Gerrits *et al.* proposed a cost aggregation method based on the mean shift-based colour segmentation of the reference image [92]. Pixels in the support and the target windows are partitioned into two disjoint sets. Pixels that belong to the same segment are given weight 1, and 0 is assigned to the remaining pixels. It is based on the assumption that pixels within the same segment are likely to have similar disparity values. Mean-shift segmentation adds computational overhead to the weighted cost aggregation.

A cost aggregation method using geodesic distance is proposed in [112]. In this adaptive weight algorithm, weight of a pixel in the correlation window is defined by computing the geodesic distance to the center pixel. Pixels having a short geodesic distance to the center pixel (*i.e.,* pixels having an approximately constant colour path to the center pixel) are given relatively high weights, whereas the pixels having long distance are assigned low weights. Performance of this method depends on the size of the support window.

Hosni *et al.* performed cost aggregation using a guided filter (GF) [11]. The computational complexity of this method depends on the size of the image and the number of labels used. De-Maeztu *et al.* proposed a cost aggregation method based on a linear model [120]. This cost aggregation method relies on the symmetric strategy, and it performs cost aggregation using both the input colour images. In this method, execution time depends on the size of the input image. The major difficulties faced by stereo correspondence methods is the computation of disparity at occluded, textureless, and discontinuities regions. Most of the existing local stereo matching algorithms suffer from all these problems,

which eventually affect the performance of the algorithms in terms of real-time implementation. In the last few decades, a number of stereo matching methods were proposed to overcome some of these problems. In view of this, we propose a new feature-based stereo matching method. The proposed method uses a local Gabor wavelet feature for matching cost computation. Subsequently, two-pass cost aggregation is performed by combined use of Kuwahara filter and median filter.

Major contributions of this chapter are highlighted as follows.

Matching cost computation:

- Gabor wavelet in spatial domain is applied to the input images for computing corresponding points in the given stereo pairs. The motivation behind using Gabor wavelet for disparity computation is as follows:
  - The simple cells of the visual cortex of mammalian brains are best modelled as a family of self-similar two-dimensional Gabor wavelets [121]. So, the extracted features by the proposed method will convey almost similar information as that of human visual system. Gabor feature can extract texture information [122].
  - Two-dimensional Gabor wavelet has good spatial localization, orientation selectivity, and frequency selectivity property. So, the features extracted in the proposed method have local and discriminative characteristics.
  - The proposed local Gabor wavelet feature differs from the existing Gabor features in two ways:
    (i) Existing Gabor feature methods apply Gabor wavelet to the entire image [123] or extracts Gabor feature at a particular point [124], whereas the proposed method applies Gabor wavelet to overlapping local image patches. Hence, the method is termed as local Gabor wavelet-based feature extraction method. The proposed method extracts feature for all the pixels in the input image. In general, local approach extracts the maximum information as compared to holistic approaches [125].
    (ii) The proposed method only uses real coefficients obtained after convolving the local image patches with Gabor wavelet, whereas existing Gabor feature-based methods use both real and imaginary coefficients *i.e.,* magnitude or phase information.
- Dimensionality of the local Gabor wavelet coefficients depends on number of orientations and scalings. Hence in the proposed method, PCA is used to reduce the dimensionality of the Gabor wavelet coefficients.

Cost aggregation:

- Cost aggregation method is implemented by filtering the matching cost volume with Kuwahara filter followed by median filter.
  - The motivation of using Kuwahara filter is that this filter can perform image smoothing by preserving the edges, and its computational complexity is $\mathcal{O}(1)$.
  - Median filter is applied to remove the blocking artifacts produced by Kuwahara filter.

Experimental evaluation of the proposed method clearly shows that the output disparity maps generated by the proposed method are qualitatively and quantitatively better than the existing methods.

3.2 Proposed Local Stereo Matching Method

The block diagram of the proposed local stereo matching algorithm is shown in Figure 3.8. There are four essential steps in the algorithm *viz.,* matching cost computation, cost aggregation, disparity computation, and finally the disparity map refinement [1].

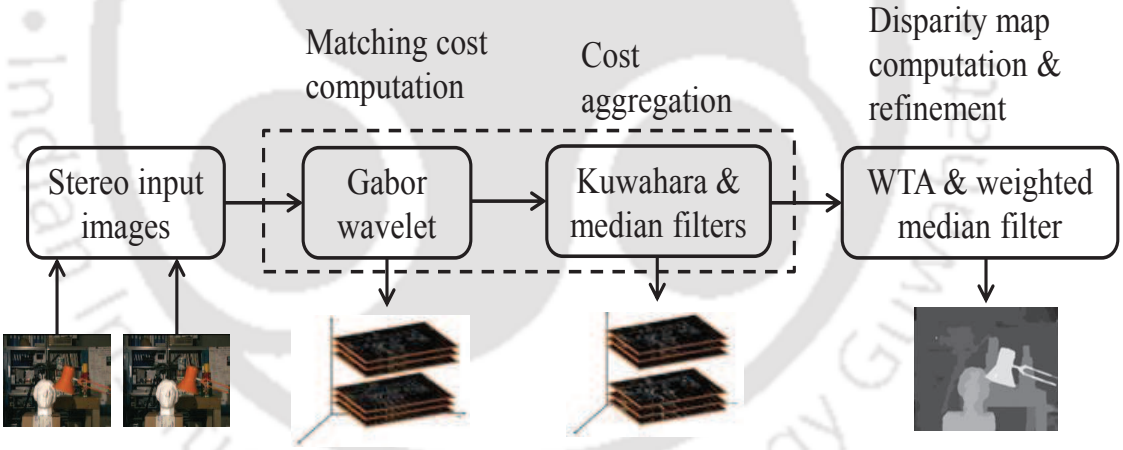


Figure 3.8: Block diagram of the proposed disparity map estimation method.

3.2.1 Matching cost computation

A feature-based stereo correspondence method using Gabor wavelet is proposed for matching cost computation. Gabor wavelet with different scales and orientations are applied over a small region around a pixel, and subsequently the extracted features are assigned to the center pixel. This step

[1]This work has been published in *IET Computer Vision 2015* (Refer item [1] in Page 135 for details)

is repeated for all the pixels of an image. Finally, these features are compared with the features in a target image to obtain the corresponding matching pixels.

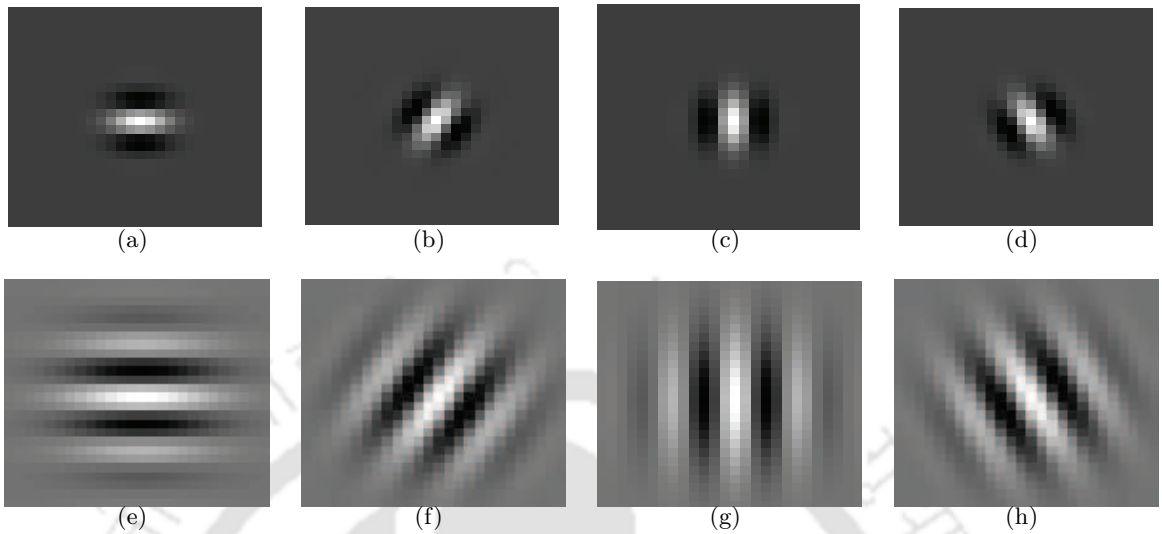


Figure 3.9: Gabor wavelet kernel (real part). (a)-(d) for scale 2; (e)-(h) for scale 5; (a) and (e) for theta 0°; (b) and (f) for theta 45°; (c) and (g) for theta 90°; (d) and (h) for theta 135°.

Gabor function approximates simple cells of the visual cortex of mammalian brains [121]. Image analysis by Gabor function approximately resembles the perception of human visual system. A two-dimensional Gabor function can be viewed as a sinusoidal plane wave modulated by a Gaussian function, which can be expressed as follows:

$$g(x,y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}\left(4x^2+y^2\right)} \left[e^{i\kappa x} - e^{-\frac{\kappa^2}{2}} \right] \tag{3.12}$$

In this Equation (3.12), $\kappa = \sqrt{2\ln 2}\left(\frac{2^{\phi}+1}{2^{\phi}-1}\right)$, where ϕ is the bandwidth in octaves. Gabor wavelet is generated by orientation and scaling of the two-dimensional Gabor function, which is given by the expression below [126]:

$$\begin{aligned} g_{mn}(x,y) &= a^{-m} g(x_a, y_a), \ a > 1, \\ x_a &= a^{-m}(x\cos\theta + y\sin\theta) \\ y_a &= a^{-m}(-x\sin\theta + y\cos\theta) \end{aligned} \tag{3.13}$$

where $\theta = \frac{n\pi}{K}$, m and n are two integers, and K is the total number of orientations. Figure 3.9 shows Gabor kernels for scales 2 and 5, while the angle of orientations is varied from 0° to 180° by an incremental step of 45°. Figure 3.10 shows the image represented using only real coefficients, only imaginary coefficients, magnitude and phase information extracted by using Gabor wavelet.

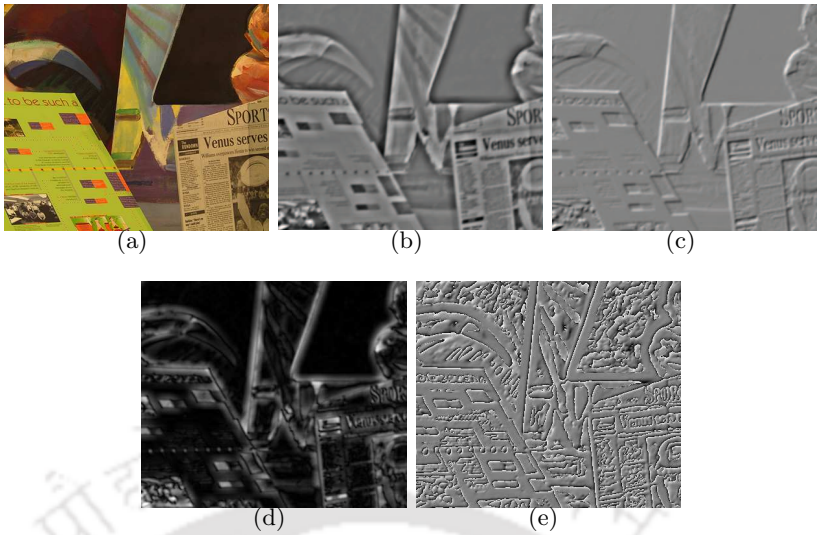


Figure 3.10: Image represented by using Gabor wavelet. (a) Left input image; (b) Image represented using only real coefficients; (c) Image represented using only imaginary coefficients; (d) Image represented using magnitude information; (e) image represented using phase information.

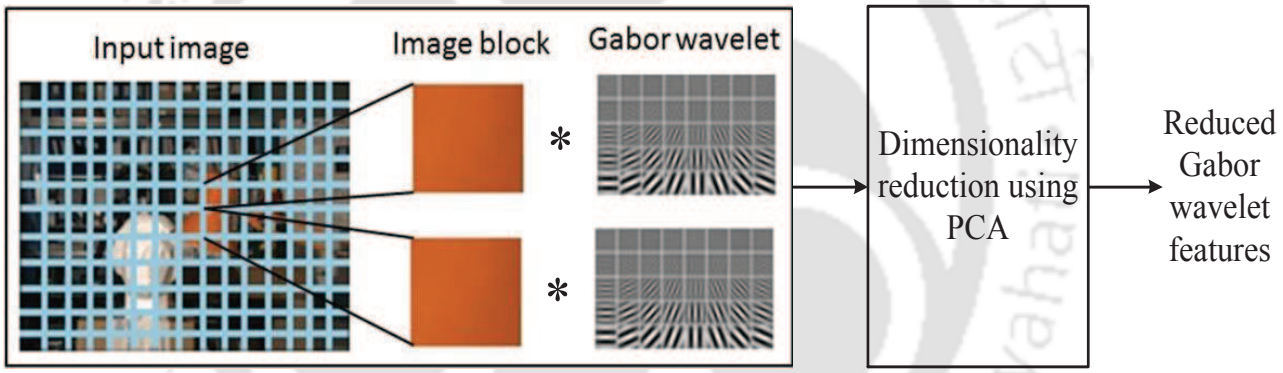


Figure 3.11: Local Gabor wavelet feature extraction.

In our proposed method, local features are extracted by applying Gabor wavelet in the spatial domain. The advantages of using Gabor wavelet in spatial domain are as follows [127]:

- Gabor wavelet in spatial domain permits image processing in a region of interest.
- Gabor wavelet in spatial domain is much faster than the conventional fast Fourier transform implementations.

In our method, local image features are extracted by convolving the image patches $\mathbf{S}_p$ with Gabor wavelet filter banks as shown in Figure 3.11, where " $*$ " denotes convolution operation. An input image I of size $M \times N$ is considered, and a small patch $\mathbf{S}_p$ with pixel $I(p_1, p_2)$ as center is convolved

with Gabor wavelet, which can be mathematically expressed as:

$$\begin{aligned}
\mathbf{G}_I^e(p_1,p_2) &= \left[\boldsymbol{\chi}_{0s} \otimes \boldsymbol{\chi}_{1s} \otimes \cdots \otimes \boldsymbol{\chi}_{(m-1)s}\right], \forall \mathbf{p} = (p_1,p_2) \\
\boldsymbol{\chi}_{rs} &= \left[vec\left(\mathbf{G}_{r1}^e\right) \otimes vec\left(\mathbf{G}_{r2}^e\right) \otimes \cdots \otimes vec\left(\mathbf{G}_{rn}^e\right)\right], s = 1,2,\cdots,n \\
\mathbf{G}_{mn}^e &= \mathbf{S}_p * \mathbf{g}_{mn}^e(x,y) \\
\mathbf{g}_{mn}^e(x,y) &= a^{-m}\mathbf{g}^e(x_a,y_a) \\
\mathbf{g}^e(x,y) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)}\left[\cos\kappa x - e^{-\frac{\kappa^2}{2}}\right]
\end{aligned} \tag{3.14}$$

where *vec* represents matrix to vector conversion operation, the symbol $\otimes$ denotes vertical concatenation of vectors, " $*$ " is a convolution operator, and $g_{mn}^e(x,y)$ is the real Gabor wavelet kernel. The number of coefficients N_c obtained after concatenation operation depends on the size of the image patch, number of orientations, and number of scalings. The input image can be almost perfectly represented with two or three orientations when the number of frequency steps per octave is one. Hence, minimum two orientations and one frequency per octave (one scaling level) are sufficient enough to extract Gabor features [121]. Same procedure is repeated for all the pixels in both the images. In our proposed method, principal component analysis (PCA) is subsequently used reduce the dimensionality of the extracted features [128] [2]. Let $\boldsymbol{\mu}_g$ and $\boldsymbol{\Sigma}$ denote mean and covariance matrix of $\mathbf{G}_I^e$. $\boldsymbol{\Phi}$ is the eigenvector of the covariance matrix $\boldsymbol{\Sigma}$. Dimensionality reduction is performed by retaining only n_c eigenvectors $\boldsymbol{\Phi}_{n_c}$ $(n_c < N_c)$ *i.e.,* the eigenvectors corresponding to the n_c largest eigenvalues are retained. This subset of eigenvectors form the basis vector of eigen subspace. Now, the input image $\mathbf{G}_I^e$ is projected onto this subspace to get the dimension reduced image $\mathbf{G}_{proj}$, which is given as:

$$\mathbf{G}_{proj} = \boldsymbol{\Phi}_{n_c}^T\left(\mathbf{G}_I^e - \boldsymbol{\mu}_G\right) \tag{3.15}$$

The obtained dimensionality reduced coefficients represent the local features, and these coefficients are termed as local Gabor wavelet feature vector (GWFV). In the proposed method, these feature vectors are used to find the correspondence between the left and right images. Gabor wavelet coefficient are complex, but the proposed method makes use of only the real coefficients. In this context, it is mentioned that the receptive field profiles of human visual system can be best modelled by either real or imaginary Gabor features [129]. Real coefficients of Gabor extracts the texture information, while imaginary coefficients represent the edge information [130]. So, only the real coefficients of Gabor wavelet are used in the proposed method (refer to Equation 3.14). Table 3.1 shows the correlation co-

[2] This work has been published in *Matrix Information Geometry 2013* (Refer item [4] in Page 135 for details)

efficients of input Teddy image with image reconstructed by using only real coefficients, only imaginary coefficients, and magnitude information for $m = 2, n = 2$ in Equation (3.13). From the correlation

Table 3.1: Correlation coefficients computed for Teddy image

Components	Correlation coefficients
Real coefficients	0.9043
Imaginary coefficients	0.3318
Magnitude of coefficients	0.8986

coefficients, it is observed that image represented using only real coefficients shows more correlation with the input image as compared to the image represented using only imaginary coefficients. Also, the reconstruction performance of real coefficients is comparable to the reconstruction performance of magnitude information. Experimentally, it is found that real coefficients alone is sufficient to produce a good disparity map. Figure 3.12 shows the disparity map of Tsukuba image produced by the proposed method using only the real coefficients, only imaginary coefficients, and using both the real and imaginary coefficients. The equations used to obtain local GWFV using only imaginary coefficients are expressed as follows:

$$\begin{aligned}
\mathbf{G}_I^o(p_1,p_2) &= \left[\boldsymbol{\chi}_{0s} \otimes \boldsymbol{\chi}_{1s} \otimes \cdots \otimes \boldsymbol{\chi}_{(m-1)s}\right], \forall \mathbf{p} = (p_1,p_2) \\
\boldsymbol{\chi}_{rs} &= \left[vec\left(\mathbf{G}_{r1}^o\right) \otimes vec\left(\mathbf{G}_{r2}^o\right) \otimes \cdots \otimes vec\left(\mathbf{G}_{rn}^o\right)\right], s = 1,2,\cdots,n \\
\mathbf{G}_{mn}^o &= \mathbf{S}_p * \mathbf{g}_{mn}^o(x,y) \\
\mathbf{g}_{mn}^o(x,y) &= a^{-m}\mathbf{g}^o(x_a,y_a) \\
\mathbf{g}^o(x,y) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left[\sin \kappa x\right]
\end{aligned} \tag{3.16}$$

The local GWFV extracted using magnitude information is given by Equation (3.17):

$$\begin{aligned}
\mathbf{G}_I^a(p_1,p_2) &= \left[\boldsymbol{\chi}_{0s} \otimes \boldsymbol{\chi}_{1s} \otimes \cdots \otimes \boldsymbol{\chi}_{(m-1)s}\right], \forall \mathbf{p} = (p_1,p_2) \\
\boldsymbol{\chi}_{rs} &= \left[vec\left(\mathbf{G}_{r1}^a\right) \otimes vec\left(\mathbf{G}_{r2}^a\right) \otimes \cdots \otimes vec\left(\mathbf{G}_{rn}^a\right)\right], s = 1,2,\cdots,n \\
\mathbf{G}_{mn}^a &= \sqrt{\left(\mathbf{G}_{mn}^e\right)^2 + \left(\mathbf{G}_{mn}^o\right)^2} \\
\mathbf{G}_{mn}^e + i\mathbf{G}_{mn}^o &= \mathbf{S}_p * \left[\mathbf{g}_{mn}^e(x,y) + i\mathbf{g}_{mn}^o(x,y)\right] \\
\mathbf{g}_{mn}^e(x,y) + i\mathbf{g}_{mn}^o(x,y) &= a^{-m}\mathbf{g}(x_a,y_a) \\
\mathbf{g}(x_a,y_a) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left[e^{i\kappa x} - e^{-\frac{\kappa^2}{2}}\right]
\end{aligned} \tag{3.17}$$

The regions enclosed by yellow rectangular boxes in Figure 3.12 are judiciously selected to demon-

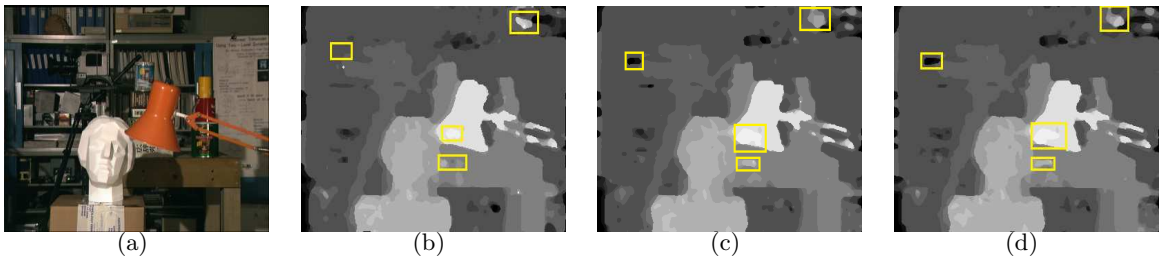

(a) (b) (c) (d)

Figure 3.12: Role of real and imaginary coefficients of Gabor wavelet on disparity map. (a) Left input image; (b) Disparity map generated using only real coefficients; (c) Disparity map generated using only imaginary coefficients; (d) Disparity map generated using both the real and imaginary coefficients.

strate the performance of disparity map estimation by using only real coefficients, only imaginary coefficients, and finally using magnitude information. Performance of disparity map estimated using only real coefficients, only imaginary coefficients, and magnitude information are evaluated, and the results are listed in Table 3.2. In this Table, *Nocc*, *all* and *disc* represents the percentage of bad pixels in the non-occluded region, entire image, and discontinuous regions respectively. It is quite

Table 3.2: Quantitative evaluation - Role of real and imaginary coefficients of Gabor wavelet on disparity map

Real part only			Avg.	Imaginary part only			Avg.	Magnitude of coefficients			Avg.
Nocc	*all*	*disc*		*Nocc*	*all*	*disc*		*Nocc*	*all*	*disc*	
4.81	5.73	14.6	8.38	5.23	6.07	14.5	8.6	5.16	6.06	14.7	8.64

evident that the disparity map computed using only the real coefficients produce almost comparable results with the disparity map computed only using the imaginary coefficients, and also using both the real and imaginary coefficients (magnitude information). On the other hand, memory requirement for overall computation is reduced by half when only the real coefficients are used [131]. That is why, only the real coefficients is used to produce the disparity maps in our proposed method. Disparity map is generated (without performing the cost aggregation and refinement) using the proposed Gabor wavelet feature vector and compared with the existing stereo matching cost methods namely SAD, SSD, NCC, adaptive NCC (ANCC) [99], rank transform [60], gradient-based method [132], and the corresponding percentages of bad pixels are shown in Table 3.3. This shows that the matching performed by the proposed feature vector outperforms the existing stereo matching cost methods. This improved performance of the proposed feature is due to the fact that the directional features are extracted at different orientations and scalings.

Table 3.3: Comparison of the proposed Gabor wavelet feature vector with existing metrics used for stereo matching

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
GWFV	7.11	9.16	24.2	5.44	6.99	37.3	13.5	22.4	33.8	8.92	18.8	23.9	17.6
Rank [60]	11.2	12.9	25.1	8.06	9.62	31.8	17.4	26.0	35.7	13.3	23.1	28.7	20.2
NCC [116]	12.1	13.9	29.0	7.67	9.25	39.1	16.4	25.0	39.7	12.9	22.7	32.6	21.7
SSD [116]	11.2	13.1	27.8	9.25	10.8	39.0	22.3	30.3	39.2	15.3	24.9	31.9	22.9
SAD [116]	10.3	12.3	22.8	10.1	11.6	35.4	24.1	31.9	36.6	19.7	28.7	31.1	22.9
Gradient [132]	11.0	13.0	30.5	11.6	13.1	44.7	18.2	26.7	41.8	13.9	23.7	34.8	23.6
ANCC [99]	12.5	14.6	34.4	25.5	26.7	46.7	38.5	44.7	51.1	20.7	29.6	37.2	31.8

3.2.2 Cost aggregation

Local stereo correspondence methods perform matching by computing match measure for a particular pixel by comparing gray or colour pixel values. Also, feature extracted from the region around a pixel of interest can be used for matching. This fixed support region results in disparity inaccuracies near boundary, and also in textureless regions. Several adaptive support window methods have been proposed with different sizes and shapes of the support window [77, 83, 87, 88, 133]. Although these methods improve the performance of stereo correspondence, they all have a common problem. The shape of the support window is not generalized, and also finding an optimal window with an arbitrary shape and size for each of the pixels is quite difficult. Hence these methods provide constraints to the shape of the support windows, and hence the estimation may be inaccurate in the boundaries. To overcome these limitations, cost aggregation is performed on the disparity space images. Cost aggregation performs smoothing of the disparity space images, and at the same time the operation must preserve the edges. Block and Gaussian filters perform smoothing by blurring of the edges. Similarly, weighted median filter also shows poor performance [83]. Computational complexity of bilateral filter depends on the support window size, which is large (35x35) to handle textureless regions. Many algorithms are proposed to increase computational speed, but these methods sacrifice quality in return of computational speed. Guided filter has edge-preserving property and a runtime independent of the filter size, but it suffers from halos near edges as shown in Figure 3.15(b) and 3.15(c) [134].

The proposed method adopts two staged cost aggregation *i.e.,* Kuwahara filter which is followed by median filtering. The edge-preserving property and runtime independent of support window size motivates us to use Kuwahara filter for cost aggregation. Median filter is subsequently applied to remove the blocking artifacts produced by Kuwahara filter. Consider $\mathbb{C}$ as the cost volume (disparity

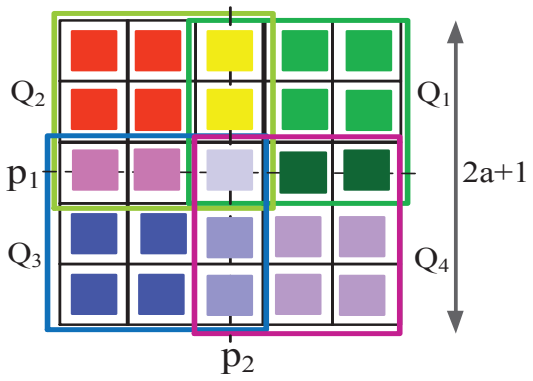


Figure 3.13: Subregions of Kuwahara filtering.

space images) of size $M \times N \times d_{max}$ and $\mathbf{C}_i$ as the i^{th} image of the matching cost volume, where $M \times N$ is the size of input image and d_{max} is the maximum allowable disparity range. Each of the disparity space images is smoothed by Kuwahara filter. Now for all $\mathbf{C}_i$, a square window $\mathbf{S}_k$ of size $2a+1$ centered around the point (p_1, p_2) in the image is considered. Subsequently, this square is partitioned into four identical subregions Q_1, Q_2, Q_3, and Q_4 as shown in Figure 3.13. Each subregion is given by:

$$Q_i(p_1,p_2) = \begin{cases} [p_1, p_1+a] \times [p_2, p_2+a], \text{ if } i=1 \\ [p_1-a, p_1] \times [p_2, p_2+a], \text{ if } i=2 \\ [p_1-a, p_1] \times [p_2-a, p_2], \text{ if } i=3 \\ [p_1, p_1+a] \times [p_2-a, p_2], \text{ if } i=4 \end{cases} \tag{3.18}$$

where " $\times$ " denotes the cartesian product [135]. Let m_i and σ_i be mean and standard deviation of the four subregions $Q_i, i = 1, ..., 4$ respectively. The output K_f of Kuwahara filter for a pixel (p_1, p_2) is given by mean m_i corresponding to the subregion having minimum standard deviation σ_i. This can be formulated as follows:

$$K_f(p_1,p_2) = \sum_i m_i(p_1,p_2) f_i(p_1,p_2) \tag{3.19}$$

where

$$f_i(p_1,p_2) = \begin{cases} 1, \ s_i(p_1,p_2) = \min\limits_k s_k(p_1,p_2) \\ 0, \ \text{otherwise} \end{cases} \tag{3.20}$$

This step is repeated for all the images in the cost volume and it is denoted as $\mathbb{C}_k$.

In the proposed method, median filter is applied to remove the blocking artifacts produced by Kuwahara filter in the cost aggregation step. Suppose $\mathbf{C}_{ki}$ is the i^{th} image of the cost volume after Kuwahara filtering. A small patch $\mathbf{S}_m$ of size $p1 \times q1$ with the pixel $C_{ki}(p_1, p_2)$ as center is considered

to find the median value *med*. Median value is obtained by arranging the intensity values in the small patch in ascending order and taking the median of these values. Finally, this value is assigned to the center pixel. This procedure is repeated for all the pixels.

$$med(p_1, p_2) = median(vec(\mathbf{S}_m)) \tag{3.21}$$

The above step is performed on all the images of $\mathbb{C}_k$ and the obtained cost aggregated volume is denoted by $\mathbb{C}_{agg}$.

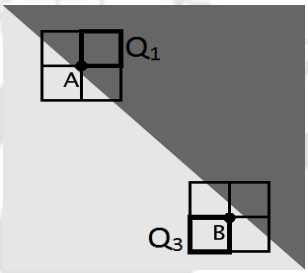


Figure 3.14: Behaviour of Kuwahara filter at boundary regions.

Figure 3.14 shows the behaviour of Kuwahara filter at boundary regions. When the center pixel (p_1, p_2) lies on the darker side of the edge (point A), the output takes the value m_1. Q_1 is the most homogeneous subregion that lies completely on the darker side of the edge having minimum variance. In this, m_1 is the local average corresponding to minimum variance σ_1 of subregion Q_1. Similarly, when the center pixel (p_1, p_2) lies on the brighter side of the edge (point B), the output takes the value m_3. This is due to the fact that the entire subregion Q_3 lies on the brighter side of the edge and this subregion is homogeneous having minimum variance σ_3. Assigning local average of a subregion having minimum variance to the center pixel guarantees edge preservation in the output. Table 3.4

Table 3.4: Comparison of the proposed method with and without median filter

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
Without MF [3]	5.08	6.02	14.5	1.99	2.79	16.4	10.3	18.4	24.7	5.62	14.5	15.7	11.3
With MF	4.81	5.73	14.6	1.66	2.15	13.0	9.18	13.4	22.3	5.03	12.1	14.8	9.89

shows the quantitative evaluation of the proposed method with and without median filter. It is seen that error is significantly reduced due to the use of median filter in the proposed method. To show the performance of the proposed cost aggregation method, guided filter is used in place of Kuwahara and median filter combination for cost aggregation [11], and the comparative result is shown in Table

3.5. Figure 3.15 shows the disparity space image for $d = 1$ filtered by guided filter with window size 3×3 (GF3) and 9×9 (GF9). It also shows the filtering by Kuwahara filter with window size 9×9. The poor performance of the guided filter is mainly due to the fact that the guided filter uses averaging strategy for overlapping windows, and hence filtering suffers from halos effect near edges which ultimately affects the smoothing operation. On the other hand, Kuwahara filter does not use averaging operation.

Table 3.5: Comparison of the proposed cost aggregation method with guided filter-based cost aggregation

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
GF3	9.53	11.1	37.0	7.02	8.29	43.6	15.9	23.5	37.3	11.3	20.3	29.1	21.2
GF9	9.55	11.2	37.4	9.59	10.7	43.9	20.1	25.6	41.2	16.0	24.4	34.1	23.7
Proposed method	4.81	5.73	14.6	1.66	2.15	13.0	9.18	13.4	22.3	5.03	12.1	14.8	9.89

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

Figure 3.15: Disparity space image filtering. (a) Disparity space image ($d = 1$); (b) Filtering by guided filter (window size - 3×3); (c) Filtering by guided filter (window size - 9×9); (d) Filtering by Kuwahara filter (window size - 9×9).

3.2.3 Disparity computation

The disparity map is obtained by determining the disparity d_p of all the pixels $\mathbf{p}$ in the reference image. This is accomplished by taking the index of the minimum value in the aggregated cost. Mathematically, the disparity value d_p of a pixel $\mathbf{p}$ is given by:

$$d_p = \arg\min_d C_{agg}(\mathbf{p}, d) \tag{3.22}$$

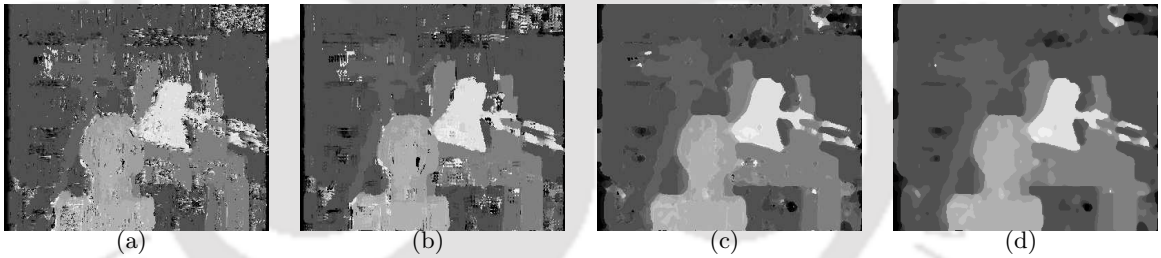
where $C_{agg}(\mathbf{p}, d)$ is the aggregated matching cost of a pixel $\mathbf{p}$ at disparity d.

3.2.4 Disparity map refinement

In the proposed method, occluded pixels are detected by using left-right consistency check. If the test fails, then a particular pixel is marked as occluded. In occlusion filling step, $min(d_l, d_r)$ is assigned to an occluded pixel, where d_l and d_r are the disparity values of neighbouring non-occluded left and right pixels. Disparity refinement is performed by a constant time weighted median filter [136]. The weights are calculated by the guided filter. The weights $w(\mathbf{p}, \mathbf{q})$ are given by:

$$w(\mathbf{p}, \mathbf{q}) = \frac{1}{|\mathcal{N}_p|^2} \sum_{\mathbf{q} \in \mathcal{N}_p} \left[1 + \left(\mathbf{I}_p - \boldsymbol{\mu}_p \right)^T \left(\boldsymbol{\Sigma}_p + \varepsilon \mathbf{U} \right)^{-1} \left(\mathbf{I}_q - \boldsymbol{\mu}_p \right) \right] \tag{3.23}$$

where $\boldsymbol{\mu}_p$ and $\boldsymbol{\Sigma}_p$ are the mean vector and the covariance matrix of all the pixels in the window $\mathcal{N}_p$, and $\mathbf{U}$ is a 3×3 identity matrix. $|\mathcal{N}_p|$ is the number of pixels in the window $\mathcal{N}_p$, and ε is a user-defined smoothness parameter. Figure 3.16 shows the output at intermediate stage of the proposed disparity map computation method. This includes the disparity map computed without cost aggregation, the disparity map obtained after cost aggregation only by Kuwahara filter, the disparity map obtained after cost aggregation using the combination of Kuwahara and median filters, and the final disparity map obtained after refinement.



