RESEARCH ON ROBUST SALIENT OBJECT EXTRACTION IN IMAGE

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL OF INFORMATION, PRODUCTION AND SYSTEMS

OF WASEDA UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

Jing Wang

2007.11

Jing Wang

Abstract

Object extraction is an essential branch of the computer vision, concerned with the analysis of images to extract useful information from the world. It covers wide range of image processing fields and plays an important role in computer vision. In this dissertation, contour detection is mainly discussed to realize object extraction, realized in three steps: edge detection, contour segment detection and object contour detection. In addition, an application of robust salient contour detection to 3D building reconstruction in aerial image is also discussed. Due to the complex scene of the natural image, it is necessary for us to develop robust contour detection methods against the noise and texture. And the computing complexity of the robust methods should also be considered.

For the problem of salient edge detection, a multi-scale transition region based edge detector is proposed. According to existence of the “transition region” in natural image, a novel gradient computing method based on the transition region is designed. The central pixels of the transition region are prominent with low gradient magnitude while their neighborhood pixels are with high gradient magnitude. Through the multi-scale analysis by a series of the windows with variable sizes, we can detect the real edge pixels based on the relatively stable gradient distribution pattern of the transition region across the scales. Experimental results show that the proposed method performs better than Canny edge detector and an eigen-analysis based color edge detector with more connected edges.

For the problem of salient contour segment detection, a scale invariance based line detector is proposed. A refined Nevatia-Babu line operator is firstly proposed based on the consecutive property of edge pixels on a line with simpler computing. Then a scale invariance based line detector is proposed based on the scale invariance property of the salient line segments on the object contour. According to the experimental results, the proposed method has almost no redundancy of the detected line segments compared with high redundancy level of the Progressive Probabilistic Hough Transform (PPHT) method

on the same experimental images. The proposed method also works more efficiently than PPHT methods, especially when the image includes many line segments.

For the problem of salient object contour detection, a Constrained Delaunay Triangulation (CDT) graph based contour detection method is proposed. A structure called the Constrained Delaunay Triangulation (CDT) graph is introduced in order to enormously reduce the computing complexity. All the CDT edges are classified into "contour link" cluster or "non-contour link" cluster according to their saliency measures. Experimental results show that the proposed method can derive contours with better detection rate and higher saliency of detected contours than the former method.

For the problem of 3D building reconstruction, the Semi-Automatic Rectangle based Dynamic Programming Stereo Matching (SAR-DPSM) algorithm is proposed based on the observation that rectangle is a basic geometrical element of the buildings. Experimental results show that the proposed SAR-DPSM performs better than Pixel-Level DPSM (PL-DPSM), while Manual Rectangle Based DPSM (MR-DPSM) performs better than the proposed SAR-DPSM. But the proposed SAR-DPSM can save averagely half of the manual working time compared with MR-DPSM with graceful performance.

In summary, the problem of robust salient object extraction and its application are discussed in this dissertation. The proposed algorithms can robustly detect salient image features from the cluttered background. And salient rectangle contour based stereo matching process can help to reduce the noise in the reconstruction data and also improve the efficiency compared with former method.

Keywords

Object Extraction, Contour Detection, Transition Region, Scale Invariance, Perceptual Grouping, Semi-Automatic Rectangle based Dynamic Programming Stereo Matching (SAR-DPSM)

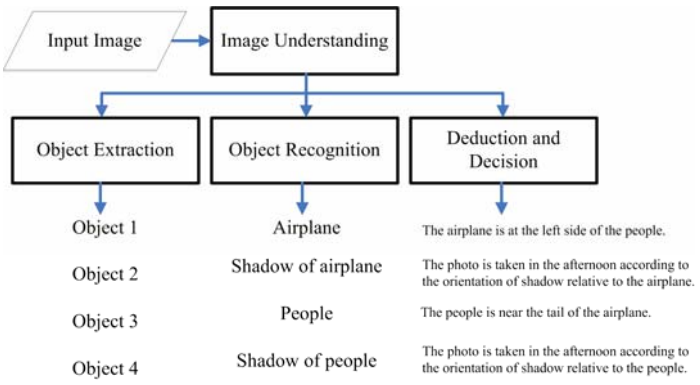
Contents

1 Introduction

1.1 Image Understanding and Object Extraction

Image understanding is the branch of the computer science concerned with the analysis of images to extract useful information from the world. It covers wide range of image processing fields and plays an important role in computer vision.

In 1998, Jim Gray mentioned in his Turing award lecture that “See as well as a person” is one of the challenging problems for the development of future computer. [1] The difficulty comes from building a model to encode the heuristic part in perception and understanding because of its abstractness. Many researchers have been working long time in order to design the cognitive computer system that can not only automatically perceive the world through visual information but also emulate the capability of human beings in visual perception. According to David Marr, “Vision is the process of discovering from images what is present in the world and where it is.”[2] It shows that computer vision system mainly deals with two problems. One is how to acquire visual information and the other is how to extract knowledgeable information from the visual information. The latter is the central work of image understanding.



(a) (b)

Figure 1.1 Description of the process of image understanding on an example image (a)

the example image (b) illustration of image understanding on the example image

In order to facilitate the information exchange in the Image Understanding (IU) community, since 1990, a project called Image Understanding Environment (IUE) was carried out[3, 4]. Its objective is to build a common object-oriented software environment of various image understanding algorithms and techniques. Compared with IUE, the Intel Open Source Computer Vision Library (OpenCV) is much more popular. It is comprised of many algorithms of image processing and computer vision, implemented in C or C++. Applications of the OpenCV library include human-computer interaction (HCI), object identification, segmentation and recognition, face recognition, gesture recognition, motion tracking, ego motion, motion understanding, structure from motion (SFM) and mobile robotics [5].

Compared with image enhancement, image de-noising and so on, image understanding is a high-level vision task since it involves the knowledge on the semantics of images. Low-level image processing deals with the original image data like intensity or color values, while high-level computer vision performs on the abstract data such as object location, object size, object shape and mutual relations among objects. Based on the information contained from the image, high-level computer vision system can decide to take certain action.

An intermediate step between the low-level and high-level vision tasks is “how to obtain object from the raw image data”. Here we call it a “Middle-Level Vision” problem. Because of the complexity of the real world and the diversity of applications, it is still an unsolved problem. The middle-level vision task includes object extraction and object recognition. From the example in Figure 1.1, it is easy to find out that robust object extraction is necessary for accurate object recognition and also the following decision and deduction according to the extracted semantic information. For various image types like aerial image, medical image, document image etc., there are quite different approaches to

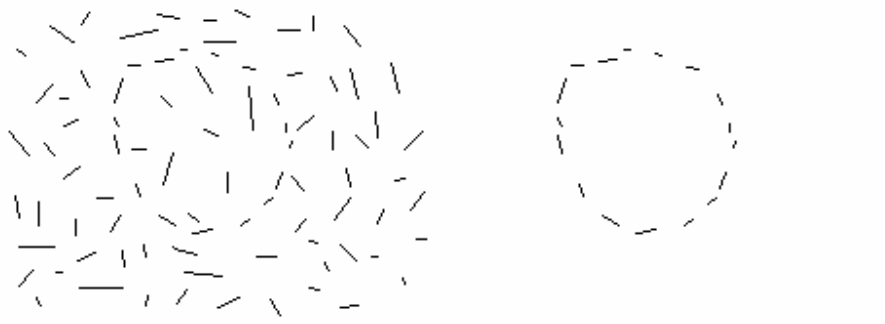
perform specific object extraction and recognition. In addition, for various applications like biometric recognition system, optical character recognition (OCR) system and geographic information system (GIS), interested objects are also very different each other. In the dissertation, we mainly discuss the object extraction in general case, independent of the image type or the application and then address two specific application cases.

Object extraction is an essential part in image understanding. There have been wide applications of object extraction in real world, such as face detection, motion estimation, robot vision systems and so on. Generally speaking, the approaches of object extraction can be classified into two parts. One is image segmentation based on the region homogeneity of object inner region and the other is contour detection based on the color or gray-level distribution change across the boundary between different objects. In this dissertation, we mainly discuss the robust contour detection methods.

The complex background scene in the real natural images makes it more difficult to accurately extract objects due to the noise and texture. Therefore it is necessary for us to research on the robust object extraction methods against noise and texture. In order to improve the accuracy of contour detection, noise filtering, multi-scale analysis and perceptual grouping have been applied in this process. However, the increasing computing complexity of the above methods compared with traditional methods asks for the reduction of the computing load and at the same time keeping the performance of the robust detection algorithms.

The digital geographic information system has attracted more and more attention in these years. Among the several topics, building modeling is one of the most important problems. Aiming at accurate building extraction to prepare for the following 3D building reconstruction, we make detailed research on robust salient contour detection in image. There are mainly three steps to extract object contours, which are edge detection, contour segment detection and contour detection. Edge pixels are firstly detected and then salient line segments on contour are extracted by linking the connective edge pixels.

After that, salient line segments are grouped into contours. For the three features discussed in the dissertation, such as edge pixels, line segments and contours, it is necessary for us to avoid the features from noise or texture but to obtain only the salient features under the complex background scene. The saliency of objects is closely related to the visual perception of Human Visual System (HVS). Human visual system is an integrated system with both optical and psychological functions. Visual cortex in the brain plays a role in higher level processing of visual image, that is, the visual perception [6, 7]. Psychological and biological researchers have been cooperating to explore the mechanism of the human visual system. Up to now, many properties have been found and made use of in computer vision. For example, the “center-surround” operations, the anisotropic filter [8], the orientation selective filter [9-13], Gestalt psychology [14-19]. Through analyzing the human visual system, we can model the visual saliency of object in many representing forms. Observing the image in Figure 1.2, we can find that there is an ellipse in the background of many short line segments because of its contour saliency.



(a) (b)

Figure 1.2 Illustration of contour saliency (a) the salient contour embedded in the texture (b) the isolated salient contour

1.2 Our Contributions

The dissertation concentrates on how to extract the objects that are salient in the sense of human visual perception. Edge pixels are firstly detected with the proposed multi-scale transition region based edge detector. Then salient line segments on contour are extracted by the proposed scale invariance based line detector. After that, salient line segments are grouped into contours through the proposed Constrained Delaunay Triangle (CDT) graph based contour detection algorithm. Based on the above results, 3D building models are reconstructed by the proposed Semi-Automatic Rectangle based DPSM (Dynamic Programming based Stereo Matching) method.

1.2.1 Algorithm for Salient Edge Detection

Different from the contour saliency and region saliency, there is no structural information in edge saliency. Only scattered independent pixels are involved as the processing elements. In visual perceptual sense, not all abrupt change in the intensity or color can stimulate the perception of human being to separate the foreground from the background. For example, the fired responses in the texture region are not meaningful for visual perception.

In order to suppress the texture or noise responses and simultaneously reinforce the contour responses, we propose a multi-scale transition region based edge detector. Firstly, the traditional template-based edge detector suffers the problem of blurring edge. The problem comes from the sum of the gradient magnitudes over the neighborhood region in the template-based edge detector. The averaging effect of the sum computing makes it difficult to assure the edge of one-pixel width. In order to solve this problem, we propose the transition region based edge detector. Through analysis, we verify that the existence of the transition region in the real natural image and also the accurate location of real edge based on the distribution characteristics of the transition region. At the same time,

we also introduce the multi-scale analysis to suppress the responses from noise or texture. Considering the computing complexity of traditional multi-scale analysis based on the Gaussian functions, we design the strategy of a series of local windows with variable sizes to realize the multi-scale analysis. Experimental results show that the proposed multi-scale transition region based edge detector can obtain connected and also accurate edges from the complex scene background.

1.2.2 Algorithm for Salient Contour Segment Detection

Contour extraction is rather a structural analysis on various components than high contrast detection in salient edge detection. There are mainly two types of salient contour extraction problems. The first case is to extract the perceptually salient contour composed of broken small line segments embedded in the texture [18], like the example image shown in Figure 1.2 (a). The other case is to group the geometrical elements into a contour curve in a natural image [21, 22], with an example shown in Figure 1.3. Both cases of contour extraction are carried out according to the rule that a closed and salient contour is the favorite. The difference is that the first type relies on statistical analysis and the second one depends on the geometrical analysis. We mainly discuss the second type of problem in the dissertation.

We can firstly extract the curve segments necessary for the contour grouping from the edge detection result. The extracted curve segments have already had some geometric information and perceptual information compared with the scattered pixels or very small short lines. At the same time, the curve segments are much easier to handle because they are in smaller number and in higher abstract level than pixels. To improve the efficiency, we propose a refined Nevatia-Babu operator [23] to finish the curve segment extraction in one step, while the traditional operator [24] has to finish it in two steps. Then we extend this method to multi-scale version and propose the scale invariance based line detector. Multi-scale approach can suppress the curve segments coming from the noise

and texture and reinforce the curve segments from the contour, and therefore prepare good data for the following perceptual grouping [23]. Here we also make simplification on the multi-scale analysis by repeatedly smoothing and sub-sampling the image to produce the multi-scale images. Experimental results show that the proposed scale invariance based line detector performs better than the former Progressive Probabilistic Hough Transform (PPHT) [51] with better detection rate and less processing time.

1.2.3 Algorithm for Salient Object Contour Detection

After extracting the salient curve segments, we would like to group them together into several closed salient contours in perceptual sense. During this process, the Gestalt laws such as the proximity, the continuity and the closure are all transformed into explicit geometric rules. Since the salient contour is always in the cluttered background, the possible combination of the curve segments is in combinatorial number [25]. It is impossible for us to compute the probability of every combination. Therefore, the optimization method is very important in this process. The approaches on perceptual grouping differentiate form each other in the ways to encode the Gestalt laws, the ways to design the objective function and the ways to obtain the optimal result.

A CDT graph based contour detection algorithm is proposed in this chapter. Most of the perceptual grouping methods are implemented on the contour segments, usually straight line segments. In order to extract salient object contours, perceptual grouping methods have to search for the suitable linking paths among all the line segments. This process can be viewed as a nearest path searching problem in the graph. The nodes of the graph are the line segments and the edges of the graph are the links between every pair of line segments. The computing complexity is rather high since the perceptual grouping method has to search for optimal path among the combinatorial number of graph edges. To address this challenge, we adopt the structure of CDT graph to form a regular structure before perceptual grouping. It can be proved that all the links for the potential

contours are included in this CDT graph. At the same time the introduction of the CDT graph can enormously reduce the computing complexity because the possible links are reduced to very low number.

On the other hand, since there are only the salient line segments detected by the proposed multi-scale contour segment detection method, we can transfer the perceptual grouping into a two-class classification problem by classifying all the CDT edges into "contour link" cluster or "non-contour link" cluster. The CDT edges are classified according to the value of their saliency measures. Here a novel saliency measure based on the Snakes model is proposed. The saliency measure includes the consideration both on the Gestalt laws and on the image information. Experimental results show that the proposed method can derive contours with better continuity and saliency than the former method under the condition of cluttered background.

1.2.4 Algorithm for 3D Building Reconstruction

After the salient object contour detection, we make further research on its application on the problem of 3D building reconstruction in the aerial image. Since the rectangle is the basic geometrical element of the building contours, we propose the Semi-Automatic Rectangle based Dynamic Programming Stereo Matching (SAR-DPSM) method to facilitate the stereo matching with rectangle features. The former research work on stereo matching does not make full use of the structural characteristics of the buildings and also suffers from the interference from the noise or texture.

In order to solve these problems, a robust stereo matching method called Semi-Automatic Rectangle based Dynamic Programming Stereo Matching (SAR-DPSM) [93] is proposed. The rectangle contours are firstly detected on the set of the salient line segments. Then the candidate rectangles are post-processed to remove the redundancy detection and also to adjust the candidate rectangle into the strict rectangle shape. After that, the detected rectangle contours are used as the stereo matching features to obtain the

3D reconstruction of the buildings in the aerial image. Experimental results show that the proposed SAR-DPSM can perform better than Pixel-Level DPSM (PL-DPSM), while the Manual Rectangle Based DPSM (MR-DPSM) performs better than the proposed SAR-DPSM. On the other hand, the proposed SAR-DPSM can save averagely half of the manual working time compared with MR-DPSM with graceful performance.

1.3 Dissertation Organization

The rest of this dissertation is organized as follows: the salient edge detection in the color image is firstly discussed in Chapter 2. Then a scale invariance based line detector is proposed in Chapter 3. In Chapter 4, CDT graph based contour detection algorithm is discussed. To evaluate the performance of the proposed methods, an application of salient object extraction, building detection and 3D reconstruction in aerial image, is introduced in Chapter 5. Finally, Chapter 6 concludes the whole dissertation.

2 Salient Edge Detection

2.1 Background and Related Work

In this chapter, salient object extraction based on edge saliency is mainly discussed. Psychological researches show that human visual system tends to recognize the differences between homogeneous regions rather than the inner region. This function of the neurons is called “lateral inhibition” [32]. According to this, salient edge detection can be deemed as one of the most fundamental problems in the research field of computer vision. Because of the complexity and diversity of natural image and uncertain definition of saliency, salient edge detection is still a challenging topic until now.

Back to 1970’s, Rosenfeld et. al started to discuss the problem of edge detection [33]. Later, a lot of operators like Sobel operator, Robert operator, Prewitt operator and LoG operator were presented. Those traditional operators provide the intensity gradient of every pixel through studying its neighborhood region. Although they are effective to detect any image contrast, it is difficult for us to obtain the precise edge location through them. The nature of template based computing leads to the edge with the width of several pixels. Additionally, the threshold necessary for binarization is also an uncertain parameter, which cannot assure the robust results.

In 1986, Canny proposed an algorithm usually called Canny edge detector [34]. It includes four steps: noise reduction, intensity gradient computing, non-maximum suppression and thresholding with hysteresis. It can produce accurate edge with single-pixel width through checking whether the gradient magnitude assumes a local maximum in the gradient direction. Under the mechanism of thresholding with hysteresis, two thresholds, high and low, are utilized to trace edges through the image. Canny edge operator is popularly deemed as a milestone in edge detection. However, there are still

distracting edges coming from noise, texture or illumination.

Since the objects in the image are in different sizes, i.e, different spatial scales, edge detection on a specific scale cannot catch the features over the whole image [35]. On the other hand, the features standing out over most of the scales are more probable corresponding to the object boundary. In comparison, the fine details like texture or noise are eliminated in the higher scales [36]. Therefore edge analysis in multiple scales is necessary.

Most of the multi-scale edge detection methods firstly compute the single scale edge detection result at multiple scales and then combine them into a synthesis result through certain heuristics [37, 38]. In this way, both the good localization in small scales and the good detection of edges in higher scales can be made full use of to achieve the final result. On the other hand, some recent researches are based on estimating optimum scales for each local neighborhood in the image [35, 36, 39]. Lindeberg [36] proposed an edge strength function with X and Y coordinates and the continuous scale “t” as its parameters. Then the optimal scale at a point can be analyzed in a differential geometric view. Tabb et al. [35] design a vector field for each pixel based on the analysis of its neighborhood. The optimum neighborhood size, that is, the optimal scale of the local object, is estimated adaptively according to the specific homogeneity level within the local regions. Besides the above, another trend in multi-scale edge detection analysis is anisotropic diffusion [40]. It can not only suppress the responses from texture or noise but also prevent smoothing around the edge locations by an adaptive Gaussian smoothing. All the above multi-scale analysis based edge detector involves complex computing not only in producing multi-scale images but also in obtaining the optimal scale information.

Although edge detection in gray level image has been well studied, salient edge detection in color images has not received the same attention. Color images hold richer information with a color vector than gray level images with a scalar gray level. In this case, vector-valued image functions are utilized in color image processing. Most of the

current color edge detectors treat the color information simply as a feature vector [41-45]. There are mainly two classes of methods to combine the three color channels. Some approaches treat the color information as a whole feature vector same as any other feature vector and compute the gradient through certain vector norm, such as the Euclidean distance or the absolute norm [41]. Other approaches obtain the gradient value in each color channel separately and then combine them together through eigen analysis on the Hessian matrix resulting from the color vector's Jacobian Matrix [42]. But there has not been any research to explore the analysis on the color space at multiple scales. In this chapter, we mainly discuss color edge detection under multi-scale analysis.