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

Figure 3.16: Intermediate results. (a) Disparity map computed without cost aggregation; (b) Disparity map obtained after cost aggregation by only Kuwahara filter; (c) Disparity map obtained after cost aggregation using the combination of Kuwahara and median filters (before refinement); (d) Final disparity map obtained after refinement.

3.3 Datasets used for Evaluation

Evaluation of stereo correspondence algorithms are accomplished using the Middlebury stereo datasets. Two datasets are used for extensive evaluation. First set comprises standard Middlebury stereo dataset [7, 58] consisting of images Tsukuba, Venus, Teddy, and Cones as shown in Figure 3.17, while the other set (2005 Middlebury stereo dataset [12, 13]) includes images - Cloth1, Books, Dolls, Laundary, Moebius, and Reindeer as shown in Figure 3.18. All the abovementioned images were

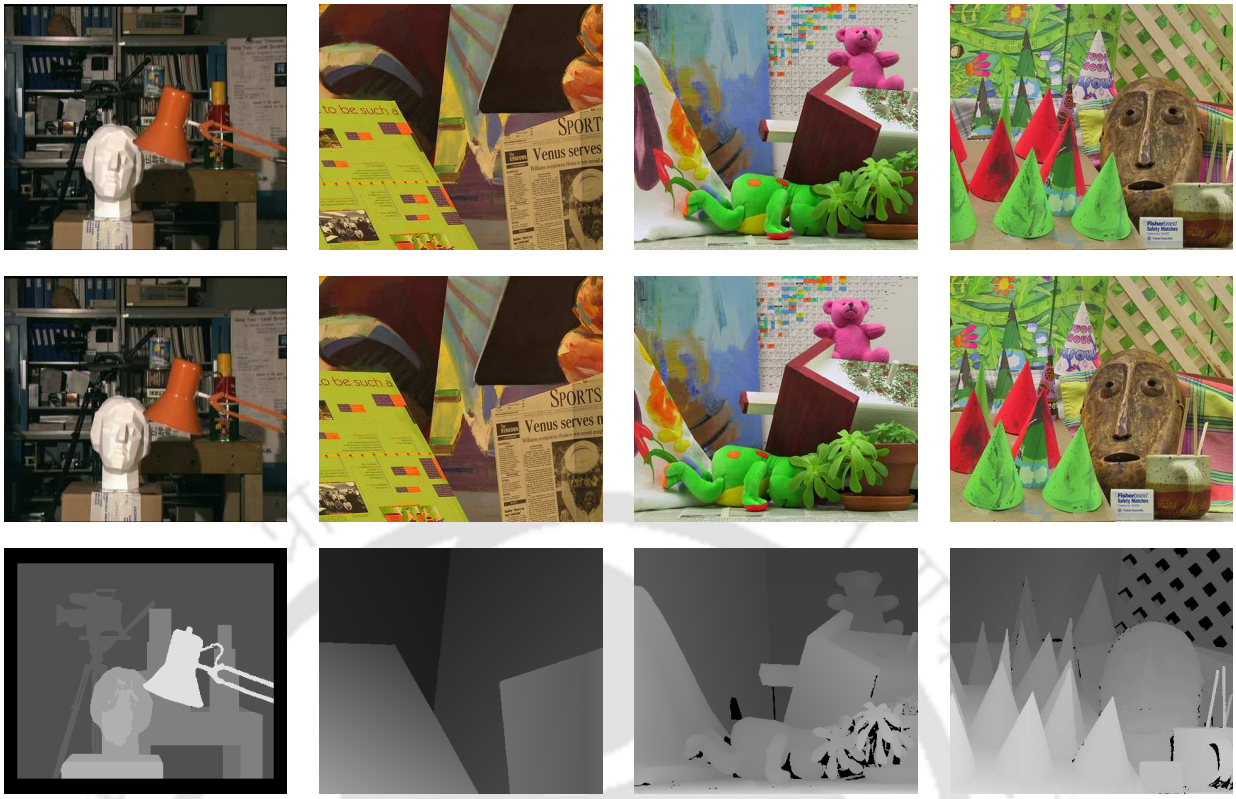


Figure 3.17: Middlebury stereo standard dataset. Left to right - Tsukuba, Venus, Teddy, and Cones images. Top to bottom - Reference images, target images, and ground truth disparity maps.

captured in a laboratory setup by using standard camera settings and normal lighting conditions. Tsukuba image contains only fronto-parallel surfaces, while Venus is a simple image with slanted surface *i.e.,* piecewise linear surfaces [137]. Teddy and Cones images are challenging compared to the above two images owing to large disparity range, complex surface shapes, textureless areas, narrow occluding objects, and ordering-constraint violations [7]. The parameters used for this dataset are shown in Table 3.6. 2005 Middlebury stereo dataset is more complicated compared to the stan-

Table 3.6: Parameters used for standard Middlebury stereo images

Images	Image size	Disparity range	Scale factor	Border pixels
Tsukuba	288×384	0-15	16	18
Venus	383×434	0-19	8	10
Teddy	375×450	0-59	4	0
Cones	375×450	0-59	4	0

dard Middlebury stereo dataset because of high disparity range, lack of textures and the presence of

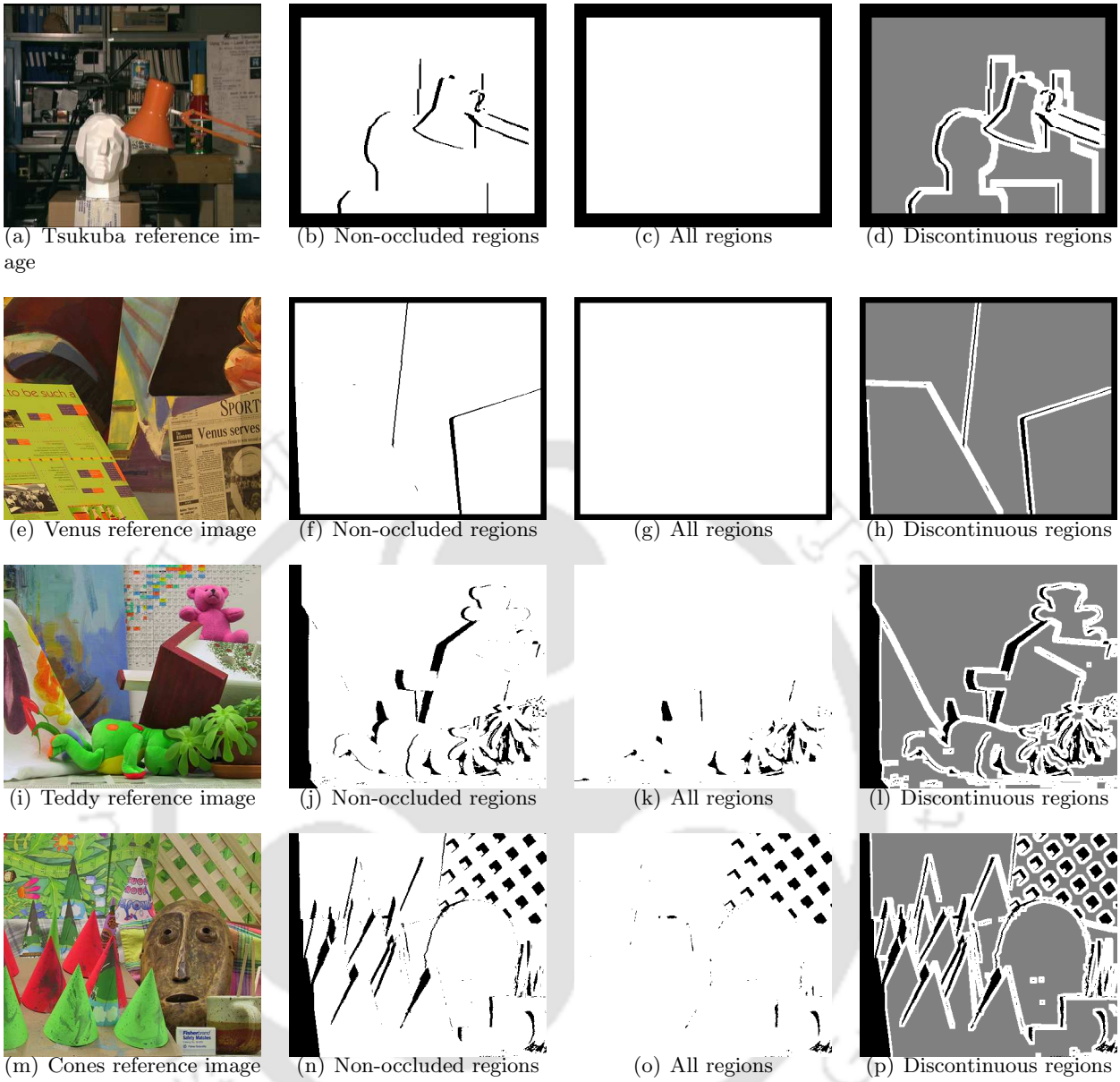
Figure 3.18: Middlebury stereo dataset (2005). Left to right - Cloth1, Books, Dolls, Laundary, Moebius, and Reindeer images. Top to bottom - Reference images, target images, and ground truth disparity maps.

complicated scene geometry. The disparity range for these images is 0-79.

3.4 Evaluation Methodology

Percentage of bad pixels [58] is widely used to quantitatively evaluate the accuracy of the stereo correspondence algorithms. It is the average number of pixels having estimated disparity values greater than the ground truth disparity values by a particular threshold. This measure is computed at three image regions namely non-occluded, all, and discontinuous regions. These three image regions are shown in Figure 3.19, where only the white regions are used for evaluation, while black and gray regions are ignored. The evaluation measures for these regions are briefly described below:

(i) Non-occluded regions : These regions comprise pixels that are present in both the stereo images, while the half-occluded regions are ignored. In addition to this, unknown regions (*i.e.,* regions for which ground truth disparity values are not available) are also not considered for evaluation. These regions are represented by $\bar{O}$. In Figure 3.19, second column shows the non-occluded regions. In these images, white-coloured regions are non-occluded, while black-coloured regions are either occluded or unknown regions.



(a) Tsukuba reference image (b) Non-occluded regions (c) All regions (d) Discontinuous regions

(e) Venus reference image (f) Non-occluded regions (g) All regions (h) Discontinuous regions

(i) Teddy reference image (j) Non-occluded regions (k) All regions (l) Discontinuous regions

(m) Cones reference image (n) Non-occluded regions (o) All regions (p) Discontinuous regions

Figure 3.19: Middlebury stereo dataset showing non-occluded, all, and discontinuous regions.

Percentage of bad matching pixels over non-occluded regions ($\bar{O}$) is given by below formula:

$$B_{\bar{O}} = \frac{1}{N_{\bar{O}}} \sum_{(p_1,p_2)\in\bar{O}} (|d_c(p_1,p_2) - d_t(p_1,p_2)| > \delta_d) \tag{3.24}$$

where $d_c(p_1,p_2)$ and $d_t(p_1,p_2)$ are the computed and ground truth disparity values respectively, $\bar{O}$ represents the pixels in the non-occluded regions, $N_{\bar{O}}$ is the number of pixels in non-occluded regions, and δ_d is the error threshold.

(ii) All regions : In all regions, both non-occluded and occluded regions are used for evaluation. In Figure 3.19, third column shows the all regions. In this figure, white colour indicates the entire image, while black colour is assigned for unknown regions.

Percentage of bad matching pixels over whole image (all) is calculated as follows:

$$B_{all} = \frac{1}{M \times N} \sum_{(p_1,p_2) \in all} (|d_c(p_1,p_2) - d_t(p_1,p_2)| > \delta_d) \tag{3.25}$$

where $d_c(p_1,p_2)$ is the computed disparity values, $d_t(p_1,p_2)$ is the ground truth disparity values, $M \times N$ is the total number of pixels in the image, and δ_d is the error threshold.

(iii) Discontinuity regions : Regions that are present nearer to disparity discontinuities are termed as discontinuous regions. Discontinuous regions used for evaluation are shown in fourth column of Figure 3.19. In this figure, white colour indicates the discontinuous regions, black colour refers to occluded regions, and gray colour is used to show the remaining image regions.

Percentage of bad matching pixels over discontinuity regions ($disc$) is calculated as follows:

$$B_{disc} = \frac{1}{N_{disc}} \sum_{(p_1,p_2) \in disc} (|d_c(p_1,p_2) - d_t(p_1,p_2)| > \delta_d) \tag{3.26}$$

where $d_c(p_1,p_2)$ and $d_t(p_1,p_2)$ are the computed and ground truth disparity values respectively, N_{disc} represents the number of pixels in discontinuous regions, and δ_d is the error threshold.

Usually the error threshold is fixed as one for evaluation. The evaluation methodology followed in this thesis is same as that adopted in Middlebury stereo evaluation [117]. Percentage of bad pixels are calculated at all the above three critical image regions for four standard Middlebury stereo images. Also, an average of all these values are considered for evaluation.

3.5 Experimental Results

The proposed method is evaluated on Middlebury stereo datasets (Tsukuba, Venus, Teddy, and Cones) [7, 58, 117], and the results are compared with some of the existing stereo matching methods. Figure 3.20 shows the qualitative comparison of the proposed method with the method proposed in [9]. The method proposed in [9] uses Gabor phase for disparity map computation. It shows that the proposed method produces distinctively superior results as compared to Gabor phase-based stereo matching method. This is due to the fact that Gabor phase-based method uses scalogram to find the disparity map and scalogram has poor time-frequency resolution. In addition to this, phase-based

methods suffer from phase non-linearities. This is due to the existence of singularities in phase signals. Additionally, we have shown the disparity map generated by [10] and [11]. The proposed method shows better performance as compared to the methods proposed in [10] and [11]. This is mainly due to the spatial distance used in ASW, which is same for all the pixels irrespective of the location of the pixels in homogeneous, high textured or discontinuous regions. Also, guided filter suffers from halos effect near edges. Table 3.7 gives quantitative comparisons of the proposed method with the method proposed in [9–11] and some other existing stereo matching methods. Additionally, experimental evaluation is done to justify the incorporation of disparity map refinement step in our proposed method.

Table 3.7: Comparison of the proposed method with existing local stereo matching methods (Error threshold = 1)

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
Proposed method	3.36	3.83	11.8	0.43	0.71	5.13	8.59	12.8	21.6	5.05	12	14.7	8.32
RTCensus [61]	5.08	6.25	19.2	1.58	2.42	14.2	7.96	13.8	20.3	4.10	9.54	12.2	9.73
Differential [138]	4.74	6.77	19.4	1.69	2.62	20.4	8.29	10.1	23.3	4.25	10.3	12.2	10.3
GF [11]	2.98	3.43	9.24	1.93	2.36	11.0	11.9	17.1	21.8	11.9	17.4	19.9	10.9
DCBGrid [87]	5.90	7.26	21.0	1.35	1.91	11.2	10.5	17.2	22.2	5.34	11.9	14.9	10.9
RINCensus [63]	4.78	6.00	14.4	1.11	1.76	7.91	9.76	17.3	26.1	8.09	16.2	17.6	10.9
BioPsyASW [139]	3.62	5.52	14.6	3.15	4.20	20.4	11.5	18.2	23.2	4.93	13.0	11.7	11.2
Without refinement	5.84	6.73	14.1	2.81	3.93	18.5	12.2	16.3	26.6	6.88	14.2	18.8	12.2
SAD+IGMCT [62]	5.81	7.14	22.6	2.61	3.33	25.3	9.79	15.5	25.7	5.08	11.5	15.0	12.5
PhaseBased [9]	4.26	6.53	15.4	6.71	8.16	26.4	14.5	23.1	25.5	10.8	20.5	21.2	15.3
SSD+MF [58]	5.23	7.07	24.1	3.74	5.16	11.9	16.5	24.8	32.9	10.9	19.8	26.3	15.7
PhaseDiff [140]	4.89	7.11	16.3	8.34	9.76	26.0	20.0	28.0	29.0	19.8	28.5	27.5	18.8
ASW [10]	6.09	8.00	8.16	9.71	11.2	18.5	21.4	29.5	32.2	23.8	32.4	32.2	19.4
LCDM+AdaptWgt [141]	5.98	7.84	22.2	14.5	15.4	35.9	20.8	27.3	38.3	8.90	17.2	20.0	19.5

Figure 3.21 shows the qualitative results for 2005 Middlebury dataset images. As discussed earlier, these images have large disparity ranges. Also, there are some non-textured image regions and complicated scene geometry in the images. Hence, this new dataset is more challenging as compared to standard stereo benchmark dataset from the point of accurate estimation of disparity. The experimental results clearly show that the proposed algorithm also gives substantially good results for the non-textured image regions and the regions having complex geometry. Also, the proposed method can tackle large disparity ranges.

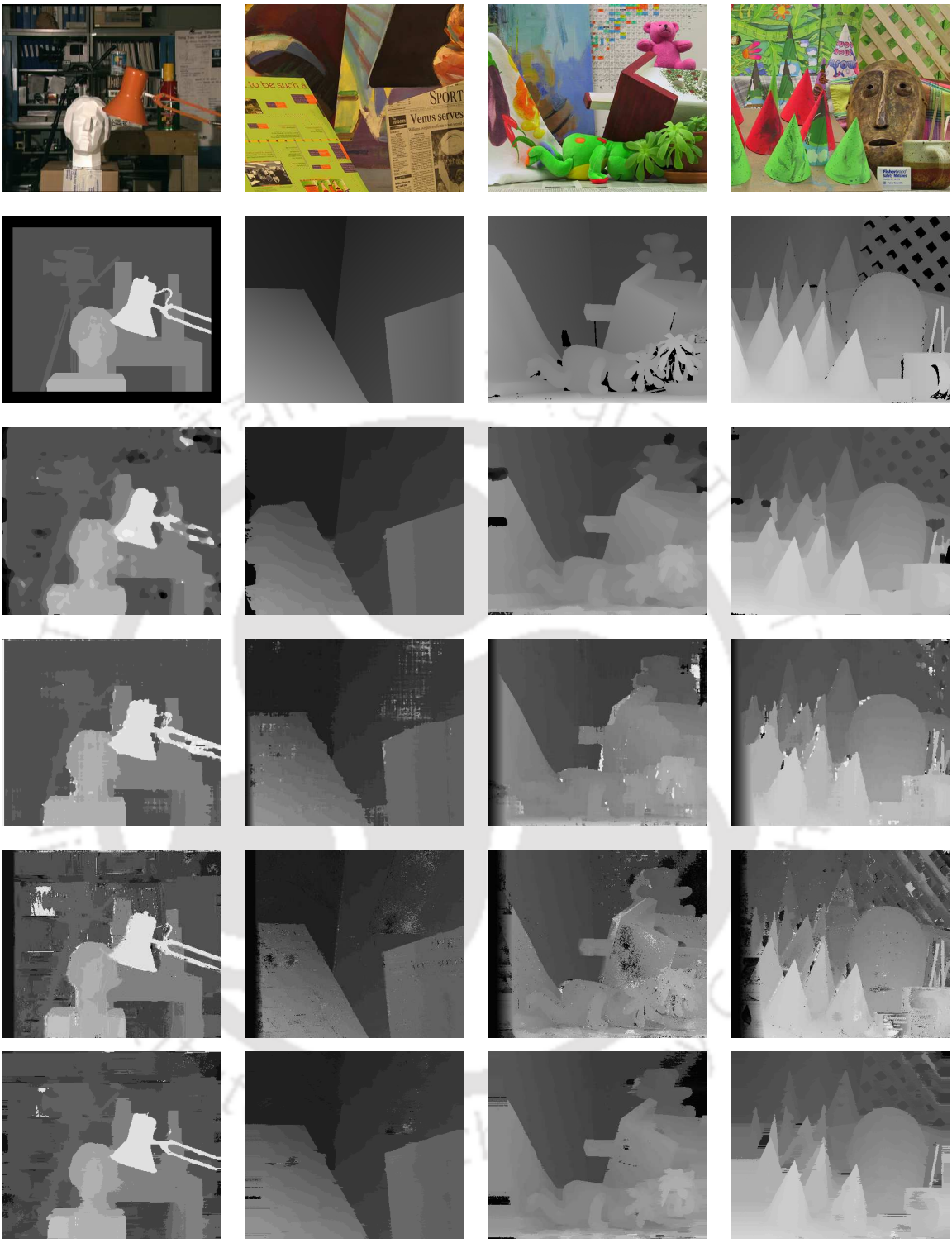


Figure 3.20: Experimental results on Middlebury datasets - Tsukuba, Venus, Teddy, and Cones. First row shows the left input images, and second row shows ground truth disparity maps corresponding to the stereo pair. Third row shows the estimated disparity maps by our proposed method. Forth row shows the disparity maps generated by the method proposed in [9], fifth and sixth rows show the disparity maps generated by the methods proposed in [10] and [11] respectively.

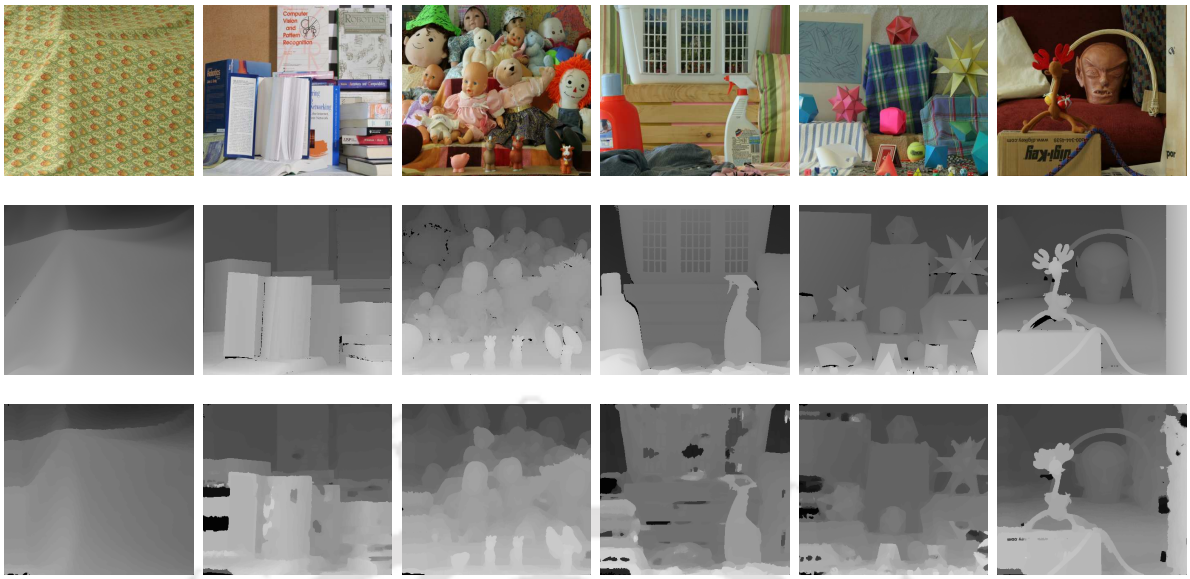


Figure 3.21: Experimental results on 2005 Middlebury datasets - Cloth1, Book, Dolls, Laundary, Moebius, and Reindeer [12,13]. First row shows the left images, second row shows ground truth disparity maps corresponding to the stereo pair, and third row shows the generated disparity maps by our proposed method.

In order to show the effects of various parameters on the accuracy of the generated disparity map, the above experiment is repeated for different values of [4]

- Local stereo window size (SW);
- Kuwahara filter window size (KWS);
- Median filter window size (MWS).

The experiment is also repeated for different numbers of

- Principal components (PC);
- Gabor wavelet filter orientations (N_{theta});
- Gabor wavelet filter scaling (N_{scale})
- **Variation of local stereo window size:** Figure 3.22 shows the percentage of errors (Nocc, all, and disc) for different values of local stereo window for Tsukuba, Venus, Teddy and Cones images. The parameters used are: SW-starting from 3×3 to 15×15, KWS - 9×9, MWS - 3×3 , %PC=20% , $N_{theta} = 2$, and $N_{scale} = 2$. As the window size increases, there are more variations in the percentage of unwanted/bad pixels in the discontinuous regions as compared to

[4]This work has been published in *EECEA 2015* (Refer item [7] in Page 135 for details)

the non-occluded regions and the entire image. Figure 3.28(a) shows the average percentage of the bad pixels. The proposed method produces significantly good results even with the smallest window. This is due to the fact that the correlation of the pixels in the smaller window is more compared to the pixels in the larger window.

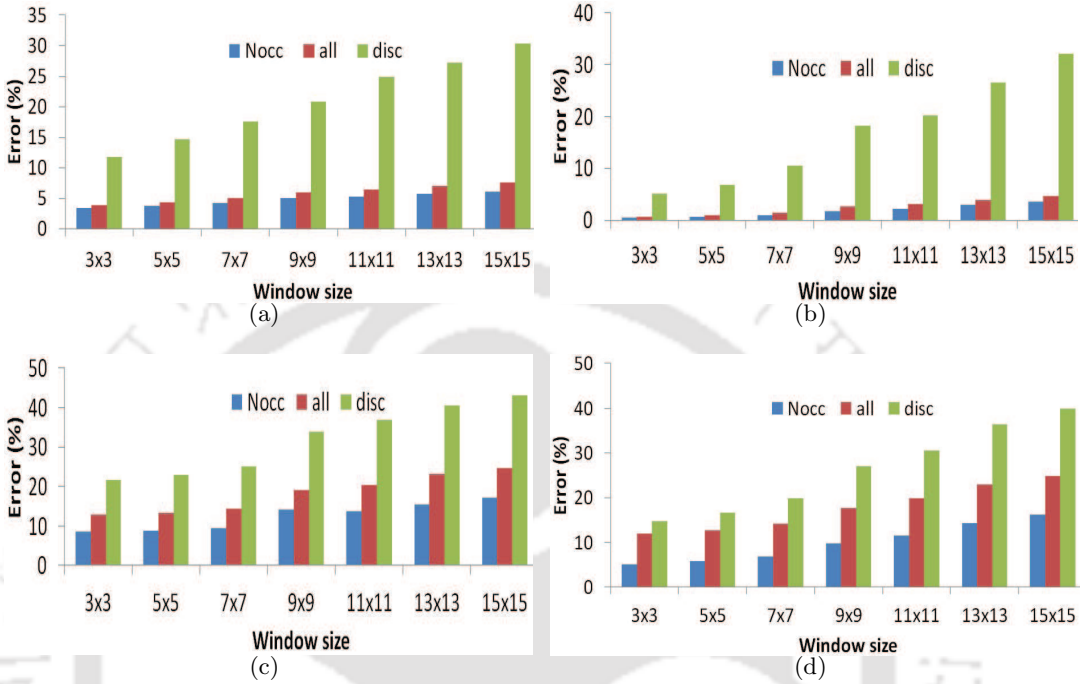


Figure 3.22: Variations of local stereo window size. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

- **Variation of Kuwahara filter window size:** The percentage of errors for Tsukuba, Venus, Teddy, and Cones images are shown in Figure 3.23 for different Kuwahara filter window sizes. The parameters used are: SW - 7×7 , KWS - 5×5,9×9, 13×13, 17×17, 21×21, and 25×25, MWS - 3×3, % PC= 20%, $N_{theta} = 2$ and $N_{scale} = 2$. In this case also, there are more variations in the percentage of unwanted/bad pixels for the bigger windows in the discontinuous regions compared to the non-occluded regions and the entire image. Figure 3.28(b) shows the average percentage of the bad pixels for different Kuwahara filter window sizes. It shows that a small window produces a detailed output image.

- **Variation of median filter window size:** Figure 3.24 shows the percentage of errors for different window sizes of the median filter. The parameters used are: SW - 3×3, KWS - 5×5, MWS - starting from 3×3 to 15×15, % PC=20% , $N_{theta} = 2$ and $N_{scale} = 2$. The average error percentage is shown in Figure 3.28(c). Similar to Kuwahara filter, a bigger median filter window blurs the edges, whereas a small median filter window retains the detailed information.

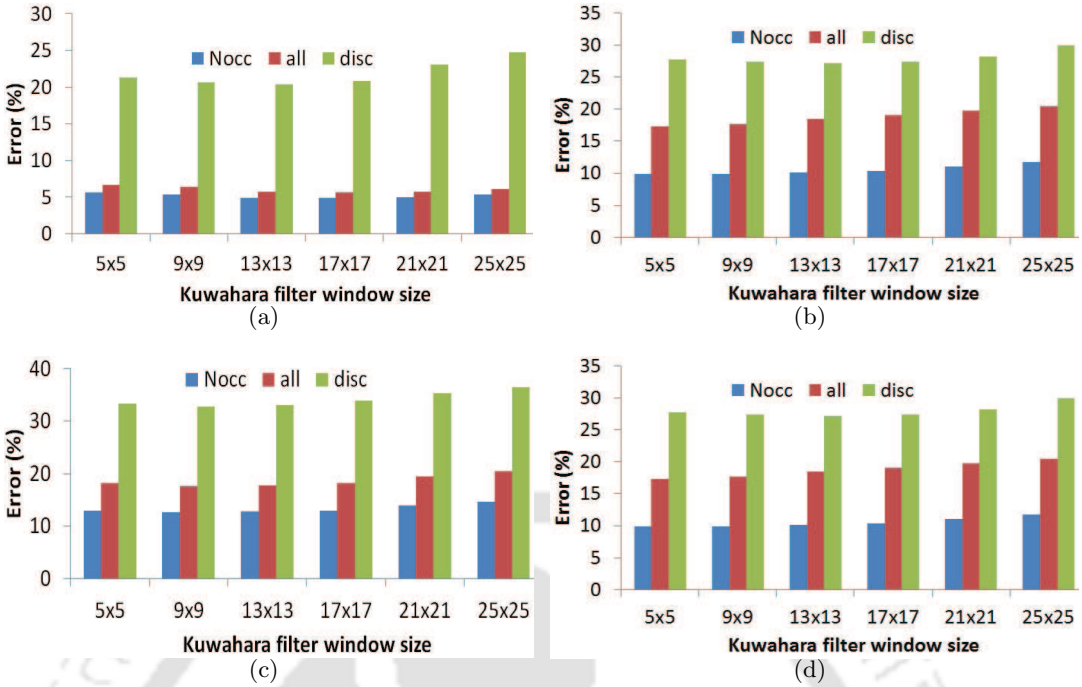


Figure 3.23: Variations of Kuwahara filter window size. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

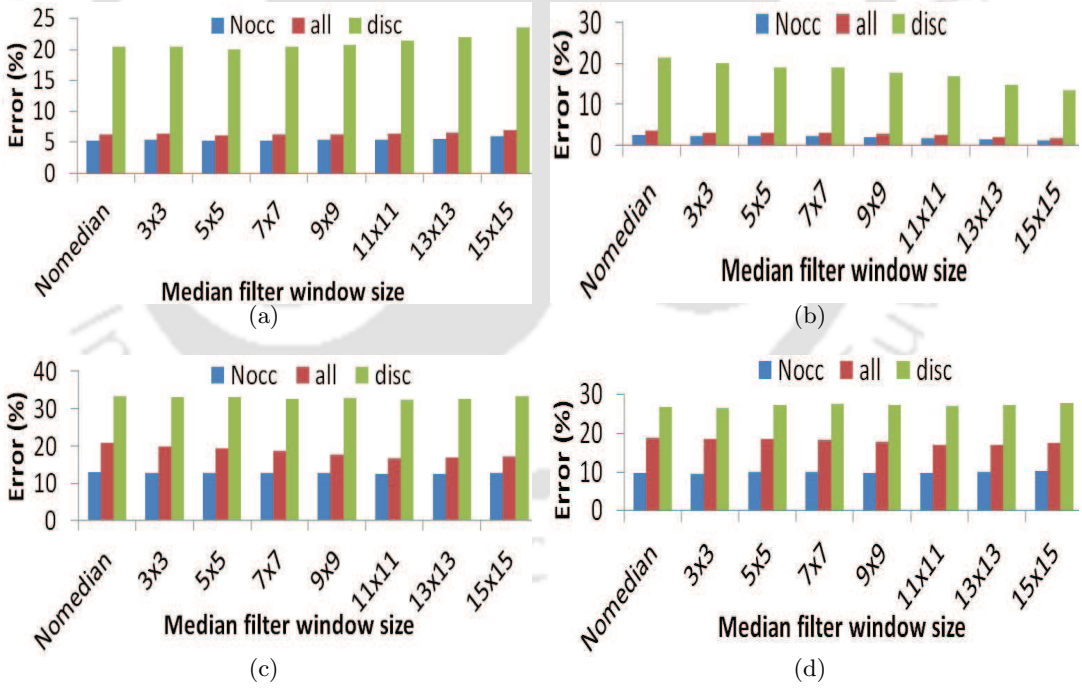


Figure 3.24: Variations of Median filter window size. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

- **Variation of number of principal components:** Figure 3.25 shows the percentage of errors for different numbers of principal components used for local stereo correspondences for all the

four images of Middlebury datasets. The parameters used are: SW - 9×9, KWS - 5×5, MWS - 3×3, no. of PC = 5, 10, 20, 50, 100, 150, 200, and all the coefficients, $N_{theta} = 2$ and $N_{scale} = 2$. Figure 11 shows that error is maximum for PC = 5 and it gradually decreases for PC = 10 and 20, after that the error does not increase significantly. Figure 14(d) shows the average percentage of bad pixels. When the Eigen values are arranged in the decreasing order, the principal components corresponding to the largest Eigen values contain more information. Figure 3.25 and 3.28(d) shows that 20 principal components corresponding to the largest Eigen values contain most important information.

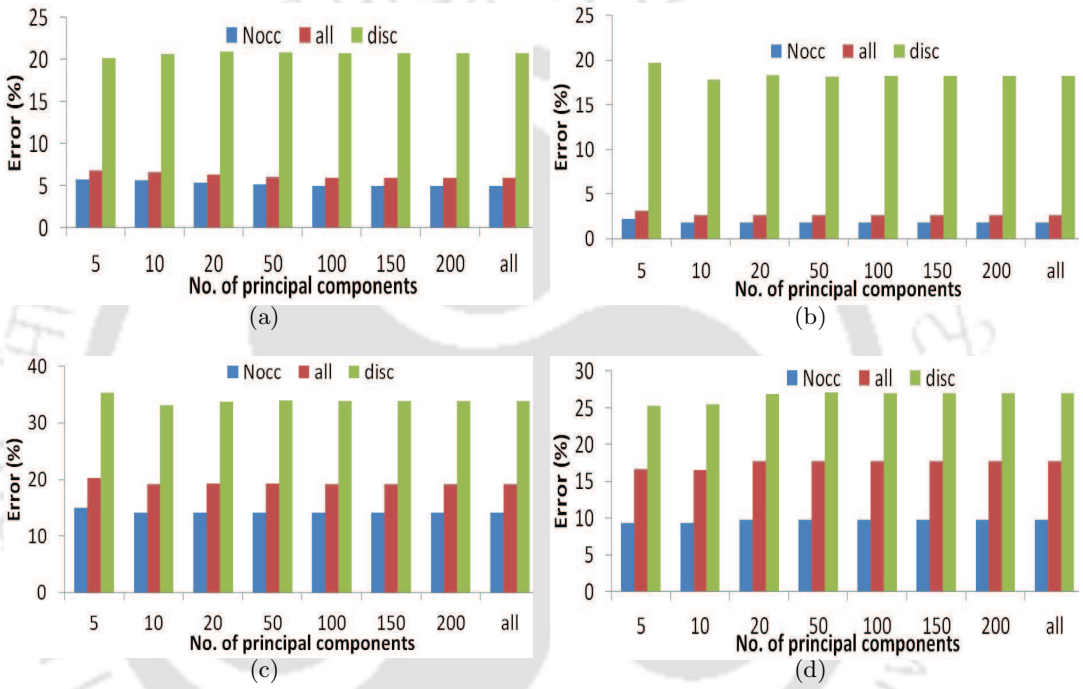


Figure 3.25: Variations of number of principal components. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

- **Variation of number of Gabor wavelet filter orientations:** The percentage of errors for four Middlebury database images for different orientations of Gabor wavelet filter is shown in Figure 3.26. The parameters used are: SW - 5×5, KWS - 5×5, MWS - 3×3, % PC= 20%, N_{theta} = 2,3,4,5, and $N_{scale} = 2$. It shows that the change in error is quite insignificant with respect to the number of orientations. Change in the average error for all the four images shown in Figure 3.28(e), which remains almost constant. This is due to the fact that the percentage of principal components remains same i.e., 20% of the total number of coefficients are used in this experiment.

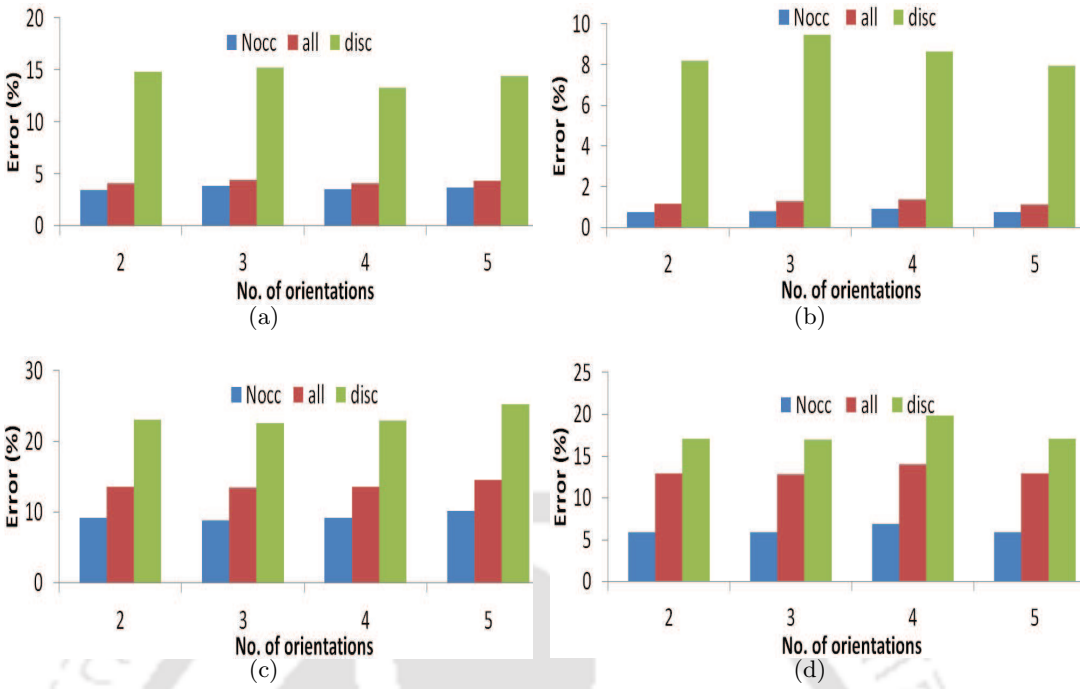


Figure 3.26: Variations of number of Gabor wavelet filter orientations. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

- **Variation of number of Gabor wavelet filter scaling:** Figure 3.27 shows the percentage of error for different numbers of filter scaling for all the four images of Middlebury datasets. The parameters used are: SW - 33×33, KWS - 5×5, MWS - 3×3, % PC= 20%, $N_{theta} = 2$ and $N_{scale} = 2$, 3, 4 and 5. It is seen that error remains almost constant as the number of scaling increases. Apparently, Figure 3.28(f) shows that the average error is more for $N_{scale} = 2$ and then decreases for $N_{scale} = 3$, 4, and 5. In this case also, the change in error is quite insignificant. This is due to the fact that the percentage of principal components remains same i.e., 20% of the total number of coefficients are used in this experiment.

3.6 Summary

In this chapter, a feature-based stereo correspondence algorithm employing a two-pass cost aggregation method is proposed. The proposed stereo correspondence method utilizes local spatial domain Gabor wavelet feature for finding the correspondences between the stereo image pair. Subsequently, PCA is used to reduce the number of local Gabor features. Kuwahara filter is used for cost aggregation as this filter has edge preserving property and run time of $\mathcal{O}(1)$. Subsequently, median filter is used to smooth the aggregated cost. Combined use of Kuwahara filter and median filter significantly improve

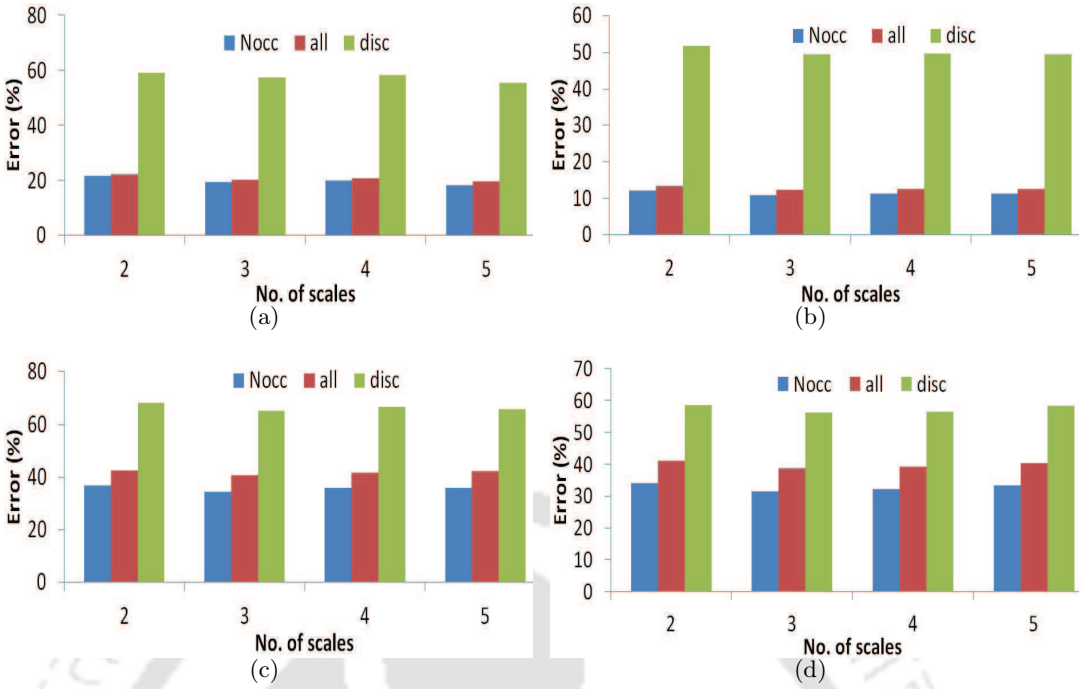


Figure 3.27: Variations of number of Gabor wavelet filter scaling. (a) Tsukuba, (b) Venus, (c) Teddy, and (d) Cones.

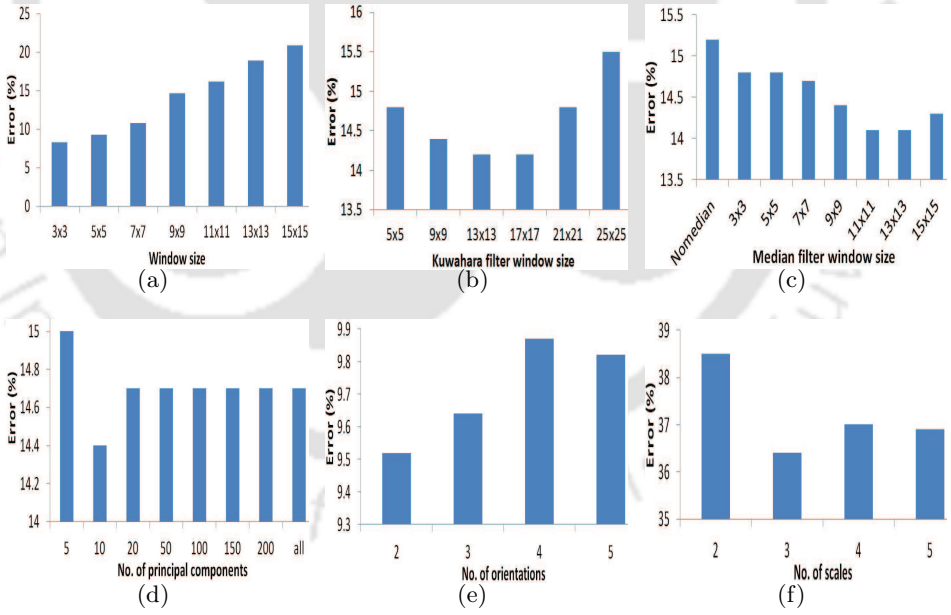


Figure 3.28: Average percentage of bad pixels. (a) Variation of local stereo window size, (b) Variation of Kuwahara filter window size, (c) Variation of Median filter window size, (d) Variation of number of principal components, (e) Number of Gabor wavelet filter orientations and (f) Number of Gabor wavelet filter scaling.

the accuracy of the estimated disparity map. Most of the existing Gabor filter stereo correspondence methods compute phase-based disparity map, whereas the proposed method extracts feature using

Gabor wavelet in the spatial domain. As the phase-based methods suffer from phase non-linearities, the results obtained by the proposed method are better than the phase-based methods. Quantitative and qualitative evaluations show that the proposed method outperforms other stereo matching methods. Our next objective is to estimate an accurate disparity map in presence of occlusion, which will be continued in Chapter 4. For this, occluded pixels are detected from the initial disparity map. The algorithm proposed in this chapter can be implemented for estimating the initial disparity map.

4

Linear Asymmetric and Weight-based Occlusion Detection and Filling

One major problem in obtaining a fine disparity map is scene occlusion. Objects present in the images are occluded on account of different camera viewpoints in a stereo vision setup, and hence it is quite difficult to get a fine disparity map. The methods which use disparity map information of two cameras (symmetric approach) to detect occluded pixels are computationally more complex. Our approach entails to detect the occluded pixels only by using single disparity map information (asymmetric approach). The behaviour of reference and target pixels are analyzed, and it is observed that the target matching pixels almost follow a linear pattern with respect to the reference image pixels. Hence, it is approximated by a linear regression model, and subsequently this model is used to detect occluded pixels in our method. Finally, a fine disparity map is obtained by incorporating a novel occlusion filling method. Experimental results show that the proposed occlusion detection method gives almost similar performance as that of the methods which use two disparity maps for detection. For occlusion filling, we utilize support weights from both the stereo images, and hence our method can give better performance.

4.1 Introduction

Binocular half-occluded pixels are a set of pixels which are visible only in one of the two stereo images (pixels visible in one image, but not in the other image). Occlusion occurs due to the placement of the cameras with respect to the scene to be captured. Occlusion usually occurs around the object boundary and scene discontinuities. In most of the methods, occluded pixels are detected only after the estimation of initial disparity map. Subsequently, occlusion filling is performed by assigning the estimated disparity value to an occluded pixel. In many existing algorithms, occluded regions are considered as noise, and hence these regions are avoided during matching. Occlusion can be detected either implicitly or explicitly. In implicit case, occlusion detection is integrated with the matching process, whereas occluded pixels are detected after performing matching (initial disparity map estimation) in case of explicit detection.

Occlusion detection methods can be broadly classified into five categories: Bimodality (BMD), Match Goodness Jumps (MGJ), Left-Right Checking (LRC), Ordering (ORD) and Occlusion constraints (OCC) [110]. In bimodality, the neighbourhood of the occluded pixels have disparity values of both non-occluded and occluded regions. Hence, the histogram of the neighbourhood disparity values would be bimodal. The ratio of the second highest peak to the highest peak (*i.e.,* peak ratio) of the histogram decides whether a pixel is occluded or not. If the peak ratio is close to one, then there are two similar peaks. This indicates the presence of bimodality in the histogram, and hence the pixel is classified as an occluded pixel on this basis. In MGJ, goodness-of-match would be more for the regions which are present in both the images. On the other hand, goodness-of-match for the occluded regions would be less. This is due to the fact that occluded regions are visible only in one image, and hence it is difficult to find the correct corresponding matching pixels. MGJ used the detected adjacent regions having low/high goodness-of-match score for occlusion detection. The general assumption for stereo vision is that both the left and right cameras approximately covers the same scene, and hence the disparity maps obtained from left image-to-right image and right image-to-left image are negatives of each other. The pixels which violate this assumption are considered as occluded pixels in LRC method. This method produces more error in noisy and textureless regions. Ordering constraint states that if a pixel A is located left to a pixel B in the left image, then the corresponding matching pixel A' is also located left to pixel B'. A pixel which does not satisfy this condition is considered as occluded. Another important consideration for occlusion detection is occlusion constraint. This constraint considers the case of transition of disparity values from one surface to another. When there is a transition from one surface to another, there is an abrupt change in the disparity values from

the occluding surface to the background. This change in disparity values corresponds to the occluded pixels in the other image and vice-versa. All the methods explained earlier give more false positives during occlusion detection. In co-operative algorithm, uniqueness constraint is introduced within the matching cost volume [142]. A pixel is classified as an occluded pixel if the maximum of the matching cost (similarity measure) for the allowable disparity range is below a pre-defined threshold. This method fails to detect the occluded pixels when the false matches of the occluded pixels give high similarity score.

In global methods, occlusion detection and filling are carried out jointly during matching. One such framework is graph cut algorithm. Based on the stereo correspondence problem to be solved, an energy function is formed which includes data, smoothness, and occlusion terms. Different disparity values are assigned to the occluded pixels, and subsequently the energy is computed. Disparity values corresponding to the minimum energy are assigned to the occluded pixels. This process is repeated until this energy function converges to a minimum value. Disparity map generated by graph cuts suffers from stair-like effects. This effect may enhance after occlusion filling. This is due to the fact that the occluded regions are not efficiently represented by the formulated energy function. In this direction, Kolmogorov and Zabih embedded uniqueness constraint in graph cut algorithm to detect occluded pixels [35]. This method outperforms many optimization methods, but the computational complexity of this method is very high. In [44], both the stereo images are segmented and the segmentation of one image is used to find the occluded region in the other image. Mean-shift segmentation used in this method adds computational overhead. Sun *et al.* proposed a belief propagation-based iterative energy minimization framework which uses visibility constraint for occlusion detection [143]. Jang and Ho used warping, cross check and luminance difference constraints for occlusion detection [109]. The computational complexity of these global algorithms are usually high.

As discussed earlier, next step of disparity map estimation after occlusion detection is occlusion filling. Occlusion filling is performed by assigning a disparity value of the nearest left non-occluded pixel to an occluded pixel [111]. In [112], occluded pixel is filled by assigning minimum of the nearest left and right disparity values of non-occluded pixels. These methods introduce horizontal streaks in the disparity map, which is polished out by weighted median filter. But, designing an appropriate filter for this purpose is quite difficult. Huq *et al.* proposed occlusion filling method based on segmentation of the reference image [14]. For an occluded pixel to be filled-up, a set of control points are selected from the neighbouring non-occluded pixels. These control points are selected on the basis of visibility, disparity gradient, and equality constraints. When the neighbourhood pixels belong to more than one background, then the control points are chosen by applying colour-based probabilistic

model. Subsequently, occluded pixels are filled using the segmented control points. This algorithm is computationally more complex, and also suffers from segmentation error.

To overcome some of the problems encountered in existing occluded pixels detection methods discussed in the literature, a novel occlusion detection algorithm is proposed in this chapter. Our proposed method uses the information of only one disparity map for occlusion detection. Finally, occluded pixel is filled by assigning the disparity value of the closest neighbouring non-occluded pixel. Our proposed occlusion filling method differs from the existing methods in the sense that a closest non-occluded pixel is selected on the basis of support weights of both the images. This is done to obtain a fine disparity map. The main contributions of this chapter can be summarized as follows:

- The characteristics of the corresponding matching pixels are analyzed, and it is found that the position of the matching pixel almost follow a linear pattern. Hence, we proposed to model this characteristics by a linear regression model. This model is subsequently employed to detect the occluded pixels.

- Occlusion detection is performed asymmetrically by using only one disparity map instead of two disparity maps as used in the symmetric methods. Our proposed asymmetric method can give almost equivalent performance as that of the symmetric methods by using a significantly simplified detection process. The proposed occlusion detection method is named as linear regression-based asymmetric occlusion detection (LAOD) method.

- Occlusion filling is done by assigning the disparity value of neighbouring non-occluded pixel to the occluded pixel. This non-occluded pixel is selected on the basis of colour similarity of the occluded pixels with the neighbouring non-occluded pixels. This similarity score is calculated by using the support weights of both left and right images. The proposed occlusion filling method is named as support weight-based occlusion filling (SWOF) method.

The proposed LAOD and SWOF methods are together denoted as linear asymmetric and support weight-based (LASW) occlusion detection and filling method. The proposed method is elaborately explained in the sections to follow.

4.2 Background

Stereo images are captured by observing three-dimensional objects by two cameras at different viewpoints. So, some of the details of the object are visible in both images, whereas some portions are totally invisible in both images. But, some portions of the object are visible only in one image in

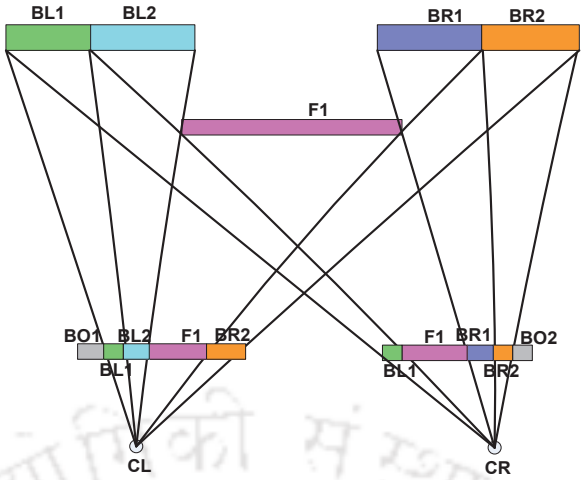


Figure 4.1: General stereo vision set-up [14].

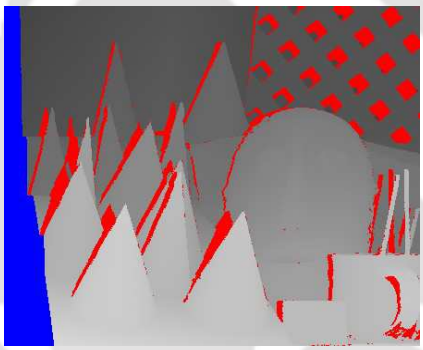


Figure 4.2: Left disparity map showing the ground truth occluded pixels. Border occlusion (blue colour), and non-border occlusion (red colour) in ground truth disparity map.

the stereo vision set-up, which is termed as occlusion. Occluded regions are visible only in one image of the stereo pair, and invisible in the other image. Occlusion occurs in both the ways *i.e.,* regions visible in the left image, but invisible in the right image and vice-versa. This is a case of half occlusion. Invisibility arises because of the scene geometry and the self and/or mutual occlusion of the objects in the scene. Thus, occlusion is one of the problems in stereo matching and accurate disparity map estimation.

Figure 4.1 shows a possible occlusion occurrence in a stereo setup. In this setup, CL and CR are the camera centers, $BL1$, $BL2$, $BR1$, and $BR2$ are the background objects present in the scene. Also, $F1$ is the foreground object, $BO1$ and $BO2$ are occlusions formed in the borders of left and right images respectively. Here, $BL1$ and $BR2$ are present in both the stereo pairs, whereas $BL2$ and $BR1$ are present only in any one of the images. In general, occlusion may be of two types, namely border and non-border occlusions. Border occlusion occurs at the border of an image, and non-border

occlusion appears anywhere inside the image. In Figure 4.1, $BL2$ and $BR1$ are the examples of non-border occlusion, while $BO1$ and $BO2$ are the examples of border occlusions. Figure 4.2 shows the border (blue colour) and non-border (red colour) occlusions for Cones image of Middlebury stereo dataset. So, non-border occlusion generally occurs in the boundary of an object when that object is located within the common field of view of two cameras. On the other hand, border occlusion in the left image occurs due to the absence of some left portion of the field of view of left camera in right image and vice-versa. This fundamental occlusion characteristics is subsequently utilized in our proposed method. Non-border occlusion can be classified as partial, self, and total occlusions, which is shown in Figure 4.3. In partial occlusion, a part of the background object is invisible in any one of the stereo images. Figure 4.4(a) shows the stereo vision setup for the case of partial occlusion. In self occlusion, a portion of an object itself occludes the object, which is shown in Figure 4.4(b). Figure 4.4(c) shows the stereo setup for total occlusion. In this case, an entire object which is visible in one image is absent in another image.

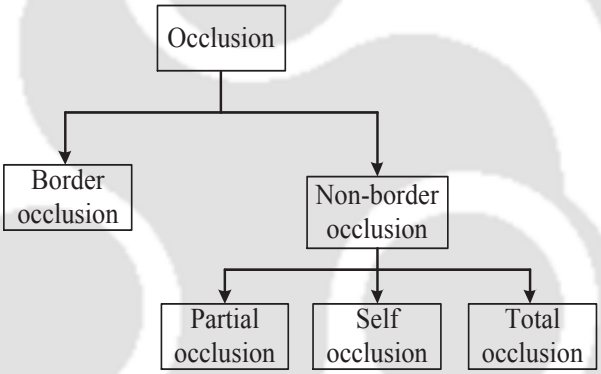
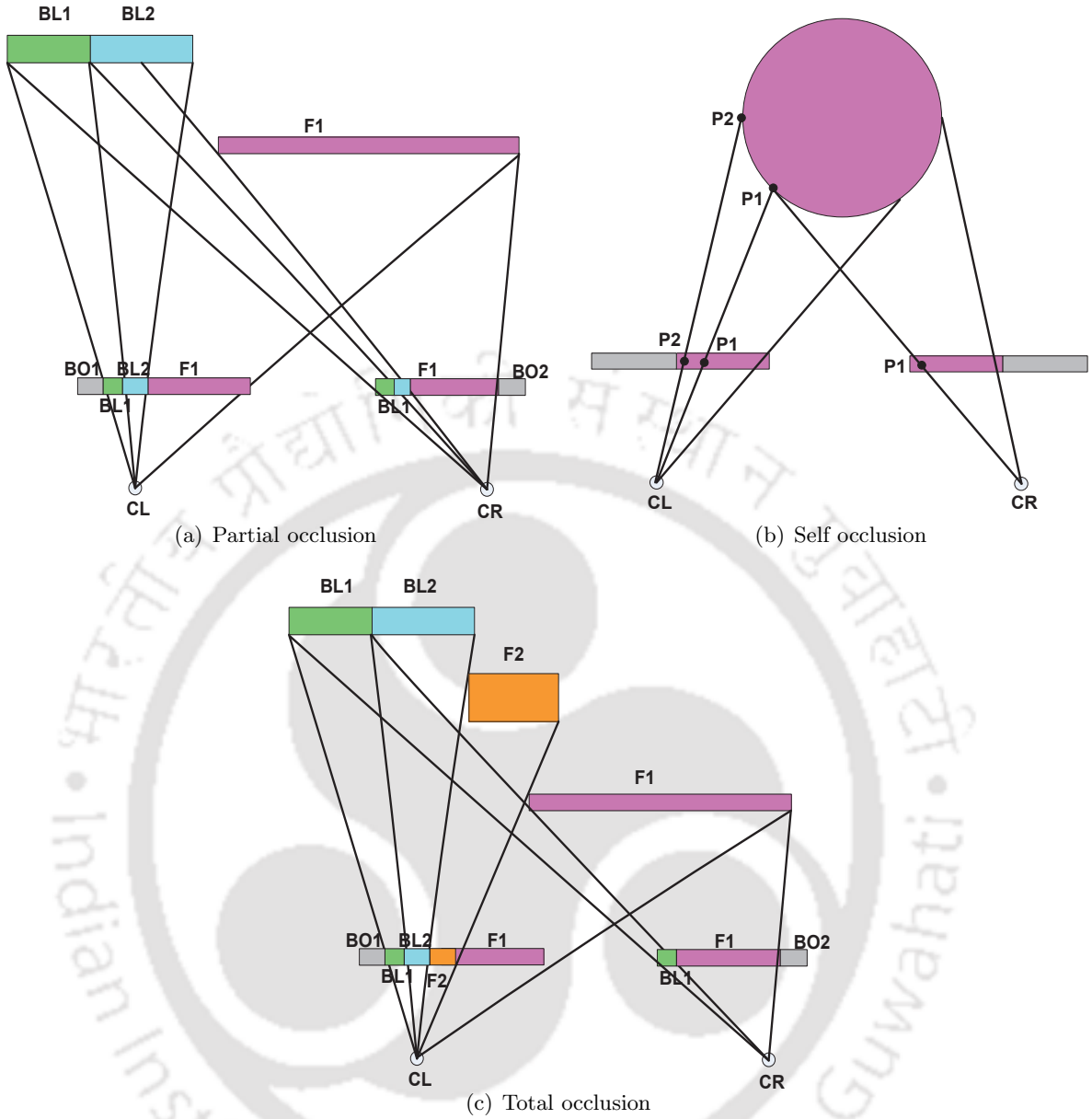


Figure 4.3: Different types of occlusion [14].

4.3 Proposed Method for Occlusion Detection and Filling

For disparity map estimation, we used four main steps in our proposed method. These steps are matching cost computation, cost aggregation, disparity map computation and disparity map refinement. In this chapter, we propose a novel scheme for disparity map estimation under occlusion, and more emphasis is given on occlusion detection and filling of the occluded pixels in the estimated disparity map [1]. All the essential steps of the proposed scheme are listed as follows.

[1]This work has been accepted for publication in *IET Computer Vision* (Refer item [2] in Page 135 for details)



(a) Partial occlusion

(b) Self occlusion

(c) Total occlusion

Figure 4.4: Stereo vision setup for different types of occlusions [14].

4.3.1 Matching cost computation

To compute matching cost, we perform pixel-wise matching. Gabor wavelet-based feature is used to find the corresponding matching pixel in the target image. A two-dimensional Gabor function can be viewed as a sinusoidal plane wave modulated by a Gaussian function, as shown below:

$$g(x,y) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left[e^{i\kappa x} - e^{-\frac{\kappa^2}{2}} \right] \tag{4.1}$$

Gabor wavelet is referred as a class of self-similar functions generated by the process of orientation and scaling of the two-dimensional Gabor function,which is given by:

$$\begin{aligned} g_{mn}(x,y) &= a^{-m} g(x_a, y_a),\ a > 1, \\ x_a &= a^{-m}(x\cos\theta + y\sin\theta) \\ y_a &= a^{-m}(-x\sin\theta + y\cos\theta) \end{aligned} \tag{4.2}$$

where $\theta = \frac{n\pi}{K}$, m and n are two integers, and K is the total number of orientations. To extract the feature for a pixel, a small neighbouring region $\mathbf{S}_p$ around the pixel is convolved with Gabor wavelet. Mathematically, this can be written as follows:

$$\begin{aligned} \mathbf{G}^e_I(p_1,p_2) &= \left[\chi_{0s} \otimes \chi_{1s} \otimes \cdots \otimes \chi_{(m-1)s}\right], \forall \mathbf{p} = (p_1,p_2) \\ \chi_{rs} &= \left[vec\left(\mathbf{G}^e_{r1}\right) \otimes vec\left(\mathbf{G}^e_{r2}\right) \otimes \cdots \otimes vec\left(\mathbf{G}^e_{rn}\right)\right], s = 0,1,\cdots,(n-1) \\ \mathbf{G}^e_{mn} &= \mathbf{S}_p * \mathbf{g}^e_{mn}(x,y) \\ \mathbf{g}^e_{mn}(x,y) &= a^{-m}\mathbf{g}^e(x_a,y_a) \\ \mathbf{g}^e(x,y) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left[\cos\kappa x - e^{-\frac{\kappa^2}{2}}\right] \end{aligned} \tag{4.3}$$

where *vec* represents matrix to vector conversion operation, the symbol $\otimes$ denotes vertical concatenation of vectors, " $*$ " is a convolution operator, and $g^e_{mn}(x,y)$ is the real Gabor wavelet kernel. In Equation 4.3, $\kappa = \sqrt{2\ln 2}\left(\frac{2^\phi+1}{2^\phi-1}\right)$, where ϕ is the bandwidth in octaves.

4.3.2 Cost aggregation

Cost aggregation involves cost volume filtering using Kuwahara filter followed by median filter. For a pixel, a small neighbourhood of size $2a+1$ centered around the pixel is partitioned into four identical subregions Q_1, Q_2, Q_3, and Q_4. Each subregion is given by:

$$Q_i(p_1,p_2) = \begin{cases} [p_1, p_1+a] \times [p_2, p_2+a], \text{ if } i = 1 \\ [p_1-a, p_1] \times [p_2, p_2+a], \text{ if } i = 2 \\ [p_1-a, p_1] \times [p_2-a, p_2], \text{ if } i = 3 \\ [p_1, p_1+a] \times [p_2-a, p_2], \text{ if } i = 4 \end{cases} \tag{4.4}$$

where "$\times$" denotes the cartesian product. Let m_i and σ_i be mean and standard deviation respectively of four subregions $Q_i, i = 1, ..., 4$. The output K_f of Kuwahara filter for a pixel (p_1,p_2) is given by mean corresponding to the subregion having minimum standard deviation. This can be formulated as

follows:

$$K_f(p_1, p_2) = \sum_i m_i(p_1, p_2) f_i(p_1, p_2) \tag{4.5}$$

where

$$f_i(p_1, p_2) = \begin{cases} 1, \ \sigma_i(p_1, p_2) = \min\limits_k \sigma_k(p_1, p_2) \\ 0, \ \text{otherwise} \end{cases} \tag{4.6}$$

Details about cost computation and aggregation can be found in [15].

4.3.3 Disparity map computation

The initial disparity map is obtained by determining the disparity value d_p of all the pixels $\mathbf{p}$ in the reference image. This is accomplished by taking the index of the minimum value in the aggregated cost of a particular pixel. Mathematically, the disparity value d_p of a pixel $\mathbf{p}$ is given by:

$$d_p = \arg\min_d C_{agg}(\mathbf{p}, d) \tag{4.7}$$

where $C_{agg}(\mathbf{p}, d)$ is the aggregated matching cost of a pixel $\mathbf{p}$ at disparity d. Disparity map generated from multiple-cameras can be used to accurately detect objects under camouflage conditions [2].

4.3.4 Proposed linear regression-based asymmetric occlusion detection (LAOD) method

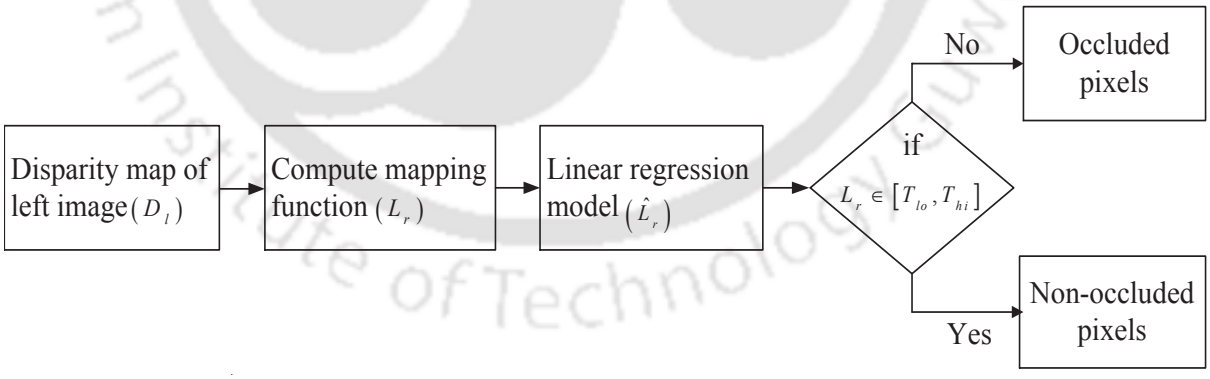


Figure 4.5: Block diagram of the proposed occlusion detection method.