We propose a multi-scale transition region based edge detector in this chapter. Firstly, the natural of transition region helps to find out the characteristics of the accurate location of the real edge pixels. Secondly, in order to simplify the multi-scale analysis, we propose the strategy of a series of local windows with variable sizes. The rest of this chapter is organized as follows: the proposed multi-scale transition region based edge detector is presented in Chapter 2.2. The experimental analysis is given in Chapter 2.3. At last, the conclusions are given in Chapter 2.4.

2.2 Multi-scale Transition Region Based Edge Detector

The proposed multi-scale transition region based edge detector combines the multiple scale analysis into the color spaces to obtain salient color edges. The working flowchart of it is illustrated in Figure 2.1.

The algorithm is designed according to the following three facts. Firstly, CIE Lu*v* is a more perceptually uniform color space than RGB color space. Secondly, we analyze the color distribution characteristics of the transition region. Thirdly, the multiple scale effect of a series of local windows with variable sizes and the multi-scale edge detection based on it are discussed. Based on the above facts, we can realize the proposed algorithm in three steps: color space transformation, single scale transition region

analysis and multi-scale color edge detection.

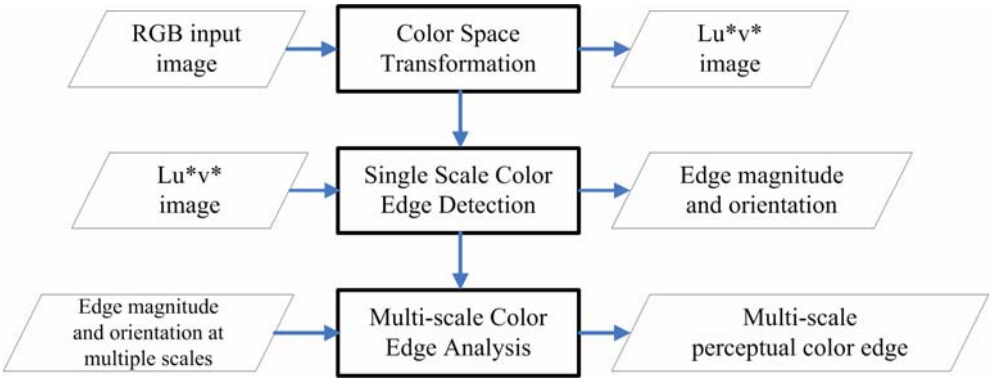


Figure 2.1 Flow chart of the proposed multi-scale perceptual color edge detection algorithm

2.2.1 Transition Region Analysis

Because the human eye has cone cells for short, middle and long wavelengths, there are usually three channels to describe the color images. In that case, a color image can be represented as a 3D vector field,

$$C(x,y) = (C_1(x,y), C_2(x,y), C_3(x,y)) \tag{2-1}$$

The three components $C_1(x,y), C_2(x,y), C_3(x,y)$ respectively hold different information. The numerical specification of color in terms of three coordinates or tristimulus values comes from the standard established by the International Commission on Illumination (CIE). We can list the most popular color spaces here and they are RGB, HIS, HSV, YUV or YCrCb, CIE XYZ, CIE LUV, CIE LAB and so on. Some of them are specially designed for certain application. For example, CMYK is designed for printing application and YUV and YCrCb for video and image compression schemes such as MPEG and JPEG [50].

RGB color model contains three additive primary colors, red, green and blue. The

decision to choose red, green and blue as primary colors is made according to the physiology of the human visual system. But with the three primary colors, it is still not enough to reproduce all colors. To address this problem, the CIE Lu*v* color space was created. The CIE Lu*v* color space was adopted by the CIE in 1976, as a simple version of CIE XYZ color space. They are more perceptual uniform scales to lightness compared with RGB color spaces since they are designed according to the direct measurements of human visual perception, that is, a small perturbation to a component value can be approximately equally perceptible of color differences. The psycho-visual principles prove that Euclidean distance in these perceptual uniform spaces is the best metric to estimate visual difference between colors [53]. In the viewpoint of vector computing, they are mathematically defined color spaces. Since we would like to utilize the Euclidean distance to measure the difference of two color vectors directly in the proposed method, CIE Lu*v* is better than the RGB color space.

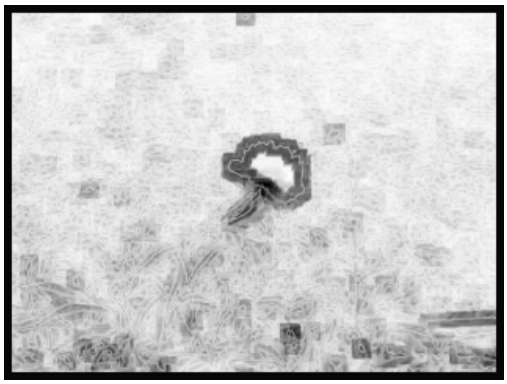
The proposed multi-scale perceptual color detection algorithm is performed in the CIE Lu*v* color space. Under the perceptual uniform properties, the local color contrast for a pixel can be computed through the following function.

$$C(i,j) = \max_{k \in N(i,j)} (d(p_{i,j}, p_k)) \tag{2-2}$$

where $C(i,j)$ is the color contrast at the pixel (i,j), $p_{i,j}$ is the color of the pixel (i,j), $N(i,j)$ is the neighborhood of pixel (i,j) and p_k is the color of one pixel in the neighborhood. $d(p_{i,j}, p_k)$ is the magnitude of the color difference between $p_{i,j}$ and p_k measured by Euclidean distance. The above color contrast, different from the usual template based gradient, is designed in order to fully capture the color difference in the spatial domain. All the pixels in the neighborhood can be viewed as a cluster of feature vectors in the color space. Then the compactness of this cluster can be well described by the radius of the minimum sphere which can contain all the data. And $C(i,j)$ is the radius of the minimum sphere. The edge strength is mainly studied in the feature space,

while the edge orientation is computed in the spatial domain.

(a)



(b) (c)

Figure 2.2 Comparison of two edge detectors (a) input image (b) edge map from the Sobel operator (c) edge map from the transition region based edge detector

Let us assume that $p_K(i_K, j_K)$ is the pixel who holds the maximal color difference with $p_{i,j}$. Then the local edge orientation at the pixel (i, j) is defined as follows.

$$\theta(i, j) = \arctan(\frac{j_K - j}{i_K - i}) \tag{2-3}$$

Since the pixel $p_K(i_K, j_K)$ captures the most different color with the pixel (i, j), the angle produced by the orient between it and the central pixel can be used to describe the

edge orient at the pixel (i, j). The value of $C(i, j)$ is normalized to $\overline{C}(i, j)$ in the range of [0, 255] to simplify the displaying. To easily observe the edges in the image, we show the edge detection result in the value of $255 - \overline{C}$. Thus the lower is the gray level; the higher is the magnitude of $C(i, j)$. The edge detection result is shown in Figure 2.2.

(a) (b)

Figure 2.3 Illustration of the “transition region” on the example image (a) the example image and a salient object marked by a black window (b) the enlarged image of the part in the marked window of (a).

But from Figure 2.2, we find out that the real edge pixels, especially the salient edge pixels on the object contour, hold lower value of $C(i, j)$. In comparison, the surrounding pixels of the precise edge pixel are with high $C(i, j)$ values. For simplicity, we call this dark region as the “edge candidate region”. At the same time, the pixels in the homogenous region all hold low value of $C(i, j)$. The edge map produced by the former template based gradient computing shows that there are continuously high gradient regions around the real edge pixel. But in the edge map produced by the $C(i, j)$ definition, there are always pixels with lower $C(i, j)$ in the edge candidate region. The

difference comes from the nature of digital image and also the different computing mechanism. As we all know, there is no absolutely sharp edge with only one pixel width in the natural image. That is to say, the perceptual edge in the image always shows itself as a transition region. If there are homogeneous regions with quite different colors besides the two sides of the transition region, the color changes gradually from one side to another side in the transition region. This phenomenon can be verified by the example image in Figure 2.3. We can see that in the enlarged part of the region around the boundary of the white flower, there is transition region between the white flower and the green grass.

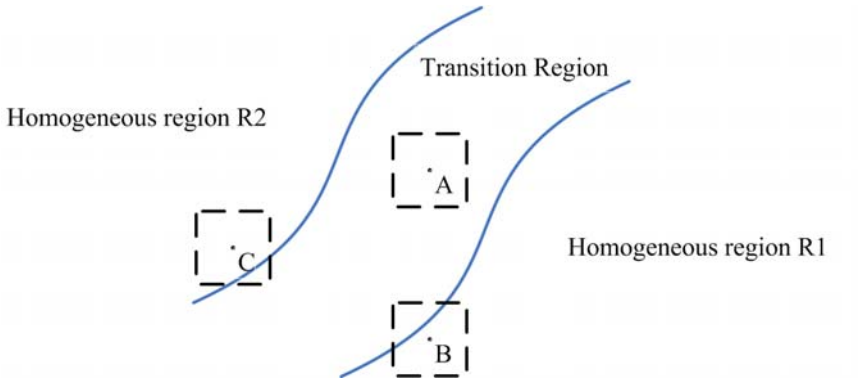


Figure 2.4 Illustration of “transition region”

A pixel in the transition region is not quite different from another, while a pixel in a homogeneous region but near to the transition region is quite different from the pixel in the transition region. Since $C(i, j)$ is designed to describe the most different color in the neighborhood, $C(i, j)$ of the pixel in the transition region is smaller than the pixel close to the transition region. In Figure 2.4, the value of $C(i_A, j_A)$ is smaller than $C(i_B, j_B)$ and $C(i_C, j_C)$. Based on the above analysis, we can derive that the most central pixels in

the transition region are more probable to hold lower $C(i, j)$ value than other pixels in the transition region. In this viewpoint, we can conclude that the pixels with lower $C(i, j)$ values in the edge candidate region are in fact the best edge location.

The definition of $C(i, j)$ makes it possible to more precisely detect edge pixel than the traditional template based gradient computing. For the template based gradient, the intensity or color differences between the central pixel and the pixels in the neighborhood are summed up to measure the edge magnitude. Under this computing, the distribution range of the gradient values is compressed. That is to say, the difference between the gradient values of transition region pixels and those of the pixels nearby the transition region is not as prominent as the difference under the $C(i, j)$ computing.

2.2.2 Multi-scale Edge Detection

A. Scale Space Theory in Computer Vision

Scale space theory originates from the necessity of describing the real world data whose components are usually at different scales. Specifically for computer vision, an unknown scene in the image may contain various interesting structures with different sizes, which poses a problem that which scale is most suitable to describe the scene. According to the physiological research, in the human visual system, the receptive field profiles recorder form the retina shows similar operations to scale space [46, 47]. This can explain the fact that we usually analyze a scene at a coarse scale at first sight and then start to explore the details.

(a) (b) (c)

Figure 2.5 Illustration of scale-space representation. (a) input image (b) scale space representation with $\sigma = 1.0$ (c) scale space representation with $\sigma = 4.0$

In order to capture all the information with various unknown scales, we have to consider the descriptions at multiple scales. In other words, a scale space description of the unknown data should include the representations over all the scales [45]. The multi-scale analysis has been applied in the edge detection for a long history [33, 37]. The spatial scales are realized by a series of Gaussian function with different variance values. As scale value increases, edge responses from the noise or texture are gradually smoothed out [48]. On the other hand, significant edges such as object boundary responses keep relatively stable across scales but the edge location becomes more and more biased from the precise location as the scale increases. This is the scale invariance property of natural image [49]. The images in Figure 2.5 illustrate the representation of the input image at multiple scales.

Based on the above scale space theory, we would like to extend the scale invariance based edge detection to the color image case. The multi-scale perceptual color edge detection algorithm is proposed. The objective is to make full use of the priority of the color information under the multiple scales. The RGB color images are firstly transformed to the CIELUV color space due to its good perceptual property [50]. Then a color edge gradient based measure is proposed, which can derive both the strength and angle of the gradient. The gradient measures obtained under different scales are

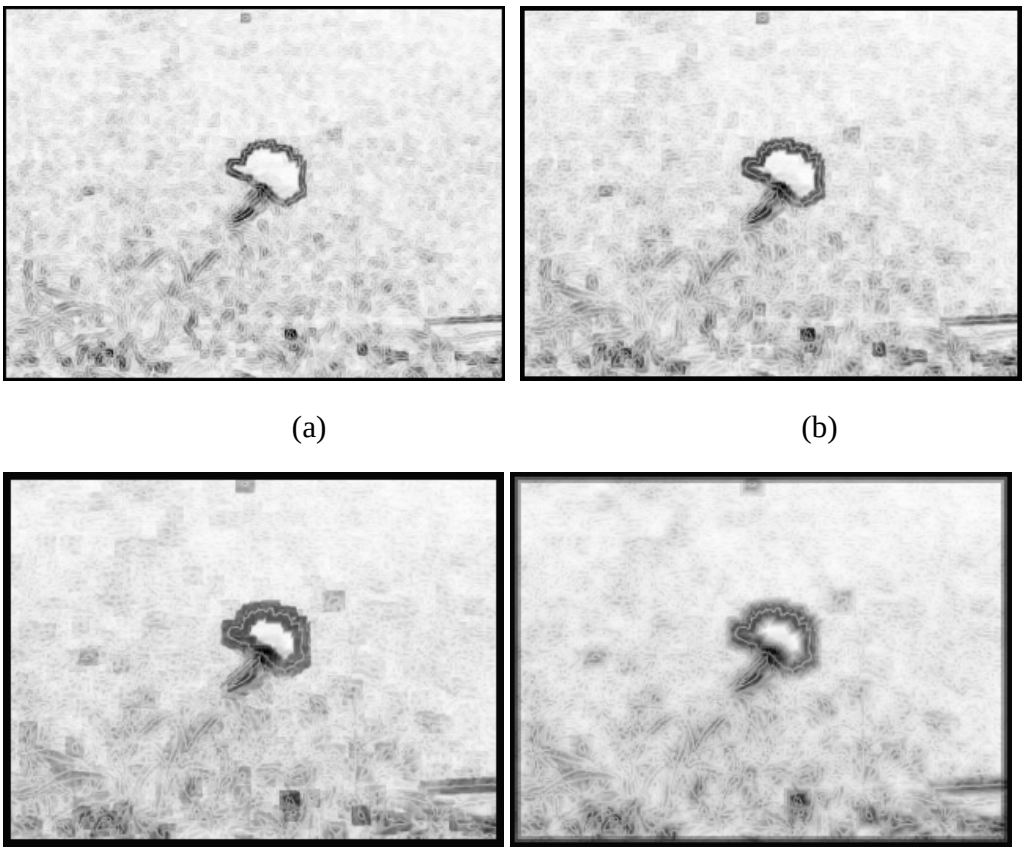
combined together in two ways: strength and direction. The scale invariant edges are extracted based on their stability both in strength and direction.

B. Multi-scale Transition Region Based Edge Detector

The proposed transition region based edge detector is built based on the study of the distribution of neighboring pixels in the color space. In some sense, it describes the texture in the neighborhood region. If the pixel (i, j) is in a homogeneous region, it can be expected that $C(i, j)$ is with small value. Otherwise, if the pixel is on the object boundary or in a textured region, $C(i, j)$ becomes larger. But the above $C(i, j)$ can only measure the color distribution in the neighborhood $N(i, j)$. In order to describe various color distributions around pixel (i, j), the neighborhood $N(i, j)$ should be carefully designed. Different color distributions, in other words, different textures need different sized windows since the window should be large enough to include the specific color distribution pattern [54]. Without the a priori knowledge about the size of textures existing in the image, $C(i, j)$ and $\theta(i, j)$ are calculated under a series of windows with different sizes, each of which aims at detecting specific texture feature under certain size. Through this way, the perceptual color edges are detected at multiple scales. The computation of $C(i, j)$ has no special requirement on the window's shape. Here the square window centered on (i, j) is chosen because of its simplicity of computing. Assume that the window series used for an image is $\{N_1, N_2, ...N_M\}$, where N_m is a $(2m+1)*(2m+1)$ window. The size of the largest window is determined by the following equation to assure the stable results.

$$M = \sqrt{\frac{\min(w, h)}{20}} \tag{2-4}$$

where w, h respectively describes the width and height of the input image. After deciding the window series, we can carry out the perceptual color edge by $C(i, j)$ and $\theta(i, j)$ at each scale. The edge detection result on each scale is shown in Figure 2.6.



(a) (b)

(c) (d)

Figure 2.6 Edge map from transition region based edge detector at various scales (a) the second scale (b) the third scale (c) the fifth scale (d) sum of all the three scales

From the edge detection results over multiple scales, we can find out that the "transition region" phenomenon exists over multiple scales as long as the window size is not larger than the size of the transition region. Though as the scale increases, the edge candidate region extends along the gradient orientation. This can be explained by the fact

that much more pixels close to the transition region are included to produce high value of $C(i, j)$. In comparison, the edge magnitude and also the location of the real edge pixel are relatively stable in the edge candidate region as the scale increases. We observe the $C(i, j)$ value of some sampled real edge pixels over scales and obtain the following table.

TABLE II-1

PERCEPUTAL EDGE MAGNITUDE OF EDGE AND NONEDGE PIXELS OVER SCALES

Pixel coordinate	Second Scale		Third Scale		Fifth Scale		Multi-Scale	
	Center	Surround	Center	Surround	Center	Surround	Center	Surround
(147,98)	111	119	105	151	97	146	104	122
(143, 103)	101	130	103	140	89	136	97	120
(192,117)	145	92	118	104	142	115	144	96

The first and the second pixels are real edge pixels in the transition region. The third pixel is in the edge candidate region, that is, near the transition region. From Table 2-1, we can find out that the $C(i, j)$ value of the transition region pixel is always lower than its surrounding pixels throughout various scales, while that of the edge candidate region is always higher than the surrounding. This assures that the location of the real edge pixels is stable across the scales. Through the stability of the real edge pixels both in location and edge magnitude, we can effectively detect the precise edge.

Based on the above analysis, we design the following multi-scale perceptual edge detector. Assume that $C(i, j, l)$ is the perceptual color edge value on the scale l. Firstly, the $C(i, j,1)$ is utilized to obtain the potential edge pixels, which are always stable in the higher scales. We call the obtained edge pixels as the set of mask. Secondly, the obtained potential edge pixels are further studied in the higher scale. Since the real edge pixels are relatively stable in the location pattern, that is, the center of the transition region with relatively small responses compared with its surrounding candidate region pixels. The following rule is designed to obtain the real edge pixels.

❖ As the size of the local window increases, the accuracy of the gradient orientation

decreases. Therefore, we make use of the $\theta(i, j,1)$ as the gradient orientation here.

- On the scale $l(l > 1)$, the real edge pixel should have smaller gradient magnitude than the 2 pixels which are closest to the gradient orientation $\theta(i, j,1)$ in the 3*3 neighbor region.
- To combine the detection results across multiple scales, voting rule is utilized to produce the final edge detection result. That is to say, for the multi-scale images of l layers, only the pixel with the voting number equal to or larger than the pre-defined threshold can be declared as the real pixel. Here we set the threshold as $l-1$.

2.3 Experimental Analysis

In this section, experiments are carried out to test the performance of the proposed multi-scale transition region based edge detector. Canny edge detector is declared as the best edge detection algorithm which can produce edges with good detection, good localization, good continuity and thinness. An eigen-analysis based color edge detector [52] proposed by Liu et al. is a fully automatic edge detection method with better performance than Canny edge detector. In this section, the proposed algorithm is compared with Canny edge detector and the eigen-analysis based color edge detector.

Similar to the eigen-analysis based color edge detector, the proposed method is also fully automatic without further setting of many parameters. Compared with the eigen-analysis based color edge detector, the proposed method can produce more connected edges and also keep the main edge information, which can be seen from the following detection results. We can find out that in the 1st, 3rd, 4th and 6th images, the eigen-analysis based color edge detector loses a lot of useful edge information due to the harsh thresholding on the gradient magnitude in its edge pixel verification step. But at the same time, the proposed method needs further improvement on the noise reduction or texture suppression since there are still a lot of responses from the texture region or noise.

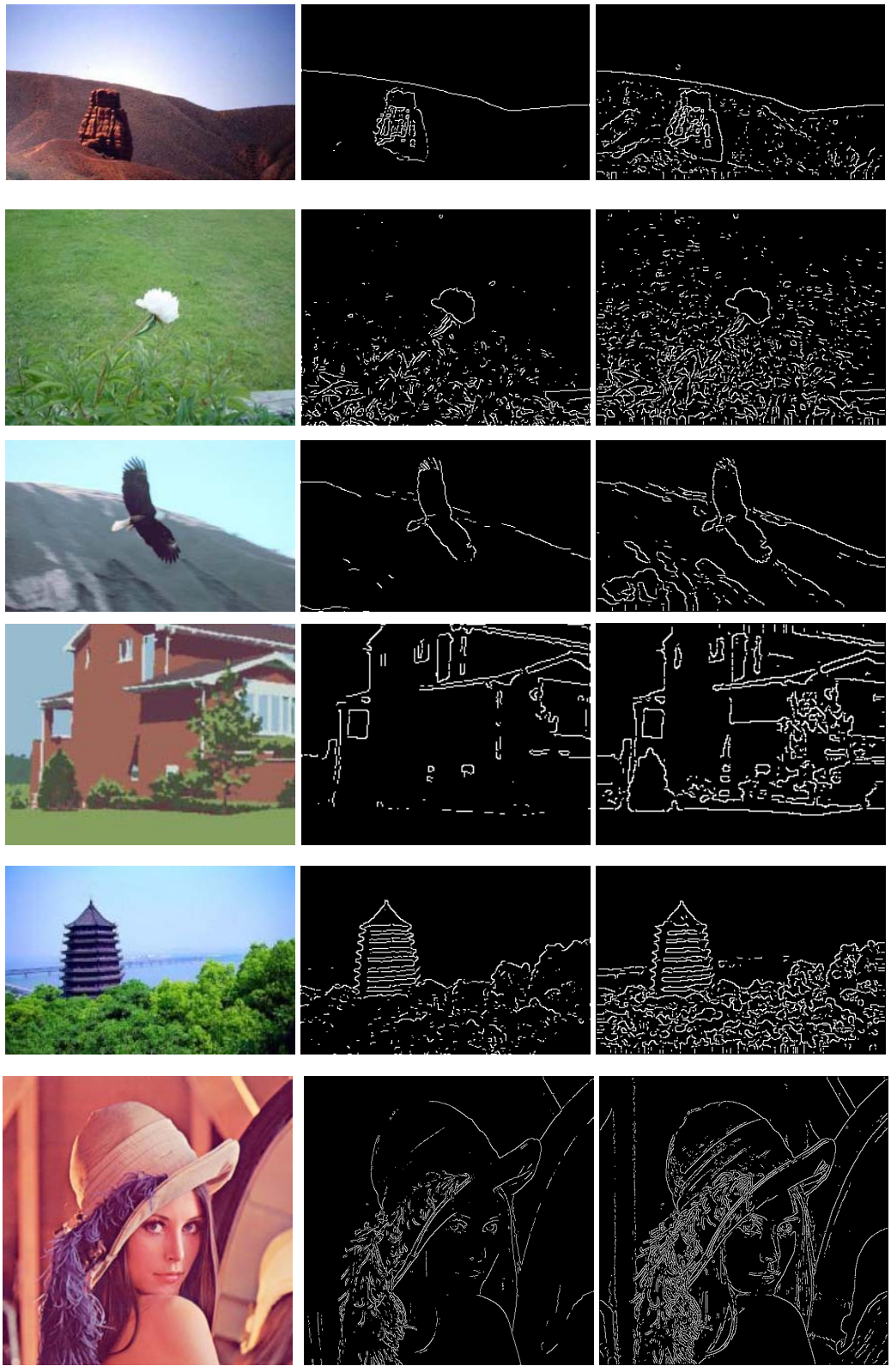


Figure 2.7 Illustration of the comparison of the proposed multi-scale transition region based edge detector and the eigen-analysis based edge detector. 1st column are the input images, the 2nd column are the edge detection results from the eigen-analysis based edge detector, and the 3rd column are the edge detection results from the proposed multi-scale transition region based edge detector. From top to bottom are the 1st, 2nd, 3rd,4th, 5th , and 6th experimental image.

Then we experiment the proposed multi-scale transition region based edge detector and the Canny edge detector on 20 natural images. To simplify the multi-scale computing, all the images are resized to 480*320 or 320*480. In Canny edge detector, when different pairs of thresholds for hysteresis thresholding are used, different edge maps are produced. We sum up the edge maps from 5 pairs of thresholds to be the final result of Canny edge detector. To quantitatively evaluate the experimental results, a fully automatic measure called the edge coherence measure proposed by Kitchen et al. is utilized [55]. It measures the local property of detected edge pixel through evaluating the continuation and thinness in its local neighborhoods.

$$E = \alpha C + (1-\alpha)T, \alpha \in [0,1] \tag{2-5}$$

$$C = L + R \tag{2-6}$$

where C and T respectively stands for the continuation measure and the thinness measure. The continuation measure further contains two parts L and R, that is, left and right continuation. The edge coherence measure is computed in a local 3* 3 windows. Larger continuation measure symbolizes good continuation of the edges along the edge gradient direction. And larger thinness measure shows less probability of producing edge with the width of more than one pixel. According to the experimental analysis from Kitchen et al., $\alpha = 0.8$ is the best parameter value and is taken in our experiments.

The edge coherence measure is calculated over each edge map from the proposed

algorithm and that from Canny edge detector. Then the average value for the 20 edge maps is taken as the final evaluation measure for two methods. Based on the experimental results, we find out that the average value for the proposed algorithm is 0.91 and that for Canny edge detector is 0.72. The larger edge coherence measure of the proposed algorithm shows that the proposed algorithm perform better than Canny edge detector in general case.

2.4 Conclusions

In this chapter, a multi-scale transition region based edge detector is proposed. The proposed method is designed according to the fact that for natural images, when the color changes from one distribution pattern to another, that is, from one texture to another texture, there is always a transition region with gradually changing pattern, but not ideally sharp change.

Based on this observation, the proposed multi-scale transition region based edge detector can detect the accurate location of the real edge pixels that are with relatively stable gradient distribution pattern of the transition region based gradient computing over scales. The transition region based gradient computing method can produce quite different edge map pattern compared with the former template based gradient computing method. The central pixels in the transition region are symbolized as the pixels with low responses surrounded by a group of pixels with large responses. Based on this pattern, the precise edge pixels can be well localized. What's more, there are usually various texture patterns in the image without a priori knowledge on their size. In order to capture all the texture patterns, we decide to combine the information from multiple scales through a series of windows with various window sizes. The information integration process over all the scales is carried out based on the stable characteristics of the edge magnitude distribution pattern surrounding the real edge pixels.

The proposed algorithm is evaluated through comparison with the well-known

Canny edge detector and also the fully automatic eigen-analysis based color edge detector. Experimental results show that the proposed method derives better detection result with better continuation and than both Canny edge detector and the eigen-analysis based color edge detector in general case.

3 Salient Contour Segment Detection

In this chapter, the problem of salient contour segment detection is mainly discussed. Because the contour segments coming from the noise or the texture interfere with both the efficiency and the accuracy of the following perceptual grouping process, it is necessary to assure that only salient contour segments are extracted before perceptual grouping to obtain the salient contours.

3.1 Background and Related Work

As a mid-level vision problem, perceptual grouping aims at linking the contour segments belonging to an object boundary together in a perceptual way under the complex backgrounds. Therefore, we discuss the salient contour detection as a two-stage problem. The first stage is to detect salient contour segment and the second is to group them into salient contours in the perceptual way. The contour segment detection is always carried out on the edge detection results. Salient edge detection results can prepare good data for salient contour segment extraction. The proposed multi-scale transition region based edge detector can satisfy this requirement. In the same sense, salient contour segments can facilitate the following perceptual grouping process.

The most popular contour segment used in current perceptual grouping algorithms is the straight line segment. There are also some algorithms using curve segments, especially when the object contour is not regular shape but random curve contour. But curve segment is hard to describe with many parameters and also the computing of good continuity between curve segments is rather complex. In comparison, salient line segment does not have such shortcomings. Straight line segments are simple to describe and also have a lot of structural information through their lengths and orients. Line segments can describe the boundary both in the regular polygonal shape and in the random curve shape.

What's more, line segments on the object contour are scale invariant features compared with those coming from noise or texture region. This is an important property for us to detect the salient contour segments. According to the above analysis, straight line segments are adopted in our salient contour detection algorithm.

Hough transform [31] is well-known for its robustness against discontinuity of line. It works on the basis of voting strategy. In image with complex scene, noise and details can lead to false peaks in Hough space. On the other hand, because the method works globally, extra processing is needed to get endpoints of lines. Aiming at the problems, original image is split into several independent sub-images to perform Hough transform respectively, called spatial decomposition [56]. The method has high computing load and is not feasible for image with cluttered backgrounds like rich texture. Another extension of the traditional Hough transform is the PPHT method[51]. But the voting based methods have the shortcoming of generating redundant line segments.

To detect salient contour segments, Grigorescu et al. [11] proposed a method called "surround suppression". By simulating the non-classical receptive field inhibition effect of the orientation selective neurons in the primary visual cortex, it can effectively suppress the texture responses and keep boundary. But the method is scale independent and cannot make full use of the multi-scale information. Another single scale method proposed by Yen et al. [12] modeled all the elements of a contour chain as a group of closely coupled neural oscillators. The oscillators synchronize to make the salient contour prominent and suppress the texture responses. All the above methods work under a center-surround interaction strategy. The affinity measure of every pair of neighboring element is modeled in different ways to compute the saliency measure. In order to address these problems, a refined Nevatia-Babu line detector is proposed to automatically detect reliable line segments based on the basic line patterns.

Compared with the above methods, multi-scale analysis can utilize more image information spreading over several scales since the best responses of sharp edges, diffuse

edges and shadow edges respectively distribute at different scales. In the multi-scale system, because of the brief and compact nature of high scale image, fitted line segments from it inherently contain the context information for the fitted lines from low scale. There has been a lot of work on multi-scale analysis of edge pixels to detect salient contours. For example, in the paper [57], the edges existing at a wide range of scales are favored and then the corresponding edge pixels are localized in the lower scales. Lindeberg [36] calculated the partial derivative of edge strength measures with respect to scale and then searched for the scales at which the edge response can reach local maximum. Nevertheless, there may be certain degree of information loss because of single scale selection. In comparison, multi-scale analysis can avoid this problem. Joshi et al. [48] extended the work of [11] to a multi-scale version with a log-Gabor filter bank. To cover the full spectrum range, many filters need to be built even if log-Gabor filters are introduced, which is often time consuming.

Different from the traditional methods, we perform the multi-scale analysis on the line segments instead of edge pixels based on scale invariance of natural image. A multi-scale image system is built to provide global information at different scales, which guides the processing of local line segments to derive salient straight line features. Compared with edge pixels, line segments hold much richer information of object contours in the form of its location, its direction and its endpoints. What's more, since all the line segments approximating real object contours have structural information of real object; it is easier to obtain salient closed contours in the following perceptual grouping process. Additionally, it is much easier to search for the corresponding element across scales for line segment than edge pixel through simple geometrical and spatial analysis. Although the method in [48] also performs multi-scale analysis, our method is different in that we utilize the sub-sampling to build an image pyramid to avoid the construction of many filters.

In summary, we propose a method to detect salient contours by studying the stability

of piecewise approximation of contours across scales, which can work in an efficient way with good performance. Compared with former methods, proposed scale invariance based line detector can work more efficiently and avoid false detections that are caused by noise and details.

3.2 Salient Line Segment Detection

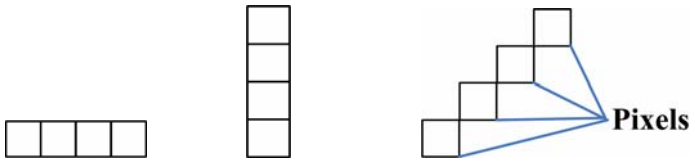
In this section, we mainly discuss the problem of detecting salient straight line segments. A refined Nevatia-Babu operator is firstly proposed to fulfill the basic straight line detection automatically [94]. Compared with the former operator, the improved operator can finish the task in only one step. Later, a multi-scale image system is built through down-sampling and smoothing. On this image system, a multi-scale line segment algorithm is implemented to suppress the lines coming from the texture or noise but also keep the salient lines on the object boundary.

3.2.1 Refined Nevatia-Babu Line Detector

In the following, we explain how to extract straight line segments on the single scale image. At first, we extract the piecewise linear approximation of contours from edge responses by the multi-scale perceptual color edge detector proposed in Chapter 2. Traditionally, the curved edge chains are firstly extracted by Nevatia-Babu edge operator [24] and then decomposed into several linear segments on high curvature points by curve partitioning algorithms such as a sequence finder [64]. We propose a revised version of Nevatia-Babu edge operator to extract linear edges in only one step. The proposed operator is designed based on the fact of the consecutive property of every edge pixel to its neighboring successive edge pixels on the line.

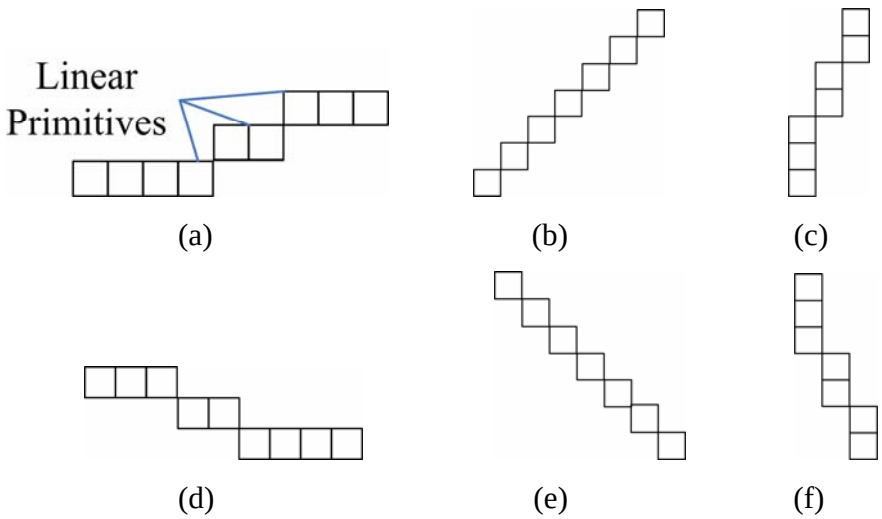
Lagunovsky et al. [65] proposed the notion of linear primitives, shown in Figure 3.1 (a), and stated that the linear primitives can represent straight lines in a binary image. In fact, we find that there are basically two other kinds of linear primitives, shown in Figure

3.2 (b) and (c). These linear primitives further constitute 6 kinds of linear primitive chains according to their slope values, depicted in Figure 3.2. The first three linear primitive chains can respectively describe the straight lines with acute angle smaller than, equal to or larger than $\pi/4$. Similarly, the last three can describe the lines with obtuse angle larger than, equal to or smaller than $3\pi/4$. Before edge tracing, we make different marks on edge pixels to symbolize that they belong to different linear primitive chain types. During the tracing process, connected linear primitives are linked to each other until a pixel with different mark is encountered. In this way, the tracer automatically stops marching forward whenever it encounters high curvature points. In the following, the details of the improved Nevatia-Babu operator are discussed. It mainly contains three steps, that is, pixel labeling, corner and isolated pixel labeling and pixel linking.



(a) (b) (c)

Figure 3.1 Illustration of three kinds of linear primitives (a) horizontal line primitive (b) vertical line primitive (c) sloped line primitive



(a) (b) (c)

(d) (e) (f)

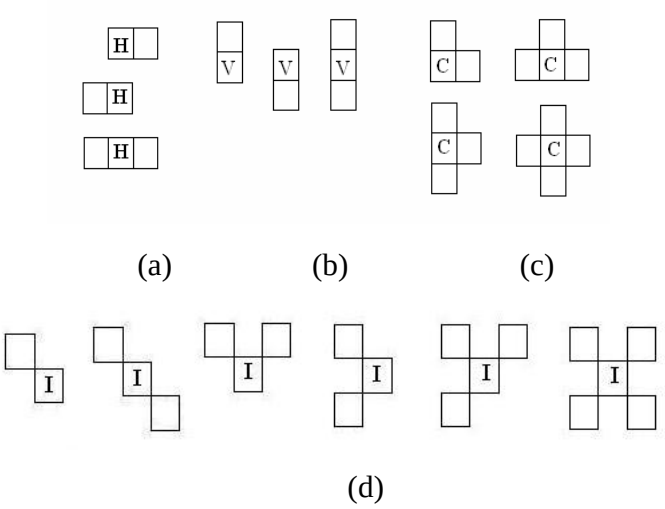
Figure 3.2 Illustration of six kinds of linear primitive chains (a, d) horizontal pattern (b, e)vertical pattern (c, f) sloped pattern

The first step is pixel labeling. Due to grid structure of digital image and one-pixel-width salient edge by the proposed edge detector, linking patterns of edge pixels on line structure can be summarized as the three basic patterns shown in Figure 3.2. According to the main orientation, we name them respectively as horizontal, vertical and sloped line pattern. Based on this, we design the following pixel labeling process to illustrate the distribution of edge pixels in the neighborhood of one pixel. With this illustration, we can easily decide the consecutive pixel for each edge pixel in the edge linking process. An edge pixel is marked with four types of labels according to the distribution of edge and non-edge pixels in its 1st neighborhood system. The definition is explained in Table 3-1 and illustrated in Figure 3.3.

TABLE III-1

DEFINITION OF FOUR TYPES OF PIXEL LABELS

Neighbor	Pixel Label Type			
	Horizontal	*Vertical*	*Corner*	*Isolated*
Vertical	Yes	No	Yes	No
Horizontal	No	Yes	Yes	No



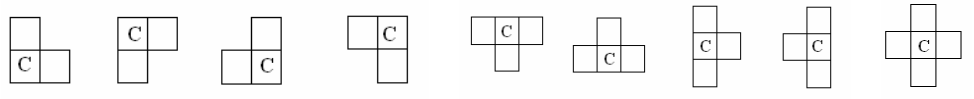
(a) (b) (c)

(d)

Figure 3.3 Illustration of four types of labels (a) horizontal label (b) vertical label (c) corner label (d) isolated label

The second step is corner and isolated pixel processing. In real edge image, not all of the line structures follow the three basic patterns because of noise. Corner and isolated pixel processing are devised to transform these general patterns into three basic patterns. In Figure 3.4, the distribution patterns of edge pixels in the 1st round neighborhood of a "corner" pixel are classified into three classes, respectively called "triple", "fourth" and "fifth" site cliques. For triple-site cliques, there are totally four patterns as "HCV", "HCC", "VCC" and "CCC", depicted in Figure 3.5. Line structures in Figure 3.5 (a-d) only contain "HCV" pattern. And they possess a main horizontal or vertical direction despite some disturbance. To transfer them into basic patterns and also keep main direction, we remove the "corner" pixel and re-label its neighbors to reach Figure 3.6(a-d). Line structure in Figure 3.5(e) contains "HCC", "VCC" and "CCC" patterns, showing a main sloped direction. By the same processing as above, we can transfer it into basic pattern shown in Figure 3.6(e). The processing in triple case can be extended to fourth and fifth site cliques.

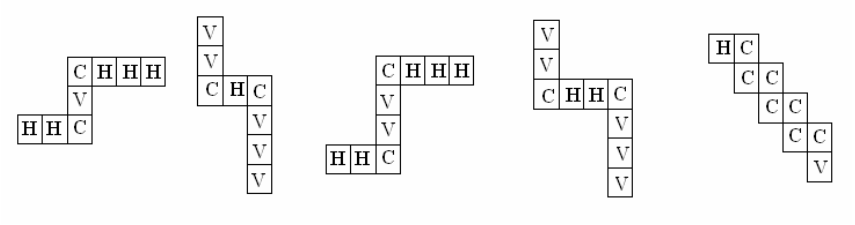
Now only the patterns in Figure 3.6(a-b) are not a basic type. "Isolated" pixel processing is necessary to be carried out. To keep the whole structure continuous, its new label can be determined according to its neighbors. In fact, not all of the existing line structures are transformed into basic patterns. Nevertheless, many of them can be transformed, which can be proved in the experimental results.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

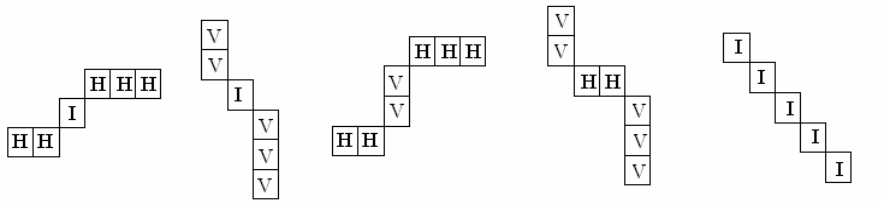
Figure 3.4 Illustration of the cliques of corner pixels' neighborhood (a-f) triple-site

cliques (g-h) fourth-site cliques (i) fifth-site cliques.