The proposed occlusion detection method simultaneously detects both the occluded pixels and the wrongly estimated disparity values. Subsequently, occlusion filling is done by replacing these values with appropriate disparity values. The main aspect of our proposed method is that occluded pixels

[2]This work has been published in *INDICON 2013* (Refer item [8] in Page 136 for details)

are detected asymmetrically. Our approach only requires the disparity map of any one of the stereo images. The proposed method is mainly based on three important fundamental characteristics of stereo vision, namely continuity, ordering and uniqueness constraints. Figure 4.5 shows the block

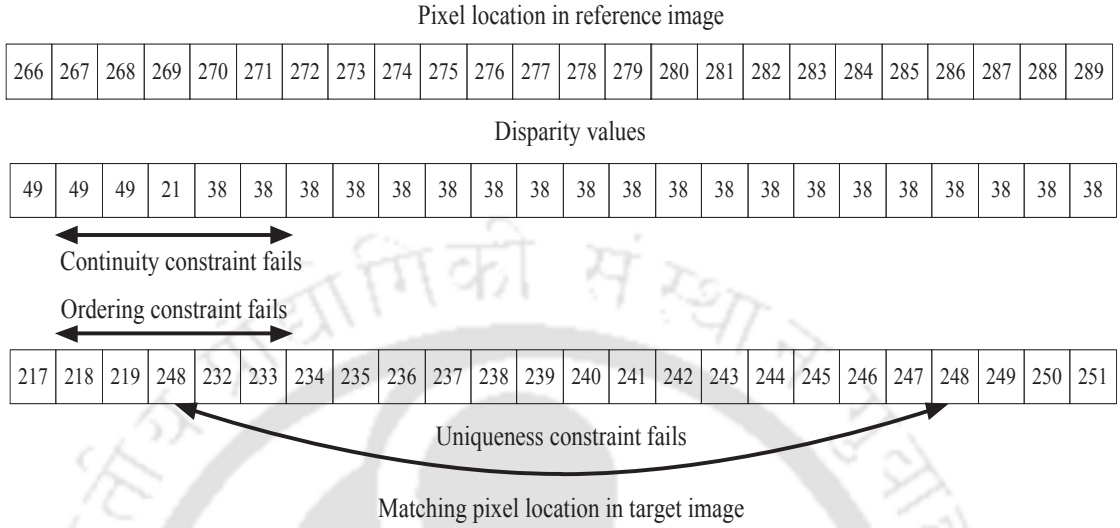


Figure 4.6: Example showing the case when a pixel not satisfies continuity, ordering, and uniqueness constraints.

diagram of proposed occlusion detection method. If I_l and I_r are the left and right stereo image pairs, then D_l and D_r are the corresponding disparity maps respectively. So, by combining the location of the pixel in the horizontal direction with its disparity value, an approximate linearly increasing function is obtained. This function is denoted as $\mathbf{L}_r$, which indicates the corresponding matching pixel in the other image. Mathematically, this function for the left reference image is given by:

$$L_r(p_1, p_2) = p_1 - D_l(p_1, p_2) \tag{4.8}$$

where p_1 is the horizontal image index. It is to be mentioned that the pixels are scanned from left to right, and hence as per Equation 4.8, the corresponding mapping function $\mathbf{L}_r$ approximately follows a linearly increasing pattern. So, $\mathbf{L}_r$ is approximated by fitting a linear function. Apparently, this linearity condition is fulfilled only if this function satisfies the three essential conditions (uniqueness, ordering, and continuity constraints). As explained earlier, pixels are occluded in the boundary of an object. So, our proposed linearity condition will not be satisfied for the occluded regions. This is the main trick of our proposed method for finding an occluded region in an image. Also, the linearity condition will be satisfied only if all these three constraints are simultaneously satisfied. This characteristics is shown by giving a numerical example as depicted in Figure 4.6. First row of this

figure shows the pixel location in the horizontal direction for the case of left to right scanning, second and third rows show the disparity values (before occlusion detection), and the corresponding linear mapping function respectively. At the occluded pixel *269*, the disparity value is *21*. This disparity value differs from the actual disparity value of *38*, which was obtained from the ground truth information. At this pixel, the corresponding mapping function does not obey uniqueness, ordering, and continuity constraints. The mapping function already has the value of *248* for the pixel location *286 i.e.,* pixel *286* is matched to *248* in right image, whereas the occluded pixel *269* also has the corresponding matching point *248*. This violates the uniqueness constraint. Again, the continuity constraint does not hold for the disparity value of *21*. Finally, the ordering constraint is also not fulfilled as the values of the mapping function within the rectangular box do not follow the same order as that of the pixel positions in the left image. So, it is clear from all these observations that our proposed mapping function does not hold all these three constraints for the occluded and/or wrongly estimated disparity value. This unique characteristics is effectively used in our method for the detection of occluded regions in an image. In our method, the mapping function $\mathbf{L}_r$ is modelled by linear regression as follows:

$$\hat{\mathbf{L}}_r = \mathbf{M}\mathbf{L}_r + \boldsymbol{\varepsilon} \tag{4.9}$$

where m_{ij} is the $(i,j)^{th}$ element of the regression matrix $\mathbf{M}$, and ε_i is the i^{th} element of the error term. If the deviation of the mapping function from the best fit function is within a threshold, then the pixel is considered as non-occluded, else occluded. Analytically, this is given by:

$$\mathbf{L}_r \in [\mathbf{T}_{lo}, \mathbf{T}_{hi}] \tag{4.10}$$

where

$$\begin{aligned} \mathbf{T}_{lo} &= \hat{\mathbf{L}}_r - \mathbf{T}_h \\ \mathbf{T}_{hi} &= \hat{\mathbf{L}}_r + \mathbf{T}_h \\ \mathbf{T}_h &= c \cdot \mathbf{g} \end{aligned} \tag{4.11}$$

Here, $\mathbf{T}_{hi}$ is the upper threshold, $\mathbf{T}_{lo}$ is the lower threshold, c is a constant, and $\mathbf{g}$ is the weight of the gradient magnitude of the disparity map, which is given by:

$$\begin{aligned} \mathbf{x}_g = \frac{\partial D_l}{\partial x},\ \mathbf{y}_g = \frac{\partial D_l}{\partial y} \\ \mathbf{g} = \exp\left(-\left(\mathbf{x}_g^2 + \mathbf{y}_g^2\right)\right) \end{aligned} \tag{4.12}$$

As the gradient operation can give discontinuity information in an image, so it also detects the discon-

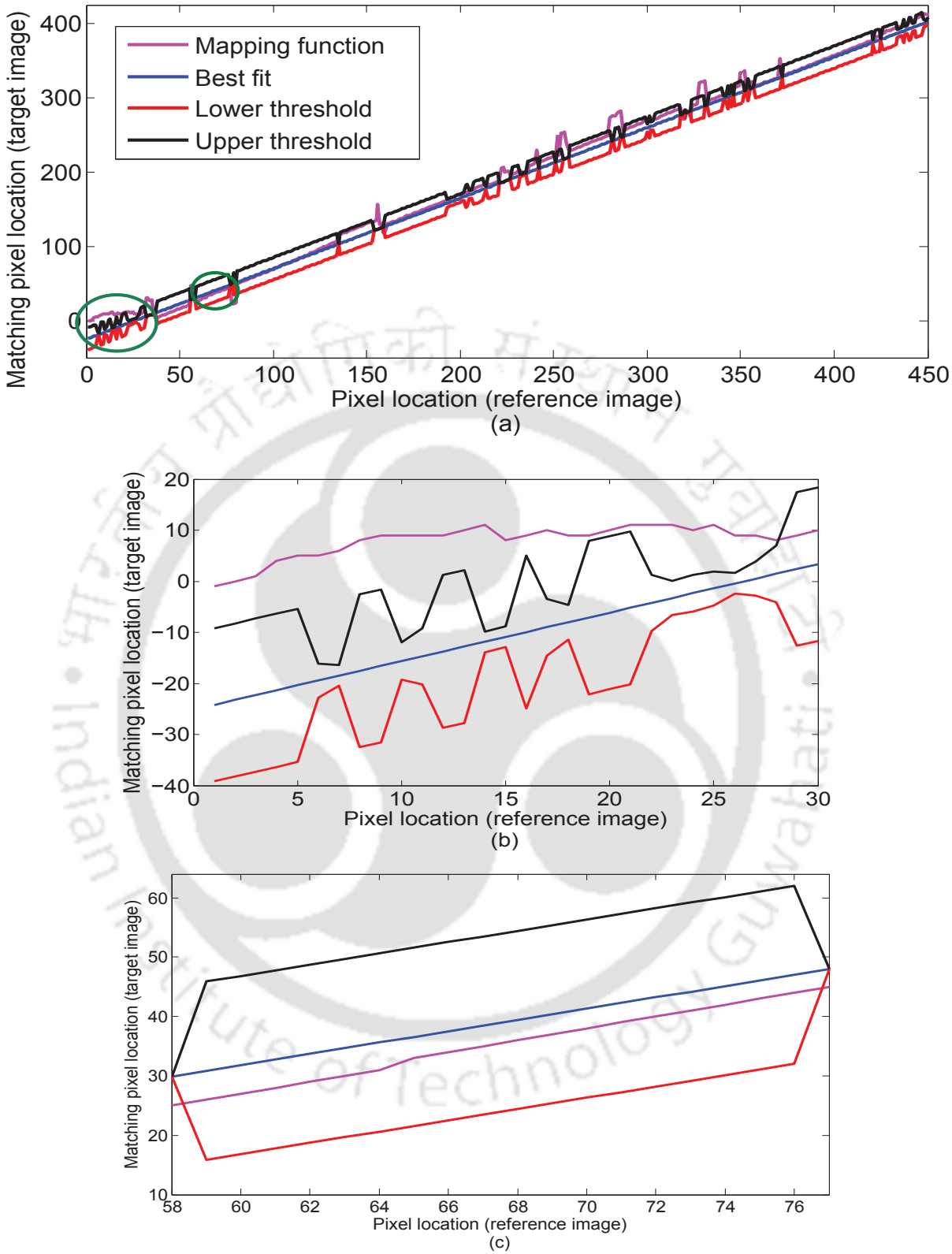


Figure 4.7: (a) Proposed mapping function along with the best fit, upper and lower thresholds for one row of Cones image; (b-c) Detection of occluded and non-occluded pixels (shown by circles in (a)) respectively by our proposed mapping function.

tinuity in the disparity map. Figure 4.7 shows the disparity values in a row of the left image. It also

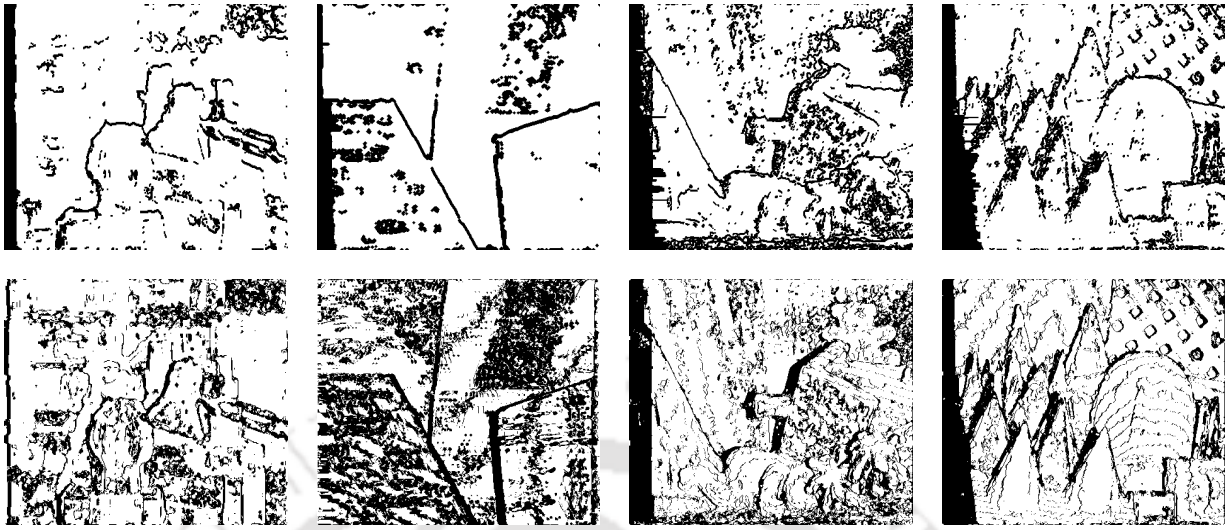


Figure 4.8: Detected occluded pixels (shown by black colour) by proposed LAOD and LRC methods. First row-proposed method, second row-LRC. Left to right-Tsukuba, Venus, Teddy and Cones.

Table 4.1: Comparison of the proposed LAOD method with LRC method

Algorithm	Tsukuba		Venus		Teddy		Cones	
	TP[3]	FP[4]	TP	FP	TP	FP	TP	FP
Proposed method	**77**	**18**	78	**11**	**78**	**14**	84	**11**
LRC	69	20	**80**	27	75	**14**	**85**	12

TP-True positive
FP-False positive

shows the corresponding mapping function $\mathbf{L}_r$, best fit of the mapping function $\hat{\mathbf{L}}_r$, upper $\mathbf{T}_{hi}$ and lower $\mathbf{T}_{lo}$ thresholds. These values are computed for occluded and non-occluded pixels. The proposed method is compared with LRC which is a popular occlusion detection algorithm. Figure 4.8 shows the performance of our proposed method and the traditional (LRC) method in terms of detected occluded pixels. Table 4.1 shows a comparison of our proposed LAOD method with the traditional occlusion detection method in terms of two measures, namely true positive and false positive. As explained earlier, symmetric methods employ disparity maps of both the stereo images for occlusion detection. The pixels having erroneous disparity values in both the estimated disparity maps influence the detection process of the occluded pixels. That is why, symmetric methods give more false positives as compared to the asymmetric methods.

4.3.5 Proposed support weight-based occlusion filling (SWOF) method

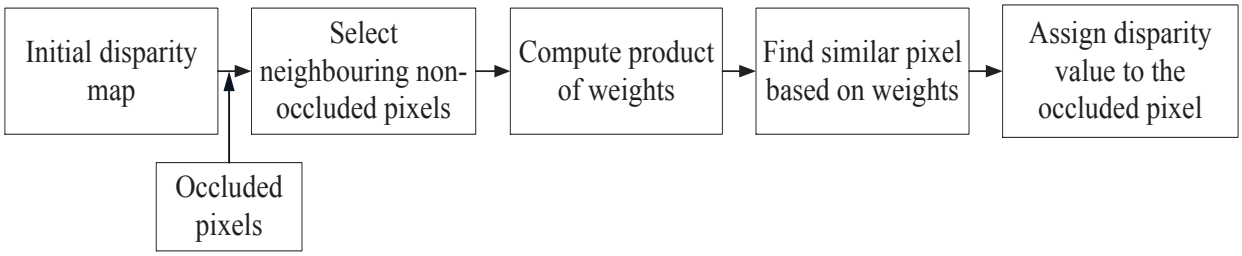


Figure 4.9: Block diagram of the proposed occlusion filling method.

The block diagram of the proposed occlusion filling method is shown in Figure 4.9. In our method, following three assumptions are considered.

- Neighbouring pixels having almost similar colour or intensity values have the same disparity value.
- Colour or intensity changes in the edges of an image create a discontinuity in the disparity map.
- Neighbouring non-occluded pixels and their corresponding matching pixels have almost similar chrominance characteristics.

In our method, occlusion filling is done by assigning a disparity value of the neighbouring non-occluded pixel which is very much similar in terms of colour to the occluded pixel. In this process, any pixel in the occluded region which is wrongly detected as non-occluded pixel, may give a high colour similarity score with the occluded pixel under consideration. To avoid this situation, non-occluded pixels can be identified from the combined weights of both the images of the stereo image pair. So, colour similarity estimation is done only with the non-occluded pixels. For a pixel in the left image, initial disparity map is used to find the corresponding matching pixel in the right image. After determining the matching pixels, support weights are calculated separately for both images. Then, product of these support weights are used to get the required information. This combined weight is subsequently used to find a neighbouring non-occluded pixel which is similar to the pixel to be filled. Finally, disparity value of the selected pixel is assigned to the pixel which is targeted for filling. Our main contribution in this step is that weights are calculated on the basis of information obtained from two stereo images. If any neighbouring occluded pixel produces colour similarity with the pixel to be filled, then that occluded pixel is given less priority amongst all the neighbouring pixels in the process of occlusion filling. The

support weight of a neighbouring pixel $\mathbf{q}$ in N_p in the left image is given by the following equation:

$$w_l(\mathbf{p},\mathbf{q}) = \exp\left(-\frac{\Delta c_{pq}}{\gamma_c}\right) \tag{4.13}$$

where Δc_{pq} is the colour dissimilarity of the pixel $\mathbf{q}$ from $\mathbf{p}$, $\mathbf{p}$ is the pixel under consideration, $\mathbf{q}$ is the non-occluded pixel in the neighbourhood region N_p, and γ_c is a constant. Similarly, support weight of a neighbouring pixel $\mathbf{q}'$ in the right image with disparity d_q is computed as follows:

$$w_r(\mathbf{p}',\mathbf{q}') = \exp\left(-\frac{\Delta c_{p'q'}}{\gamma_c}\right) \tag{4.14}$$

where $\mathbf{p}'$ and $\mathbf{q}'$ are the pixels corresponding to the pixels $\mathbf{p}$ and $\mathbf{q}$ with disparity values d_p and d_q respectively. To minimize the influence of invalid pixels *i.e.,* pixels which are occluded, the above weights are combined to get the final weight as follows:

$$\begin{aligned} w(\mathbf{p},\mathbf{q}) &= w_l(\mathbf{p},\mathbf{q}) w_r(\mathbf{p}',\mathbf{q}') \\ &= \exp\left(-\frac{\Delta c_{pq}}{\gamma_c}\right) \exp\left(-\frac{\Delta c_{p'q'}}{\gamma_c}\right) \end{aligned} \tag{4.15}$$

So in our method, the final weight is estimated on the basis of information obtained from two images. This procedure eliminates the chance of assigning a disparity value of neighbouring occluded pixel to the pixel to be filled. Detected non-border occluded pixels are filled from left-to-right, while border pixels are filled in the reverse direction *i.e.,* from right-to-left. Figure 4.10 illustrates our method for final weight computation with the help of two images. Figures 4.10(a) and 4.10(c) show an image patch for reference and target images respectively, and their corresponding weights are shown in Figures 4.10(b) and 4.10(d) respectively. Pixel A in Figure 4.10(a) corresponds to pixel A in Figure 4.10(c) with a disparity value d_a, whereas pixel B corresponds to pixel C for a disparity value d_a. It is observed that pixel A finds a similar corresponding matching pixel in the target image. Since the pixel B is occluded, unique correspondence is not obtained for this pixel. So, the influence of pixel B on the estimation of final weight should be minimized. If only one reference image is considered for weight computation, then weight of pixel A will have erroneous contribution of pixel B. This leads to the assignment of wrong disparity value to an occluded pixel. In contrary to this approach, if both the images are used for weight computation, then the influence of pixel B is minimized. As shown in Figure 4.10(e), the combined weight for the pixel B is less as compared to the weight shown in Figure 4.10(b). In Figures 4.10(f), Figures 4.10(g) and Figures 4.10(h), we used three arrows, where the leftmost arrow shows the non-occluded pixel, middle arrow shows the pixel to be filled, and third

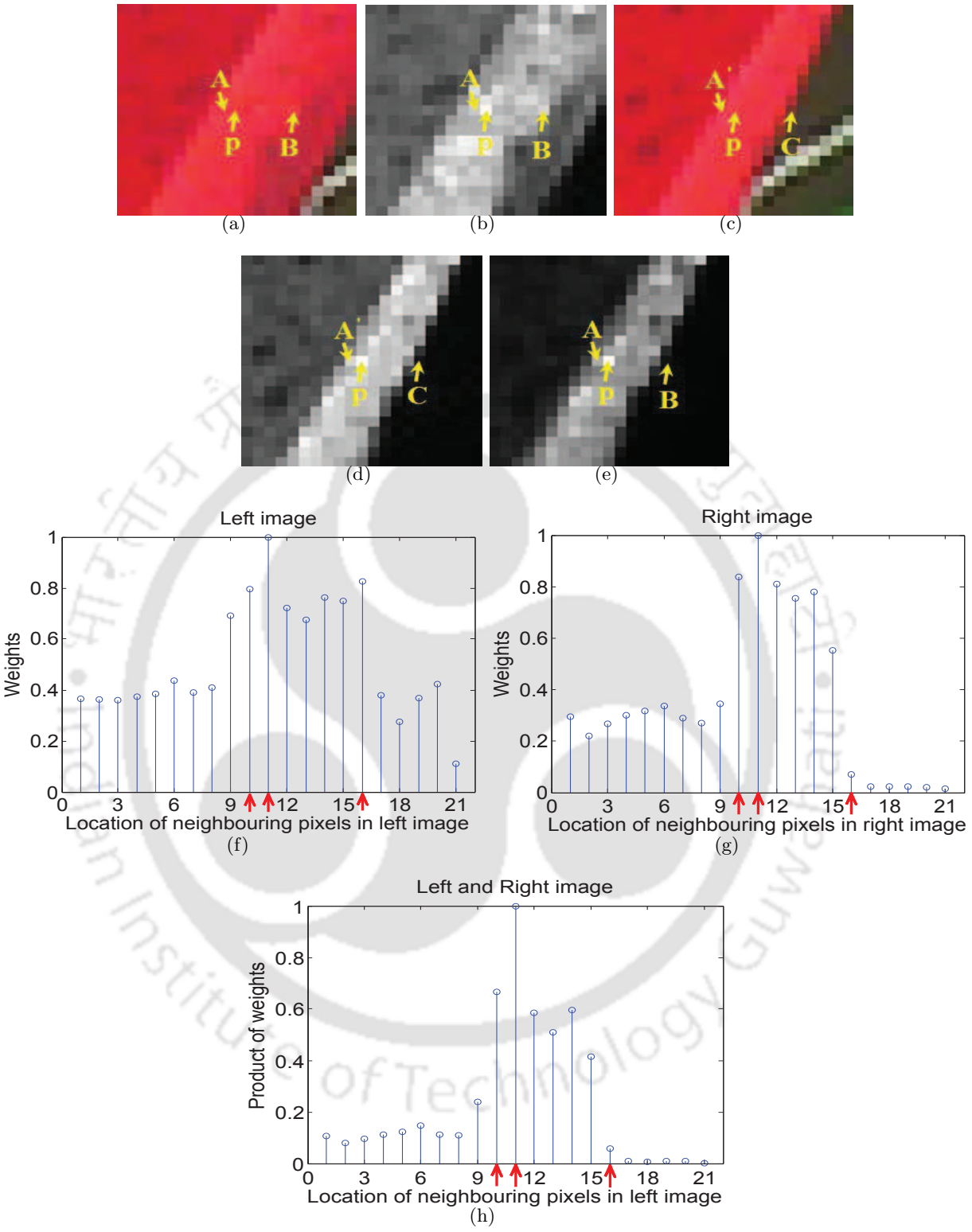


Figure 4.10: Illustration of our proposed scheme for determining combined weights. (a) Reference image; (b) Support weight of (a); (c) Target image; (d) Support weight of (c); (e) Product of weights; (f) Center row of (b); (g) Center row of (d); (h) Center row of (e).

arrow shows the occluded pixel. When only one reference image is used, weight of the occluded pixel as shown by the rightmost arrow is more, but the final weight corresponding to the occluded pixel B is less. This illustration clearly highlights our contribution in weight computation by using both the stereo images for occlusion filling. Table 4.2 shows a quantitative comparison of the proposed

Table 4.2: Comparison of the proposed LASW method with LNDA [11] (Error threshold=1)

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
Proposed method	2.34	2.57	12.6	0.40	0.66	3.99	7.73	14.1	20.3	5.21	10.9	14.3	7.93
LNDA	5.84	6.73	14.1	2.81	3.93	18.5	12.2	16.3	26.6	6.88	14.2	18.8	12.2

LASW method with LNDA method used in [11]. The values *Nocc*, *all* and *disc* shown in the Table 4.2 represent the percentage of bad pixels in the non-occluded region, the percentage of bad pixels in the entire image, and the percentage of bad pixels in the discontinuous regions respectively.

4.3.6 Disparity refinement

Finally, disparity refinement is performed by a constant time weighted median filter [136]. The weights are calculated by a guided filter. The weights $w_{p,q}$ are given by:

$$w(\mathbf{p},\mathbf{q}) = \frac{1}{|\mathcal{N}_p|^2} \sum_{\mathbf{q}\in\mathcal{N}_p} \left[1 + \left(\mathbf{I}_p - \boldsymbol{\mu}_p\right)^T (\boldsymbol{\Sigma}_p + \varepsilon\mathbf{U})^{-1} \left(\mathbf{I}_q - \boldsymbol{\mu}_p\right)\right] \tag{4.16}$$

where $\boldsymbol{\mu}_p$ and $\boldsymbol{\Sigma}_p$ are the mean vector and the covariance matrix of all the pixels in the window $\mathcal{N}_p$, $\mathbf{U}$ is a 3×3 identity matrix. $|\mathcal{N}_p|$ is the number of pixels in the window $\mathcal{N}_p$, and ε is a user-defined smoothness parameter.

4.4 Experimental Results

The performance of the proposed occlusion detection and filling (LASW) method is evaluated using Middlebury datasets (Tsukuba, Venus, Teddy, and Cones) [7,58]. Figure 4.11 shows the disparity maps estimated by our proposed method and the method proposed in [15]. These two methods differ only in the process of occlusion detection and filling, and all other steps of disparity map estimation remain same. First row in this figure shows the input left image, second row shows the ground truth disparity maps, third and fourth rows show the disparity maps generated by our proposed method and its corresponding error image respectively. This error image gives an indication of the bad pixels present

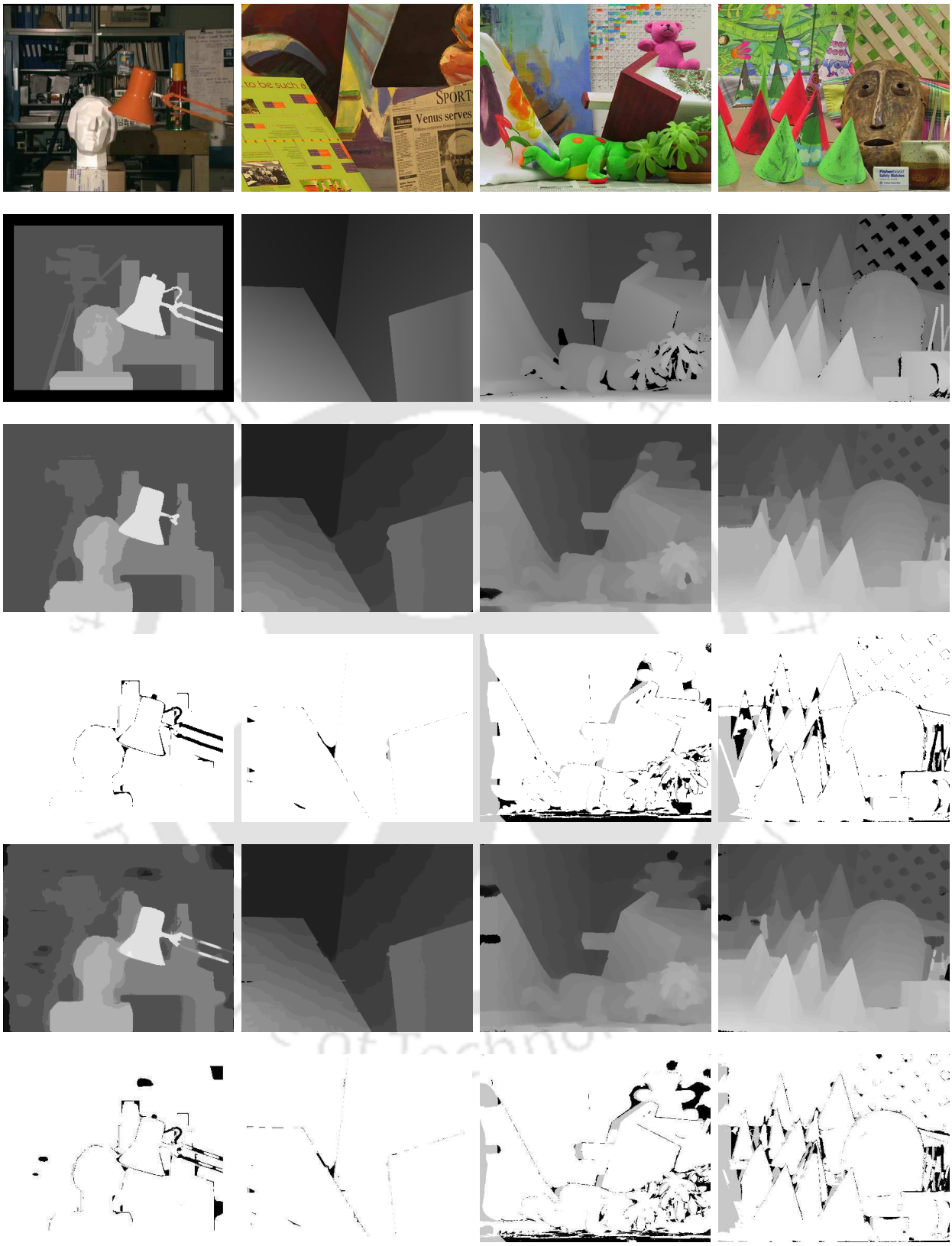


Figure 4.11: Comparison of disparity maps estimated by our proposed LASW method and the method proposed in [15]. Top to bottom - Left image, ground truth disparity maps, disparity maps generated by our method, error image which indicates the bad pixels in the disparity maps, disparity maps generated by the method proposed in [15], and its corresponding bad-pixel image. Left to right - Tsukuba, Venus, Teddy, and Cones images.

in the estimated disparity maps. The disparity maps and its corresponding bad pixel image generated by the method proposed in [15] are shown in fifth and sixth rows of figure respectively. Table 4.3 shows a comparison of the proposed method with some of the existing stereo matching methods in terms of *Nocc*, *all*, and *disc*. It is seen that the proposed method produces significantly better results as compared to the methods listed in Table 4.3. As shown in Table 4.4, the performance of our LASW method is compared with LNDA without using the refinement step. The proposed occlusion detection

Table 4.3: Comparison of the proposed LASW method with the existing stereo matching methods (Error threshold = 1)

Algorithms	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
Proposed method	2.34	2.57	12.6	0.40	0.66	3.99	7.73	14.1	20.3	5.21	10.9	14.3	7.93
GW [15]	3.36	3.83	11.8	0.43	0.71	5.13	8.59	12.8	21.6	5.05	12	14.7	8.32
TensorVoting [144]	3.79	4.79	8.86	1.23	1.88	11.5	9.76	17.0	24.0	4.38	11.4	12.2	9.25
ConvexTV [145]	3.61	5.72	18.0	1.16	2.50	12.4	6.10	15.7	16.8	3.88	14.4	11.5	9.30
GenModel [146]	2.57	4.74	13.0	1.72	3.08	16.9	6.86	15.0	19.2	4.64	14.9	11.4	9.50
RTCensus [61]	5.08	6.25	19.2	1.58	2.42	14.2	7.96	13.8	20.3	4.10	9.54	12.2	9.73
ReliabilityDP [47]	1.36	3.39	7.25	2.35	3.48	12.2	9.82	16.9	19.5	12.9	19.9	19.7	10.7
GF [11]	2.98	3.43	9.24	1.93	2.36	11.0	11.9	17.1	21.8	11.9	17.4	19.9	10.9
DCBGrid [87]	5.90	7.26	21.0	1.35	1.91	11.2	10.5	17.2	22.2	5.34	11.9	14.9	10.9
CSBP [57]	2.00	4.17	10.5	1.48	3.11	17.7	11.1	20.2	27.5	5.98	16.5	16.0	11.4
H-Cut [147]	2.85	4.86	14.4	1.73	3.14	20.2	10.7	19.5	25.8	5.46	15.6	15.7	11.7
TreeDP [46]	1.99	2.84	9.96	1.41	2.10	7.74	15.9	23.9	27.1	10.0	18.3	18.9	11.7
SAD-IGMCT [62]	5.81	7.14	22.6	2.61	3.33	25.3	9.79	15.5	25.7	5.08	11.5	15.0	12.5
SSD+MF [58]	5.23	7.07	24.1	3.74	5.16	11.9	16.5	24.8	32.9	10.9	19.8	26.3	15.7
LCDM+AdaptWgt [141]	5.98	7.84	22.2	14.5	15.4	35.9	20.8	27.3	38.3	8.90	17.2	20.0	19.5

Table 4.4: Comparison of the proposed LASW method (without refinement) with LNDA

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
GW+PM	3.21	3.37	17.3	0.41	0.71	4.29	7.70	14.1	20.8	5.59	11.6	15.5	8.71
GW+LNDA	5.84	6.73	14.1	2.81	3.93	18.5	12.2	16.3	26.6	6.88	14.2	18.8	12.2

and filling method can take any disparity map as an initial estimate. This initial disparity map can be generated by any popular disparity map estimation algorithms which essentially have two important steps: matching cost computation and cost aggregation. For this analysis, the initial disparity map is estimated by the method proposed in [11], and subsequently this initial estimate is processed by our proposed occlusion detection and filling LASW method. Cost for a pixel $\mathbf{p}$ in image $\mathbf{I}_l$ having

matching pixel in another image $\mathbf{I}_r$ with disparity value d is given by:

$$C(\mathbf{p}, d) = (1-\alpha) \cdot \min\left[\left|I_l(p1, p2) - I_r(p1-d, p2)\right|, \tau_1\right] + \alpha \cdot \min\left[\left|\nabla_x I_l(p1, p2) - \nabla_x I_r(p1-d, p2)\right|, \tau_2\right] \tag{4.17}$$

Here, ∇_x is the gradient in the x direction, α balances the colour and gradient terms, and τ_1, τ_2 are truncation values. Cost aggregation is performed by a guided filter as follows:

$$C_{agg}(\mathbf{p}, d) = \sum_{\mathbf{q} \in \mathcal{N}_p} w(\mathbf{p}, \mathbf{q}) C(\mathbf{q}, d) \tag{4.18}$$

Here, the weights are calculated using Equation (4.16). This evaluation is shown in Table 4.5. In this Table, the overall disparity map estimation method is shown as a combination of Step 1 + Step 2. Step 1 shows the methods by which the initial disparity map is estimated. For this, Gabor wavelet (GW) [15] and Guided filter (GF) [11] are considered. Similarly for Step 2, either the proposed occlusion detection and filling method (LASW) or LNDA [11] is used. Performance evaluation of our occlusion detection and filling method with LNDA for initial disparity map generated by GW and GF is illustrated in Figure 4.12. So, the information shown in Table 4.5 can be roughly visualized in Figure 4.12. Table 4.6 shows a comparison of percentage of occluded pixels in an image that are incorrectly filled by the proposed method and the existing occlusion filling methods [14]. Each cell of Table 4.6 has two rows. The upper row shows the error percentage, while the lower row shows the rank of performance for the particular error measure. The term score mentioned in Table 4.6 denotes the average of these ranks. Here, initial disparity map is obtained by using the ADCensus method proposed in [148]. The main difference between our proposed occlusion detection method and the

Table 4.5: Performance estimation for the cases when different initial disparity map estimation methods are used with our proposed LASW method and LNDA method for finding a disparity map

Algorithm	Tsukuba			Venus			Teddy			Cones			Avg. %
	Nocc	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	*Nocc*	*all*	*disc*	
GW+PM	2.34	2.57	12.6	0.40	0.66	3.99	7.73	14.1	20.3	5.21	10.9	14.3	7.93
GF+PM	2.62	2.83	14.2	0.39	0.70	3.62	7.98	13.7	20.6	5.45	11.3	14.6	8.17
GW+LNDA	3.36	3.83	11.8	0.43	0.71	5.13	8.59	12.8	21.6	5.05	12	14.7	8.32
GF+LNDA	2.98	3.43	9.24	1.93	2.36	11.0	11.9	17.1	21.8	11.9	17.4	19.9	10.9

SDOD method [16] is the employment of a linear fitting model in our proposed method. As discussed earlier, the behaviour of reference and target pixels are analyzed using epipolar, uniqueness, ordering, and continuity constraints in our proposed method. From the analysis, it is observed that the target

Table 4.6: Percentages of errors in occlusion filling by our proposed SWOF method and other methods (NDA, DIS, WLS, and SLS) [14]

Algorithm	Tsukuba	Venus	Teddy	Cones	Score
NDA	13.20	5.60	44.88	45.93	3.25
	5	4	1	3	
DIS	9.3	4.87	50.99	49.56	3.25
	2	1	5	5	
WLS	12.31	5.52	45.00	48.11	3.25
	4	3	2	4	
SLS	10.45	4.94	47.73	41.99	2.75
	3	2	4	2	
LASW	7.93	20.73	45.36	36.44	2.5
	1	5	3	1	

NDA-Neighbours Disparity Assignment
DIS-Diffusion in Intensity Space
WLS-Weighted Least Squares
SLS-Segmentation-based Least Squares

matching pixels approximately follow a linear pattern with respect to the reference pixels. Hence, it is approximated by a linear regression model (best or linear fit), and subsequently this model is used to detect occluded pixels in our proposed method. So, our linear regression model is also suitable for detecting the occluded pixels in horizontally slanted surfaces as compared to SDOD method. In the next paragraph, we have explained how our method is also suitable for detection of occluded pixels in a horizontally slanted surface.