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

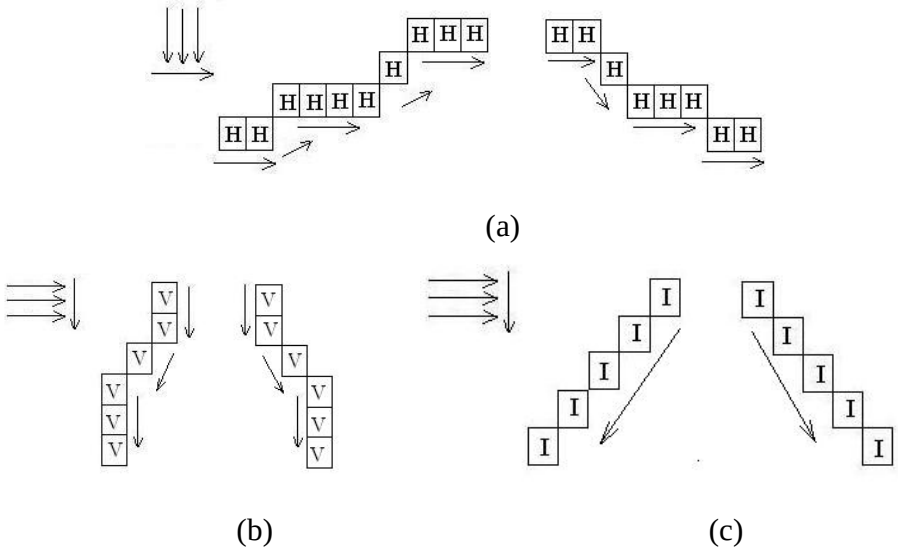
Figure 3.5 Illustration of corner pixel's neighborhood distribution patterns (a, b) "HCV" pattern (c, d) "HCV" pattern (e) "HCC" pattern, "VCC" pattern and "CCC" pattern



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

Figure 3.6 Illustration of processing result of the example patterns in Figure 3.5 after corner and isolated corner pixel processing (a-e) result respectively for the pattern in Figure 3.5 (a-e)

The third step is pixel linking. Now continuous chains in edge map are mostly transformed into three basic patterns. With the basic patterns, it is necessary for us to inspect only three neighbors of an edge pixel to get its successor. Afterwards, consecutive pixels can be linked under simple rules. The pixel inspecting rule and linking rule are visually described in Figure 3.7. Take horizontal line case for example, we search for "horizontal" labeled pixels in the image firstly from up to down then from left to right. For every such pixel, its right, right upper and right lower neighbors are studied one by one. If a neighbor holds the same label, it is decided as the successor.



(a)

(b) (c)

Figure 3.7 Illustration of linking pixels into line segments (a) horizontal line segment (b) vertical line segment (c) sloped line segment

3.2.2 Scale Invariance Based Line Detector

As we have discussed, straight line segments are used in our method as the basic contour segment. Besides the fact that straight line segments are easy to describe and also have rich structural information to describe various shaped boundary, line segments have another very important property, that is, scale invariance. This property makes line segments applicable in multi-scale contour segment detection problem. In this section, a scale invariance based line detector is proposed [95, 96].

A. Scale invariance of line segment

From the perceptual principle of Human Visual System (HVS), we know that significant features of object shape are relatively stable across scales. An example of this is the scale-invariant feature transform (SIFT) key-points [66], which are widely applied in shape matching and object recognition. Visual information along the contour is

concentrated on the points with high magnitude of curvature especially those with negative curvature [67]. The curve segments between those key-points, as the complementary elements of the key-points, are also scale invariant. The curve segments are further approximated by several line segments to simplify the computing. This approximation does not cause heavy loss of information during salient contour detection.

Contours are represented as piecewise line segment approximation and saliency of line segments is computed by combining multi-scale information. The scale invariance of the piecewise approximated contour can be validated in several aspects. From biological viewpoint, we can catch the coarse shape of an object in some distance away. When we approach the object, its boundary is becoming clearer and more details are prominent to reflect the sharp changes. During this process, our impression of the contour shape has been gradually intensified as the spatial scale becomes smaller. In theory, empirical distribution of line segments between high curvature points follows the power law [49], which implies the scale invariance characteristics. In summary, object contours can be represented as piecewise line segment approximation and saliency of contours can be evaluated by combining multi-scale information of these approximated line segments.

To utilize multi-scale information, we firstly build an image pyramid by repeatedly smoothing and sub-sampling the image. Then the salient line segments from real object contours are strengthened and those from texture and noise responses are suppressed through studying their stability across scales. By this way, we can extract salient line segments from the background scene.

B. Image Pyramid Construction

Here we utilize a computing and memory efficient way to build the multi-scale image pyramid. Burt [68] has proposed an efficient way by repeatedly smoothing the image with a Gaussian filter and then sub-sampling the image by simply averaging over a k*k pixel region. In our method, we take the variance value σ as 1.0 and the scaling factor as

2*2. On one hand, image decimation can accelerate the algorithm and also save memories on storing the multi-scale image representation. On the other hand, Konishi et al. [69] proved that after image decimation there is almost no information loss, through comparing the Chernoff information on edge localization of the decimated image with that of the un-decimated image smoothed by a filter with the same effective scale. This conclusion assures reliable edge detection result from the image pyramid. Because of the smoothing and sub-sampling process, the effect of noise and low contrast is gradually reduced as the scale increases. In contrast, the line segments form real object contours are stable across all the scales.

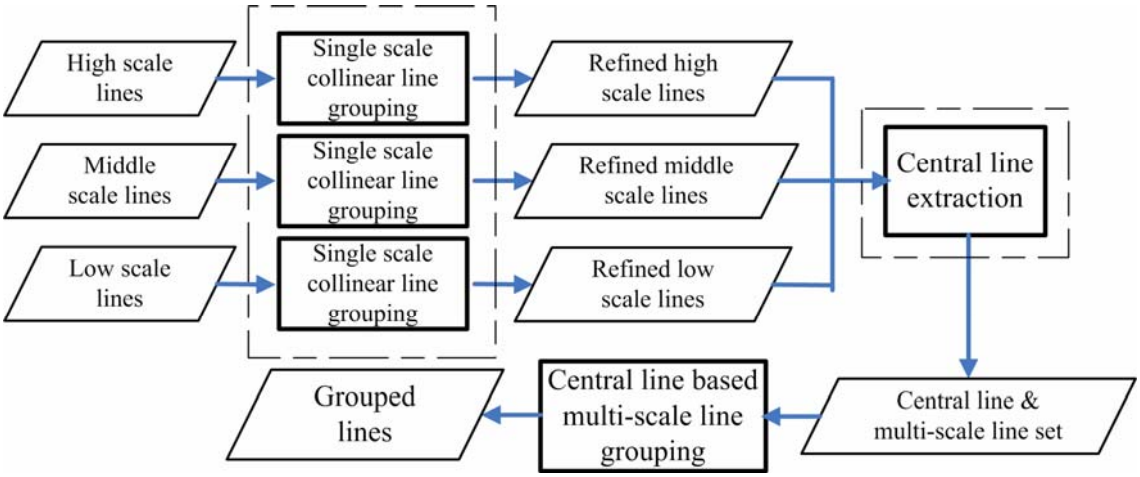


Figure 3.8 Flowchart of the multi-scale line segment detection process

C. Salient Line Segment Extraction on Image Pyramid

Based on the above analysis, the problem of salient line segment detection can be fulfilled through multi-scale analysis. Firstly, line segments are detected at each scale of the image pyramid. Then a collinear line grouping (CLG) [70] process is implemented on extracted line segments at each scale to improve the connectivity and the saliency. Finally, the stability of line segments across scales is studied to extract only the salient ones on the object contours. The flowchart of this process is illustrated in Figure 3.8.

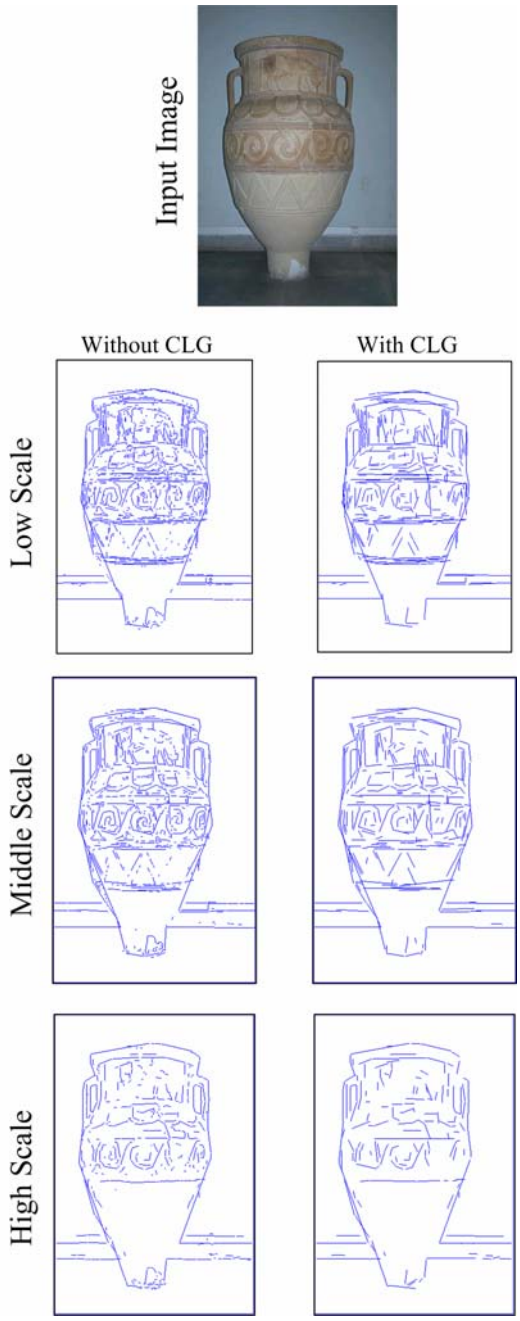


Figure 3.9 Comparison of line segment maps with and without Collinear Line Grouping (CLG) process at different different scales. The first row is the input image. The second, third and fourth row are line segment maps respectively at the low scale, middle scale and high scale. From the second to the fourth row, the first column is the line segment map

without CLG processing and the second column is the line segment map with CLG processing.

After detecting line segments at each scale, a collinear line grouping process is implemented respectively on the detected line segments to further improve the connectivity of contour approximation. For polygonal shaped objects, the contour is directly composed of several line segments. The fragmentation of line segments are lessened through collinear line grouping. For curve shaped objects, approximating line segments of an object contour are separated by high curvature points. The collinear line grouping improve the connectivity of line segments in the smooth curve part but do not influence the line segments in the high curvature curve part. At the same time, through the collinear line grouping, we can remove the short line segments not collinear to any other line segments. These kinds of line segments usually are in the texture region. We can summarize that both the connectivity and the saliency of the line segment result are improved after the collinear line grouping. Experimental results shown in Figure 3.9 can prove this.

In the following, we explain in detail how to extract salient contours by analyze the piecewise line segment approximation of contours across multiple scales. Note that line segments extracted on higher scale are all projected back to the original image space by zooming with the same scaling factor as image decimation.

Let us firstly study the relation among the approximated contours of an object across scales. According to the above analysis, line segments from real object contours are stable across scales. For every such line segment, it is highly probable to find another line segment in different scale, which is close to it and with similar direction, called the corresponding line segment. But the line segments from background texture do not have such characteristics. As the scale increases, the responses of noise and texture are gradually suppressed while the responses of the real objects remain. Hence the

approximating line segments in the highest scale construct the general shape of the real objects. In comparison, the approximating line segments in the lower scales are closer to the location of real objects. To utilize both the general shape information and accurate location information, the following process is designed to extract salient contours.

Firstly, for every line segment in the highest scale, we search for its corresponding lines in the lower scales and collect them into a line set. Then a new line segment is constructed based on the information of both the lower scale corresponding line set and the highest scale line segment. If the lower scale corresponding line set is empty for a highest scale line segment, it is deemed as unreliable and is removed.

We mainly depend on the spatial distance and the angle difference between two line segments to determine whether they are corresponding to each other or not. We firstly define $S(l_l)$ and $S(l_h)$ to be respectively the line set in a low scale and a high scale. $N(l_h)$ is defined as the corresponding line of l_h. In our algorithm, a lower scale line segment l_l is considered as a neighbor of a higher scale line segment l_h if and only if it satisfies the following three requirements.

- Firstly, one of the endpoints of l_l is close enough to either of the endpoints of l_h.
- Secondly, the orientation difference between l_h and l_l should not be too large.
- Thirdly, the length of l_l should not be too small to avoid the influence of line segments from noise or background texture.

Based on the requirements, we define the scaled distance between endpoints $P(l_h,l_l)$, the orientation difference $O(l_h,l_l)$ and the voting length $D(l_h,l_l)$ as follows.

To include all the possible corresponding line segments into the line set, if one of the endpoints of l_l is close to one of the endpoints of l_h, it is accepted as one candidate. So the following proximity line segment searching process is implemented twice respectively centering on each endpoint of l_h. Here we define the two endpoints of l_l as

A_{l_l}, B_{l_l}, then the endpoints of l_h are A_{l_h}, B_{l_h}. And we take proximity line segment searching process on A_{l_h} as example.

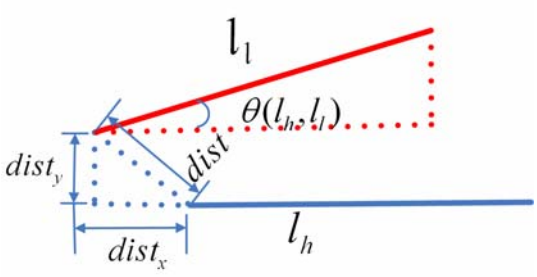


Figure 3.10 Sketch Map of Vector Decomposition

$$dist = \min(\overline{A_{l_h} A_{l_l}}, \overline{A_{l_h} B_{l_l}}) \tag{3-1}$$

$$dist = \sqrt{dist_x^2 + dist_y^2} \tag{3-2}$$

$$P(l_h, l_l) = \sqrt{(\frac{dist_x}{scale_x})^2 + (\frac{dist_y}{scale_y})^2} \tag{3-3}$$

$$dist_x \le scale_x, dist_y \le scale_y \tag{3-4}$$

Firstly, one of the two endpoints closer to A_{l_h} can be determined by their Euclidean distances by Equation 3-1, assuming it as A_{l_l} Then we decompose the vector between those two endpoints into two components on the directions parallel and perpendicular to the direction of the line segment l_h, illustrated in Figure 3.10. The norm of the two decomposed vectors are respectively $dist_x, dist_y$, which are scaled by different parameters $scale_x$ and $scale_y$. Note that Equation 3-4 assures the spatial proximity constraint on the two line segments.

$$scale_x = \lambda L_h, 0 < \lambda \le 1 \tag{3-5}$$

$$scale_y = perpend\_th \tag{3-6}$$

From Equation 3-5, we can find that $scale_x$ is set to be related with the length of the line segment l_h to confine the possible bias on its direction. The weight λ is set as 0.5 in experiment. $scale_y$ is a threshold on the perpendicular distance between the two line segments, set as 5 here.

The orientation difference $O(l_h, l_l)$ is defined as the sine value of $\theta(l_h, l_l)$, the angle between the two line segments. It is also illustrated in Figure 3.10. A threshold is set to assure the two line segments satisfy the orientation similarity.

$$O(l_h, l_l) = \left|\sin(\theta(l_h, l_l))\right| \tag{3-7}$$

$$O(l_h, l_l) \le orient\_th \tag{3-8}$$

The voting length $D(l_h, l_l)$ is defined as Equation 3-9. The length of l_l is weighted by considering the effect of the scaled distance $P(l_h, l_l)$. If two line segments are closer to each other, the original length is much more weighted.

$$D(l_h, l_l) = L_l(1 - P(l_h, l_l)) \tag{3-9}$$

According to Equation 3-4 and Equation 3-8, we can select all the line segments corresponding to l_l to form $N(l_h)$, the corresponding line set.

The corresponding line set of a highest scale line l_h reflects its stability across scales. If the line set of l_h is empty, we can make the decision that it does not belong to any real object contour and remove it. For the non-empty corresponding line set, all the line segments in it are ordered according to their voting length $D(l_h, l_l)$. The larger voting length means more importance in constructing new line segments. In this way, long line segments close to l_h are much more favored. After ordering, the first line segment of the line set is deemed as the most important one. The new line segment is constructed by

determining one point on it, its direction and its range.

- To assure accurate localization of contour, the middle point of the first line segment is determined as one point on the newly constructed line segment l_h^{new}.
- Then the direction of l_h^{new} is calculated by averaging the angles of all the line segments in the line set and the line segment l_l. Since there is an orientation threshold to assure that all the line segments are not much biased from l_h, the averaging orientation is also in the acceptance range. Then a straight line containing l_h^{new} can be determined with one point and a direction.
- To obtain the two endpoints of l_h^{new}, all the line segments in the corresponding line set and the highest scale line segment l_l are projected to the line containing l_h^{new}. Then the furthest two projecting points are selected as the endpoints.

The newly constructed line segment contains both the information of the corresponding line segments from low scale and that from high scale. Its existence reflects the stability of the line segments belonging to the real object contours. On one hand, if the corresponding line set $N(l_h)$ is empty, l_h is not stable in lower scale. On the other hand, if a lower scale line segment l_l is not selected as the corresponding line, l_l is not stable in higher scale. In both cases, they are deemed as useless line segments. Only the newly constructed line segments are output as the approximation of salient contours.

3.3 Experimental Analysis

In order to illustrate the performance of the proposed method, we make the following experiments. Firstly, the proposed scale invariance based line detector is compared with the PPHT method. Secondly, to verify the scale invariance analysis on suppression of the noise or texture responses, we make further analysis on the single scale and multi-scale

line detector by the experimental results in the building contour analysis of aerial image.

A. Comparison on PPHT Method and Proposed Method

Considering the voting nature of the PPHT method, it has high possibility of generating redundant line segments. At the same time, the voting based method always has high computing load and therefore spends lots of time in processing. Compared with PPHT method, the proposed scale invariance based line detector performs line extraction based on the scale invariance property of the object contours and also the connectivity property of the edge pixels on the basic line patterns. Therefore, we can expect less redundancy of the line detection result because of the one to one correspondence between the connective edge pixels and the line pattern. At the same time, the direct linking of these edge pixels is rather a simple process and therefore has low computing complexity.

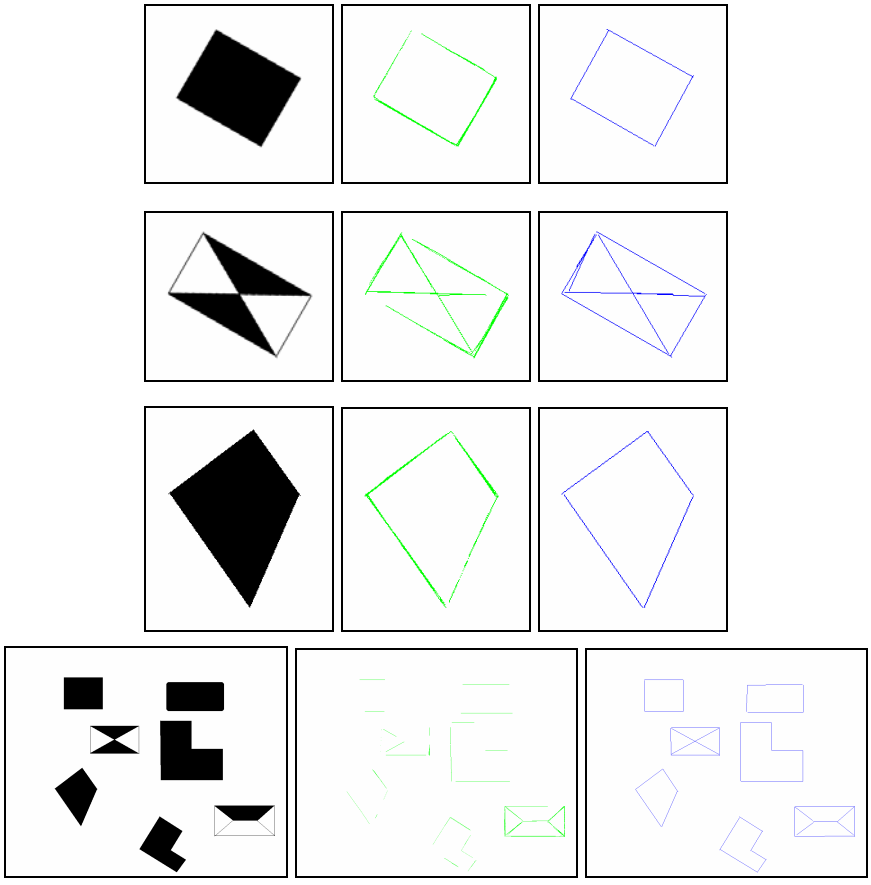


Figure 3.11 Illustration of the comparison of the proposed scale invariance based line detector and the PPHT method. The 1st column are the input images, the 2nd column are the line detection results from the PPHT method, and the 3rd column are the line detection results from the proposed scale invariance based line detector. From top to bottom are the 1st, 2nd, 3rd, and 4th experimental image.