Table 4.7: Performance of SDOD method in the region of horizontally slanted surface

Pixel location in reference image	60	61	62	63	64	65	66	67	68	69	70	71
Disparity value	21	21	21	22	22	22	23	23	23	23	24	24
Pixel location in target image	30	40	41	41	42	43	43	44	45	46	46	47
Visible/occluded pixels	Visible	Visible	Occluded	Visible	Visible	Occluded	Visible	Visible	Visible	Occluded	Visible	Visible
Reason	Both the pixels 62 and 63 in the reference image corresponds to pixel 41 in the target image. Subsequently, ordering and photometric constraints are used to decide which pixel is occluded, and which pixel is visible. These constraints fail to detect all these pixels shown above as visible.											

In SDOD method, uniqueness, ordering, and photometric constraints are used to detect the occluded pixels. In this method, these constraints are sequentially applied to detect the occluded pixels. In our LAOD method, these constraints are jointly used to mathematically model the target matching pixels by a linear fit. But in the presence of horizontally slanted surface/object in a scene, the object appears horizontally stretched in one image as compared to other image. During stereo correspondence, this may result in correspondence of M number of pixels in one image to N number of pixels in other image. This results in labelling of $|M - N|$ pixels as occluded. As the horizontally slanted

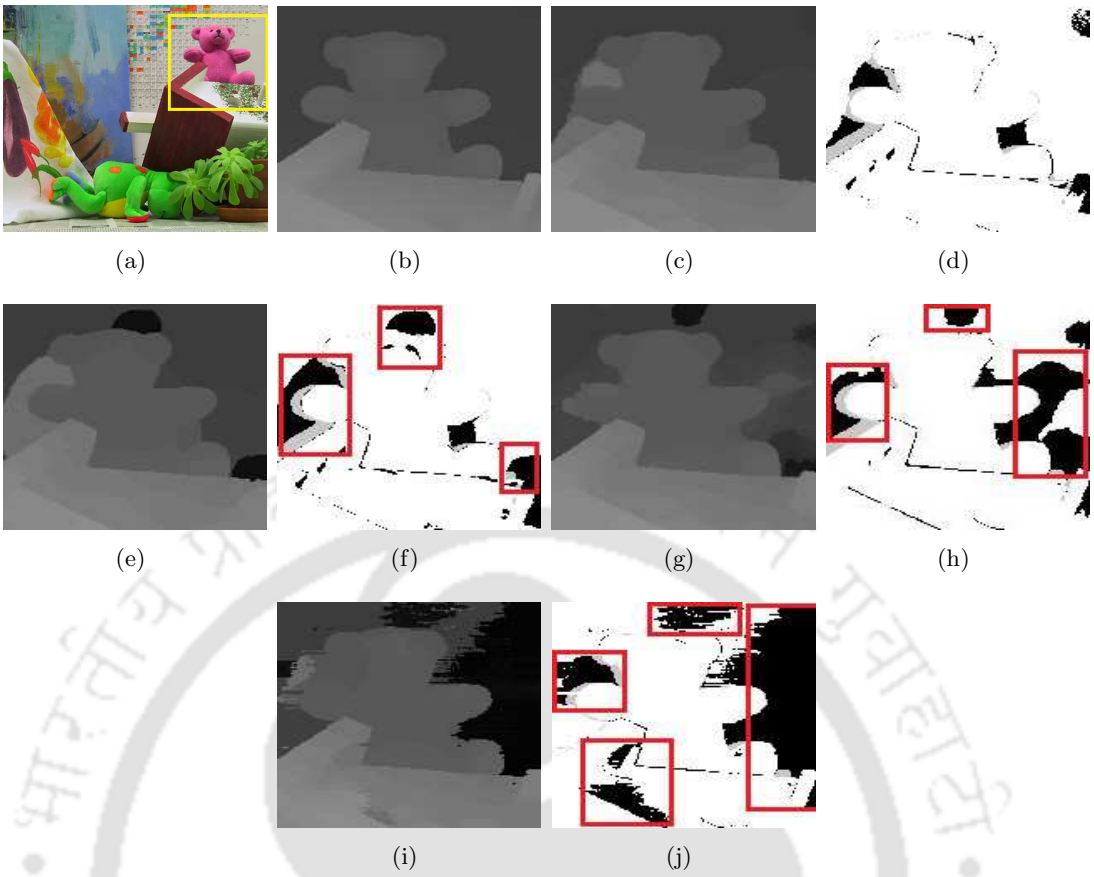


Figure 4.12: Performance comparison of our LASW method with LNDA. (a) Teddy image; (b) Ground truth disparity map of the image patch; (c) Disparity map obtained by using GW+LASW; (d) GW+LASW error estimated using the images (b) and (c); (e) Disparity map obtained by using GF+LASW; (f) GF+LASW error estimated using the images (b) and (e); (g) Disparity map obtained by using GW+LNDA; (h) GW+LNDA error estimated using the images (b) and (g); (i) Disparity map obtained by using GF+LNDA; (j) GF+LNDA error.

object is visible in both the images, the interleaved matching pixels are present in the target image as well, but not in the integer coordinates. Thus, the $|M - N|$ visible pixels are wrongly detected as occluded. Hence, the uniqueness constraint which imposes one-to-one correspondence may not be appropriate for occluded pixels detection from a scene having horizontally slanted surfaces [143]. Similarly, ordering and photometric constraints are not also be appropriate to detect the occluded pixels in this scenario. The behaviour of SDOD method in a region of horizontally slanted surface is shown by giving a numerical example in Table 4.7. Here, few pixels which belong to a slanted surface are detected as occluded.

In our proposed LAOD method, it is observed that the target matching pixels approximately follow a linear pattern even in the presence of horizontally slanted object in the scene. In the regions of horizontally slanted surfaces, many pixels in the reference image corresponds to a single pixel in

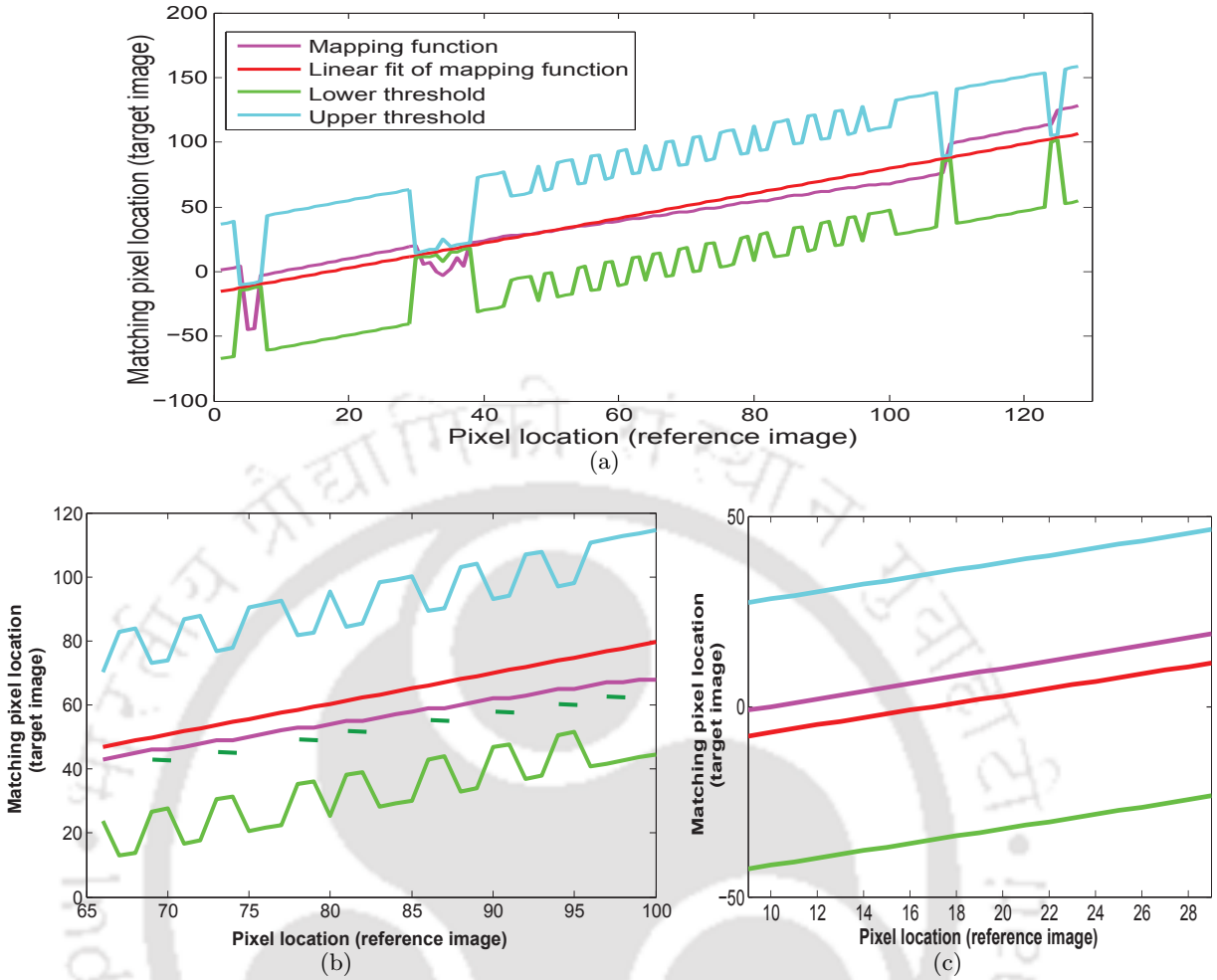


Figure 4.13: Proposed mapping function. (a) Proposed mapping function along with the best fit, upper and lower thresholds; (b) Regions showing many pixels in reference image mapping to a single pixel in the target image (blue rectangle in (a)); (c) Regions showing a pixel in reference image mapping to a pixel in the target image (brown rectangle in (a)).

the target image. This leads to small variations in the disparity values of these regions. Hence, there is a small deviation of the mapping function from the linear pattern, and this deviation is taken care by the best fit. In addition to this, the deviation of the mapping function from the best fit is small, and lies within the thresholds. So, this consideration helps in detecting the pixels of the horizontally slanted surfaces as visible. On the other side, there is a drastic change in the disparity values for occluded regions. This leads to comparatively large deviation of the mapping function from the best fit, and hence the mapping function falls outside the estimated thresholds. This procedure enables us to detect the occluded pixels more accurately. The proposed mapping function along with the best fit, upper, and lower thresholds is shown in Figure 4.13. In this, a region of horizontally slanted surface is shown by a blue coloured rectangle in Figure 4.13(a). The correspondence of many pixels in the

reference image to a single pixel in the target image is shown by a red coloured short line parallel to the mapping function (purple graph) in Figure 4.13(b). Figure 4.13(b) is an enlarged version of a portion of the mapping function of Figure 4.13(a). Figure 4.13(c) shows mapping of a single pixel in the reference image to a single pixel in the target image along with the best fit and thresholds (shown by a brown coloured rectangle in Figure 4.13(a)). This figure is also an enlarged version of the initial portion of the mapping function of Figure 4.13(a). Figure 4.14 shows the proposed mapping function

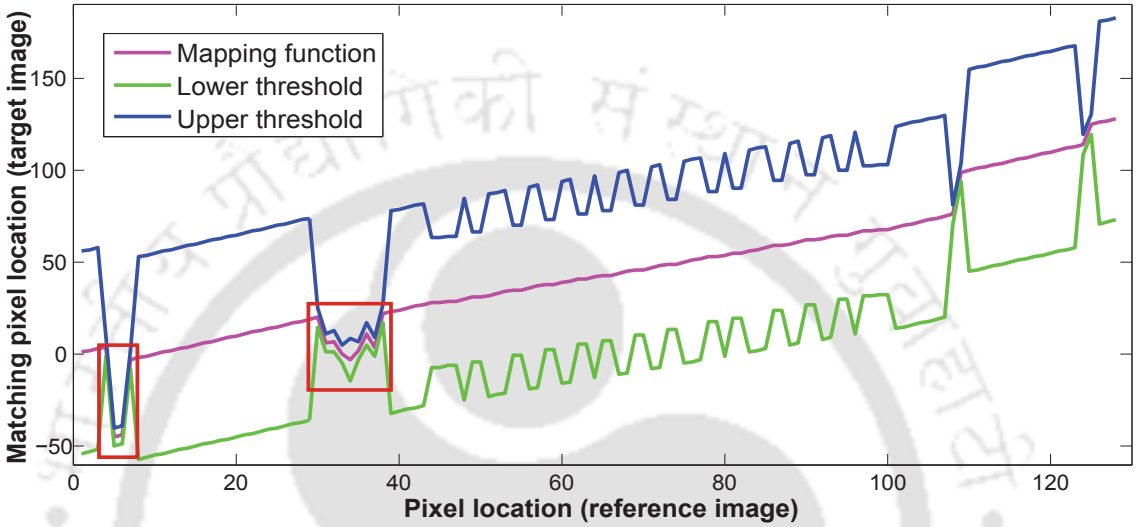


Figure 4.14: Proposed mapping function along with the upper and lower thresholds.

along with the upper and lower thresholds. Here, upper and lower thresholds are obtained using the mapping function instead of best fit. This figure shows that the mapping function always falls (both visible and occluded pixels) in between the thresholds, as these thresholds are calculated from the mapping function. Hence, thresholds calculated as above fail to detect the occluded pixels (shown by the brown coloured boxes in Figure 4.14). This illustration shows the importance of the best fit in estimating the thresholds. These thresholds are subsequently used for detection of the occluded pixels.

Figure 4.15 shows the detection of occluded pixels in presence of horizontally slanted surface by our proposed LAOD method and SDOD method. Figure 4.15(a) and Figure 4.15(b) show the reference and target stereo images having horizontally slanted surface. This surface is horizontally stretched in one image as compared to the other image. Figure 4.15(c) shows the generated disparity map. The occluded regions are marked by the red coloured boxes. Occluded pixels detected by SDOD method and our LAOD method are shown in Figure 4.15(d) and Figure 4.15(e) respectively. It is quite evident that the proposed LAOD method is able to detect the occluded pixels more accurately as compared

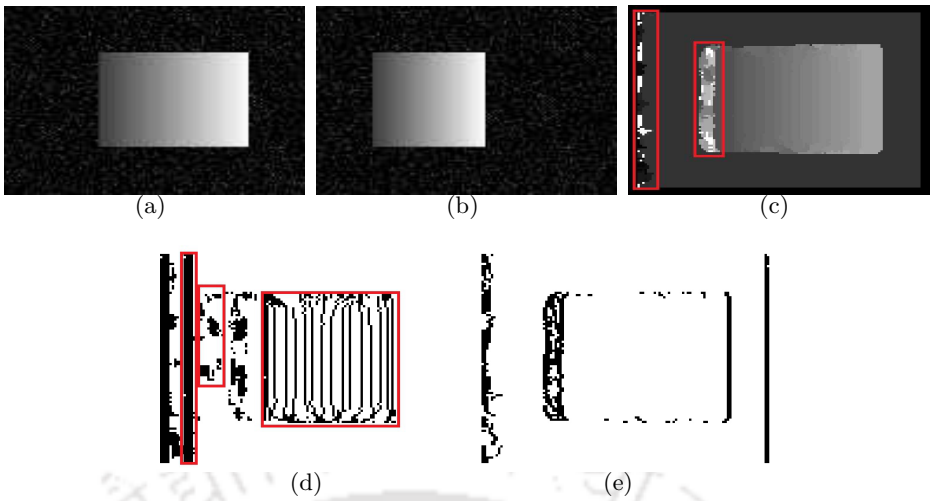


Figure 4.15: Detection of occluded pixels (shown in black colour). (a) Left image; (b) Right image; (c) Disparity map; (d) Occluded pixels detected by SDOD method [16]; (e) Occluded pixels detected by our proposed method.

to SDOD method. Additionally, SDOD method detects many visible pixels as occluded pixels which is shown by the red coloured boxes in Figure 4.15(d).

4.5 Summary

Occlusion is one major challenge in estimating a fine disparity map. In order to obtain an accurate disparity map, the occluded pixels need to be effectively detected. Subsequently, an appropriate disparity value is assigned to a detected occluded pixel. Our proposed occlusion detection method is presented in this chapter. This method simultaneously detects both the occluded pixels and the pixels having wrong disparity values. Our proposed occlusion detection method only makes use of single disparity map for detecting the occluded pixels as against the methods which use two disparity maps. Apparently, computational burden is reduced as the comparison is done only with a single disparity map. Also, the existing single disparity map-based occlusion detection method detects a candidate set of occluded pixels, and some of the visible pixels may be present in that set. For such a case, more false positives are obtained. On the other hand, our method can give comparatively less false positives as our method is proposed only to detect the occluded pixels. So, the chance of inclusion of visible pixels in the set of occluded pixels would be less. In last part of this chapter, our proposed occlusion filling scheme is described. Our proposed occlusion filling method employs both the stereo images for assigning appropriate disparity values to the occluded pixels. On account of this, assignment of an appropriate disparity value to a detected occluded pixel would be reasonable. Experimental results presented in this chapter show the efficacy of our proposed occlusion detection and filling methods.

5

Performance Analysis of Gabor Wavelet for Extracting Informative and Efficient Features

Features extracted by Gabor wavelet have similar information as visualized by the receptive field of simple cells in the visual cortex of the mammalian brains. This motivates us to analyze Gabor features to evaluate their effectiveness in representing an image. In this chapter, two major characteristics of Gabor features are established viz., (i) Real coefficients of Gabor wavelet alone is sufficient enough to represent an image; and (ii) Local Gabor wavelet features with overlapping regions represent an image more accurately as compared to the global Gabor features and the local features extracted for the non-overlapping regions. The efficacy and effectiveness of these findings are evaluated by reconstructing the original image using the extracted features, and subsequently the reconstructed image is compared with the original image. Experimental results show that the local Gabor wavelet features extracted from overlapping regions represent an image more efficiently than the global and non-overlapping region-based features. Experimental results also show that the real coefficients alone is sufficient enough to represent an image more accurately as compared to the imaginary and magnitude information.

5.1 Introduction

Feature extraction is an important and active research topic of Computer Vision. Also, many multimedia applications such as face recognition, texture image classification, image indexing and retrieval employees Gabor wavelet for feature extraction [149, 150]. Some of these applications need a disparity map which can be obtained from stereo correspondence [151–153]. The disparity map gives an additional information for these applications. Again, feature extraction may be local or a global process. Global features extract a prominent information of an image, whereas local features extract a detailed information. Depending on the applications, either local or global features may be employed. Gabor filter was first proposed by Daugman, and it is extended for two-dimensional domain to extract the information of image textures [154, 155]. This is done by convolving an image with the Gabor wavelet kernel. Some of the applications which use a Gabor wavelet for feature extraction, and subsequent pattern classification are discussed in this context. In general, the feature vector is the concatenation of features extracted by a Gabor filter for different orientations and scales [156]. Another way of extracting Gabor features is by applying gray level co-occurrence matrix over the Gabor wavelet convolved images. In another approach, covariance matrix is calculated for all the Gabor filtered images [157, 158]. Finally, non-duplicated values of the covariance matrix are used as features. The simplicity and success of local binary pattern (LBP) in many applications motivates the researchers to use LBP on Gabor filtered images [159]. Another recent approach is the adaptation of fractal signature of magnitude coefficients for efficient texture feature extraction [160]. Zhang *et al.* used histogram of Gabor phase pattern (HGPP) as a feature for face recognition [161]. Xu *et al.* extracted Gabor features from depth and intensity images, and subsequently used this feature for face recognition [162]. Jahanbin *et al.* obtained Gabor features separately from co-registered range and portrait image pairs at fiducial points [124]. These two features are merged and used for face recognition. Yang *et al.* also used Gabor wavelet for face classification [163]. This method takes care the cases of occlusion by constructing a compact occlusion dictionary from Gabor features. In [164], facial expression representation model is proposed using the statistical characteristics of training images. Texture and shape information are used to measure the similarity between the testing images and the facial expression models. Zhang *et al.* proposed a FER system with a capability of handling occlusion [165]. A set of face templates are extracted from the Gabor filtered images using Monte Carlo algorithm. Extracted features are robust to occlusion.

Gabor wavelet is also used for image indexing and retrieval [166]. Input image is decomposed by Gabor wavelet at different scales and orientations. The obtained coefficients contain certain redundant

information. Edge detection using simplified Gabor wavelet is presented in [167]. In this method, initially input image is convolved with quantized imaginary Gabor filter by considering two orientations and one scale. Shen and Jia proposed a three-dimensional Gabor wavelet for hyperspectral image classification [168]. This three-dimensional Gabor wavelet is convolved with an input image to obtain the feature vector. Two-dimensional Gabor wavelet-based automatic retinal vessel segmentation is proposed in [169]. In this method, image is filtered by a two-dimensional Gabor wavelet with different orientations and scales. At a particular scale, maximum value of the coefficients for all the possible orientations is taken for each pixel. This procedure is repeated for all the scales to form the feature vector.

It has to be noted that the abovementioned methods use magnitude information of Gabor wavelet, which require both real and imaginary coefficients. In this chapter, an extensive analysis of Gabor filter properties is presented. It has been established that the real coefficients of a Gabor filter alone can be more effectively used to extract necessary information of an image in place of magnitude information. To validate our claim, local Gabor wavelet features are extracted for all the image pixels from the overlapping neighbouring regions. The performance of this local Gabor wavelet feature is compared with the global Gabor features. Additionally, the performance of the above local feature is also compared with the local features extracted from non-overlapping regions.

Jones and Palmer mentioned that an optimal performance of two-dimensional Gabor filter can be obtained by using real part of the filter in [170]. Further, Daugman proposed a new feature extraction method by considering elementary two-dimensional Gabor functions [171]. In this paper, a neural network is employed to achieve this task. In these two papers, authors simply mentioned "real part of the complex Gabor function is a good fit to the receptive field weight functions found in simple cells in a cats striate cortex". They did not establish their claim either experimentally and/or analytically. But in our analysis, extensive experimental evaluations are done to validate the effectiveness of the real coefficients of the Gabor features. It is also observed from our experimental analysis that real coefficients can give almost similar performance as that given by the magnitude information for the applications like stereo correspondence. Additionally, extensive experimental investigations are done for analyzing the characteristics of Gabor features for synthetic illumination changes and real radiometric variations. Based on the above literatures and observations, major contributions of this chapter can be summarized as follows:

- Real coefficients of Gabor wavelet alone are sufficient enough to represent an image. Our claim is validated by considering three different cases: (i) Global Gabor features, (ii) Local Gabor features extracted from overlapping regions, and (iii) Local Gabor features extracted from non-

overlapping regions. In all these cases, it is observed that real coefficients alone can represent an image more efficiently as compared to the Gabor imaginary coefficients. Also, the real coefficients can represent an image almost in the similar manner as compared to the magnitude information.

- Three different features (global Gabor features, local Gabor features for overlapping regions and local Gabor features for non-overlapping regions) are analyzed. It is again observed that local Gabor features for overlapping regions can represent an image more accurately compared to other two counterparts.

- Robustness of all the three Gabor features are analyzed for radiometric variations in a scene, and we found that the real coefficients of local Gabor features for overlapping regions are more robust as compared to the Gabor features extracted from the imaginary part or magnitude information. Also, this method is significantly better than the local Gabor features for non-overlapping regions and the global features.

5.2 Basics of Gabor Wavelet

Two-dimensional Gabor functions are Gaussian modulated complex sinusoids, which is given by [121]:

$$\begin{aligned} g(x,y) &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left[(\cos \kappa x + i \sin \kappa x) - e^{-\frac{\kappa^2}{2}} \right] \\ &= \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} \left(\cos \kappa x - e^{-\frac{\kappa^2}{2}} \right) + i \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{8}(4x^2+y^2)} (\sin \kappa x) \\ &= g^e(x,y) + i g^o(x,y) \end{aligned} \tag{5.1}$$

In Equation (5.1), $g^e(x,y)$ and $g^o(x,y)$ represent the real and imaginary part of $g(x,y)$.

5.3 Global Gabor Wavelet Feature (GGWF) Extraction

Global features can be used to represent the entire image, while local features can represent a particular region of an image. In most of the cases, the dimension of global features are less than the dimension of the input image. So, the global feature extraction methods may be considered as the dimensionality reduction techniques. Global features are fast to compute [1].

Let us consider an image I of size $M \times N$. In order to obtain the GGWF, the input image I is convolved with the Gabor function which is tuned to different scales and orientations, and the obtained

[1] This work has been accepted for publication in *Multimedia Tools and Applications (Springer)* (Refer item [3] in Page 135 for details)

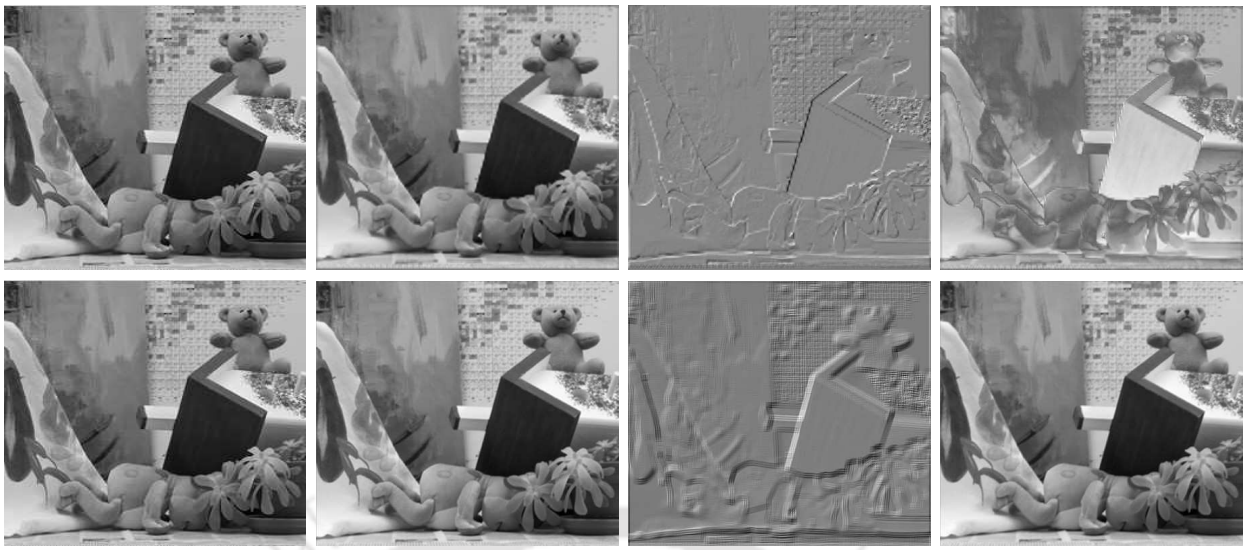


Figure 5.1: Gabor wavelet filtered images. First row - Global Gabor filtered images (GGWF) and second row - local Gabor filtered images (LGWF) for overlapping regions. Input image, image represented only using the real coefficients, image represented only using the imaginary coefficients, and the image represented using the magnitude information are shown from the left to right in this figure.

coefficients represent the global feature F. Mathematically, it is given by:

$$F_{rs} = I * g_{rs}, \forall r = 0, 1, \cdots, (m-1), s = 0, 1, \cdots, (n-1) \tag{5.2}$$

where "$*$ "is the convolution operator. For a Gabor kernel of size $N_g \times N_g$, the filtered image F_{mn} is given by:

$$\begin{aligned} F_{mn} &= I * g_{mn} \\ &= \sum_{k=1}^{N_g} \sum_{l=1}^{N_g} I(x-k, y-l) g_{mn}(k, l) \end{aligned} \tag{5.3}$$

g_{mn} is obtained by rotating $g(x, y)$ for n^{th} angle at m^{th} scale. Since Gabor function is complex, the above Equation (5.3) can be separated into real and imaginary parts given by:

$$F_{mn}^{real} = I * g_{mn}^{e}(x, y) \tag{5.4}$$

$$F_{mn}^{imag} = I * g_{mn}^{o}(x, y) \tag{5.5}$$

g_{mn}^{e} and g_{mn}^{o} are the real and imaginary part of g_{mn}. The magnitude of the Gabor filtered output is calculated as follows:

$$F_{mn}^{mag} = \sqrt{(F_{mn}^{real})^2 + (F_{mn}^{imag})^2} \tag{5.6}$$

First row of Figure 5.1 shows images represented using global Gabor wavelet features. In this figure, input image, image represented using the real coefficients, image represented using the imaginary coefficients, and the image represented using the magnitude information for $m = 2, n = 2$ in Equations (5.2), (5.4), (5.5), and (5.6) are shown from left to right. Approximate reconstruction of an input

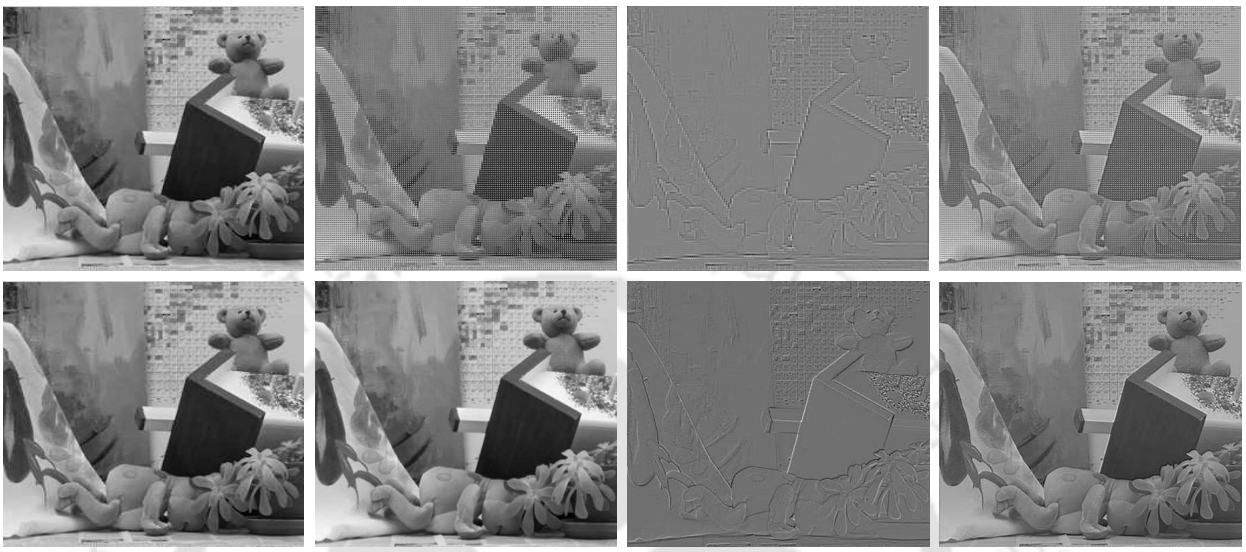


Figure 5.2: Reconstruction of original image with GGWF and LGWF. First row shows the reconstructed image using GGWF, and the second row shows the reconstructed image using LGWF for overlapping regions. Input image, image reconstructed only using the real coefficients, image reconstructed only using the imaginary coefficients, and the image reconstructed using the magnitude information are shown from the left to right in this figure.

image is possible by the linear superposition of the bases weighted by the wavelet coefficients [121]. The reconstructed image I^{recon} is given by:

$$I_{rs}^{recon} = \langle F_{rs}, g_{rs} \rangle \, g_{rs} \tag{5.7}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product. Teddy image reconstructed using the extracted global features is shown in first row of Figure 5.2. From both Figure 5.1 and 5.2, it is observed that the image represented/reconstructed using the real coefficients and the magnitude are almost visually similar to the input image, whereas image represented using the imaginary coefficients significantly visually different as compared to the image represented/reconstructed by both real and magnitude informations. Image reconstructed using only the real coefficients is similar to the image reconstructed using the magnitude information, but memory requirement is reduced by half when only the real coefficients are used [131]. This is due to the fact that both real and imaginary information are needed to be stored to extract the magnitude information, which is not the case for storing only the real information.

5.4 Local Gabor Wavelet Feature (LGWF) Extraction

Local features represent a region of interest of an image. Generally, these local features are obtained by considering the gray-scale value and/or colour information of a pixel. A good local feature should uniquely represent a particular point in an image. Local features are generally used for the applications such as stereo matching, object tracking, three-dimensional calibration, and three-dimensional reconstruction. In this chapter, local features are extracted using both overlapping and non-overlapping regions. The procedure of LGWF extraction, and subsequent reconstruction of the original image by using the extracted features is discussed below. Figure 5.3 shows the block diagram

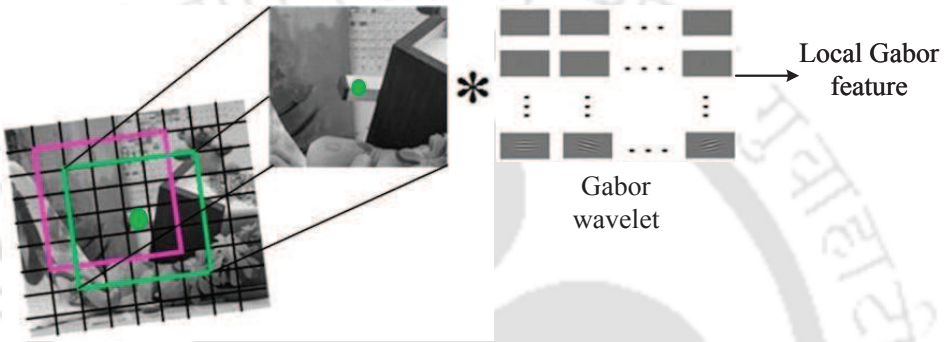


Figure 5.3: Local Gabor wavelet (overlapping region) feature extraction.

of Gabor wavelet-based local feature extraction from overlapping regions, where the square (pink and green colour) represents two local image patches. In this figure, the pixel for which the feature is extracted is shown by a small green circle. It is seen that the neighbouring regions of the two patches have a common region and hence, this process is called as feature extraction from overlapping regions. Let us consider an image I of size $M \times N$. To find the local feature vector for the pixel $I(p_1, p_2)$, a certain neighbourhood $\mathbf{S}_p$ is considered. This patch is convolved with the Gabor kernel g_{mn} for different orientations and scales, which can be represented as:

$$\begin{aligned} \mathbf{G}^l &= \mathbf{S}_p * \mathbf{g}^l_{rs},\ l \in \{e, o\} \\ \boldsymbol{\chi}_{rs} &= \begin{bmatrix} \mathbf{G}^l_{r0} & \mathbf{G}^l_{r1} & \cdots & \mathbf{G}^l_{r(n-1)} \end{bmatrix} \\ \mathbf{G}^l_I(p_1, p_2) &= vec \begin{bmatrix} \boldsymbol{\chi}_{0s} & \boldsymbol{\chi}_{1s} & \cdots & \boldsymbol{\chi}_{(m-1)s} \end{bmatrix} \end{aligned} \tag{5.8}$$

This procedure is repeated for all the pixels of an image, and each pixel has a set of Gabor coefficients. The images represented using the real, imaginary, and the magnitude information for Teddy image with $m = 2, k = 2$ are shown in second row of Figure 5.1. Reconstruction of the original image using the real coefficients, imaginary coefficients, and both real and imaginary coefficients (magnitude part)

of local Gabor wavelet are performed by using the below equation:

$$I_L^{recon}(p_1, p_2) = \frac{1}{M \times N} \left[\sum_{r,s} \langle F_L, g_{rs} \rangle g_{rs} \right] \tag{5.9}$$

where $M \times N$ is the size of reconstructed image region for a pixel. Figure 5.2 shows the images which are reconstructed using the extracted local features from the overlapping regions. The reconstructed images using the real coefficients, imaginary coefficients, and the magnitude information by using Equation (5.9) are shown in second row of Figure 5.2. An input image is partitioned into subregions of size $u_1 \times v_1$ to find the LGWF from the non-overlapping regions. In this case, the extracted coefficients characterize the entire region, whereas in the case of overlapping regions, the extracted coefficients characterize a particular pixel. Equations (5.8 - 5.9) with slight modifications can also be employed to represent and reconstruct the original image for non-overlapping regions.