TABLE III-2

NUMBER OF DETECTED LINE SEGMENTS OF PPHT AND PROPOSED METHOD AND THE GROUND TRUTH LINE NUMBER

	1	*2*	*3*	*4*
PPHT	85	112	97	118
Proposed method	4	10	4	39
Ground Truth	4	6	4	39

TABLE III-3

COMPARISON OF REDUNDANCY LEVEL OF LINE DETECTION RESULTS OF PPHT AND PROPOSED METHOD

	1	*2*	*3*	*4*
PPHT	20.25	17.67	23.25	2.02
Proposed method	0	0.67	0	0

TABLE III-4

COMPARISON OF PROCESSING TIME OF PPHT AND PROPOSED METHOD (SECONDS)

	1	*2*	*3*	*4*
PPHT	2	7	2	80
Proposed method	1	1	1	3

We perform the PPHT method and the proposed scale invariance based line detector on four synthetic images that contains line segments. The detection results from the two methods are compared in Figure 3.11. At the same time, the number of the detected line segments and also the redundancy level of the line detection results of the PPHT method and the proposed method are firstly compared, which are shown in the Table 3-2 and Table 3-3. Then the comparison result of their processing time is shown in Table 3-4. From the line detection result shown in Figure 3.11, we can find out that the PPHT method sometimes detect several line segments overlapping with each other for one ground truth line segment. At the same time, the detected line segments from PPHT are not with good connectivity. Some of them lose part of the line segments in the results of

the 1st, 2nd and 3rd example images. Others lose almost total line segments, for example, the line segments in the 4th example image. In comparison, the proposed scale invariance based line detector can detect all the line segments in good connectivity and with almost no repeating detection. Only in the 2nd example image there are repeating detections of one line.

In Table 3-2, the number of detected line segments by PPHT and the proposed line detector are shown. Compared with the ground truth, we can find out that results from PPHT have high redundancy for every example image. But the results from the proposed method have almost no redundancy for all the example images except the 2nd image. In addition, the redundancy level measure is computed based on the following equation.

$$r = (N - N_0) / N_0 \tag{3-10}$$

where N_0 is the number of the ground truth line segments and N is the number of the detected line segments. The results from PPHT method have high redundancy level from 2.02 to 23.25, while the results from the proposed method have very low redundancy level in the 2nd example image as 0.67 and no redundancy in other images.

What's more, the comparison on the processing time of the PPHT method and the proposed method shows that the proposed method has much better efficiency. Note that as the number of line segments increases in the image, the superiority of the proposed method is more prominent. This property assures the good computing efficiency of the proposed method even for the large sized image or very complex image. The proposed method can finish the line detection in 3 seconds for an image including 39 line segments while the PPHT spends 80 seconds. Compared with the proposed method, the PPHT method spends at least 2 times and at most 26 times of the processing time of the proposed method.

B. Single Scale and Multi-scale Analysis

Considering the application of the contour detection to the building extraction application,

we make further study on the performance of the proposed method in the aerial image. We would like to find out the performance of the proposed method under the condition that the number of line segments in the aerial image are relatively huge compared with the above synthetic images. At the same time, we also make verification on the necessity of multi-scale analysis to suppress the responses from noise or texture and assure the detection of salient line segments.

The aerial images containing many man-made objects such as the buildings have rich information of line segments. Therefore, the aerial images are good experimental images for evaluation of line segment detection algorithm. In this experiment, we utilize five aerial images, respectively with the size of 500*461, 951*741, 2500*2500, 2500*3000, 3600*3800.

TABLE III-5

COMPARISON ON PROCESSING TIME BETWEEN SINGLE SCALE AND MULTI-SCALE LINE SEGMENT DETECTION METHODS (SECONDS)

Algorithm	Image Size				
	*500*461*	*951*741*	*2500*2500*	*2500*3000*	*3600*3800*
Single scale	1	2	14	19	20
Multi-scale	2	6	103	159	237

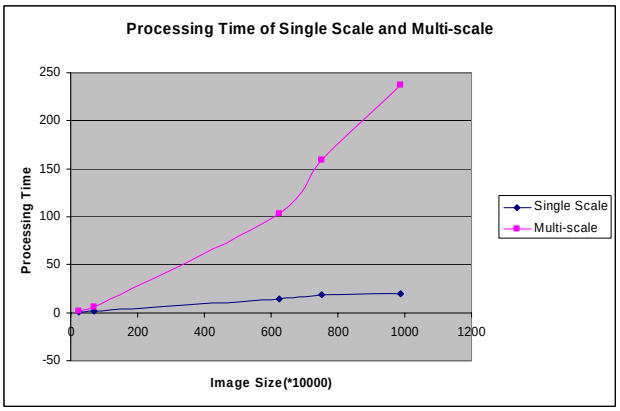


Figure 3.12 Comparison on the processing time between single scale and multi-scale contour segment detection algorithms

The main computing burden of multi-scale method, different from the single scale method, lies in the step of transferring information across levels. In order to compare the computing efficiency of the proposed two methods, we design the following experiments. The processing time of five testing aerial images under single scale and that under multi-scale cases are provided in Table 3-5 and illustrated in Figure 3.12.

From the comparison, we can see that both the computation complexity of the proposed single scale and that of the multi-scale contour segment detection algorithm are proportional to image size. In Hough transform, the angle interval should be small enough for accuracy. In that case, before we can get the accumulating peaks, every pixel should vote many times. In our method, the single scale method takes only three times traversing. And then, in multi-scale method, the processing element changes from pixels to line segments, whose magnitude is reduced a lot. From the above analysis, it is clear that our method is more efficiency than voting method like Hough transform.

We select one testing image to compare the performance of the single scale and multi-scale line detection algorithm and show it in Figure 3.13. The blue lines in Figure 3.13(b, c) are respectively extracted by single and multi-scale method. From them we can observe that multi-scale method can suppress the line segments coming from the texture region effectively and also keep the salient line segments on the house contours. What's more, detected line segments from multi-scale method are with good connectivity, high accuracy in position and orientation. At the same time, the corners of some intersected lines can be reconstructed. To further evaluate the improvement from single to multiple scales, we design the following two measures.

- The Percentage of Removed Lines (PRL) to the total lines on the highest level is to measure the effectiveness of removing noise and textured lines.
- Mean value of integrity of every integral line produced by collinear grouping is to show the improving connectivity of lines. Integrity is defined as the reciprocal of the number of collinear lines on top level.

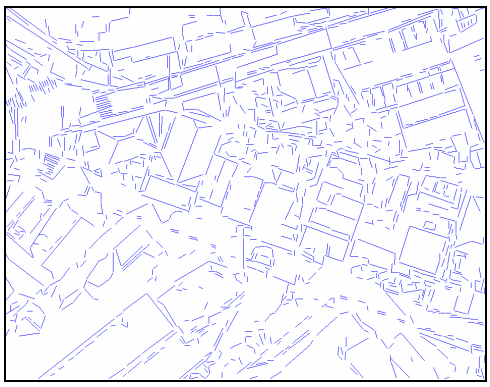
From Table 3-6, we can find that for all the images, at least 30 percent of textured lines are removed and especially 67 percent for the 3rd testing image are achieved. And the integrity measure value for images of different sizes and contents are mostly around 0.5. This shows that salient lines are averagely broken into two segments in the result from the single scale method.

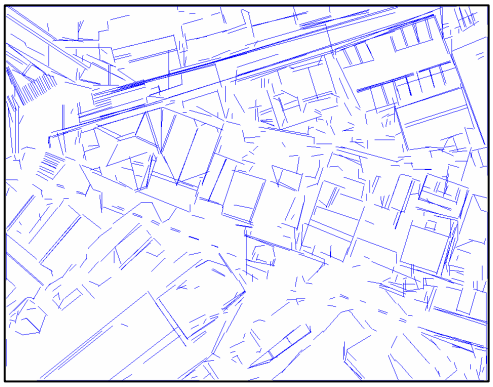
TABLE III-6

EVALUATION ON THE IMPROVEMENT FROM SINGLE SCALE TO MULTI-SCALE LINE SEGMENT DETECTION METHOD

Evaluation measures	Image Size				
	*500*461*	*951*741*	*2500*2500*	*2500*3000*	*3600*3800*
PRL	34.54%	58.55%	67.37%	53.87%	59.79%
Multi-scale	0.39	0.48	0.50	0.47	0.46

(a)





(b) (c)

Figure 3.13 Comparison on performance between single scale and multi-scale contour segment detection algorithms (a) input image (b, c) line segments extracted respectively

by the single scale and multi-scale contour segment detection algorithm, shown in blue lines

According to the above analysis, we can conclude that although multi-scale contour line segment detection algorithm is a little more time consuming than the single scale algorithm, the multi-scale algorithm can produce salient line segments more robustly in the cluttered background.

3.4 Conclusions

In this chapter, a scale invariance based line detector is proposed based on a refined Nevatia-Babu line detector. Experimental results show that the proposed scale invariance based line detector can obtain line detection results with less redundancy in less processing time compared with the PPHT method. We also find out that the proposed method perform better than the single scale based line detector in the building detection analysis of aerial image, with fewer responses from noise or texture. This can prepare good data for the following contour detection step.

According to the consecutive property of the edge pixels on a line, we propose the refined Nevatia-Babu line detector to finish the basic line extraction automatically in only one step. The proposed line detector mainly includes three steps, that is, pixel labeling, corner and isolated pixels processing and pixel linking. Due to the simplicity of the computing, the refined line detector can efficiently detect the line segments in the image even for the image in large size. On the contrary, the traditional Nevatia-Babu line detector has to compute the curvature in order to detect the turning points on the curve traced in the first step and then cut the traced curve into segments on the turning points. Experimental results show that the refined line detector is rather efficient and can finish the line segment detection in 20 seconds for an aerial image with size of 3600*3800.

Although can detect all the possible line segments in the image, there are some line

segments coming from the texture region detected at the same time. In order to suppress the responses from the texture region and also keep the responses from the object contour, a multi-scale extension of the refined Nevatia-Babu line detector is designed. A multi-scale image pyramid is firstly built by repeatedly smoothing and sub-sampling the input image. Then the refined Nevtia-Babu line detector is performed at each image scale to extract basic line segments, which are further processed by collinear line grouping to improve the continuity of the line segments and suppress the responses from the texture region. According to the scale invariance property, line segments on the object contour are stable across the scales. Then salient line segments can be easily extracted through combining the information from multiple scales. Compared with the single scale detector, the multi-scale is a little more time-consuming but can produce better result. Large percentage of the line segments from texture region is suppressed and the extracted line segments are with better continuity.

Compared with the PPHT method, the proposed method can obtain better line detection results. According to the experimental results on the synthetic images, the results from PPHT method have high redundancy level from 2.02 to 23.25, while the proposed method have no redundancy in 3 example images and only 0.67 redundancy level in one example image. The comparison on the processing time of the two methods also shows that the proposed method works more efficiently than PPHT methods, especially for the case when the image includes many line segments.

4 Salient Object Contour Detection

In this chapter, perceptual grouping based salient contour detection is mainly discussed. Contour is very important feature for object detection and recognition. As we have stated before, object contour can facilitate the process of object extraction and description.

4.1 Background and Related Work

For the computer vision system, salient contours are extracted by linking contour elements on the same object boundary together into a closed curve. By this means, scattered contour elements are also classified into different clusters, each of which represents an object contour. In this sense, the grouping of contour elements involves two parts: clustering and linking. Clustering process needs to check the closeness of the contour elements between each other, while linking process has to decide the consecutive order of the contour elements in the same cluster. Since the contour segments are grouped based on the perceptual rules derived from the physiology theory, this grouping process is often called as “perceptual grouping” or sometimes as “contour integration”.

Researches on psychological, physiological, psychophysical and anatomical prove that perceptual grouping can be realized by the mediating of primary visual cortex, the first stage in visual cortical processing of the human visual system. The primary visual cortex involves the contextual influences to measure perceptual contour saliency in the process of contour integration [58]. Many methods for contour extraction in unconstrained backgrounds have been published in the past decades. There are two kinds of most widely used features. One is the principles of perceptual grouping such as proximity, curvilinear property, convexity and closure. Another is the image information such as the high gradient on the contour and the homogeneity inside the contour.

Perceptual grouping is one of the most popular ideas in the field of contour

extraction [19, 21, 26, 59, 60]. Besides the local features of proximity, curvilinear property and convexity, the global feature of contour closure can greatly help to improve the contour extraction [59]. What's more, the method proposed in [26] firstly introduced the region homogeneity into the convex grouping, which realizes the combination of both region and boundary information. Perceptual grouping is carried out mainly on the perceptual rules called as "Gestalt laws", which declare that the contour elements belonging to the same contour should satisfy proximity, similarity, closure, symmetry, continuity, collinearity, cocircularity and parallelism [15, 16]. In summary, when smoothly continuous curve can be formed along a path through certain contour elements, it can be declared as a salient visual contour which can be extracted directly from the background.

The proper definition of contour saliency measure is very important to grasp all the salient contours in the scene and not to include any noise or texture responses. The term of "saliency" was firstly used by Shashua and Ullman [61] in the literature. They defined a particular measure quantified the properties of smoothness and length. The saliency of an element is iteratively updated by the saliencies of all neighboring elements in its local network. Later this method was extended by Alter and Basri [62] to address the problem that the maximal saliency measure may not be consistent to the most visually salient curve because of the local and greedy nature of the measure. Guy and Medioni [29] proposed a non-iterative method denoted as "tensor voting". After the scale of neighborhood is determined, neighboring elements communicate by voting for each other and then all votes for an element are collected to infer its predominant orientation. Some saliency measures are modeled explicitly under the psychology theory studying the human vision system. In [21], Estrada et al. proposed a multi-scale contour grouping method. Candidate contours are firstly extracted at a coarse scale and then used as prior knowledge of the object to generate spatial priors at finer scales. Thus the coarse contour estimation can be refined progressively to extract detailed object contours. The multi-

scale method outperforms single-scale contour extraction methods like the ratio contour algorithm (RC) [80], the grouping method of Estrada and Jepson (EJ) [22]. Another type of salient contour extraction algorithm is the active contour model (ACM) based method. In [63], self-organizing model (SOM) is integrated with ACM to form the batch-SOM (BSOM) method, which can avoid the contour "leak" into the weak or broken edges and also detect multiple objects in the scene. In [30], a new multi-scale saliency function is designed according to the Minimal Description Length (MDL) principle.

Although there has been much research on this topic, because the complexity of the cluttered and texture background in real-world scene and no prior knowledge about the shape of the target object in general case, problem of salient contour extraction is still a challenging problem. Shortcomings of many current methods include their inability to exclude non-contour segments from the potential contour segment set before segment linking step to form more connected object contours. Many contour extraction approaches directly perform perceptual grouping on the linear contour segment set, and as a consequence, the accuracy and the efficiency of the contour extraction are both influenced. At the same time, the computing complexity of these methods is very high due to the graph based computing on the networks of possible links between line segments.

In order to address this challenge, we present the Constrained Delaunay Triangulation (CDT) graph based contour detection algorithm. Different from the former methods, the CDT graph can be built on the line segments without scale parameter but the property of CDT graph can assure that the potential contours are included in the graph. In addition, CDT graph is a more compact structure than the scattered edge pixels or scattered line segments, which can greatly reduce the computing complexity due to the combinatorial number of possible links between line segments. Since the CDT graph is built on salient line segments, perceptual grouping can be transferred to a two-class classification algorithm on the links between salient line segments. By classifying all the

CDT edges into “contour link” cluster or “non-contour link” cluster, we can simplify the process to avoid the computing in combinatorial complexity. At the same time, saliency measure is designed under the Snakes model, which encodes both the Gestalt laws and the image information naturally in the energy function. The constraints related to Gestalt laws include the convexity constraints and the curve smoothness. The image constraints comprise the high color gradient on the boundary and the homogeneous color distribution inside the enclosed region. Compared with the former methods, the proposed method is more robust and efficient to detect salient contours in the complex scene.

4.2 Constrained Delaunay Triangle (CDT) Graph Based Contour Detector

Through multi-scale contour segment detection algorithm, salient contour segments most probably on the object contours are extracted. In order to group them into connected closed contours, perceptual grouping is implemented according to the Gestalt laws. Different perceptual grouping methods vary in the mechanisms of encoding the Gestalt laws and in the optimization rules to obtain the optimal object contours. Some methods address the problem of extracting the best contour in an image scene. But this is not applicable to the case when there are multiple interested objects in the image. The algorithms proposed in this chapter do not have a bias on the number of objects and thus can simultaneously extract multiple objects from an image.

In this section, the Snakes model based perceptual grouping algorithm is proposed. The Snakes model based algorithm can naturally encode the Gestalt laws and the image information into the evolving function. The probability of a CDT edge to be a “contour link” is then represented by the Snakes energy on the line segment group formed by the CDT edge and the two line segments linked by it. The probability can be viewed as a saliency measure for the central CDT edge to decide whether it is a “contour link” in the classification process. Note that the proposed method is performed on the salient line segments embedded in the CDT graph, instead of the edge pixels in the traditional Snakes

models. Experimental results show that the proposed method can perform in a robust way even when the interested objects are in the cluttered backgrounds.

4.2.1 CDT Graph on Salient Line Segments

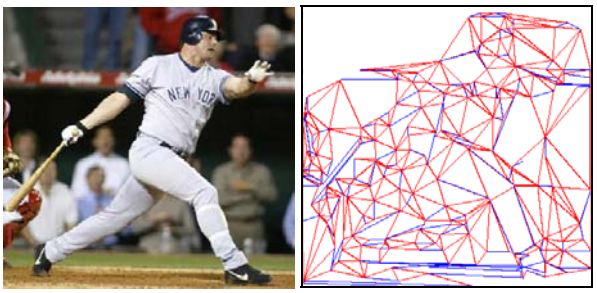
By means of the proposed multi-scale line segment detection algorithm, only salient line segments are extracted from the input image. Perceptual grouping further search for the proper paths among these salient line segments to link them in order and form the salient object contours. Without the distracting line segments from the noise or texture region, perceptual grouping can also be viewed as a two-class classification algorithm on all the linking paths formed by all the pairs of line segments. As we all know, the number of all the possible line segment pairs is in combinatorial complexity and thus it is impossible to obtain the absolute optimal solution in limited time. Under this circumstance, the CDT graph is introduced in our algorithm. As a compact structure with simple description, each line segment is linked with its neighboring line segments in a triangular structure. What' more, it can be proved that all possible object contours are included in this graph. Then salient contours can be further extracted by the means of contour saliency measure. In this sense, the adoption of CDT graph in perceptual grouping can enormously reduce the size of basic data set.

The Delaunay Triangulation (DT) of a point set is the unique triangulation such that no point is inside the circum-circle of any triangle. The Dealaunay triangulation of a point set is also the dual graph of the Voronoi tessellation for this point set. The Constrained Delaunay Triangulation (CDT) is a modified version of the DT, in which the user-defined edges should be included in the graph. Since the DT is unique for a point set, the CDT always includes some un-Delaunay edges. According to Ren et al., the CDT holds many good properties inherited from the DT and is popular in geometric modeling and finite element analysis [49].

In the proposed method, the constrained Delaunay triangulation (CDT) graph is built

over the salient line segments set. Since non-contour line segments have already been removed in the multi-scale analysis, our CDT has the following special properties.

- ❖ All the potential contours are included in the CDT graph. There are two kinds of them, salient line segments and "contour link", that is, the CDT edges as links between consecutive line segments on the contour.
- ❖ Only the linking CDT edges between two salient line segments are possible to be "contour link" CDT edges.
- ❖ CDT graph is much more standard in structure description to formalize the potential linking segments on the contour breaks.



(a) (b)

Figure 4.1 Illustration of the CDT graph with an example image (a) input image (b) CDT graph built on the extracted salient line segments, where blue segments correspond to the salient line segments extracted by the first step and red segments correspond to the CDT edges.

Concerning the second property, we would like to make further explanation. Since the point set of current CDT include all the starting and ending points of every salient lien segment, every CDT edge is between two salient line segments. The second property here emphasizes that the probability of a CDT edge to be a "contour link" is only related to its left and right salient line segment neighbors among its left and right neighboring segment sets. Concerning the third property, compared with former perceptual grouping methods, the number of potential linking paths is greatly reduced through the structure of CDT

graph. In Figure 4.1, an example of CDT graph building on the salient line segments is given.

Ren et. al proposed a Supervised Contour Completion (SCC) approach using conditional random fields [49]. CDT graph is also utilized in their approach to extract the CDT edges on the contour. Our method is different from this method in the following three aspects. Firstly, CDT graph is directly built on the overall line segment set in SCC while on the salient line segment set in our approach. Secondly, since there are not only contour segments but also non-contour segments in gradient edges, both Gradient edges (G-edges) and Completion edges (C-edges) need to be classified to decide their final state in SCC. In comparison, only the C-edges are necessary to be labeled as “contour link” or “non-contour link” in the proposed extraction system. Thirdly, SCC is a supervised method to learn useful contour patterns from the human marked contours while our method is an unsupervised method, directly classifying the C-edges into two classes according to their local geometrical and image features.

4.2.2 CDT Graph Based Contour Detection

Because the non-contour line segments have been removed before CDT graph construction, the contour completion in this stage can be transferred as a two-class classification problem. All the CDT edges are labeled as “contour link” or “non-contour link” according to their feature set and local context.

The principle of the CDT edge classification is to decide whether a CDT edge is a correct and effective completion on a contour break between two salient line segments. Therefore, for every CDT edge, the two salient line segments on its both sides should also be considered when evaluating the probability of the CDT edge to be a contour segment. That is, “GCG” structure of every C-edge is analyzed. Here we use the same definition same as the paper [49] to name the salient line segment as “G-edge” and the CDT edge as “C-edge”. The properties of the “GCG” structure are then one part of the

feature set of the C-edge.

If the C-edge is a contour segment, its “GCG” structure is one part of the object contour curve. As we all know, smooth curves with high image gradients on every pixel and homogeneous inner region are always favored to be extracted as the object contours. The properties of the “GCG” structure include the rules according to the Gestalt laws and the image information.

Based on the Gestalt laws, two line segments can be linked into contour curve when they satisfy the rules of proximity and continuity. The proximity rules are represented as the convexity measure in the proposed algorithm. And the continuity rules are symbolized as the external energy of the Snakes model. At the same time, the image information in the surrounding of the central CDT edge should also be considered. It includes the color gradient measure on the boundary and the color homogeneous measure in the enclosed region. The color gradient measure here can be directly represented by the external Snakes energy. In certain sense, we can deem the above measures constitute an extended Snakes energy.

As a well-known contour extraction method, the Snakes model, or Active Contour Model (ACM), performs on the edge pixels under an energy optimization scheme. Given an appropriate initial state, the given curve can evolve itself under the rules of minimizing the curve energy including both the internal energy and the external energy. Here we do not intend to obtain the final optimized curve of the “GCG” structure. Our aim is to illustrate the stability of the studied “GCG” structure through the extended Snakes energy. Therefore, the extended Snakes energy of every “GCG” structure on its next state after only one evolution is taken as the evaluation value. Note that the Snakes energy here should be normalized by the length sum of the three line segments, in order to avoid a bias to produce overly small “GCG” structures.

A. Convexity Measure

A salient convex grouper proposed by Jacob [71] is robust to viewpoints, image scale, partial occlusion and clutters. The input of the salient convex grouper is oriented line segments. Each line segment corresponds to two oriented vectors reverse to each other. The output is the possible convex groups and convex cycles in the scene. For a convex group, a measure called Length-Gap Ratio (LGR) is then defined to assure its saliency. Only the convex groups with LGR greater than fixed threshold are valid in the extraction. Here we utilize this LGR measure to evaluate the convexity of a "GCG" structure.

$$C = \frac{L_{\mathrm{ij}}}{L_{\mathrm{ij}} + L_i + L_j} \tag{4-1}$$

where L_{ij} is the length of the CDT edge between two salient line segments l_i and l_j and L_i, L_j are respectively the lengths of l_i and l_j. The smaller the convexity measure is, the two line segments are more possible to be linked under the Gestalt laws.

B. External Snakes Energy Measure

The three-line-segment "GCG" structure is taken as a potential contour curve parameterized as follows.

$$v(s) = (x(s), y(s)) \tag{4-2}$$

The Snakes internal energy addresses the smoothness of the contour curve. It is composed of two parts, elastic energy and bending energy. The smaller the external energy S is, the curve fitted by the studied "GCG" structure is smoother.

$$S = \frac{E_{\mathrm{int}\,ernal}}{\int_s ds},\ E_{\mathrm{int}\,ernal} = E_{elastic} + E_{bending} \tag{4-3}$$

$$E_{elastic} = \frac{1}{2}\int_s \alpha(s)\left|v_s\right|^2 ds,\ v_s = \frac{dv(s)}{ds} \tag{4-4}$$

$$E_{bending} = \frac{1}{2}\int_s \beta(s)\left|v_{ss}\right|^2 ds \tag{4-5}$$

C. Internal Snakes Energy Measure

As we all known, the higher the boundary gradient is, the closer the current curve is to the object boundary. External energy is defined in the following form so that it can take smaller values when the contour is on the object contour. The larger the internal energy G is, the studied "GCG" structure is closer to the object boundary.

$$G = \frac{E_{external}}{\int_s ds},\ E_{external} = \frac{1}{2}\int_s E_{image}(v(s))ds \tag{4-6}$$

$$E_{image}(x,y) = \left|\nabla\left(G_\sigma(x,y) * I(x,y)\right)\right|^2 \tag{4-7}$$

where E_{image} is the square of the edge magnitude from the filtered image by standard Gaussian function.

D. Region Homogeneous Measure

For the gray-level image, the standard deviation value of the intensity values is often used to measure the homogeneous degree of an image region. In addition, the measure M was utilized to evaluate the intensity variation in the enclosed region of a contour [26].

$$M = \frac{\iint_R \left|\nabla(I(x,y))\right| dxdy}{\iint_R dxdy} \tag{4-8}$$

where R is the enclosed region and $\left|\nabla(I(x,y))\right|$ stands for the gradient magnitude at the pixel (x,y). But the above measures are not suitable to describe the distribution variation in color images. To address this problem, a homogeneous measure, Color Distribution Evenness, specially designed for color distribution is accepted in our perceptual grouping algorithm. Assume that in the enclosed region R, there are M different colors and the number of pixels with color of C_m is n_m. In this case, the spatial centroid of all the pixels whose color is C_m can be obtained as P_{C_m}. The distance between the P_{C_m} and P_{C_m}, the geometrical centroid of the enclosed region R, can be used to evaluate the evenness of the

color C_m's distribution in this region. The overall homogeneity of this region can then be measured by the average evenness value of all the possible colors.

$$H = \frac{\sum_{m=1}^{M} n_m \left\| P_{C_m} - P_R \right\|_2}{\sum_{m=1}^{M} n_m} \tag{4-9}$$

where $\| \ \|_2$ stands for the Euclidean distance. The smaller the average distance H is, the more homogeneous the enclosed region is.

The convexity measure, external and internal Snakes energy of the "GCG" structure used here is favored when they have small values, while region homogeneous measure is on the contrary. At the same time, they are assumed to be independent information sources. According to the above analysis, we design the following weight function of every "GCG" structure. The above measures are respectively normalized to the range of [0,1].

$$w_{AB} = \exp\left(- \frac{C^2}{2\sigma_C^2} \right) \exp\left(- \frac{(1-S)^2}{2\sigma_S^2} \right) \exp\left(- \frac{G^2}{2\sigma_G^2} \right) \exp\left(- \frac{H^2}{2\sigma_H^2} \right) \tag{4-10}$$

where C, S, G and H are respectively the convexity measure, external Snakes energy, internal Snakes energy and region homogenous measure. σ_H, σ_S, σ_G, σ_H the are respective variance. w_{AB} is then the saliency measure of the "GCG" structure including the C-edge, L_{AB}, with the endpoints of "A" and "B". w_{AB} describes the on-contour probability of the studied CDT edge.

Besides the property of the "GCG" structure including the studied C-edge, the feature set of the C-edge also includes its local context. On one endpoint, the studied C-edge is related to a unique salient line segment. The line segment set of all the C-edges starting from this unique salient line segment forms part of the local context of the studied C-edges. In this way, every C-edge has two neighboring C-edge sets, called as

"the left neighbor set" and "the right neighbor set". They are expressed as S_{Left} and S_{Right}.

If a C-edge can well complement the break between two salient line segments, it is expected to have high probability value either in the left or in the right neighbor set. Thus high left probability or high right probability is necessary to declare a C-edge to be a contour segment. We design the following decision rule based on the above analysis. In this way, all the CDT edges are classified into the contour segment set and the non-contour segment set.

$$M_{left} = \begin{cases} 1, & w_{AB} = \max\limits_{S_i}\left(w_{AS_i}\right) \\ 0, & \text{otherwise} \end{cases} \tag{4-11}$$

$$M_{right} = \begin{cases} 1, & w_{AB} = \max\limits_{K_j}\left(w_{K_jB}\right) \\ 0, & \text{otherwise} \end{cases} \tag{4-12}$$

$$S_{contour} = \left\{AB \middle| M_{left} = 1, \quad or \quad M_{right} = 1\right\} \tag{4-13}$$

4.3 Experimental Analysis

Seven benchmark images from the Berkeley Segmentation Dataset (BSD) [20] are used in our experiments. Their corresponding human-marked boundary image is taken as ground truth. Figure 4.2 shows the seven images and their ground truth images. BSD contains 300 images each with 5 human-marked boundary maps. Here we show all the 5 ground truth images in one image. All the test images are free from any noise. In each image, there are one or more real objects in perceptual sense, which are also illustrated by the boundary maps marked by human beings. Some images contain considerable texture in the scene and most others are in rather simple background. The size of these benchmark images are all 481*321 pixels. Since image decimation is implemented during image pyramid construction, for these benchmark images, the image pyramid used in the process of multi-scale salient line segment detection is composed of 3 scale levels.

The CDT graph based contour detection method is carried out on the salient line

segments to explore how to link these line segments into continuous contours. The conventional line tracing algorithm [65] is often used to obtain the contours comprised of line segments. Compared with it, the proposed method can provide much more reliable line fitting result with better connectivity and saliency for perceptual grouping process. To prove the improvement, detection rate and saliency of the result from traditional line tracing method and the proposed method is compared. Both measures are computed through comparing the result image with the ground truth image.

A. Detection Rate

For every pixel on a detected line segment, a neighborhood region is defined. If one contour pixel from the ground truth image is in this neighborhood, the detected pixel is deemed as correct detection, i.e., true positive. The detection rate can then be calculated by the F-Measure [72] in Equation 4-16. Larger F-Measure values reflect the better detection result.

$$P = \frac{card(X_{tp})}{card(X_{tp}) + card(X_{fp})} \tag{4-14}$$

$$R = \frac{card(X_{tp})}{card(X_{tp}) + card(X_{tn})} \tag{4-15}$$

$$F = PR/(\alpha P + (1-\alpha)R) \tag{4-16}$$

where $card(X)$ denotes cardinality of a set. X_{tp}, X_{tn}, X_{fp} is respectively the set of true positives, true negatives, false positives. α is set to 0.5 in our experiments. Since there are 5 ground truth images for every benchmark image, F-Measure is computed on every ground truth image and then all the values are averaged to get the final F-Measure.

The improved percentage here is computed by Equation 4-17. From Table 4-1, it is clear that the proposed method can detect more reliable contours for all of the benchmark images than the former method. The F-Measure value is improved in different degrees for 7 test images and the largest improvement reaches 68% for the 1st test image.

$$P = \frac{F_{new} - F_{old}}{F_{old}} * 100\% \tag{4-17}$$

Both F-Measures from former method and the proposed method become smaller or larger together with each other for different images. This shows that F-Measure has close relation to the image content. If we sort the image according to their F-Measure, we can coarsely classify them into three clusters.

- ❖ In the 1st and 3rd images, the structure of the real object contours is simple and every part of the contours is loosely related to each other.
- ❖ In the 4th, 6th and 7th images, the inner structure of real objects is more complex but such details are not emphasized too much in the ground truth image compared with the 1st and 3rd image. And the real object contours in the three images are all surrounded with much more background texture.
- ❖ In the 2nd and 5th images, there is less background texture and every part of the real object contours are strongly related to each other.

TABLE IV-1

COMPARISON ON F-MEASURE BETWEEN FORMER AND PROPOSED METHOD

Image number	*1*	*2*	*3*	*4*	*5*	*6*	*7*
Former method	0.35	0.73	0.43	0.51	0.63	0.51	0.52
Proposed Method	0.59	0.84	0.70	0.71	0.77	0.67	0.70
Improved percentage	68%	15%	63%	39%	22%	31%	35%

Through the above analysis, we can conclude that rich background texture can deteriorate the detection accuracy. On the other hand, the structural information of the contour can help the detection method to obtain more complete and accurate contours. Additionally, as the F-Measure becomes smaller, the improved percentage grows larger. This shows that for 1st and 3rd images, the structural information on the real object contours are explored by the proposed method and then utilized to improve the accuracy of detection. The improvement percentage on the detection rate of the proposed method over the former method is at least 15% and at most 68% for the experimental image.

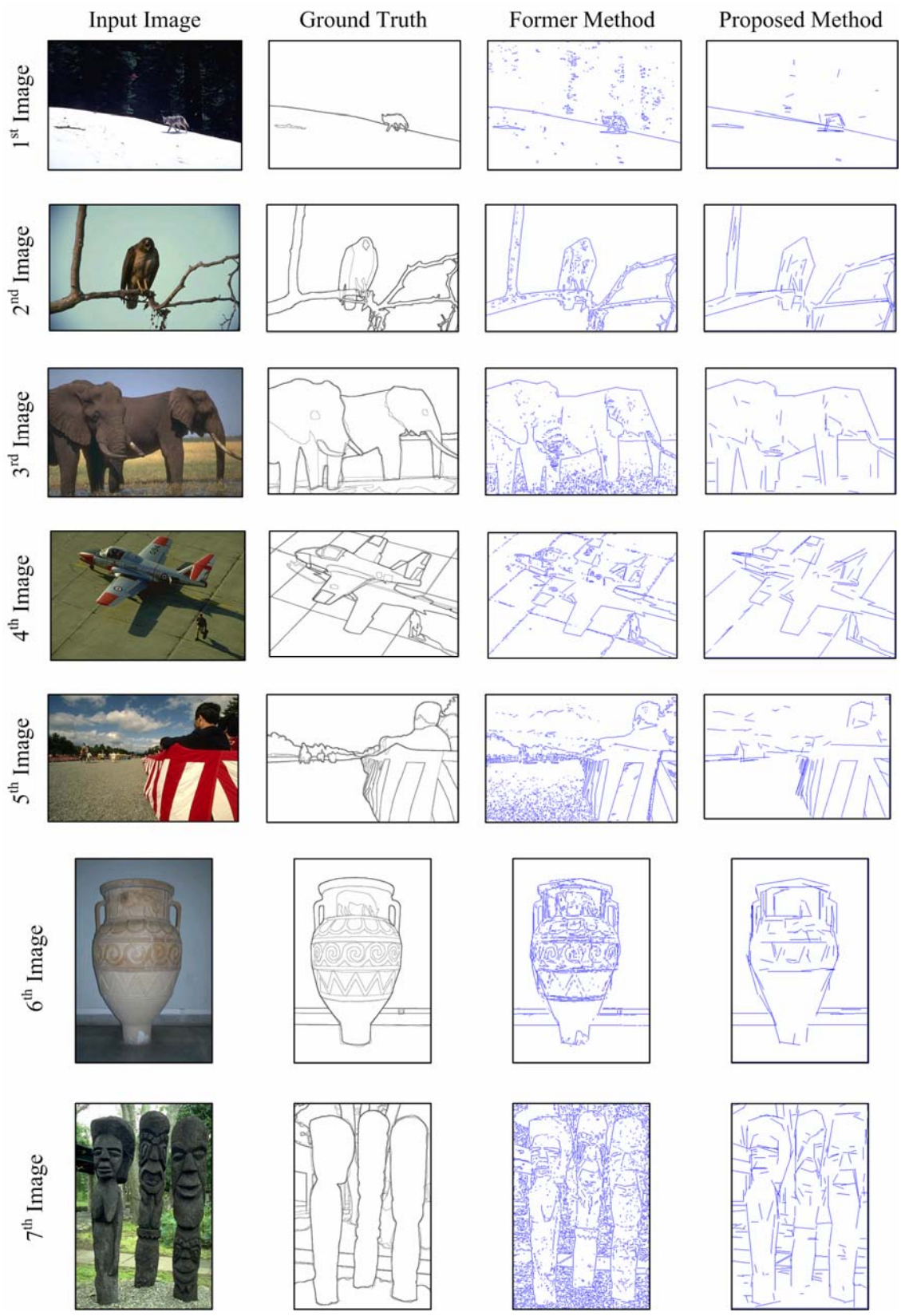

Input Image
Ground Truth
Former Method
Proposed Method
1st Image
2nd Image
3rd Image
4th Image
5th Image
6th Image
7th Image

Figure 4.2 Comparison on the performance between the former method and the proposed method. There are totally 7 example images shown here, from the above to the bottom. The first column is the input image. The second column is the ground truth image. The third column is the line segment map from former method. The fourth column is the line segment map from the proposed method.

B. Saliency Measure

The saliency measures of every contour pixel is evaluated by the number of times that it is deemed as true positive when the detected result is compared with all the 5 ground truth images. To simply the computing, this counting number is further normalized by 5 to get a value in the range from zero to one. Then the saliency measure of one line segment and further the whole detection result is defined by Equation 4-18.

$$S = \frac{\sum_{i=1}^{card(l)} S(l^i)}{card(l)}, \quad S(l) = \frac{\int_l s(t)dt}{\int_l dt} \tag{4-18}$$

where $s(t)$ is the saliency of a pixel on the line segment l. From Table 4-2, we can find out that the contours extracted by the proposed method are much more salient than the former method. In five out of seven test images, the improvement of saliency even exceeds 50%. The improvement percentage on the overall saliency of the detected contours from the proposed method over the former method is at least 18% and at most 165% for the experimental image.

We further analyze the saliency distribution of all the line segments in result images through the histogram of saliency measure. In Figure 4.3, the saliency measure histograms for seven testing images are shown. Having studied the saliency histogram comparison figures, we can find out the following phenomenon. Most of the line segments detected by the former method have low saliency measures but only a few are salient contours. In contrast, the line segments detected by the proposed method

concentrate in the high saliency region. We can also figure out that the proposed method increase the percentage of high saliency lines segments and reduce that of low saliency line segments for every test image. This reflects the proposed method can obtain more salient contours than the former method.