5.5 Experimental Results

To evaluate the performance of the extracted Gabor features, Middlebury stereo images are used [7, 12, 13, 58, 172]. Experimental evaluations are performed by considering different window sizes, various orientations, and scales[2]. Additionally, performance of these features are evaluated for synthetic illumination changes (gain change, bias change, gamma, and vignetting changes). Also, these evaluations are performed for real radiometric changes which include the change in exposure and light source. Metrics used for these evaluations are mean square error (MSE), correlation coefficient (CC), universal quality index (UQI or QI), and structural similarity index (SSI) [173]. Mathematically, these metrics can be expressed as follows:

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left(I\left(i,j\right) - J\left(i,j\right) \right)^2 \tag{5.10}$$

where I is an input image and J is the reconstructed image.

$$CC = \frac{\text{cov}(I, J)}{\sigma_I \sigma_J} \tag{5.11}$$

[2]This work has been published in *Smart innovations, systems and technologies 2016* (Refer item [6] in Page 135 for details)

Table 5.1: Comparison (by MSE and CC) of the local features (both overlapping and non-overlapping regions) for different window sizes

Window	MSE(overlapping)			MSE(non-overlapping)			CC(overlapping)			CC(non-overlapping)		
size	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
9×9	623	1718	817	912	2678	1120	0.901	0.724	0.901	0.846	0.198	0.846
11×11	681	1345	739	890	3293	1373	0.891	0.742	0.891	0.858	0.202	0.858
13×13	955	2391	1009	591	3320	1191	0.871	0.574	0.871	0.886	0.106	0.886
15×15	953	2818	1106	769	2742	1177	0.872	0.491	0.872	0.851	0.202	0.851

where $cov(I, J)$ denotes the covariance between the input and reconstructed images. The symbols σ_I and σ_J are the standard deviation of the input and reconstructed images respectively.

$$\text{QI} = \frac{\sigma_{IJ}}{\sigma_I \sigma_J} \frac{2\bar{I}\bar{J}}{\left(\bar{I}\right)^2 + \left(\bar{J}\right)^2} \frac{2\sigma_I \sigma_J}{\sigma_I^2 + \sigma_J^2} \tag{5.12}$$

where σ_{IJ} represents the standard deviation of the product of I and J images.

$$\bar{I} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} I\left(i, j\right), \bar{J} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} J\left(i, j\right)$$

$$SSI = \frac{\left(2\bar{I}\bar{J} + c_1\right)\left(2\sigma_{IJ} + c_2\right)}{\left(\bar{I}^2 + \bar{J}^2 + c_1\right)\left(\sigma_I^2 + \sigma_J^2 + c_2\right)} \tag{5.13}$$

Lesser values of MSE and larger values of CC, QI, and SSI indicate better similarity of the reconstructed image with the original image. For evaluation, images are reconstructed using both the global and local features which are extracted from both overlapping and non-overlapping regions. The reconstructed images are normalized in the range of [0-1]. Figure 5.4 shows the image represented using the real coefficients, image represented using the imaginary coefficients, and the image represented using the magnitude information extracted from both global and overlapping regions. In the following Figures and Tables, *real*, *imag* and *mag* denote the real, imaginary, and magnitude information of the extracted features, while OL, NO and G represent overlapping, non-overlapping, and global regions.

5.5.1 Different window sizes

Table 5.1 and Table 5.2 show the performance of local features which are extracted from overlapping and non-overlapping regions for different window sizes. Performance of the features for overlapping regions decreases with the increase of window size, whereas the performance of features extracted from non-overlapping regions increases with the increase of window size. For overlapping regions,

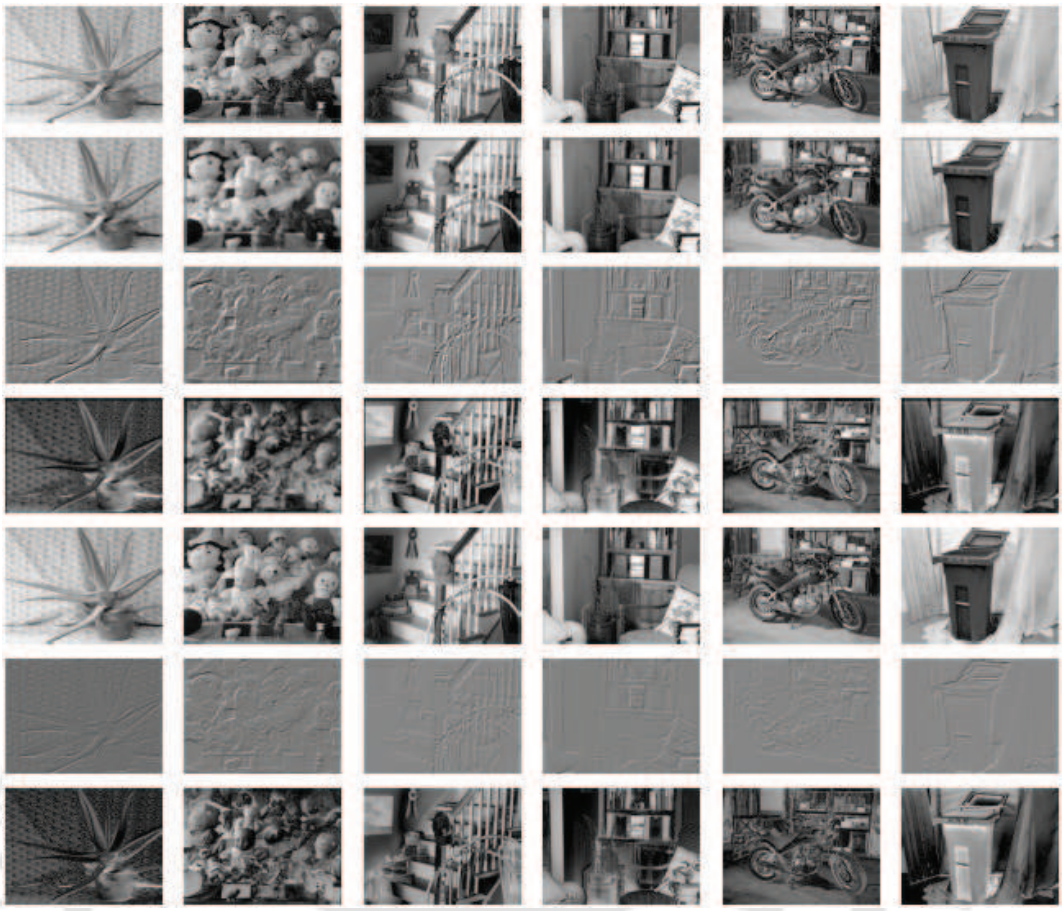


Figure 5.4: Experimental results for some additional images; First row - input image, second, third and fourth rows show features extracted from overlapping regions; Fifth, sixth ,and seventh rows show features extracted from global regions; Second and fifth rows show the image represented by using real coefficients, third and sixth rows show the image represented by using imaginary coefficients; fourth and seventh rows show the image represented by using magnitude information; Left to right - Aloe, Dolls, Hoops, Livingroom, Motorcycle, and Recycle images.

feature vectors are extracted for each of the pixels of an image. So if the window size is increased, many neighbouring pixels influence the feature vector of the center pixel. The extracted features can characterize a pixel more effectively when all the neighbouring pixels belong to a homogeneous region. On the other hand, the feature vector cannot effectively represent a pixel for the case when some of the neighbouring pixels belong to different regions. So, the reconstruction error increases with the increase of window size. For non-overlapping regions, a feature vector is extracted for each of the image patches. When the window size is increased, it encloses more number of pixels. These pixels in turn provide additional information to the extracted feature of the patch. Hence, the extracted feature vector will be able to effectively represent an image patch.

Table 5.2: Comparison (by SSI and QI) of the local features (both overlapping and non-overlapping regions) for different window sizes

Window	SSI(overlapping)			SSI(non-overlapping)			QI(overlapping)			QI(non-overlapping)		
size	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
9×9	0.732	0.338	0.606	0.614	0.178	0.556	0.708	0.285	0.572	0.600	0.121	0.538
11×11	0.689	0.367	0.581	0.605	0.189	0.470	0.662	0.311	0.544	0.584	0.140	0.444
13×13	0.635	0.192	0.484	0.716	0.181	0.562	0.605	0.123	0.438	0.705	0.125	0.542
15×15	0.626	0.121	0.445	0.623	0.176	0.516	0.596	0.045	0.394	0.608	0.114	0.493

Table 5.3: Comparison (by CC) of the global and local features (both overlapping and non-overlapping regions) for different number of orientations

No. of	CC(overlapping)			CC(non-overlapping)			CC(global)		
orientations	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
1	0.968	0.897	0.968	0.886	0.077	0.886	0.800	0.274	0.800
2	0.968	0.927	0.968	0.886	0.106	0.886	0.902	0.365	0.902
3	0.968	0.927	0.968	0.885	0.106	0.885	0.971	0.370	0.971
4	0.968	0.927	0.968	0.885	0.106	0.885	0.973	0.370	0.973
5	0.968	0.927	0.968	0.885	0.106	0.885	0.971	0.370	0.971
6	0.968	0.927	0.968	0.885	0.106	0.885	0.971	0.370	0.971
7	0.968	0.927	0.968	0.885	0.106	0.885	0.971	0.370	0.971
8	0.968	0.927	0.968	0.885	0.106	0.885	0.971	0.370	0.971

5.5.2 Different number of orientations

The performance of GGWF and LGWF (both overlapping and non-overlapping regions) for different number of orientations $K = 1, 2, 3, 4, 5, 6, 7,$ and 8 (refer Equation (4.2)) are shown in Table 5.3, and in Figure 5.5. For local features, there is no significant change in the performance for more number of orientations, whereas global feature shows better performance with the increase of number of orientations. The entire image used for global feature extraction has more pixel variations as compared to the pixel variations in the image patches which is used for local feature extraction. So, global feature is able to represent these variations more efficiently with the increase of number of orientations. In case of the local feature, few number of orientations are sufficient enough to represent the pixel variations in the image patches. This finding is illustrated for global features in Figure 5.6. Figure 5.6 shows the reconstructed Cones image for 2, 4, 6, and 8 number of orientations. An image can be reconstructed more accurately with more number of orientations.

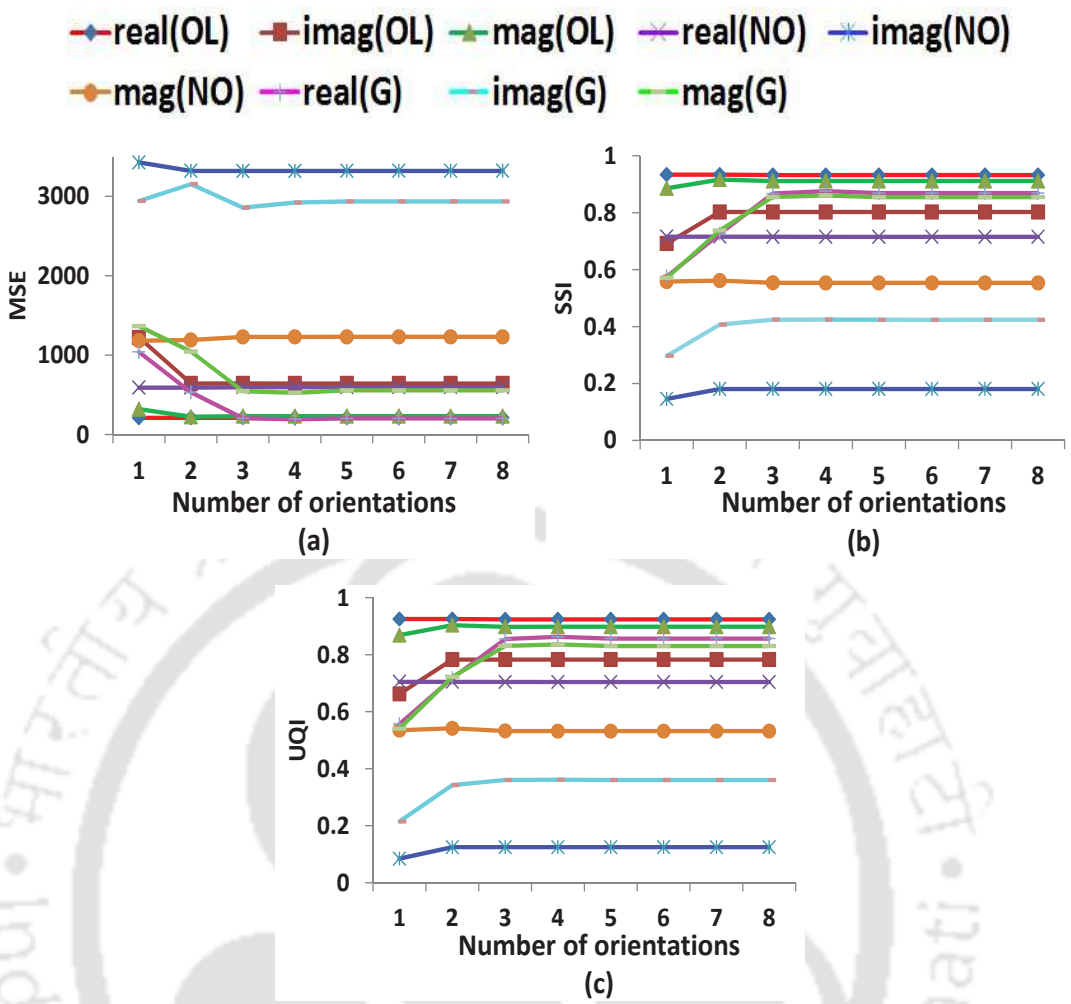
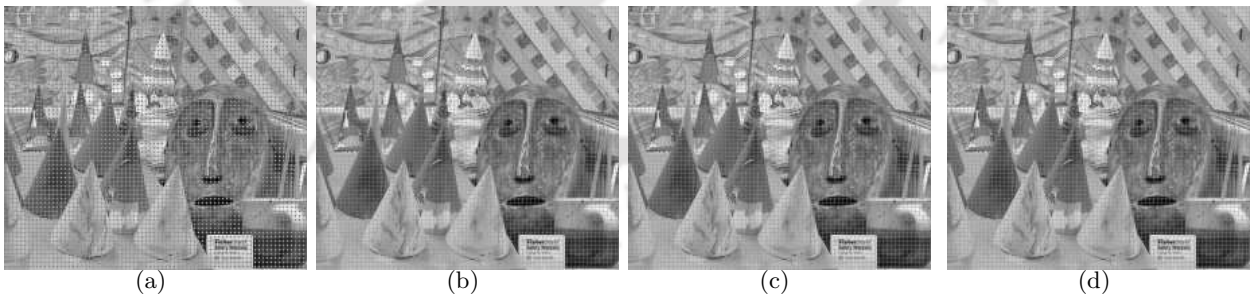


Figure 5.5: Comparison by (a) MSE; (b) SSI; (c) UQI of global and local features (both overlapping and non-overlapping regions) for different numbers of orientations.



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

Figure 5.6: Image reconstructed using global feature for (a) Two; (b) Four; (c) Six; (d) Eight number of orientations.

5.5.3 Different number of scalings

The performance of Gabor features are also evaluated for different scales as shown in Table 5.4 and Table 5.5. When the number of scale is 1, the performance of LGWF (overlapping regions) is comparable to the global features. On the other hand, LGWF (overlapping regions) performs better as compared to the global features when the number of scales is 2. Few number of scales are able to represent the pixel variations in the image patch used for local feature extraction. Hence, the error value remains almost same for different scales. However for global features, high frequency information is lost with the increase of number of scales. So, an image reconstructed using these low resolution images can produce an approximate original image. Hence, the more number of scales increases the error. This finding is illustrated in Figure 5.7.

Table 5.4: Comparison of the global and local features (both overlapping and non-overlapping regions) for number of scales = 1

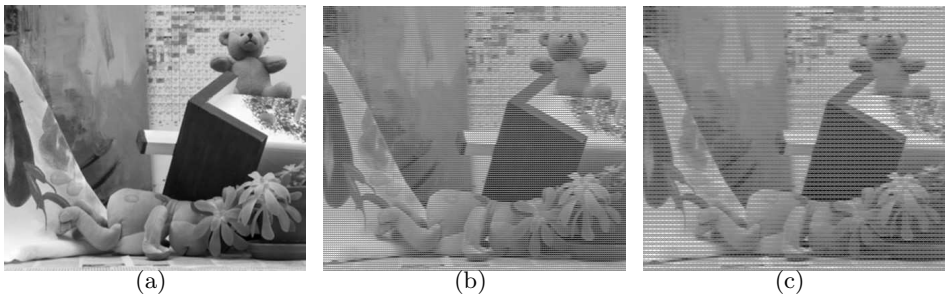
Metrics	Overlapping			Non-overlapping			Global		
	real	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
MSE	211	642	223	591	3320	1191	127	3130	398
CC	0.968	0.927	0.968	0.886	0.106	0.886	0.987	0.322	0.987
QI	0.925	0.782	0.903	0.705	0.125	0.542	0.951	0.314	0.896
SSI	0.933	0.803	0.916	0.716	0.181	0.562	0.956	0.380	0.909

Table 5.5: Comparison of the global and local features (both overlapping and non-overlapping regions) for number of scales = 2

Metrics	Overlapping			Non-overlapping			Global		
	real	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
MSE	211	642	223	591	3320	1191	533	3153	1046
CC	0.968	0.927	0.968	0.886	0.106	0.886	0.902	0.365	0.902
QI	0.925	0.782	0.903	0.705	0.125	0.542	0.718	0.342	0.722
SSI	0.933	0.803	0.916	0.716	0.181	0.562	0.726	0.407	0.738

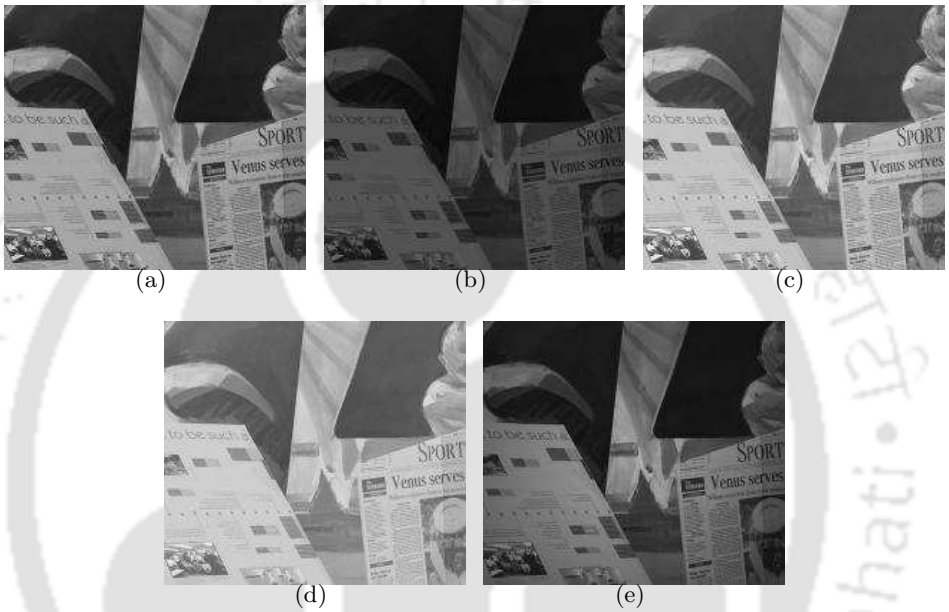
5.5.4 Synthetic illumination changes

The performance of Gabor features are investigated for illumination changes. To incorporate synthetic illumination changes, left image is kept unaltered, while the intensity values of the pixels of the right image are altered. Illumination change may be global or local. Global illumination changes can further be classified as linear or non-linear. To evaluate the performance of these features under varying illumination conditions, the intensity values of the pixels of the right image are varied



(a) (b) (c)

Figure 5.7: Image reconstructed using global feature for (a) One; (b) Two; (c) Three number of scales.



(a) (b) (c)

(d) (e)

Figure 5.8: (a) Input image; synthetically illuminated Venus image by (b) Multiplicative factor; (c) Additive factor; (d) Gamma factor; (e) Vignetting effect.

synthetically using the formula given by:

$$I_o = 255\left(\frac{m_f I_i + a_f}{255}\right)^{\gamma_f} \tag{5.14}$$

where I_i is the input image and I_o is the synthetically varied image of I_i. m_f, a_f, and γ_f are the multiplicative (gain change), additive (bias change), and gamma factors respectively with $m_f, \gamma_f > 0$ [174]. In the above formula, multiplicative and additive factors represent a linear global change, whereas gamma factor denotes a non-linear global change. Additionally, local change is represented by using a vignetting function. The synthetically illuminated images are normalized in the range of [0-255]. Figure 5.8 shows different synthetic illumination changes applied to the Venus image. Figure

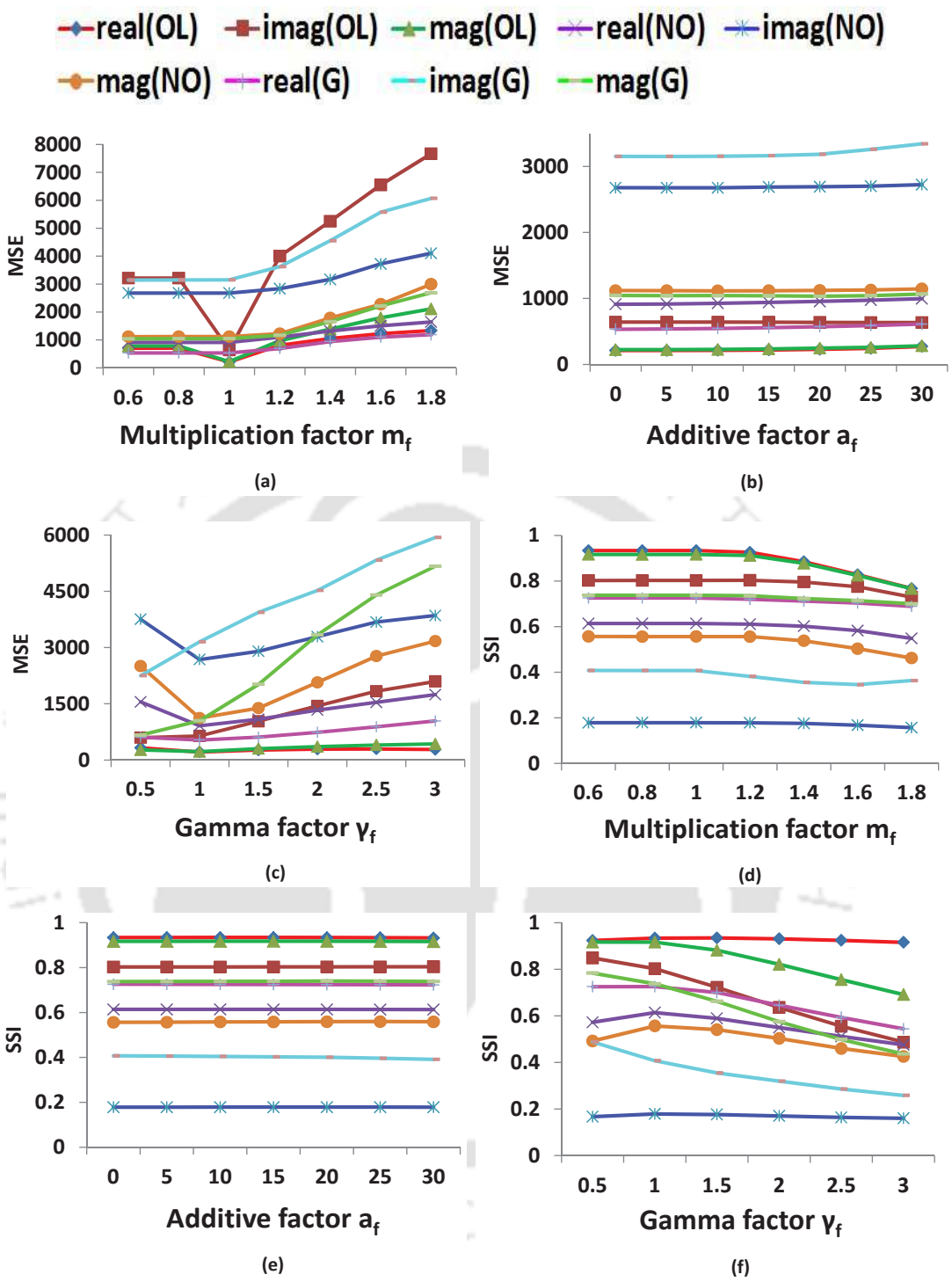


Figure 5.9: Comparison of features (global, local - both overlapping and non-overlapping regions) by MSE and SSI for synthetic illumination changes obtained by multiplicative, additive, and gamma factors. (a)-(c) MSE; (d)-(f) SSI.

5.9, Figure 5.10, Table 5.6, Table 5.7, and Table 5.8 show the performance of Gabor features for different values of multiplicative, additive, and gamma factors. Figure 5.12 and Table 5.9 show the

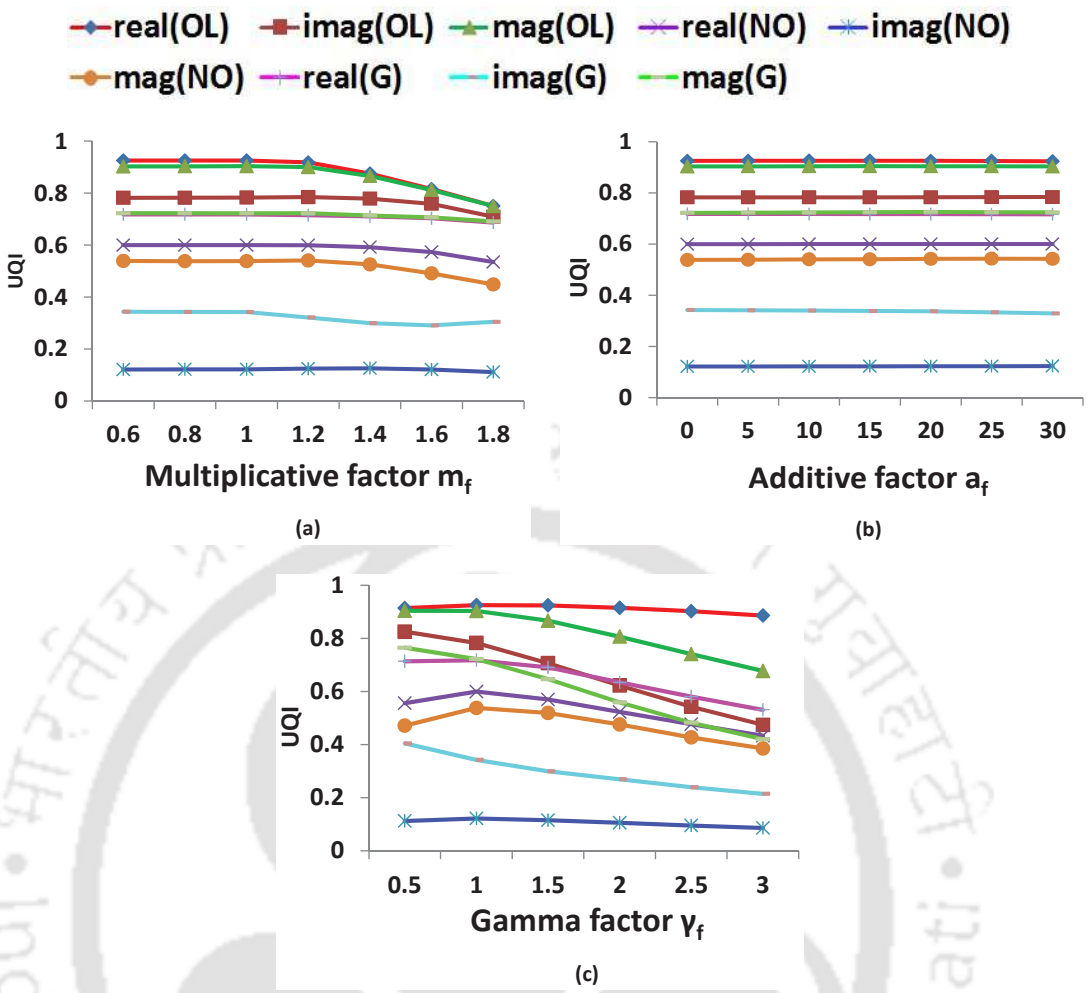


Figure 5.10: Comparison of features (global, local - both overlapping and non-overlapping regions) by UQI for synthetic illumination changes obtained by multiplicative, additive, and gamma factors. (a) Multiplicative factor; (b) Additive factor; (c) Gamma factor.

results for local illumination variations. The abovementioned figures and tables show that these Gabor features are affected by illumination variations. This effect is because of the inherent properties of convolution operation. To establish our claim, an image is synthetically illuminated for $m_f = 2$. Figure 5.11 shows the comparison of the image reconstructed by using the features extracted from the synthetically varied image with the image reconstructed by using the features extracted from the original image. Figure 5.11(a) and Figure 5.11(d) show the original and reconstructed images respectively, whereas Figure 5.11(b) and Figure 5.11(e) show the synthetically illuminated image and its corresponding reconstructed image respectively. Figure 5.11(c) shows the difference image between Figure 5.11(a) and Figure 5.11(b), while Figure 5.11(f) shows the difference image between Figure 5.11(d) and Figure 5.11(e). These results show that the changes in the intensity values of the original

Table 5.6: Comparison of correlation coefficients of all the features (global, local - both overlapping and non-overlapping regions) for synthetic illumination change by multiplicative factor

Multiplicative	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
0.6	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
0.8	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
1	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
1.2	0.968	0.926	0.968	0.845	0.198	0.845	0.902	0.366	0.902
1.4	0.963	0.922	0.963	0.841	0.196	0.841	0.902	0.365	0.902
1.6	0.951	0.910	0.951	0.830	0.192	0.830	0.902	0.362	0.902
1.8	0.929	0.889	0.929	0.811	0.186	0.811	0.901	0.362	0.901

image alter the extracted Gabor features, which finally affects the intensity values of the corresponding reconstructed image. Similar explanations can be given for the cases when an image is synthetically illuminated by considering the vignetting effect, additive, and gamma factors.

Table 5.7: Comparison of correlation coefficients of all the features (global, local - both overlapping and non-overlapping regions) for synthetic illumination change by additive factor

Additive	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
0	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
5	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
10	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
15	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
20	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
25	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
30	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902

Table 5.8: Comparison of correlation coefficients of all the features (global, local - both overlapping and non-overlapping regions) for synthetic illumination change by gamma factor

Gamma	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
0.5	0.969	0.927	0.969	0.836	0.194	0.836	0.902	0.367	0.902
1	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902
1.5	0.967	0.925	0.967	0.839	0.198	0.839	0.901	0.369	0.901
2	0.965	0.921	0.965	0.824	0.195	0.824	0.900	0.376	0.900
2.5	0.963	0.918	0.963	0.804	0.190	0.804	0.898	0.385	0.898
3	0.961	0.914	0.961	0.782	0.185	0.782	0.897	0.395	0.897

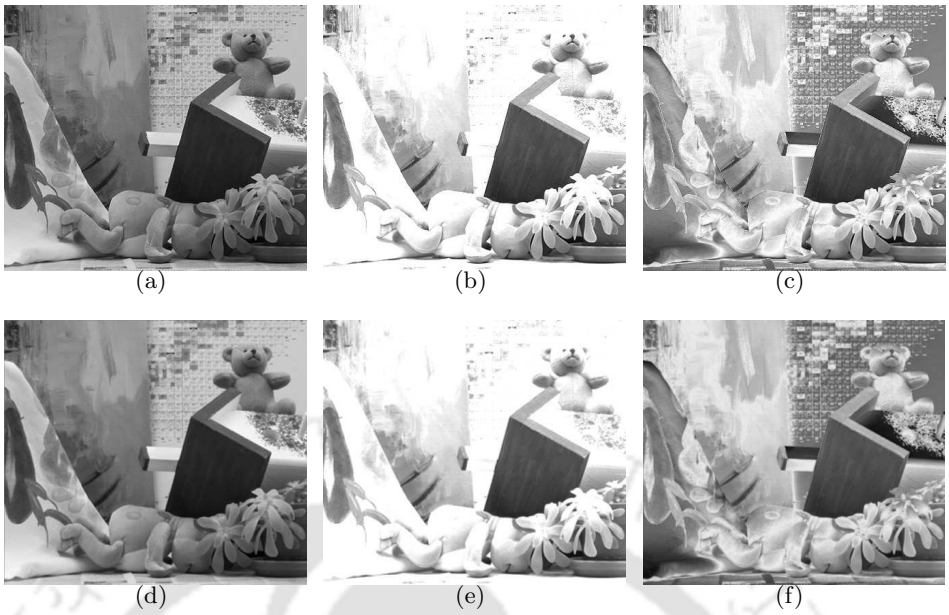


Figure 5.11: Effect of synthetic illumination variations by a multiplicative factor. (a) Original image; (b) Original image synthetically illuminated; (c) Difference of (a) and (b); (d) Image reconstructed using (a); (e)Image reconstructed using (b); (f) Difference of (d) and (e).