TABLE IV-2

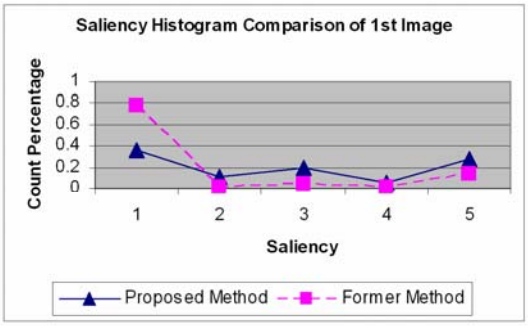
COMPARISON ON SALIENCY MEASURE BETWEEN FORMER AND PROPOSED METHOD

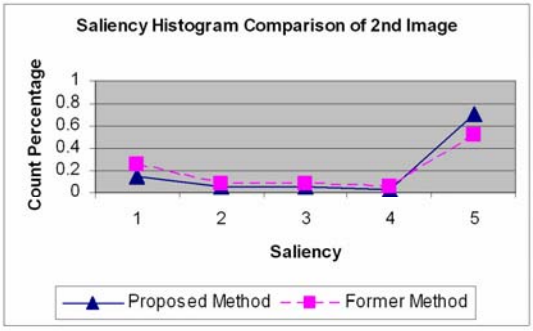
Image number	*1*	*2*	*3*	*4*	*5*	*6*	*7*
Former method	0.17	0.61	0.29	0.28	0.55	0.31	0.31
Proposed Method	0.45	0.76	0.49	0.55	0.65	0.63	0.46
Improved percentage	165%	24%	69%	96%	18%	103%	48%

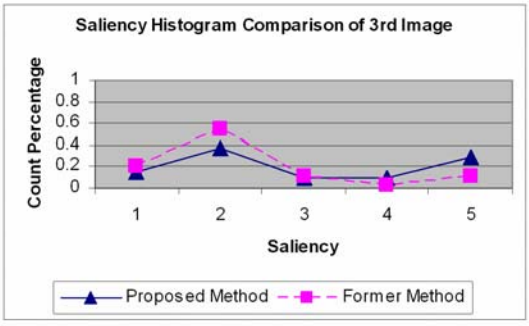
Based on the above analysis on detection rate and saliency, we can conclude that the proposed method improve the saliency of extracted object contours. Both the detection result images of the former method and the proposed method are shown in Figure 4.2. For these test images, a lot of responses from noise or background texture prevail in the result from the former method, while in the result from the proposed method, such responses are almost cleaned. Simultaneously, some structural information of the boundary shape is lost in the former method result because of the disturbance of noise or low contrast on the real object boundary. But the proposed method can restore this information. This phenomenon is especially prominent in the 6th and 7th test image. In the 6th image, the two horizontal lines near the handles are detected as densely clustered short line segments by the former method. In contrast, the two lines are restored by the proposed method. In the 7th image, the three objects are surrounded by rich texture. The line segments of real object contours are covered by many short line segments from texture background. And the connectivity of the contours is also greatly destroyed. The proposed method effectively removes the texture background responses and also improves the connectivity of the contours.

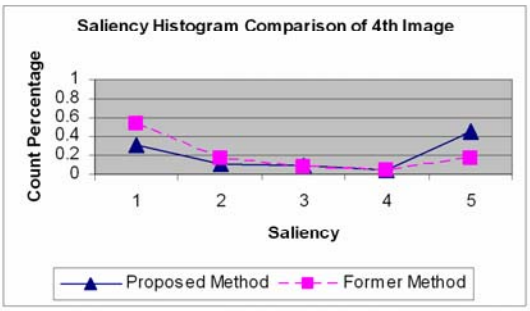
In summary, the proposed CDT graph based contour detection method can effectively extract salient contours even in the cluttered background with rich texture.

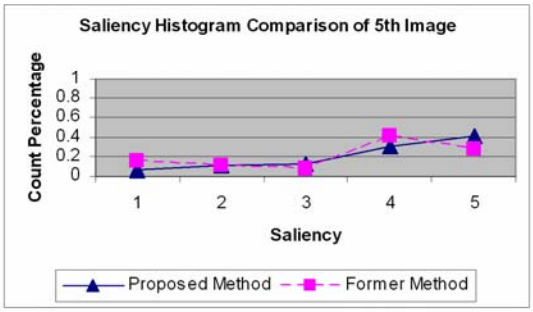
The salient contours with better connectivity and saliency produced by the proposed method make a good preparation for the following object description and further object recognition.

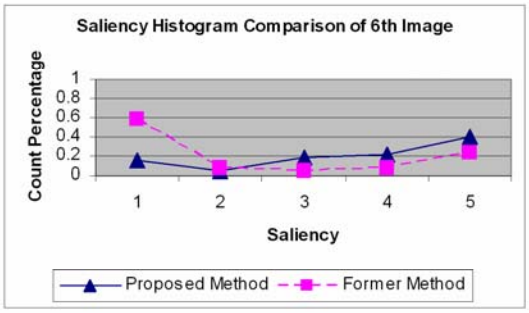












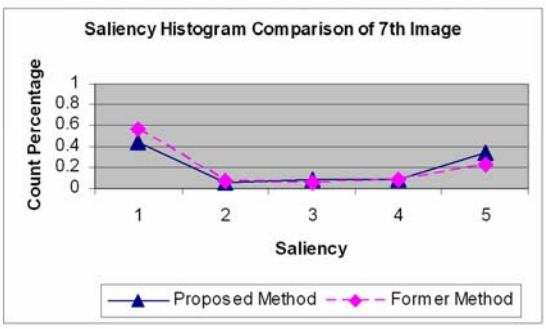


Figure 4.3 Comparison on the saliency histogram between the former method and the proposed method. From the above to the bottom, and for each row from the left to the right, the comparison figure respectively for the 1^{st}, 2^{nd}, …, 7^{th} testing image are shown in order. In each figure, the blue line is the saliency histogram curve for the line segment detection result from the proposed method, while the red line is for the former method.

4.4 Conclusions

In this chapter, a CDT graph based contour detection algorithm is proposed. Most of the perceptual grouping methods are implemented on the contour segments, usually straight line segments. In order to extract salient object contours, perceptual grouping methods have to search for the suitable linking paths among all the line segments. This process can be viewed as a nearest path searching problem in the graph. The nodes of the graph are the line segments and the edges of the graph are the links between every pair of line segments. The computing complexity is rather high since the perceptual grouping method has to search for optimal path among the combinatorial number of graph edges. To address this challenge, we adopt the structure of CDT graph to form a regular structure before perceptual grouping. It can be proved that all the links for the potential contours are included in this CDT graph. At the same time the introduction of the CDT graph can enormously reduce the computing complexity because the possible links are reduced to very low number.

On the other hand, since there are only the salient line segments detected by the proposed multi-scale contour segment detection method, we can transfer the perceptual grouping into a two-class classification problem by classifying all the CDT edges into “contour link” cluster or “non-contour link” cluster. The CDT edges are classified according to the value of their saliency measures. Here a novel saliency measure based on the Snakes model is proposed. The saliency measure includes the consideration both

on the Gestalt laws and on the image information. The “GCG” structure formed by a CDT edge and the two line segments on its both sides can be evaluated by a combined measure including the convexity measure, the external Snakes energy, the internal Snakes energy and the region homogeneous measure. Experimental results show that the proposed method can derive contours with better continuity and saliency than the former method under the condition of cluttered background on the given 7 experimental images. The improvement percentage on the detection rate of the proposed method over the former method is at least 15% and at most 68% for the experimental image. The improvement percentage on the overall saliency of the detected contours from the proposed method over the former method is at least 18% and at most 165% for the given example images.

5 Object Contour Based 3D Building Reconstruction

Since contour is an effective feature for the object description and recognition, we mainly discuss how to detect salient contour in the image from the cluttered background to locate the interested objects in the dissertation. In the following, the application of salient contour detection is discussed.

Contour detection is an essential part of many computer vision problems such as image understanding [73] and object recognition [74]. Recently there are also increasing applications of contours in object tracking [75-77] and image coding [78] because contour is a concise and effective form to describe objects in the scene. In the field of aerial image understanding, man-made object detection and description is very important. In this chapter, we mainly discuss the application of salient contour detection to the problem of stereo matching based building reconstruction. It is expected that the introduction of the salient contours can facilitate the process of reconstruction.

5.1 Background and Related Work

Performance of stereo matching based 3D building reconstruction in the aerial image is closely related to matching features. There are mainly three kinds of stereo matching methods: intensity based matching, area based matching and feature based matching [79-80]. Intensity based matching methods directly study the intensity profiles of two images to find the correspondence [81]. Area based matching methods explore the correspondence between the image regions based on some similarity measures [82-83]. Feature based matching methods introduce many higher level features in the matching process such as edge element [84-86], line segment [87], curve segment [88], corner [89-90], circle and ellipse, texture region, and so on. To improve performance, some researchers also proposed to combine a collection of feature types. For example, Lim and

Binford [91] used a hierarchy of features varying from edges, curves, to surfaces and bodies for high-level attribute matching. Wei and Hirzinger [80, 92] proposed to include intensity gradients together with intensity to perform stereo matching.

Although there has been much research on feature based stereo matching [84-90], there is not a systematic method to perform 3D building reconstruction in urban scene due to the complexity of the cluttered background and the various building structures. Shortcomings of many current methods include their inability to exclude features generating from noise and texture, and shortage in losing features due to shadow or occlusion.

In order to address this challenge, we propose a robust stereo matching method called SAR-DPSM method [93]. Firstly, non-contour features are excluded to produce the salient line segments. Secondly, since the rectangle is the basic polygonal shape for the buildings, rectangle regions are respectively extracted in two images based on the salient line segments. Thirdly, the candidate rectangle regions from the left and right image are matched with each other by human observers. After that, stereo matching is performed based on these matched line segments. The introduction of the rectangle regions can reduce the noise level in reconstruction result. On the whole, the proposed method combines the features of rectangle region, contour of the rectangles and intensity together to fulfill the matching process.

5.2 Rectangle Region Based Stereo Matching

The proposed SAR-DPSM method is mainly based on the following three facts:

- The boundary of most buildings in the image is in polygonal shape or in curved contour which can be decomposed into rectangles.
- The area and the height to width ratio of the rectangles regions corresponding to buildings are in certain range in the given aerial image.
- The rectangle regions corresponding to the buildings are intensity homogeneous

inside.

Based on the above facts, we divide the rectangle based stereo matching process into two stages. Firstly, the candidate rectangles are detected based on the local information of salient line segments and then are filtered according to their height to width ratio and the proposed saliency measure. Secondly, the candidate rectangles from two images are matched with each other by human observers and DPSM is then implemented on the basis of matching line segment list.

Our work is superior to past work since in the proposed method, most of the non-contour line segments are suppressed before region analysis and reconstruction. This can not only improve the accuracy and efficiency of candidate rectangle extraction, but also simplify the following region based stereo matching. In the following, we explain and state the details of the proposed method.

5.2.1 Salient Rectangle Contour Detection

Rectangle contours are detected on the set of salient line segments. The detected rectangle contours are then used as the stereo matching features to finish the 3D building reconstruction. The rectangle detection includes mainly steps. Firstly, the salient line segments are grouped by the semantic rules of the rectangle structure. Secondly, post-processing is carried out to filter the redundant rectangle detection and also adjust the candidate rectangle contours to strict rectangle shape.

A. Rectangle Detection from Salient Line Segments

Before rectangle detection, salient line segments are detected by the proposed scale invariance based line detector described in Chapter 3. Then candidate rectangles are extracted based on Gestalt laws of proximity, parallelism and perpendicularity on the group of line segments which can form the rectangles.

There has been much research in the problem of rectangle detection. For example, Krishnamachari et al. [27] proposed a MRF based method which can estimate the probability of every boundary line segment belonging to a contour and then generate potential rectangles in the image. Although the application of the rectangle detection in [27, 28] is also rooftop detection, the method cannot be well applied to our problem because of the following reasons.

The MRF based rectangle detection method mainly makes use of the Gestalt principles like the proximity, the parallel property and the perpendicular property among the boundary line segments. Any line segment that satisfies the above property with its neighbor line is kept as a candidate boundary line of certain rectangle; otherwise it is removed. This idea works effectively when rectangle regions are sparsely distributed. But in urban scene, the houses are very close to each other, thus a single line segment may have many neighboring lines satisfying the above principles. In this case, boundary lines belonging to every potential rectangle cannot be effectively segregated. And then, all the other methods on probability analysis of line segments in local region are not suitable for our current application.

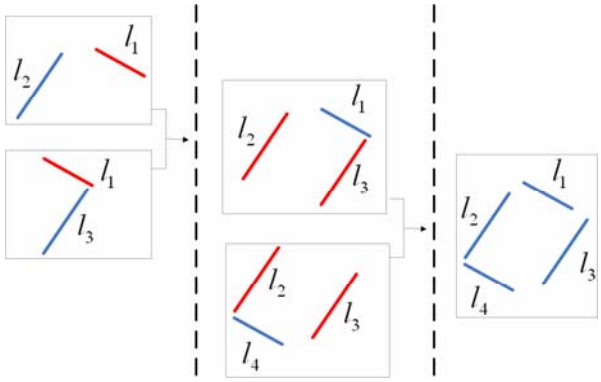


Figure 5.1 Illustration of candidate rectangle detection process

To address the challenge, we propose the following rectangle detection method based on both the Gestalt principles and the intensity information. Then a saliency measure is

designed to evaluate the existing probability of every candidate rectangle. The process of the proposed rectangle detection method is illustrated in Figure 5.1.

In the first step, we produce a perpendicular neighbor line set $N(l_1)$ for every line segment l_1. When a line segment l_2 satisfies the following principles, it belongs to $N(l_1)$.

- Firstly, the smallest distance between the endpoints of l_1 and l_2 is under certain threshold.
- Secondly, the angle difference between l_1 and l_2 should be close enough to 90 degree.

The above two principles assure the proximity and the perpendicular property. Now we have a lot of "L" shaped corners of the potential rectangles.

In the second step, the "|_|" shaped structures are generated from two "L" shaped corners. Assume the endpoints of l_1 are (A_1, B_1), for every two line segments l_2 and l_3 in $N(l_1)$,

- Firstly, l_2 is closer to A_1 and l_3 is closer to B_1, i.e., l_2 and l_3 are respectively closer to each endpoint of l_1.
- Secondly, a convexity based measure developed from l_1, l_2 and l_3 is under certain threshold.

The breaks among l_1, l_2 and l_3 are measured by a convexity measure described in Equation 5-1, where l_{ij} is the length of the break between l_i and l_j, and L_i is the length of l_i. Then the convexity measure should satisfy Equation 5-2 to assure the proximity among l_1, l_2 and l_3.

$$LGR = \frac{l_{12} + l_{13}}{l_{12} + l_{13} + L_1 + L_2 + L_3}. \tag{5-1}$$

$$LGR < threshold_{\lg r}. \tag{5-2}$$

In the third step, complete rectangles with four boundary lines are produced from two "|_|" shaped structure. When l_2 and l_3 are both perpendicular to l_1 to form a "|_|" shaped structure, we search for another line segment l_4 that can form another "|_|" shaped structure with both l_2 and l_3. Then l_1 and l_4 are parallel with each other. To declare an effective rectangle detection from l_1, l_2, l_3 and l_4, the perpendicular distance between l_1 and l_4 should satisfy Equation 5-3 to assure a compact rectangle.

$$\lambda L < d < (1+\lambda)L, L = \max(L_2, L_3), 0 < \lambda < 1. \tag{5-3}$$

Here d is the perpendicular distance between l_1 and l_4. λ is an adjusting parameter. λ is empirically set as 0.2 in our experiments.

Having detected the candidate rectangles, we further select the ones which are most probably corresponding to the buildings based on a saliency measure. Three measures such as convexity measure, region homogeneity and boundary gradient are incorporated into the saliency measure.

Firstly, a convexity measure "C" is defined in Equation 5-4, similar to the "LGR" in Equation 5-1. The smaller the convexity measure is, the better the proximity among the four line segments. Secondly, the region homogeneity measure "H" is defined as the variance of the intensity of the pixels inside the rectangle region. Smaller "H" implies smoother region. Thirdly, the boundary gradient measure "G" is defined as the average gradient of all the pixels on the boundary line segments. When a candidate rectangle has higher "G", it means that the location of its boundary lines is rather precise. All the above three measures are normalized to the range of [0, 1] and then combined into the saliency measure "S" defined in Equation 5-5. Then higher saliency measure "M" symbolizes more salient candidate rectangle.

$$C = \frac{l_{12} + l_{13} + l_{42} + l_{43}}{l_{12} + l_{13} + l_{42} + l_{43} + L_1 + L_2 + L_3 + L_4}. \tag{5-4}$$

$$S = w_1(1-C) + w_2(1-H) + w_3 G, w_1 + w_2 + w_3 = 1. \quad (5\text{-}5)$$

Besides the above saliency measure, we also set a threshold on the height to width ratio to exclude rectangles in elongated shape.

With the saliency measure and the height to width ratio, we perform the experiment on a test image. In Figure 5.2(b), the most 100 salient rectangles satisfying the given height to width ratio threshold are shown. From the result, we can see that some detected candidate rectangles are not strictly regular rectangles. At the same time, the detected rectangles are overlapped with each other on a building. To simplify the processing of rectangle based stereo matching, we have to take some post-processing on the detected rectangles.

(a) (b)

Figure 5.2 Illustration of the saliency measure on an example image to illustrate the saliency measure (a) input image (b) the most salient 100 rectangles according to the proposed saliency.

B. Post-processing on Detected Rectangles

The post-processing on detected rectangles includes two parts. Firstly, all the detected rectangles are adjusted to strict rectangle shapes. Secondly, among the rectangles overlapping with each other on the same spot of a building, only the most salient one is

kept.

Some candidate rectangles are not in strict rectangle shape because the acceptance range on the forming angle of two perpendicular line segments is set as from $\pi/2-\pi/12$ to $\pi/2+\pi/12$. In order to adjust them into strict rectangle shapes, the following steps are implemented. We investigate the four corners of a rectangle and pick up the angle most different from $\pi/2$. Then on the most biased corner, a new line is generated which can form better perpendicular pairs with its two neighboring line segments. We illustrate this process in Figure 5.3. $l_4^{'}$ is the newly generated line segment to replace l_4.

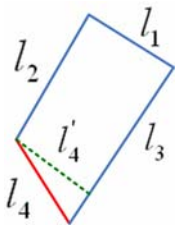


Figure 5.3 Illustration of the shape adjusting process on the basic candidate rectangle

TABLE V-1

NUMBER OF CANDIDATE RECTANGLES BEFORE AND AFTER OVERLAPPING RECTANGLE PROCESSING

Image Index	Before Processing	After Processing	Reduced Percentage
1	269	20	93%
2	37	7	81%
3	189	49	74%
4	3014	391	87%

In urban area, the buildings are always very close to each other. According to the construction rule, the neighboring buildings usually have parallel boundary edges. The above phenomena lead to the overlapping detection of many rectangles on a spot. To reduce the redundancy in rectangle detection, we decide to keep a single rectangle on each building region. In order to simplify the computing complexity, we directly compare the four corners of every two candidate rectangles. If all of the four corners of a rectangle are respectively close enough to the four corners of another rectangle, the less salient

rectangle is removed. This processing is implemented not only before the shape adjusting of candidate rectangles but also after it. This processing is rather meaningful for the following rectangle based stereo matching since it saves a lot of redundant computing on the overlapping rectangles. To illustrate this, the numbers of candidate rectangles with and without this processing are compared in Table 5-1. From Table 5-1, it is clear that the overlapping rectangle processing can greatly reduce the redundancy in rectangle detection for all the experimental images.

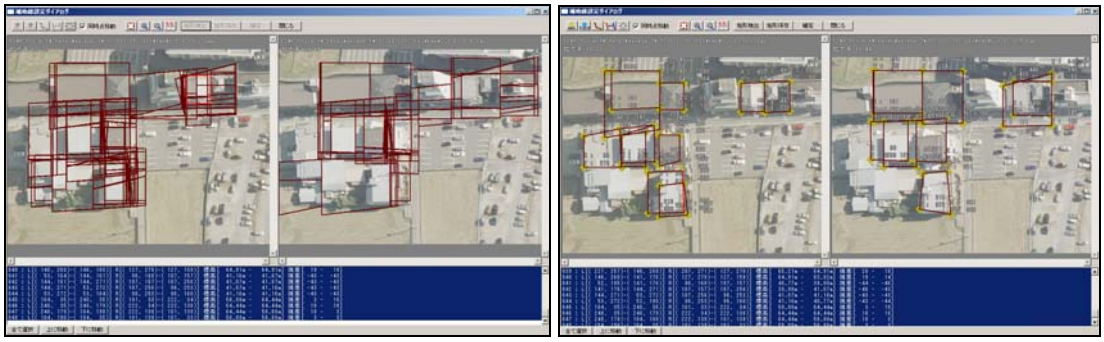
5.2.2 Rectangle Contour Based Stereo Matching

In this section, the details of the rectangle region based stereo matching are explained. A semi-automatic method is designed. Firstly, the correspondence between the rectangles in the left and the right image is marked by the observers. Then a refined DPSM process is carried out based on the corresponding line segments from two corresponded rectangles.