Table 5.9: Comparison of correlation coefficients of all the features (global, local - both overlapping and non-overlapping regions) for synthetic radiometric change by vignetting effect

Multiplicative	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
0.4	0.875	0.845	0.875	0.759	0.180	0.759	0.812	0.354	0.812
0.6	0.934	0.896	0.934	0.814	0.191	0.814	0.869	0.361	0.869
0.8	0.962	0.920	0.962	0.839	0.197	0.839	0.895	0.364	0.895
1	0.968	0.927	0.968	0.846	0.198	0.846	0.902	0.365	0.902

Table 5.10: Comparison of correlation coefficients for real radiometric change for different camera exposures

Multiplicative	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
0/0	0.955	0.895	0.955	0.837	0.224	0.837	0.896	0.407	0.896
0/1	0.947	0.889	0.947	0.829	0.223	0.829	0.888	0.409	0.888
0/2	0.929	0.875	0.929	0.813	0.222	0.813	0.871	0.416	0.871
1/0	0.951	0.887	0.951	0.833	0.222	0.833	0.892	0.395	0.892
1/1	0.953	0.892	0.953	0.834	0.227	0.834	0.894	0.416	0.894
1/2	0.943	0.886	0.943	0.824	0.228	0.824	0.884	0.431	0.884
2/0	0.939	0.870	0.939	0.822	0.217	0.822	0.881	0.380	0.881
2/1	0.949	0.882	0.949	0.830	0.225	0.830	0.891	0.407	0.891
2/2	0.949	0.886	0.949	0.829	0.230	0.829	0.891	0.432	0.891

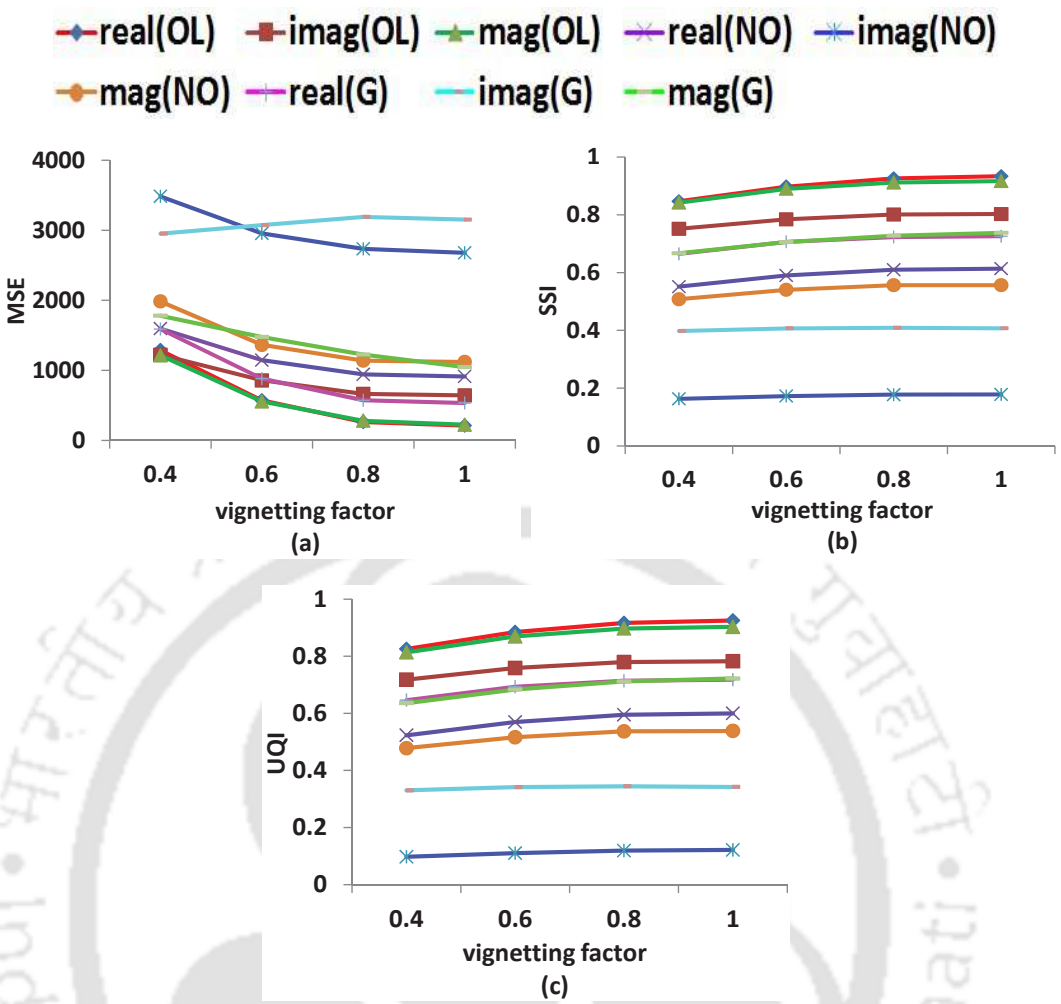


Figure 5.12: Comparison by (a) MSE; (b)SSI; (c) UQI of all the features (global, local - both overlapping and non-overlapping regions) for synthetic illumination change by vignetting effect.

5.5.5 Real radiometric changes

Many multimedia applications need features which are robust to radiometric changes[3]. One such application is finding a stereo correspondence. The images used so far for the performance evaluation of Gabor features are captured under the same lighting condition and the same camera settings. Hence, new dataset which was captured under different lighting conditions and different camera exposures are also used [12,13]. This dataset is shown in Figure 5.13. The change of camera exposure is a global transformation which is similar to the global brightness change. This effect is similar to gain change or multiplicative factor in synthetic illumination variations [175]. Different light sources produce many local radiometric variations in the captured images. The performance of Gabor features for exposure

[3]This work has been published in *ICCSP 2013* (Refer item [9] in Page 136 for details)

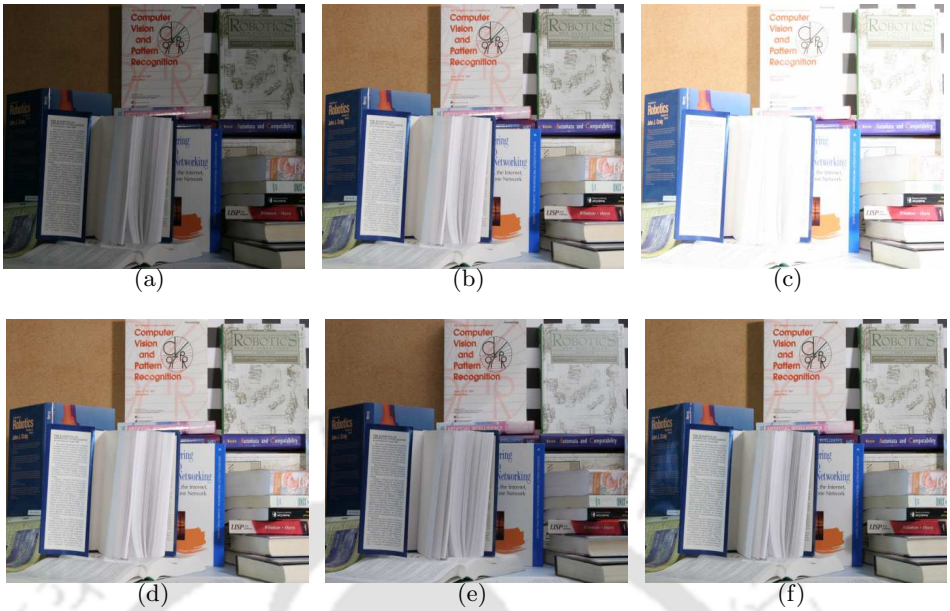


Figure 5.13: "Books" image for three different exposures and lighting conditions: (a) Exposure 1; (b) Exposure 2; (c) Exposure 3; (d) Lighting 1; (e) Lighting 2; (f) Lighting 3.

and lighting changes can be seen in Figure 5.14 and Figure 5.15. Quantitative evaluations of these effects are shown in Table 5.10 and Table 5.11. So, it is observed that Gabor features obtained for real radiometric variations show almost similar characteristics as that of the Gabor features for synthetically illuminated variations. An input image for "Exposure a" which is used to extract the features, while the reference image which is used to evaluate the reconstructed image is taken for "Exposure b". This is denoted as "a/b" in Figure 5.14 and Table 5.10. The reconstructed image corresponds to the image taken for "Exposure a". Similarly, an input image for "Lighting a" which is used to extract the features, while the reference image which is used to evaluate the reconstructed image is taken for "Lighting b". This is denoted as "a/b" in Figure 5.15 and Table 5.11. The reconstructed image corresponds to the image taken for "Lighting a". From all the above experimental results, it is concluded that the real part of the feature extracted from the overlapping regions represents the original image more efficiently than the imaginary part, and the real part of the feature gives almost similar performance as that of the magnitude information. Real Gabor filter extracts texture information, while imaginary Gabor filter extracts the edge information [176]. Hence, real Gabor filter is sufficient to represent an image. Additionally, it is observed that the local features extracted from the overlapping regions can represent an image more efficiently as compared to the features extracted from both global and non-overlapping regions. In the local feature extraction method from overlapping regions, the pixel for which the feature is extracted is given more weight as compared to

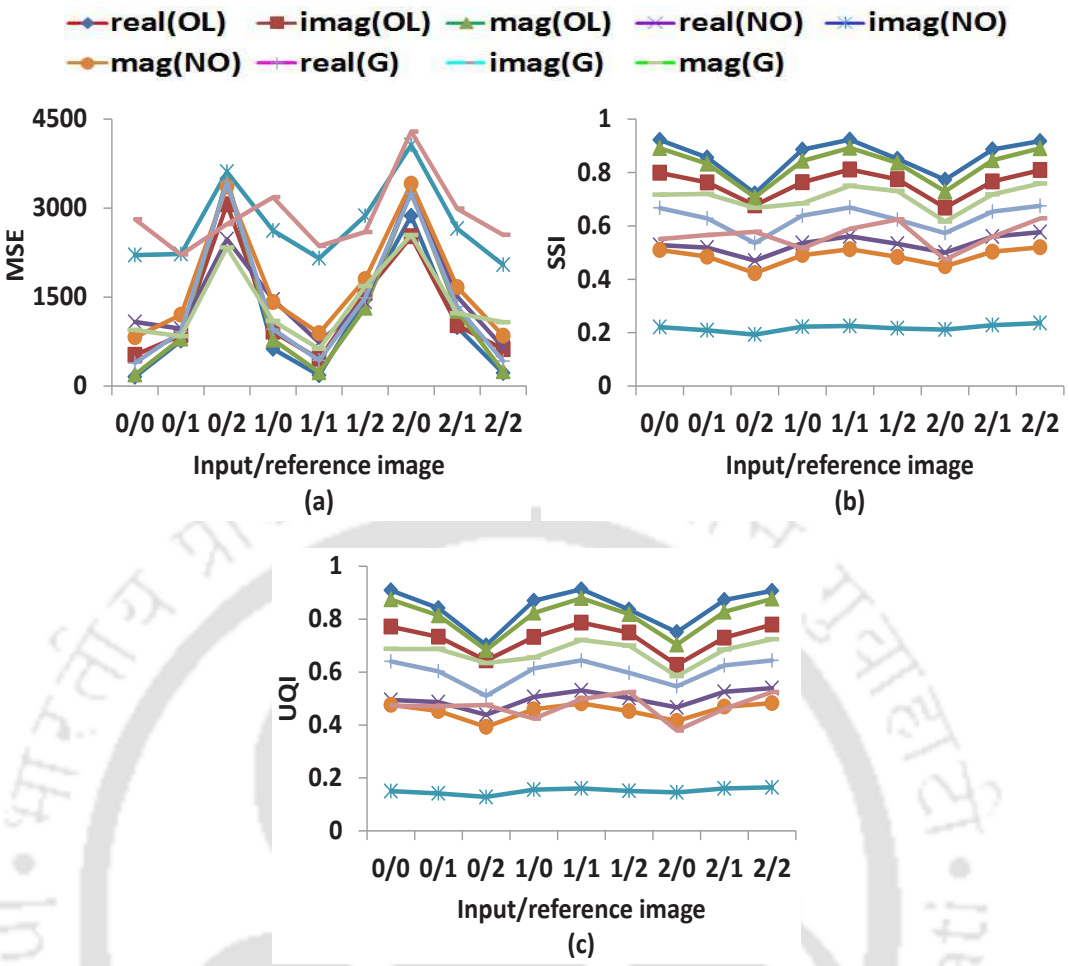


Figure 5.14: Comparison by (a) MSE; (b) SSI; (c) UQI of all the features (global and local - both overlapping and non-overlapping regions) for real radiometric change for different camera exposures.

its neighbouring pixels by the Gabor function during the convolution operation. That is why the local features extracted from the overlapping regions perform better than the other two Gabor features.

Table 5.11: Comparison of correlation coefficients for real radiometric change for different light sources

Multiplicative	CC(overlapping)			CC(non-overlapping)			CC(global)		
factor	*real*	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
1/1	0.979	0.943	0.979	0.855	0.169	0.855	0.906	0.328	0.906
1/2	0.902	0.862	0.902	0.790	0.157	0.790	0.838	0.282	0.838
1/3	0.947	0.910	0.947	0.827	0.163	0.827	0.840	0.287	0.840
2/1	0.907	0.884	0.907	0.795	0.158	0.795	0.835	0.299	0.835
2/2	0.976	0.945	0.976	0.858	0.177	0.858	0.906	0.312	0.906
2/3	0.928	0.903	0.928	0.811	0.164	0.811	0.877	0.307	0.877
3/1	0.950	0.918	0.950	0.832	0.164	0.832	0.878	0.315	0.878
3/2	0.924	0.886	0.924	0.811	0.163	0.811	0.859	0.291	0.859
3/3	0.978	0.943	0.978	0.855	0.171	0.855	0.844	0.290	0.844

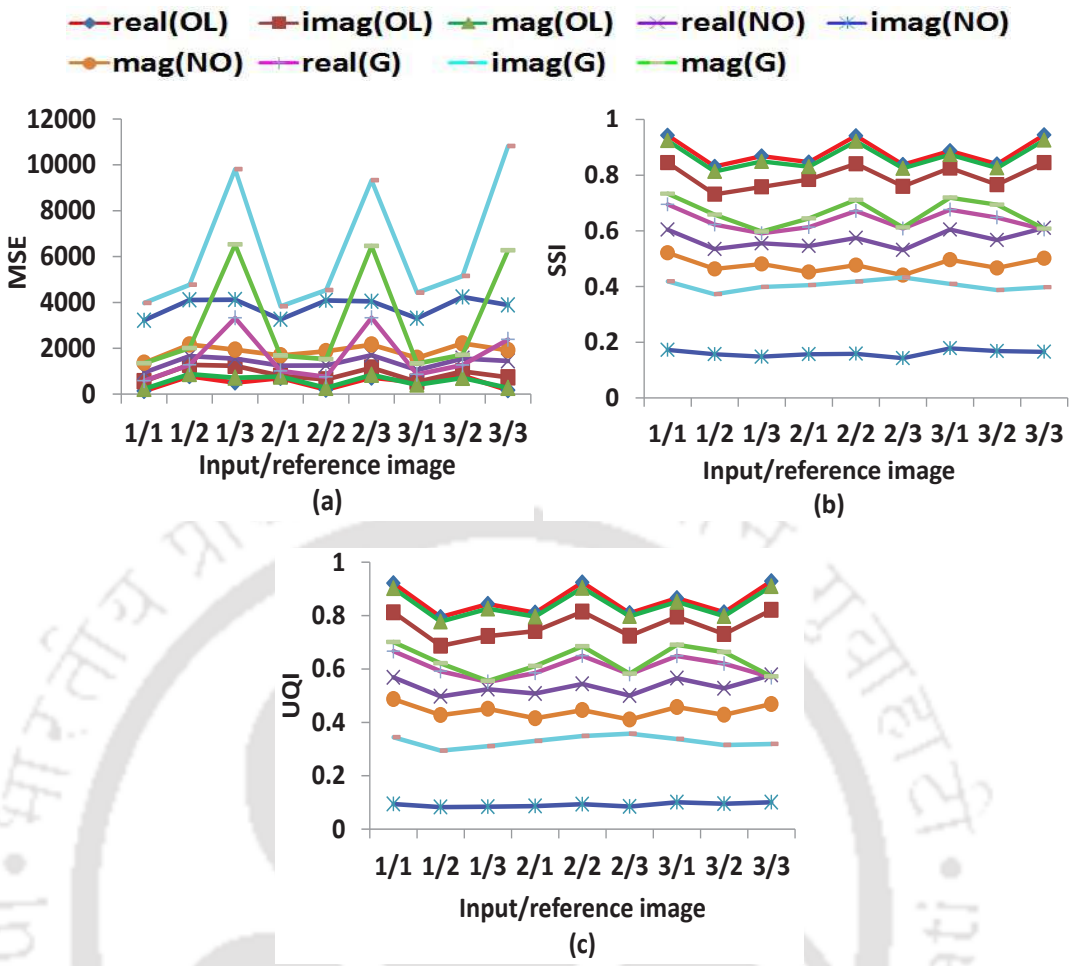


Figure 5.15: Comparison by (a) MSE; (b) SSI; (c) UQI of all the features (global and local - both overlapping and non-overlapping regions) for real radiometric change for different light sources.

5.5.6 Performance evaluation of Gabor features for stereo correspondence

Stereo correspondence is considered as one application to investigate the performance of three Gabor-based extracted features. As explained earlier, the additional information obtained in the form of disparity map from stereo correspondence may be used in many multimedia applications such as face and facial expression. The efficacy of these features are evaluated from the computed disparity map [15]. For this, mean-square error is used for evaluating the estimated disparity map[4]. Table 5.12 gives a quantitative comparison of all the three Gabor features for stereo correspondence. In Table 5.12, *real*, *imag*, and *mag* represents real coefficients, imaginary coefficients and magnitude information used as feature for computing the disparity map.

[4]This work has been published in *DIPDMWC 2015* (Refer item [10] in Page 136 for details)

Table 5.12: Gabor features applied for stereo correspondence

LGWF (overlapping)			LGWF (non-overlapping)			GGWF		
real	*imag*	*mag*	*real*	*imag*	*mag*	*real*	*imag*	*mag*
24.97	33.33	28.4	63.45	69.89	65.56	58.82	77.25	49.51

5.6 Summary

Feature plays an important role in several image processing and computer vision applications. Hence, feature extraction is an active research topic in computer vision. Gabor wavelet-based extracted features are used for a number of computer vision and multimedia applications namely face recognition, facial expression recognition, iris recognition, texture classification and segmentation. The extracted feature should represent an image more accurately. Existing algorithms use magnitude information of Gabor wavelet for feature extraction, which requires both real and imaginary coefficients.

In this chapter, the performance of three Gabor wavelet feature extraction methods namely global and local (overlapping and non-overlapping regions) features extracted using Gabor wavelet are compared. The performance of these features are also evaluated for different window sizes, different number of orientations, different scalings, and also for radiometric changes. The metrics used for evaluation are MSE, CC, QI, and SSI. Experimental results show that among the three features, LGWF (overlapping) performs better compared to the other two counterparts. In addition to this, it is also showed that the real coefficients represent the original image more accurately compared to imaginary coefficients, and the magnitude information. So use of only the real coefficients reduces the computational complexity and memory requirement, which are the basic requirements for different multimedia applications. Real coefficients of LGWF (overlapping regions) are more robust to the radiometric changes compared to the remaining features mentioned in this chapter.

6

Conclusions

In this final chapter, we present a summary of the major contributions of the work reported in the previous chapters of this thesis. Moreover, we point out a few areas that may be explored for further progress of the present research.

6.1 Summary

Disparity map estimation of a stereo image pair is an important research topic in the field of computer vision. This thesis investigated the issues involved in the problem of accurate disparity map estimation, and proposed methods to address some of these issues. The proposed disparity map estimation method employs Gabor wavelet in spatial domain to extract local features. The real part of this feature is only used in the proposed stereo matching process. Gabor feature conveys almost similar information as perceived by a human visual system. The accuracy of Gabor wavelet features in representing an image is experimentally studied for different Gabor filter parameters. Additionally, the behaviour of Gabor features are analyzed for radiometric variations. Kuwahara filter shows edge preserving property, and this filter is applied in our method to smoothen out the matching cost. Furthermore, the characteristics of this filter is analyzed for discontinuous image regions. For estimating a disparity map in presence of occlusions, an asymmetric occlusion detection method is proposed, which employ only one disparity map. The behaviour of reference and target pixels are analyzed, and the characteristics is observed that the target matching pixels with respect to reference pixels almost follow a linear pattern. Hence, it is approximated by a linear regression model, and this model is used to detect the occluded pixels in our method. Subsequently, occlusion filling is implemented by assigning a disparity value of a neighbouring non-occluded pixel to an occluded pixel. This non-occluded pixel is selected on the basis of colour similarity of an occluded pixel with the neighbouring non-occluded pixels. This similarity score is calculated by using the support weights of both left and right images.

The main points of the work reported in this dissertation may be summed up as follows:

- In the **Introduction**, we elaborated the concepts of image formation in a single camera-based setup and stereo vision setup. The basic principle of stereo correspondence, and the major issues associated with the estimation of an accurate disparity map are addressed. Photometric variations, disparity discontinuities, and occlusions are identified as the major obstacles in obtaining an accurate disparity map.

- In **Chapter 2**, we elaborated the problems associated with stereo correspondence estimation. A state-of-the-art review of the existing literature on disparity map estimation methods was presented. The review also included methods of finding disparity map estimates in presence of occlusions. Disparity map estimation methods can be broadly classified as global and local methods. The performance of the global methods depends on the formulation of the energy function. These methods are performed in an iterative manner, and are hence time consuming.

On the other hand, selection of an appropriate window size is a major challenge in the case of local methods. Larger window blurs the object boundaries, while smaller window produces unreliable results in the low textured regions. Again, the phase difference between the Gabor convolved stereo images are employed for estimating disparity map in Gabor phase-based methods. In this context, one important research challenge is the disparity map estimation in presence of occlusions. Occlusion occurs due to the presence of a portion of the scene in one image, while absent in the other image. Hence, finding of the matching pixels for all the pixels in the occluded regions is a difficult task. The most widely used occlusion detected methods detect the occluded pixels based on the uniqueness constraint. In these methods, disparity maps of both the stereo images are employed. This increases computational burden. Global algorithms perform this task by incorporating an occlusion term in the energy function along with data and smoothness terms. These methods compute disparity map in an iterative manner. After detecting the occluded pixels, the next step is the assignment of an appropriate disparity value to a detected pixel.

- The proposed disparity map estimation method is extensively discussed in **Chapter 3**. Our method employs Gabor wavelet to extract local features for finding corresponding matching pixels. The reason behind using Gabor wavelet for feature extraction is that the image analysis by Gabor function resembles the perception of the human visual system. Next step in our proposed method is cost aggregation, which is performed by employing Kuwahara filter. Kuwahara filter preserves disparity discontinuities in the estimated disparity map. Experimental results show that the performance of our proposed method is better than Gabor phase-based stereo matching methods. This is due to the fact that Gabor phase-based disparity map estimation methods suffer from phase singularities. Additional experimental results are shown for a dataset having large disparity values. It is observed that the proposed method can tackle large disparity range, and produces substantially good results for the non-textured image regions and the regions having complex geometry.

- In **Chapter 4**, we proposed a novel occlusion detection method which makes use of only one disparity map for detecting the occluded pixels in a stereo image pair. Our method can detect the occluded pixels and/or the pixels having wrong disparity values accurately only by using one disparity map. In our method, a mapping function is employed, and this function is modelled by a linear regression function. The deviation of the mapping function from the modelled linear function gives an indication of the presence of occluded pixels. The proposed mapping function

is obtained from the initial disparity map. Similarly, the upper and lower thresholds required for taking a decision are calculated on the basis of gradient information of the initial disparity map. The thresholds are selected in such a way that for occluded pixels the mapping function lies outside the region bounded by these two thresholds. Incorporation of these steps in our proposed method helps in detecting the occluded pixels more accurately. The next step of our method is filling of the detected occluded pixels. In general, occlusion filling is performed by assigning of an appropriate disparity value of a neighbouring non-occluded pixel to a pixel targeted for filling. Neighbouring non-occluded pixels are selected based on the colour information obtained from one of the images of the stereo image pairs. This approach fails when the pixel to be filled and the neighbouring occluded pixel have similar colour characteristics. This drawback is overcome in our method by employing both the images of stereo image pairs. For this, weights are computed for both the images separately, and subsequently they are combined to get the final weight. The disparity value of the neighbouring pixel corresponding to the highest combined weight is finally assigned to the pixel to be filled. The accuracy of our proposed occlusion filling method is significantly better as both the left and right images are considered in this process. But, the computational complexity of this approach is high as compared to single image based occlusion filling. Experimental results show that the proposed occlusion detection method gives almost similar performance as that of the standard LRC method even with the help of one disparity map. The performance of our proposed occlusion filling method is also better than some of the well known occlusion filling methods such as neighbor disparity assignment, diffusion in intensity space, weighted least square, and segmentation-based least squares methods. Additionally, the disparity maps generated by the existing stereo matching algorithms can also be directly utilized for occlusion detection, and subsequent filling by our proposed methods.

- In **Chapter 5**, we experimentally observed the performance of our employed Gabor filter for different essential parameters. Most of the well-established pattern recognition algorithms use magnitude information of Gabor wavelet for feature extraction. For this, information of both real and imaginary coefficients are needed. In this chapter, it is validated that the real coefficients of Gabor filter alone are sufficient to represent an image for stereo matching. To compute the real coefficients, only the real part of the Gabor filter needs to be stored in the memory. On the other hand, both the real and imaginary parts of Gabor filter need to be stored separately to compute the magnitude information. Additionally, the outputs obtained by the convolution operation with both real and imaginary parts of Gabor filter need to be stored separately to compute the

magnitude information. But for computing the real coefficients, the outputs obtained by the convolution operation with the real part of the Gabor filter only need to be stored. That is why, memory requirement is reduced by half when only the real coefficients are used. In earlier literatures, it was also mentioned that an optimal performance of two-dimensional Gabor filter can be obtained by using real part of the filter. But, there is no concrete experimental validations in this regard. But in our analysis, an experimental evaluation using two-dimensional Gabor wavelet suggests that the real coefficients of Gabor function is sufficient to represent an image for the applications like stereo matching. In this chapter, the performance of local Gabor wavelet features for both overlapping and non-overlapping regions are evaluated. These comparisons are done by considering different window sizes, different number of orientations, and different scales. Also, performance of these features are analyzed for radiometric changes. The metrics used for performance comparisons are MSE, CC, QI, and SSI. Experimental results show local Gabor wavelet features (overlapping regions) performs better as compared to the local Gabor wavelet features (non-overlapping regions) and global Gabor wavelet features. Additionally, it is shown that the real coefficients of Gabor filter represent an image more accurately as compared to the imaginary coefficients.

6.2 Possible Extensions

Although the presented work terminated in a working disparity map estimation algorithm for a stereo image pair, many issues arose during the work. Some of these issues require further research. The future research objectives are enumerated below.

- To combine Gabor phase with Gabor real coefficients, and subsequently use this combination as a feature for stereo matching. The robustness of this feature can be further analyzed for illumination variations.

- To employ piece-wise linear model to improve the performance of proposed occlusion detection method. Additionally, incorporation of a non-linear regression model may improve the performance of our occlusion detection algorithm.

- To develop a disparity map estimation method which can also consider foreshortening effect by employing Gabor features at different scales.

- To extend the proposed disparity map estimation for multiple camera-based setup.

List of Publications

Journal Publications

[1] Malathi. T and M.K. Bhuyan, "Estimation of disparity map of stereo image pairs using spatial domain local Gabor wavelet", IET computer vision, vol. 9, no. 4, pp. 595–602, 2015.

[2] Malathi. T and M.K. Bhuyan, "Asymmetric occlusion detection and filling scheme for the estimation of stereo disparity map", IET computer vision, DOI 10.1049/iet-cvi.2015.0214, Online ISSN 1751-9640, Available online: April 2016.

[3] Malathi. T and M.K. Bhuyan, "Performance analysis of Gabor wavelet for extracting most informative and efficient features", Multimedia tools and applications, Springer, DOI 10.1007/s11042-016-3414-2, Available online: April 2016.

Book Chapters

[4] M.K. Bhuyan and Malathi. T, "Review of the application of matrix information theory in video surveillance", Matrix information geometry, Springer-Verlag, pp. 293–321, 2013.

[5] N. Mishra, M. K. Bhuyan, T. Malathi, Y. Iwahori and R. J. Woodham, "Pixel-wise background segmentation with moving camera", Pattern recognition and machine intelligence, Lecture notes in computer science, Springer Berlin Heidelberg, vol. 8251, pp. 423–429, 2013.

[6] Malathi. T and M.K. Bhuyan, "Local Gabor wavelet-based feature extraction and evaluation", Smart innovations, systems and technologies, Springer-Verlag, vol. 43, pp. 181–189, 2016.

Conference Publications

[7] M.K. Bhuyan and Malathi. T, "Performance evaluation of local Gabor wavelet-based disparity map computation", in Proc. International conference on electrical, electronics, computer engineering and their applications (EECEA'15), pp. 79–92, 2015.

[8] Malathi. T and M.K. Bhuyan, "Foreground object detection under camouflage using multiple camera-based codebooks", in Proc. Annual IEEE Indian conference (INDICON'13), pp. 1–6, 2013.

[9] Malathi. T and M.K. Bhuyan, "Multiple camera-based codebooks for object detection under sudden illumination change", in Proc. International conference on communication and signal processing (ICCSP'13), pp. 310–314, 2013.

[10] Malathi. T and M.K. Bhuyan, "Disparity map estimation using local Gabor wavelet under radiometric changes", International conference on digital information processing, data mining, and wireless communications (DIPDMWC'15), pp. 148–156, 2015.

A

Appendix

A.1 Detailed explanation to obtain mother wavelet of Gabor filter

Gabor function is a Gaussian modulated complex sinusoidal. The most general 2D complex Gabor function is given by [121]:

$$\psi\left(x, y, \xi_0, \nu_0, x_0, y_0, \rho, \theta, \sigma, \beta\right) = \frac{1}{\sqrt{\pi\sigma\beta}} \exp\left(-\left(\frac{\left((x-x_0)\cos\theta + (y-y_0)\sin\theta\right)}{2\sigma^2} + \frac{\left(-\left((x-x_0)\sin\theta + (y-y_0)\cos\theta\right)\right)}{2\beta^2}\right)\right) \cdot \exp\left(i\left(\xi_0\left(x-x_0\right) + \nu_0\left(y-y_0\right) + \rho\right)\right) \quad \text{(A.1)}$$

In the above equation, the first term in the right hand side (RHS) is the elliptical Gaussian function and the second term is the complex sinuoidal function. The filter is centered at $(x = x_0, y = y_0)$ in the spatial domain, and at $(\xi = \xi_0, \nu = \nu_0)$ in the spatial frequency domain. σ and β are the standard deviations of an elliptical Gaussian along the x and y axes. θ is the orientation of the filter, rotated counter-clockwise around the origin. ρ is the absolute phase of an individual filter. There are, therefore, eight degrees of freedom in the general Gabor function: $\xi_0, \nu_0, \theta, \rho, \sigma, \beta, x_0, y_0$.

The above equation is simplified by setting the spatial location of the filters center $(x_0 = 0, y_0 = 0)$ and the absolute phase ρ of the filter to 0. The above Equation (A.1) can then be expressed as:

$$\psi\left(x, y, \xi_0, \nu_0, \theta, \sigma, \beta\right) = \frac{1}{\sqrt{\pi\sigma\beta}} \exp\left(-\left(\frac{\left(x\cos\theta + y\sin\theta\right)^2}{2\sigma^2} + \frac{\left(-x\sin\theta + y\cos\theta\right)^2}{2\beta^2}\right)\right) \cdot \exp\left(i\left(\xi_0 x + \nu_0 y\right)\right) \quad \text{(A.2)}$$

The Fourier Transform of the simplified complex-valued Gabor function (Equation (A.2)) is shown below:

$$\hat{\psi}\left(\xi, \nu, \xi_0, \nu_0, \theta, \sigma, \beta\right) = 2\sqrt{\pi\sigma\beta} \exp\left(-\frac{1}{2}\left[\left(\left(\xi-\xi_0\right)\cos\theta + \left(\nu-\nu_0\right)\sin\theta\right)^2\sigma^2 + \left(\left(\xi-\xi_0\right)\sin\theta + \left(\nu-\nu_0\right)\cos\theta\right)^2\beta^2\right]\right) \quad \text{(A.3)}$$

where ξ and ν are the spatial frequencies in radians per unit length along x and y. The number of degrees of freedom can be further reduce by implying constraints on σ, β and θ in terms of ξ_0 and ν_0 according to the physiological findings. The constraints are as follows:

(i) *The aspect ratio $\frac{\beta}{\sigma}$ of the elliptical Gaussian envelope is 2:1.* Hence, Equation (A.2) can be

rewritten as:

$$\psi\left(x, y, \xi_0, \nu_0, \theta, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{8\sigma^2}\left[4(x\cos\theta + y\sin\theta)^2 + (-x\sin\theta + y\cos\theta)^2\right]\right) \cdot \exp\left(i\left(\xi_0 x + \nu_0 y\right)\right) \quad \text{(A.4)}$$

(ii) *The plane wave with frequency* (ξ_0, ν_0) *tends to have its propagating direction along the short axis of the elliptical Gaussian.* The elliptical Gaussian rotates analogous to the plane wave rotation. Hence, the center frequency (ξ_0, ν_0) of the filter is related to the rotation angle θ of the modulating Gaussian which is given by: $\xi_0 = \omega_0 \cos\theta$ and $\nu_0 = \omega_0 \sin\theta$ where $\omega_0 = \sqrt{\xi_0^2 + \nu_0^2}$. Inposing this constraint into Equation (A.4) yields:

$$\psi\left(x, y, \xi_0, \nu_0, \theta, \sigma\right) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{8\sigma^2}\left[4(x\cos\theta + y\sin\theta)^2 + (-x\sin\theta + y\cos\theta)^2\right]\right) \cdot \exp\left(i\left(x\omega_0\cos\theta + y\omega_0\sin\theta\right)\right) \quad \text{(A.5)}$$

(iii) *The half-amplitude bandwidth of the frequency response is about 1 to 1.5 octaves along the optimal orientation.* The relationship between σ and ω_0 can be expressed as:

$$\sigma = \frac{\kappa}{\omega_0} \quad \text{(A.6)}$$

where, $\kappa = \sqrt{2\ln 2}\left(\frac{2^\phi + 1}{2^\phi - 1}\right)$. Here, ϕ is the bandwidth in octaves. Incorporating this constrain in Equation (A.5) gives:

$$\psi\left(x, y, \omega_0, \theta\right) = \frac{\omega_0}{\sqrt{2\pi}\kappa} \exp\left(-\frac{\omega_0^2}{8\kappa^2}\left[4(x\cos\theta + y\sin\theta)^2 + (-x\sin\theta + y\cos\theta)^2\right]\right) \cdot \exp\left(i\left(x\omega_0\cos\theta + y\omega_0\sin\theta\right)\right) \quad \text{(A.7)}$$

where, $\theta = \arctan\frac{\nu_0}{\xi_0}$ and κ is fixed for Gabor wavelets of a particular bandwidth. The whole family can be translated to any spatial position (x_0, y_0). In order to make the Gabor filters into admissible wavelets, we need to introduce the following constraint.

(iv) *Admissible wavelets are functions having zero mean.* The sine component of the complex-valued Gabor filter has zero mean, but its cosine component has nonzero mean (DC response). The DC response can be computed from its Fourier transform Equation (A.3), with $\xi = 0$ and $\nu = 0$

given by:

$$\hat{\psi}\left(\xi=0,\nu=0,\nu_0\right)=\sqrt{8\pi}\sigma\exp\left(-\left(\frac{\kappa^2}{2}\right)\right) \tag{A.8}$$

A family of admissible 2D Gabor wavelets can be obtained by subtracting this DC response (Equation (A.8)) from the Gabor filter (Equation (A.7)),

$$\psi\left(x,y,\omega_0,\theta\right)=\frac{\omega_0}{\sqrt{2\pi}\kappa}\exp\left(-\frac{\omega_0^2}{8\kappa^2}\left[4(x\cos\theta+y\sin\theta)^2+(-x\sin\theta+y\cos\theta)^2\right]\right)$$
$$\cdot\left[\exp\left(i\left(x\omega_0\cos\theta+y\omega_0\sin\theta\right)\right)-\exp\left(-\left(\frac{\kappa^2}{2}\right)\right)\right] \tag{A.9}$$

Each of these two families of Gabor wavelets can be generated by rotation and dilation (affine group) of the mother Gabor wavelet which is as follows:

$$\psi\left(x,y\right)=\frac{1}{\sqrt{2\pi}}\exp\left[-\frac{1}{8}\left(4x^2+y^2\right)\right]\cdot\left[\exp\left(i\kappa x\right)-\exp\left(-\left(\frac{\kappa^2}{2}\right)\right)\right] \tag{A.10}$$

A.2 Linear Regression

Linear regression models the relationship between the dependent and independent variables. This model is a linear combination of the set of coefficients and the independent variables. These coefficients are known as regression coefficients. This model can be used to predict the dependent variable. The aforesaid models are called as linear models. A linear regression model which has a single independent variable is called a simple linear regression. A straight line is fit through a cloud of data points. This straight line is a best fit when the prediction error is as small as possible. Prediction error is the deviation of the actual data points from the predicted value *i.e.,* least-square approach. In other words, error is the squared sum of the vertical distance between the data points and the fitted straight line. Suppose we have a set of n points (X_i, Y_i), our goal is to find the best fit $\hat{Y}_i=\alpha+\beta X_i$, such that the sum of errors (Q) *i.e.,* $\sum\left(Y_i-\hat{Y}_i\right)^2$ is minimized.

$$Q=\sum_{i=1}^{n}\left(Y_i-\hat{Y}_i\right)^2=\sum_{i=1}^{n}\left(Y_i-\alpha-\beta X_i\right)^2 \tag{A.11}$$

Q is minimized at the values of α and β for which $\partial Q/\partial\alpha=0$ and $\partial Q/\partial\beta=0$. For the first case, differentiating Equation (A.11) with respect to α, we get:

$$\frac{\partial Q}{\partial\alpha}=\sum_{i=1}^{n}-2\left(Y_i-\alpha-\beta X_i\right)=2\left(n\alpha+\beta\sum_{i=1}^{n}X_i-\sum_{i=1}^{n}Y_i\right)=0 \tag{A.12}$$

After simplification,

$$\alpha = \bar{Y} - \beta\bar{X} \tag{A.13}$$

where, $\bar{X}$ and $\bar{Y}$ are the mean of X and Y respectively. When minimizing Q with respect to β *i.e.,* differentiating Equation (A.11) with respect to β, we get:

$$\frac{\partial Q}{\partial \beta} = \sum_{i=1}^{n} -2X_i \left(Y_i - \alpha - \beta X_i\right) = \sum_{i=1}^{n} -2\left(X_iY_i - \alpha X_i - \beta X_i^2\right) = 0 \tag{A.14}$$

Substituting the expression for α from Equation (A.13) into Equation(A.14), we get:

$$\sum_{i=1}^{n} \left(X_iY_i - X_i\bar{Y} + \beta X_i\bar{X} - \beta X_i^2\right) = 0 \tag{A.15}$$

Separating the above Equation (A.15) into two terms, we get:

$$\sum_{i=1}^{n} \left(X_iY_i - X_i\bar{Y}\right) - \beta \sum_{i=1}^{n} \left(X_i^2 - X_i\bar{X}\right) = 0 \tag{A.16}$$

From the above Equation (A.16), β can be obtained which is shown below:

$$\beta = \frac{\sum\limits_{i=1}^{n} \left(X_iY_i - X_i\bar{Y}\right)}{\sum\limits_{i=1}^{n} \left(X_i^2 - X_i\bar{X}\right)} \tag{A.17}$$

For simplification, Equation (A.17) can be written as follows:

$$\beta = \frac{\sum\limits_{i=1}^{n} \left(X_iY_i - X_i\bar{Y}\right) + \sum\limits_{i=1}^{n} \left(\bar{X}\bar{Y} - Y_i\bar{X}\right)}{\sum\limits_{i=1}^{n} \left(X_i^2 - X_i\bar{X}\right) + \sum\limits_{i=1}^{n} \left(\bar{X}^2 - X_i\bar{X}\right)} \tag{A.18}$$

$$\left[\because \sum_{i=1}^{n} \left(\bar{X}^2 - X_i\bar{X}\right) = 0 \text{ and } \sum_{i=1}^{n} \left(\bar{X}\bar{Y} - Y_i\bar{X}\right) = 0\right]$$

Rewriting the above Equation (A.18) gives:

$$\beta = \frac{\frac{1}{n}\sum\limits_{i=1}^{n} \left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{\frac{1}{n}\sum\limits_{i=1}^{n} \left(X_i - \bar{X}\right)^2} \tag{A.19}$$

Finally, β is given by:

$$\beta = \frac{Cov(X, Y)}{Var(X)} \tag{A.20}$$

Here, $Cov(X, Y)$ is the correlation between X and Y, and $Var(X)$ is the variance of X.

Bibliography

[1] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis and machine vision.* Cengage Learning, 2007.

[2] S. Mattoccia. (2011, May) Stereo vision: Algorithms and applications. [Online]. Available: http://vision.deis.unibo.it/~smatt/Seminars/StereoVision.pdf

[3] C. J. Prabhakar and K. Jyothi, "Segment-based stereo correspondence of face images using wavelets," in *Proc. International Conference on Signal and Image Processing (ICSIP)*, 2012, pp. 79–89.

[4] J. Ralli, "Fusion and regularisation of image information in variational correspondence methods," Ph.D. dissertation, Univ. of Granada, 2012.

[5] C. Unger and N. Navab. (2009) Stereo matching. [Online]. Available: http://campar.in.tum.de/twiki/pub/Chair/TeachingWs09Cv2/3D_CV2_WS_2009_Stereo.pdf

[6] S. Tijmons, "Stereo vision for flapping wing MAVs-Design of an obstacle avoidance system," Master's thesis, Delft Univ. of Technology, 2012.

[7] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 195–202.

[8] B. Cyganek and J. P. Siebert, *An introduction to 3D Computer Vision techniques and algorithms.* John Wiley and sons, 2009.

[9] S. El-Etriby, A. Al-Hamadi, and B. Michaelis, "Dense stereo correspondence with slanted surface using phase-based algorithm," in *Proc. IEEE International Symposium on Industrial Electronics (ISIE)*, 2007, pp. 1807–1813.

[10] K.-J. Yoon and I. S. Kweon, "Adaptive support-weight approach for correspondence search," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 4, pp. 650–656, 2006.

[11] A. Hosni *et al.*, "Fast cost-volume filtering for visual correspondence and beyond," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 2, pp. 504–511, 2013.

[12] D. Scharstein and C. Pal, "Learning conditional random fields for stereo," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.

[13] H. Hirschmuller and D. Scharstein, "Evaluation of cost functions for stereo matching," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8.

[14] S. Huq, A. Koschan, and M. Abidi, "Occlusion filling in stereo: Theory and experiments," *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 688–704, 2013.

[15] T. Malathi and M. K. Bhuyan, "Estimation of disparity map of stereo image pairs using spatial domain local Gabor wavelet," *IET Computer Vision*, vol. 9, no. 4, pp. 595–602, 2015.

[16] D. Min and K. Sohn, "Cost aggregation and occlusion handling with WLS in stereo matching," *Image Processing, IEEE Transactions on*, vol. 17, no. 8, pp. 1431–1442, 2008.

[17] R. Zhang *et al.*, "Shape from shading: A survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 8, pp. 690–706, 1999.

[18] M. Clerc and S. Mallat, "The texture gradient equation for recovering shape from texture," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 4, pp. 536–549, 2002.

[19] Y.-P. Wang, S. L. Lee, and K. Toraichi, "Multiscale curvature-based shape representation using B-spline wavelets," *Image Processing, IEEE Transactions on*, vol. 8, no. 11, pp. 1586–1592, 1999.

[20] I. Akhter *et al.*, "Trajectory space: A dual representation for nonrigid structure from motion," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 7, pp. 1442–1456, 2011.

[21] R. R. Sahay and A. N. Rajagopalan, "Dealing with parallax in shape-from-focus," *Image Processing, IEEE Transactions on*, vol. 20, no. 2, pp. 558–569, 2011.

[22] M. Brannstrom, E. Coelingh, and J. Sjoberg, "Model-based threat assessment for avoiding arbitrary vehicle collisions," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 11, no. 3, pp. 658–669, 2010.

[23] H. Cheng, H. Chen, and Y. Liu, "Topological indoor localization and navigation for autonomous mobile robot," *Automation Science and Engineering, IEEE Transactions on*, vol. 12, no. 2, pp. 729–738, 2015.

[24] C. Teuliere and E. Marchand, "A dense and direct approach to visual servoing using depth maps," *Robotics, IEEE Transactions on*, vol. 30, no. 5, pp. 1242–1249, 2014.

[25] N. Uchiyama *et al.*, "Model-reference control approach to obstacle avoidance for a human-operated mobile robot," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 10, pp. 3892–3896, 2009.

[26] M. C. Yip *et al.*, "Tissue tracking and registration for image-guided surgery," *Medical Imaging, IEEE Transactions on*, vol. 31, no. 11, pp. 2169–2182, 2012.

[27] R. Richa *et al.*, "Vision-based proximity detection in retinal surgery," *Biomedical Engineering, IEEE Transactions on*, vol. 59, no. 8, pp. 2291–2301, 2012.

[28] S. Treuillet, B. Albouy, and Y. Lucas, "Three-dimensional assessment of skin wounds using a standard digital camera," *Medical Imaging, IEEE Transactions on*, vol. 28, no. 5, pp. 752–762, 2009.

[29] B. Kamolrat *et al.*, "3D motion estimation for depth image coding in 3D video coding," *Consumer Electronics, IEEE Transactions on*, vol. 55, no. 2, pp. 824–830, 2009.

[30] S. Hodges and B. Richards, "Looking for a cheaper robot: Visual feedback for automated PCB manufacture," Ph.D. dissertation, University of Cambridge, 1996.

[31] M. Whitehorn *et al.*, "Stereo vision in LHD automation," *Industrial Applications, IEEE Transactions on*, vol. 39, no. 1, pp. 21–29, 2003.

[32] F. Rovira-Mas, Q. Zhang, and J. F. Reid, "Stereo vision three-dimensional terrain maps for precision agriculture," *Computers and Electronics in Agriculture*, vol. 60, no. 2, pp. 133–143, 2008.

[33] V. Leemans, B. Dumont, and M. Destain, "Assessment of plant leaf area measurement by using stereo vision," in *Proc. International Conference on 3D imaging (IC3D)*, 2013, pp. 1–5.

[34] T. Kemppainen and A. Visala, "Stereo vision based tree planting spot detection," in *Proc. International Conference on Robotics and automation (ICRA)*, 2013, pp. 739–745.

[35] V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2001, pp. 508–515.

[36] ——, "Multi-camera scene reconstruction via graph cuts," in *Proc. European Conference on Computer Vision (ECCV)*, 2002, pp. 82–96.

[37] O. Woodford *et al.*, "Global stereo reconstruction under second order smoothness priors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 12, pp. 2115–2128, 2009.

[38] M. Bleyer and C. Breiteneder, "Stereo matching: State-of-the-art and research challenges," in *Advanced Topics in Computer Vision*, 2013, pp. 143–179.

[39] M. H. Lin and C. Tomasi, "Surfaces with occlusions from layered stereo," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 8, pp. 1073–1078, 2004.

[40] M. Bleyer and M. Gelautz, "Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions," *Signal Processing: Image Communication*, vol. 22, no. 2, pp. 127–143, 2007.

[41] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 11, pp. 1222–1239, 2001.

[42] P. Felzenszwalb and D. Huttenlocher, "Efficient belief propagation for early vision," *International Journal of Computer Vision*, vol. 70, no. 1, pp. 41–54, 2006.

[43] C. L. Zitnick *et al.*, "High-quality video view interpolation using a layered representation," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 600–608, 2004.

[44] Y. Deng *et al.*, "A symmetric patch-based correspondence model for occlusion handling," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 1316–1322.

[45] P. N. Belhumeur, "A Bayesian approach to binocular steropsis," *International Journal of Computer Vision*, vol. 19, no. 3, pp. 237–260, 1996.

[46] O. Veksler, "Stereo correspondence by dynamic programming on a tree," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 384–390.

[47] M. Gong and Y.-H. Yang, "Near real-time reliable stereo matching using programmable graphics hardware," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 924–931.

[48] ——, "Real-time stereo matching using orthogonal reliability-based dynamic programming," *Image Processing, IEEE Transactions on*, vol. 16, no. 3, pp. 879–884, 2007.

[49] X. Chang *et al.*, "Real-time accurate stereo matching using modified two-pass aggregation and Winner-take-all guided dynamic programming," in *Proc. IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, 2011, pp. 73–79.

[50] V. Lempitsky, C. Rother, and A. Blake, "Logcut - efficient graph cut optimization for Markov random fields," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.

[51] V. Lempitsky *et al.*, "Fusion moves for Markov random field optimization," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 8, pp. 1392–1405, 2010.

[52] A. Zureiki, M. Devy, and R. Chatila, "Stereo matching using reduced-graph cuts," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2007, pp. 237–240.

[53] Y. Wang, C. Tung, and P. Chung, "Efficient disparity estimation using hierarchical bilateral disparity structure based graph cut algorithm with a foreground boundary refinement mechanism," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 5, pp. 784–801, 2013.

[54] J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 7, pp. 787–800, 2003.

[55] T. Yu *et al.*, "Efficient message representations for belief propagation," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.

[56] L. Wang, H. Jin, and R. Yang, "Search space reduction for MRF stereo," in *Proc. European Conference on Computer Vision (ECCV)*, 2008, pp. 576–588.

[57] Q. Yang, L. Wang, and N. Ahuja, "A constant-space belief propagation algorithm for stereo matching," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 1458–1465.

[58] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7–42, 2002.

[59] N. Einecke and J. Eggert, "A two-stage correlation method for stereoscopic depth estimation," in *Proc. IEEE International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2010, pp. 227–234.

[60] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. European Conference on Computer Vision (ECCV)*, 1994, pp. 151–158.

[61] M. Humenberger *et al.*, "A fast stereo matching algorithm suitable for embedded real-time systems," *Computer Vision and Image Understanding*, vol. 114, no. 11, pp. 1180–1202, 2010.

[62] K. Ambrosch and W. Kubinger, "Accurate hardware-based stereo vision," *Computer Vision and Image Understanding*, vol. 114, no. 11, pp. 1303–1316, 2010.

[63] L. Ma *et al.*, "A modified census transform based on the neighborhood information for stereo matching algorithm," in *Proc. IEEE International Conference on Image and Graphics (ICIG)*, 2013, pp. 533–538.

[64] W. S. Fife and J. K. Archibald, "Improved census transforms for resource-optimized stereo vision," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 1, pp. 60–73, 2013.

[65] S. Perri, P. Corsonello, and G. Cocorullo, "Adaptive census transform: A novel hardware-oriented stereovision algorithm," *Computer Vision and Image Understanding*, vol. 117, no. 1, pp. 29–41, 2013.

[66] Z. Lee, J. Juang, and T. Q. Nguyen, "Local disparity estimation with three-moded cross census and advanced support weight," *Multimedia, IEEE Transactions on*, vol. 15, no. 8, pp. 1855–1864, 2013.

[67] D. Geiger, B. Ladendorf, and A. Yille, "Occlusions and binocular stereo," *International Journal of Computer Vision*, vol. 14, no. 3, pp. 211–226, 1995.

[68] S. B. Kang, R. Szeliski, and J. Chai, "Handling occlusions in dense multi-view stereo," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001, pp. 103–110.

[69] A. Fusiello, V. Roberto, and E. Trucco, "Efficient stereo with multiple windowing," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1997, pp. 858–863.

[70] S. S. Intille and A. F. Bobick, "Disparity-space images and large occlusion stereo," in *Proc. European Conference on Computer Vision (ECCV)*, 1994, pp. 179–186.

[71] H. Hirschmuller, P. R. Innocent, and J. Garibaldi, "Real-time correlation-based stereo vision with reduced border errors," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 229–246, 2002.

[72] M. Okutomi, Y. Katayama, and S. Oka, "A simple stereo algorithm to recover precise object boundaries and smooth surfaces," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 261–273, 2002.

[73] J. Jeon, C. Kim, and Y. Ho, "Sharp and dense disparity maps using multiple windows," in *Proc. IEEE Pacific Rim Conference on Multimedia (PCM)*, 2002, pp. 1057–1064.

[74] S. A. Adhyapak, N. Kehtarnavaz, and M. Nadin, "Stereo matching via selective multiple windows," *Journal of Electronic Imaging*, vol. 16, no. 1, p. 013012, 2007.

[75] O. Veksler, "Stereo matching by compact windows via minimum ratio cycle," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2001, pp. 540–547.

[76] ——, "Fast variable window for stereo correspondence using integral images," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 556–561.

[77] T. Kanade and M. Okutomi, "A stereo matching algorithm with an adaptive window: Theory and experiment," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 9, pp. 920–932, 1994.

[78] Y. Boykov, O. Veksler, and R. Zabih, "A variable window approach to early vision," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 12, pp. 1283–1294, 1998.

[79] R. Yang and M. Pollefeys, "Multi-resolution real-time stereo on commodity graphics hardware," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003, pp. 211–217.

[80] Y. Zhao and G. Taubin, "Real-time stereo on GPGPU using progressive multi-resolution adaptive windows," *Image and Vision Computing*, vol. 29, no. 6, pp. 420–432, 2011.

[81] R. Yang, M. Pollefeys, and S. Li, "Improved real-time stereo on commodity graphics hardware," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, 2004, p. 36.

[82] R. K. Gupta and S. Cho, "Window-based approach for fast stereo correspondence," *IET Computer Vision*, vol. 7, no. 2, pp. 123–134, 2013.

[83] A. Hosni, M. Bleyer, and M. Gelautz, "Secrets of adaptive support weight techniques for local stereo matching," *Computer Vision and Image Understanding*, vol. 117, no. 6, pp. 620–632, 2013.

[84] M. Ju and H. Kang, "Constant time stereo matching," in *Proc. International Conference on Machine Vision and Image Processing (IMVIP)*, 2009, pp. 13–17.

[85] K. Zhang *et al.*, "Joint integral histograms and its application in stereo matching," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2010, pp. 817–820.

[86] F. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 829–836.

[87] C. Richardt *et al.*, "Real-time spatiotemporal stereo matching using the dual-cross bilateral grid," in *Proc. European Conference on Computer Vision (ECCV)*, 2010, pp. 510–523.

[88] S. Mattoccia, S. Giardino, and A. Gambini, "Accurate and efficient cost aggregation strategy for stereo correspondence based on approximated joint bilateral filtering," in *Proc. Asian Conference on Computer Vision (ACCV)*, 2009, pp. 371–380.

[89] Q. Yang *et al.*, "Full-image guided filtering for fast stereo matching," *Signal Processing Letters, IEEE*, vol. 20, no. 3, pp. 237–240, 2013.

[90] ——, "A novel guided image filter using orthogonal geodesic distance weight," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2013, pp. 1207–1211.

[91] X. Huang, G. Cui, and Y. Zhang, "An improved filtering for fast stereo matching," in *Proc. IEEE International Conference on Pattern Recognition (ICPR)*, 2014, pp. 2448–2452.

[92] M. Gerrits and P. Bekaert, "Local stereo matching with segmentation-based outlier rejection," in *Proc. Canadian Conference on Compututer and Robot Vision (CRV)*, 2006, p. 66.

[93] F. Tombari, S. Mattoccia, and L. Stefano, "Segmentation-based adaptive support for accurate stereo correspondence," in *Proc. IEEE Pacific-Rim Symposium on Image and Video Technology (PSIVT)*, 2007, pp. 427–438.

[94] F. Tombari *et al.*, "Near real-time stereo based on effective cost aggregation," in *Proc. IEEE International Conference on Pattern Recognition (ICPR)*, 2008, pp. 1–4.

[95] V. Muninder, U. Soumik, and G. Krishna, "Robust segment-based stereo using cost aggregation," in *Proc. British Machine Vision Conference (BMVC)*, 2014, pp. 1–11.

[96] T. D. Sanger, "Stereo disparity computation using Gabor filters," *Biological Cybernatics*, vol. 59, no. 6, pp. 405–418, 1988.

[97] D. Fleet, A. Jepson, and M. Jenkin, "Phase-based disparity measurement," *CVGIP: Image Understanding*, vol. 53, no. 2, pp. 198–210, 1991.

[98] M. H. Ouali, D. Ziou, and C. Laurgeau, "Dense disparity estimation using Gabor filters and image derivatives," in *Proc. IEEE International Conference on 3-D Digital Imaging and Modeling (3DIM)*, 1999, pp. 483–489.

[99] Y. S. Heo, K. M. Lee, and S. U. Lee, "Robust stereo matching using adaptive normalized cross-correlation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 33, no. 4, pp. 807–822, 2011.

[100] C. Fookes, M. Bennamoun, and A. Lamanna, "Improved stereo image matching using mutual information and hierarchical prior probabilities," in *Proc. IEEE International Conference on Pattern Recognition (ICPR)*, 2002, pp. 937–940.

[101] J. Kim, V. Kolmogorov, and R. Zabih, "Visual correspondence using energy minimization and mutual information," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2003, pp. 1033–1040.

[102] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, 2008.

[103] A. Spoerri and S. Ullman, "The early detection of motion boundaries," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 1987, pp. 209–218.

[104] J. J. Little and W. E. Gillett, "Direct evidence for occlusion in stereo and motion," *Image and Vision Computing*, vol. 8, no. 4, pp. 328–340, 1990.

[105] R. P. Wildes, "Direct recovery of three-dimensional scene geometry from binocular stereo disparity," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 8, pp. 761–774, 1991.

[106] B. L. Anderson and K. Nakayama, "Toward a general theory of stereopsis: Binocular matching, occluding contours, and fusion," *Psychological review*, vol. 101, no. 3, pp. 414–445, 1994.

[107] A. Luo and H. Burkhardt, "An intensity-based cooperative bidirectional stereo matching with simultaneous detection of discontinuities and occlusions," *International Journal of Computer Vision*, vol. 15, no. 3, pp. 171–188, 1995.

[108] R. Trapp, S. Drüe, and G. Hartmann, "Stereo matching with implicit detection of occlusions," in *Proc. European Conference on Computer Vision (ECCV)*, 1998, pp. 17–33.

[109] W. Jang and Y. Ho, "Discontinuity preserving disparity estimation with occlusion handling," *Journal of Visual Communication and Image Representation*, vol. 25, pp. 1595–1603, 2014.

[110] G. Egnal and R. P. Wildes, "Detecting binocular half-occlusions: Empirical comparisons of five approaches," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 8, pp. 1127–1133, 2002.

[111] Q. Yang *et al.*, "Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 3, pp. 492–504, 2009.

[112] A. Hosni *et al.*, "Local stereo matching using geodesic support weights," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 2093–2096.

[113] R. Szeliski *et al.*, "A comparative study of energy minimization methods for Markov random fields with smoothness-based priors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 6, pp. 1068–1080, 2008.

[114] M. F. Tappen and W. T. Freeman, "Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2003, pp. 900–906.

[115] T. Meltzer, C. Yanover, and Y. Weiss, "Globally optimal solutions for energy minimization in stereo vision using reweighted belief propagation," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 428–435.

[116] M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 8, pp. 993–1008, 2003.

[117] Middlebury stereo evaluation. [Online]. Available: http://vision.middlebury.edu/stereo/eval/

[118] J. Braux-Zin, R. Dupont, and A. Bartoli, "A general dense image matching framework combining direct and feature-based costs," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 185–192.

[119] P. Pinggera, T. P. Breckon, and H. Bischof, "On cross-spectral stereo matching using dense gradient features," in *Proc. British Machine Vision Conference (BMVC)*, 2012, pp. 1–12.

[120] L. De-Maeztu *et al.*, "Linear stereo matching," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 1708–1715.

[121] T. S. Lee, "Image representation using 2D Gabor wavelets," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 10, pp. 959–971, 1996.

[122] O. Pichler, A. Teuner, and B. J. Hosticka, "A comparison of texture feature extraction using adaptive Gabor filtering, pyramidal and tree structured wavelet transforms," *Pattern Recognition*, vol. 29, no. 5, pp. 733–742, 1996.

[123] H. Hu, "Enhanced Gabor feature based classification using a regularized locally tensor discriminant model for multiview gait eecognition," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 7, pp. 1274–1286, 2013.

[124] S. Jahanbin, H. Choi, and A. C. Bovik, "Passive multimodal 2-D+3-D face recognition using Gabor features and landmark distances," *Information and Forensics Security, IEEE Transactions on*, vol. 6, no. 4, pp. 1287–1304, 2011.

[125] K. Okajima, "The Gabor function extracts the maximum information from input local signals," *Neural Networks*, vol. 11, no. 3, pp. 435–439, 1998.

[126] S. Bhagavathy, J. Tesic, and B. S. Manjunath, "On the rayleigh nature of Gabor filter outputs," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2003, pp. 745–748.

[127] O. Nestares *et al.*, "Efficient spatial-domain implementation of a multiscale image representation based on Gabor functions," *Journal of Electronic Imaging*, vol. 7, no. 1, pp. 166–173, 1998.

[128] M. K. Bhuyan and T. Malathi, "Review of the application of matrix information theory in video surveillance," in *Matrix Information Geometry*, 2012, pp. 293–321.

[129] D. A. Pollen and S. F. Ronner, "Visual cortical neurons as localized spatial frequency filters," *Systems, Man, and Cybernetics, IEEE Transactions on*, vol. 13, no. 5, pp. 907–916, 1983.

[130] R. Navarro and A. Tabernero, "Gaussian wavelet transform: Two alternative fast implementations for images," *Optical Signal Processing*, pp. 67–82, 1991.

[131] R. L. Panetta, C. Liu, and P. Yang, "A pseudo-spectral time domain method for light scattering computation," in *Light scattering reviews*, 2013, pp. 139–188.

[132] T. Twardowski, B. Cyganek, and J. Borgosz, "Gradient based dense stereo matching," in *Proc. International Conference on Image Analysis and Recognition (ICIAR)*, 2004, pp. 721–728.

[133] Q. Yang, K.-H. Tan, and N. Ahuja, "Real-time O(1) bilateral filtering," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 557–564.

[134] K. He, J. Sun, and X. Tang, "Guided image filtering," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 6, pp. 1397–1409, 2013.

[135] G. Papari, N. Petkov, and P. Campisi, "Artistic edge and corner enhancing smoothing," *Image Processing, IEEE Transactions on*, vol. 16, no. 10, pp. 2449–2462, 2007.

[136] Z. Ma *et al.*, "Constant time weighted median filtering for stereo matching and beyond," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 49–56.

[137] R. Szeliski and R. Zabih, "An experimental comparison of stereo algorithms," in *Proc. International Workshop on Vision Algorithms: Theory and Practice*, 1999, pp. 1–19.

[138] M. Samadi and M. F. Othman, "A new fast and robust stereo matching algorithm for robotic systems," in *International Conference on Computing and Information Technology (IC2IT)*, 2013, pp. 281–290.

[139] L. Nalpantidis and A. Gasteratos, "Biologically and psychophysically inspired adaptive support weights algorithm for stereo correspondence," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 457–464, 2010.

[140] S. El-Etriby, A. K. Al-Hamadi, and B. Michaelis, "Dense depth map reconstruction by phase difference-based algorithm under influence of perspective distortion," *International Journal of Machine Graphics and Vision*, vol. 15, no. 3, pp. 349–361, 2006.

[141] L. Nalpantidis and A. Gasteratos, "Stereo vision for robotic applications in the presence of non-ideal lighting conditions," *Image and Vision Computing*, vol. 28, no. 6, pp. 940–951, 2010.

[142] C. L. Zitnick and T. Kanade, "A cooperative algorithm for stereo matching and occlusion detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 7, pp. 675–684, 2000.

[143] J. Sun *et al.*, "Symmetric stereo matching for occlusion handling," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005, pp. 399–406.

[144] P. Mordohai and G. Medioni, "Stereo using monocular cues within the tensor voting framework," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 6, pp. 968–982, 2006.

[145] T. Pock *et al.*, "A convex formulation of continuous multi-label problems," in *Proc. European Conference on Computer Vision (ECCV)*, 2008, pp. 792–805.

[146] C. Strecha, R. Fransens, and L. Van Gool, "Combined depth and outlier estimation in multi-view stereo," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 2394–2401.

[147] D. Miyazaki, Y. Matsushita, and K. Ikeuchi, "Interactive shadow removal from a single image using hierarchical graph cut," in *Proc. Asian Conference on Computer Vision (ACCV)*, 2009, pp. 234–245.

[148] X. Mei *et al.*, "On building an accurate stereo matching system on graphics hardware," in *Proc. IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 467–474.

[149] L. Lin *et al.*, "Representing and recognizing objects with massive local image patches," *Pattern Recognition*, vol. 45, no. 1, pp. 231–240, 2012.

[150] Z. Peng *et al.*, "Deep boosting: Joint feature selection and analysis dictionary learning in hierarchy," *Neurocomputing*, vol. 178, pp. 36–45, 2016.

[151] A. Ali, "A 3D-based pose invariant face recognition at a distance framework," *Information Forensics and Security, IEEE Transactions on*, vol. 9, no. 12, pp. 2158–2169, 2014.

[152] J. Liebelt, J. Xiao, and J. Yang, "Robust AAM fitting by fusion of images and disparity data," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006, pp. 2483–2490.

[153] S. Kosov *et al.*, "Rapid stereo-vision enhanced face detection," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 1221–1224.

[154] J. G. Daugman, "Two-dimensional spectral analysis of cortical receptive field profiles," *Vision research*, vol. 20, no. 10, pp. 847–856, 1980.

[155] ——, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *Journal of the Optical Society of America A: Optics, Image Science, and Vision*, vol. 2, no. 7, pp. 1160–1169, 1985.

[156] O. Rajadell, P. Garca-Sevilla, and F. Pla, "Scale analysis of several filter banks for color texture classification," in *International Symposium on Visual Computing*, 2009, pp. 509–518.

[157] J. Y. Tou, Y. H. Tay, and P. Y. Lau, "Gabor filters and grey-level co-occurrence matrices in texture classification," in *Proc. MMU International Symposium on Information and Communications Technologies*, 2007, pp. 197–202.

[158] ——, "Gabor filters as feature images for covariance matrix on texture classification problem," in *Advances in Neuro-Information Processing*, 2009, pp. 745–751.

[159] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

[160] A. G. Zuñiga, J. B. Florindo, and O. M. Bruno, "Gabor wavelets combined with volumetric fractal dimension applied to texture analysis," *Pattern Recognition Letters*, vol. 36, pp. 135–143, 2014.

[161] B. Zhang *et al.*, "Histogram of Gabor phase patterns (HGPP): A novel object representation approach for face recognition," *Image Processing, IEEE Transactions on*, vol. 16, no. 1, pp. 57–68, 2007.

[162] C. Xu *et al.*, "Automatic 3D face recognition from depth and intensity Gabor features," *Pattern Recognition*, vol. 42, no. 9, pp. 1895–1905, 2009.

[163] M. Yang *et al.*, "Gabor feature based robust representation and classification for face recognition with Gabor occlusion dictionary," *Pattern Recognition*, vol. 46, no. 7, pp. 1865–1878, 2013.

[164] X. Xie and K.-M. Lam, "Facial expression recognition based on shape and texture," *Pattern Recognition*, vol. 42, no. 5, pp. 1003–1011, 2009.

[165] L. Zhang, D. Tjondronegoro, and V. Chandran, "Random Gabor based templates for facial expression recognition in images with facial occlusion," *Neurocomputing*, vol. 145, pp. 451–464, 2014.

[166] H. A. Moghaddam and M. N. Dehaji, "Enhanced Gabor wavelet correlogram feature for image indexing and retrieval," *Pattern Analysis and Applications*, vol. 16, no. 2, pp. 163–177, 2013.

[167] W. Jiang, K.-M. Lam, and T.-Z. Shen, "Efficient edge detection using simplified Gabor wavelets," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 4, pp. 1036–1047, 2009.

[168] L. Shen and S. Jia, "Three-dimensional Gabor wavelets for pixel-based hyperspectral imagery classification," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 49, no. 12, pp. 5039–5046, 2011.

[169] J. V. Soares *et al.*, "Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification," *Medical Imaging, IEEE Transactions on*, vol. 25, no. 9, pp. 1214–1222, 2006.

[170] J. P. Jones and L. A. Palmer, "An evaluation of the two-dimensional Gabor filter model of simple receptive fields in cat striate cortex," *Journal of Neurophysiology*, vol. 58, no. 6, pp. 1233–1258, 1987.

[171] J. Daugman, "Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 36, no. 7, pp. 1169–1179, 1988.

[172] D. Scharstein *et al.*, "High-resolution stereo datasets with subpixel-accurate ground truth," in *Proc. German Conference on Pattern Recognition (GCPR)*, 2014, pp. 31–42.

[173] C. P. Loizou *et al.*, "Quality evaluation of ultrasound imaging in the carotid artery based on normalization and speckle reduction filtering," *Medical and Biological Engineering and Computing*, vol. 44, no. 5, pp. 414–426, 2006.

[174] M. Mohamed *et al.*, "Illumination-robust optical flow using a local directional pattern," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 24, no. 9, pp. 1499–1508, 2014.

[175] H. Hirschmüller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 9, pp. 1582–1599, 2009.

[176] A. Kumar and G. Pang, "Fabric defect segmentation using multichannel blob detectors," *Optical Engineering*, vol. 39, no. 12, pp. 3176–3190, 2000.