Throughout rectangle detection, except geometrical and intensity analysis, almost no other a priori knowledge of buildings is introduced, which leads to the following phenomenon. Some candidate rectangles are parts of certain building regions, and others may be the combination of several buildings close to each other. On the spot of a building, there may be several candidate rectangles partly overlapped with each other. Among them, the observers select only one candidate rectangle that best matches a building region on the boundary. Then it is easy to determine the corresponding rectangle in another image by spatial proximity. In Figure 5.4, we illustrate the process of rectangle correspondence.

Assume that there are N rectangles respectively in the left and right image. Among them, C_i^L and C_i^R are respectively the ith rectangle in the left and right image. Then we can find out the correspondence between the four boundary lines of C_i^L and C_i^R. In this way,

we can build a list of matching line segments.



(a) (b)

Figure 5.4 Illustration of the rectangle correspondence process in SAR-DPSM. (a) detected candidate rectangles (b) selected rectangles for correspondence

Let us assume that any two corresponding line segments are l_l and l_r. The left and right images are taken under the constraints that the cameras are parallel, which assures that the corresponding pixels in the two images lie on the same horizontal scan-line. In pixel-level DPSM (PL-DPSM) [70], two horizontal lines with the same "y" coordinate are firstly introduced in two images and then corresponding pixel pairs are searched on the two horizontal lines. In our rectangle based stereo matching, since we have already obtained the correspondence between l_l and l_r, for any possible "y" coordinate on both lines, a pair of corresponding pixels $P_l(x_l, y)$ and $P_r(x_r, y)$ can be firstly obtained. Then stereo matching can be carried out in two ranges independently. One range is from the most left point $(0, y)$ to the corresponding point, and another is from the corresponding point to the most right point $(w-1, y)$. Here w defines the width of the input images. In this case, the accuracy of the reconstructed 3D data is improved, especially on the building boundaries. This can be proved by the experimental data in Section 3.

In DPSM, the normalized correlation between the pixels from two images is utilized

to measure the probability of their correspondence.

$$corr_y(x_l, x_r) = \frac{\sum_{i,j=-w}^{w}\left(L(x_l + i, y + j) - \overline{L}\right)\cdot\left(R(x_r + i, y + j) - \overline{R}\right)}{\sqrt{\sum_{i,j=-w}^{w}\left(L(x_l + i, y + j) - \overline{L}\right)^2 \cdot \sum_{i,j=-w}^{w}\left(R(x_r + i, y + j) - \overline{R}\right)^2}}. \tag{5-6}$$

$$\overline{L} = \frac{1}{w^2}\cdot\sum_{i,j=-w}^{w} L(x_l + i, y + j) \tag{5-7}$$

$$\overline{R} = \frac{1}{w^2}\cdot\sum_{i,j=-w}^{w} R(x_r + i, y + j). \tag{5-8}$$

Here $corr_y(x_l, x_r)$ defines the normalized correlation between one pixel (x_l, y) from left image and another pixel (x_r, y) from right image. During the computing, the pixels in the neighbour window of (x_l, y) and (x_r, y) are also taken into consideration. $L(x, y)$ and $R(x, y)$ define the intensity of the pixel (x, y) respectively in the left and right image. And $\overline{L}$ and $\overline{R}$ are the average intensity in the respective neighbour windows.

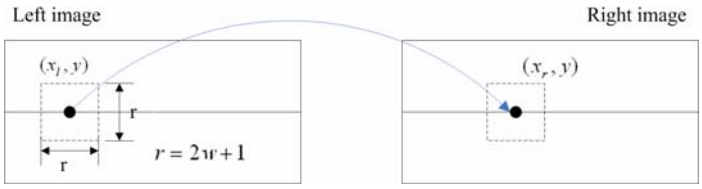
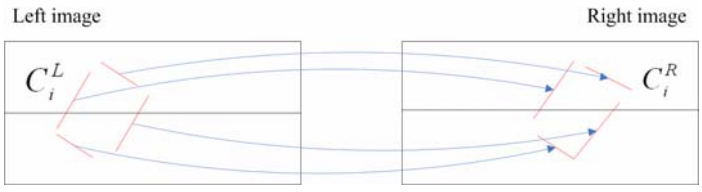
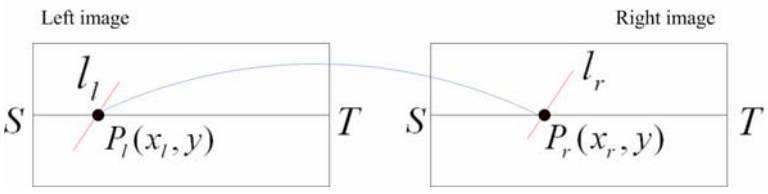


Figure 5.5 Illustration of the basic stereo matching process



(a)



(b)

Figure 5.6 Illustration of the refined rectangle based stereo matching process (a) list of matching line segments produced from the corresponding rectangles in the left and right image (b) refined matching process with a pair of matched pixel from matching line segments

Based on the given camera parameters and the correspondence between pixels, we can obtain the parallax information and then achieve the depth on every pixel. The basic stereo matching between the left and right images can be illustrated by Figure 5.5, while the refined stereo matching based on the candidate rectangles is illustrated in Figure 5.6.

5.3 Experimental Analysis

To assess the effectiveness, the proposed SAR-DPSM method has been applied to 55 pairs of aerial images. Each pair includes a left image and a right image, both in the size of 640*480. They are image blocks from a pair of large aerial images.

Besides our proposed algorithm, two other approaches are also used in the comparative experiments. The first one is the PL-DPSM method. The second one is the MR-DPSM method. In MR-DPSM method, the rectangle regions are manually marked by the observers. In comparison, the SAR-DPSM method can automatically detect the rectangle regions. In Figure 5.7, the process of manual rectangle detection and manual rectangle correspondence in MR-DPSM method is illustrated on two example image pairs. In our implementation, the reconstruction process after determining the pixel correspondence is all the same for the above three methods.

The first experiment aims to compare the accuracy of the reconstructed data by different methods. Because of the error in stereo matching, there is usually heavy noise on object boundary. Here we design two measures to evaluate the noise level in the reconstructed data.

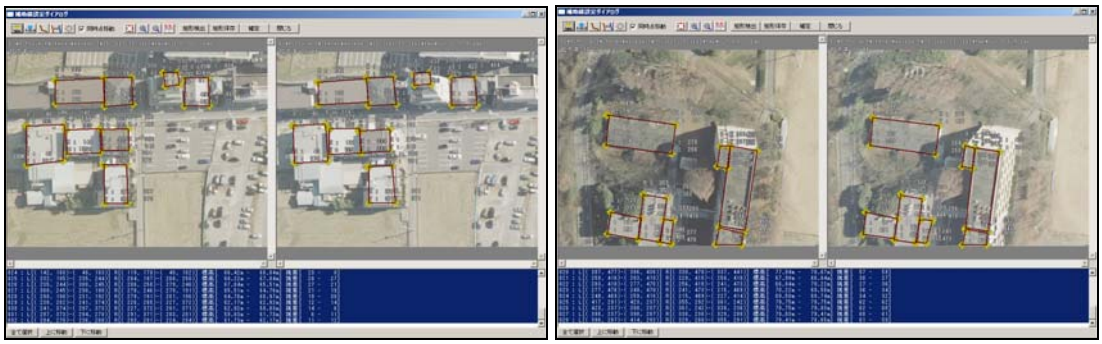


Figure 5.7 Illustration of the process of manual rectangle detection and correspondence in MR-DPSM on two example pairs.

According to the stereo matching process described in Section 2, two potential matching pixels in the left and right image always stay on the same y coordinate. Then the noise in the reconstructed data mostly exhibits as the discontinuity of the depth in horizontal direction. The coarse boundary of the objects reflects this phenomenon. To measure such noise, we implement Harr-Wavelet transform on the reconstructed data. The scaling function $\Phi(t)$ and wavelet function $\Psi(t)$ used in our experiment are shown as follows.

$$\Phi(t) = \begin{cases} 1, & 0 \le t < 1 \\ 0, & otherwise \end{cases}. \tag{5-9}$$

$$\Psi(t/2) = \begin{cases} 1, & 0 \le t < 1 \\ -1, & 1 \le t < 2. \\ 0, & otherwise \end{cases} \tag{5-10}$$

The wavelet coefficient in the vertical component is then normalized to [0,255] and assigned to each pixel. In this way, we obtain a noise image that includes the high frequency element in the horizontal direction of the reconstructed data. The noise image is firstly thresholded by 1 to a binary image, where the background is set to 0 and the noise is set to 255. The flowchart of this process is illustrated in Figure 5.8.

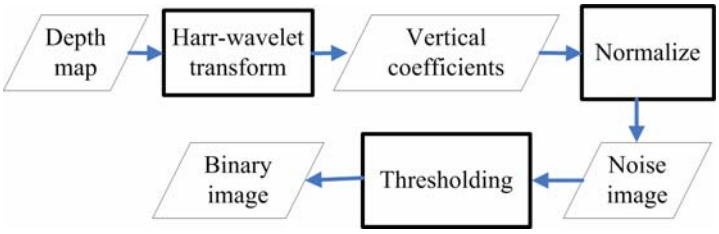


Figure 5.8 Illustration of noise level measure computing process

TABLE V-2

PERCENTAGE OF THE ENTIRE DATA WITH BETTER PERFORMANCE FOR THE FIRST ALGORITHM OVER THE SECOND ALGORITHM

	PM_{Left}	PM_{Right}	PD_{Left}	PD_{Right}
MR-DPSM vs. PL-DPSM	81.80%	89.09%	98.18%	96.36%
SAR-DPSM vs. PL-DPSM	72.73%	70.91%	85.45%	83.64%
MR-DPSM vs. SAR-DPSM	63.64%	70.91%	85.45%	87.27%

TABLE V-3

REDUCED PERCENTAGE OF NOISE LEVEL MEASURES FOR THE FIRST ALGORITHM OVER THE SECOND ALGORITHM

	RM_{Left}	RM_{Right}	RD_{Left}	RD_{Right}
MR-DPSM vs. PL-DPSM	3.79%	3.90%	16.60%	17.35%
SAR-DPSM vs. PL-DPSM	3.45%	3.21%	8.00%	7.91%
MR-DPSM vs. SAR-DPSM	7.31%	7.29%	50.14%	42.66%

The discontinuity of the depth in horizontal direction is now reflected by the horizontal bar with several consecutive noise pixels. The length of such horizontal bar, i.e., the number of consecutive noise pixels, can be taken as a measure of the noise level. Globally, we compute the average "M" and the standard deviation "D" of the length of all the horizontal bars in an image to evaluate the noise level. Smaller "M" and "D" means lower noise level and hence better accuracy of the reconstructed data.

At first, the two rectangle-based methods, MR-DPSM and SAR-DPSM, are compared with the pixel-level based method PL-DPSM. Since the candidate rectangles provide more information on building region, the two rectangle-based methods, proposed SAR-DPSM method and MR-DPSM method can achieve better performance than PL-DPSM in most of the test images. Table 5-2 shows that in what percentage of the entire test image pairs, the two rectangle based methods can achieve better performance than

PL-DPSM under different noise level measures. PM_{Left}, PM_{Right}, PD_{Left}, PD_{Right} are respectively the percentage of the entire data when MR-DPSM or SAR-DPSM can achieve better performance than PL-DPSM on the measure of M_{Left}, M_{Right}, D_{Left}, D_{Right}. Among them, M_{Left}, M_{Right} are respectively the average of horizontal bar length in the left and right image. D_{Left}, D_{Right} are respectively the standard deviation of horizontal bar length in the left and right image. In Table 5-2, we can find out that proposed SAR-DPSM method can achieve better accuracy in at most 85.45% of the entire test data, while MR-DPSM method can achieve in at most 98.18% of all the data. Compared with proposed SAR-DPSM method, MR-DPSM method can achieve in at most 87.27% of all the data.

We further provide the reduced percentage on the value of different noise level measures for the two rectangle-based methods compared with PL-DPSM. In Table 5-3, RM_{Left}, RM_{Right}, RD_{Left}, RD_{Right} define the reduced percentage on the value of each measure like M_{Left}, M_{Right}, D_{Left}, D_{Right} compared with PL-DPSM method. On the average, MR-DPSM performs better than SAR-DPSM. The proposed SAR-DPSM method can achieve at most 8.00% improvement among the entire test data compared with PL-DPSM, while MR-DPSM method can achieve at most 17.35% improvement. Compared with proposed SAR-DPSM method, MR-DPSM method can achieve at most 50.14% improvement.

From Table 5-2 and Table 5-3, it is also clear that for both of the rectangle based methods, improvement on D_{Left}, D_{Right} is much more evident than on M_{Left}, M_{Right}. It is because the information of rectangle helps to improve the accuracy of reconstruction on building boundaries, the distribution of the noise level value becomes less scattered.

In order to show the performance of the three methods in different cases, we also compare the value of M_{Left} and D_{Left} on the left image of 10 example image pairs by three methods in Figure 5.9 and Figure 5.10. These left images are shown in Figure 5.11. The block indices, from one to ten, stand for the image blocks “4_26, 5_18, 6_6, 6_16, 7_24,

8_15, 8_28, 9_17, 10_5, 22_27" in order. Since every image is a block cut from a large aerial image, the block index stands for the location of the image block. That is to say, "4_26" block is horizontally the 4th and vertically the 26th block in the whole image.

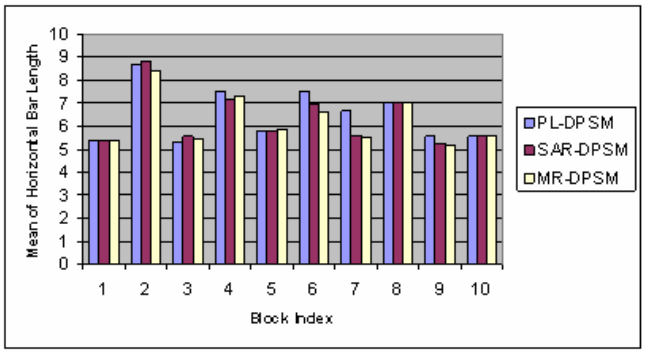


Figure 5.9 Comparison on M_{Left} among three methods of PL-DPSM, SAR-DPSM and MR-DPSM.

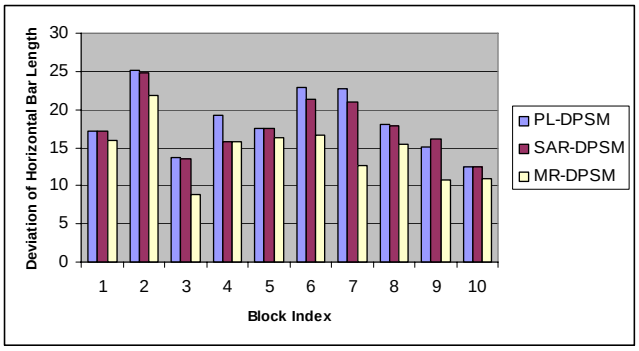


Figure 5.10 Comparison on D_{Left} among three methods of PL-DPSM, SAR-DPSM and MR-DPSM.

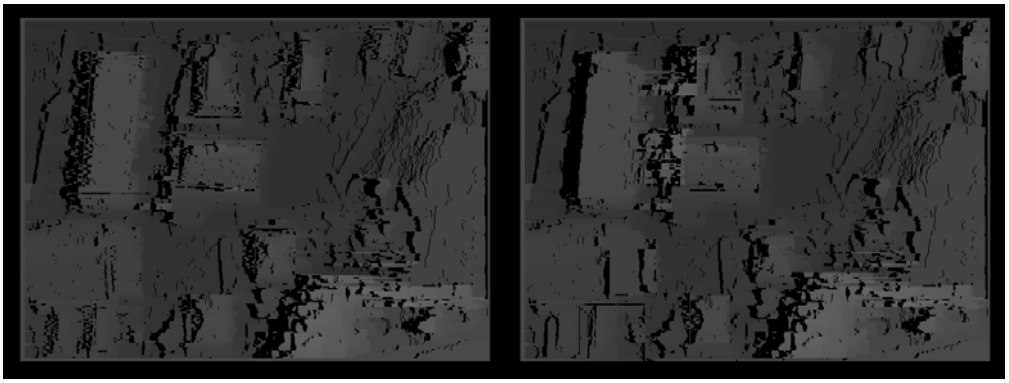
We can see that MR-DPSM performs better than SAR-DPSM in most cases since there is more human instruction in MR-DPSM. There are two cases when MR-DPSM and SAR-DPSM do not perform better than PL-DPSM. When the rectangle detection module fails to mark any rectangle region in a test image, the performance of PL-DPSM and SAR-DPSM are the same, like in "4_26" image block. In the image block of "5_18, 6_6,

7_24, 9_17, 22_27", MR-DPSM or SAR-DPSM performs worse than PL-DPSM. After observing these images, we find out that in "5_18, 6_6, 9_17", there are many small houses. In "7_24 and 22_27", there are buildings in irregular shape. These conditions lead to wrong detection of rectangles and thus less accurate reconstructed data. In Figure 5.12 and Figure 5.13, the reconstructed data of the three methods in the image block "4_26" and "6_16" is shown.

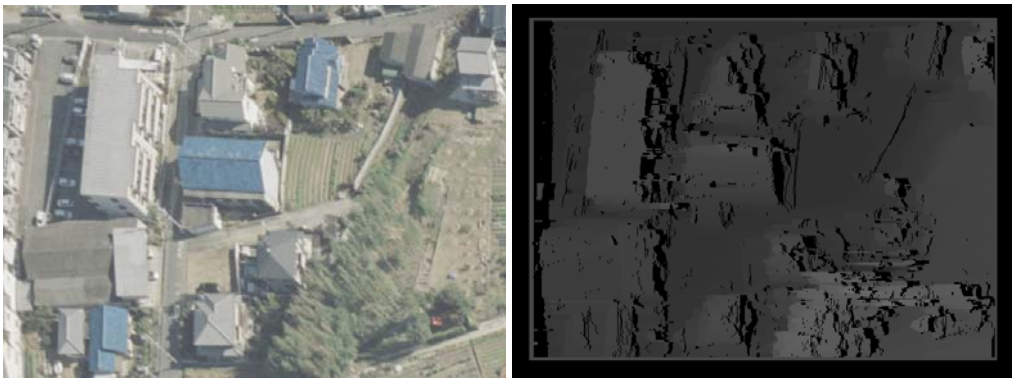
Figure 5.11 Illustration of the 10 example image of the left image among the 55 testing image pairs. From left to right and then from the above to below, they are "4_26, 5_18, 6_6, 6_16, 7_24, 8_15, 8_28, 9_17, 10_5, 22_27" in order.

According to the above analysis, MR-DPSM can achieve better performance than the proposed SAR-DPSM in most cases. On the other hand, there is more work for the observers in MR-DPSM. To illustrate the advantage of the proposed SAR-DPSM method in efficiency, the second experiment is carried out on comparing the manual working time in SAR-DPSM and MR-DPSM.

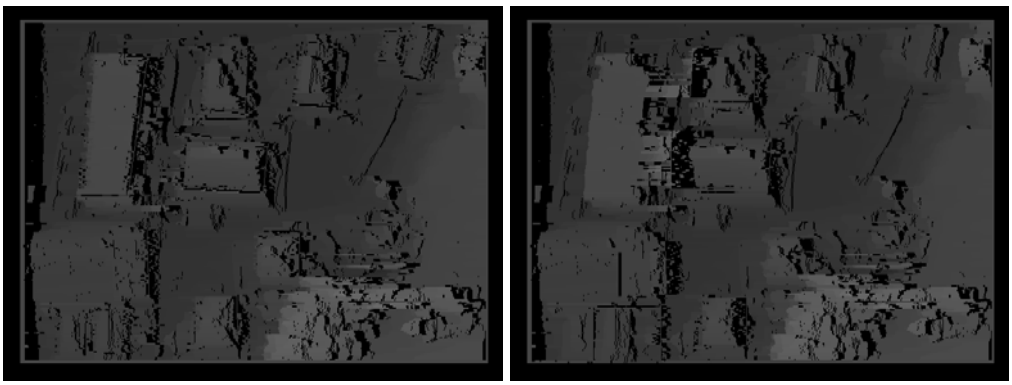
(a) (b)



(c) (d)

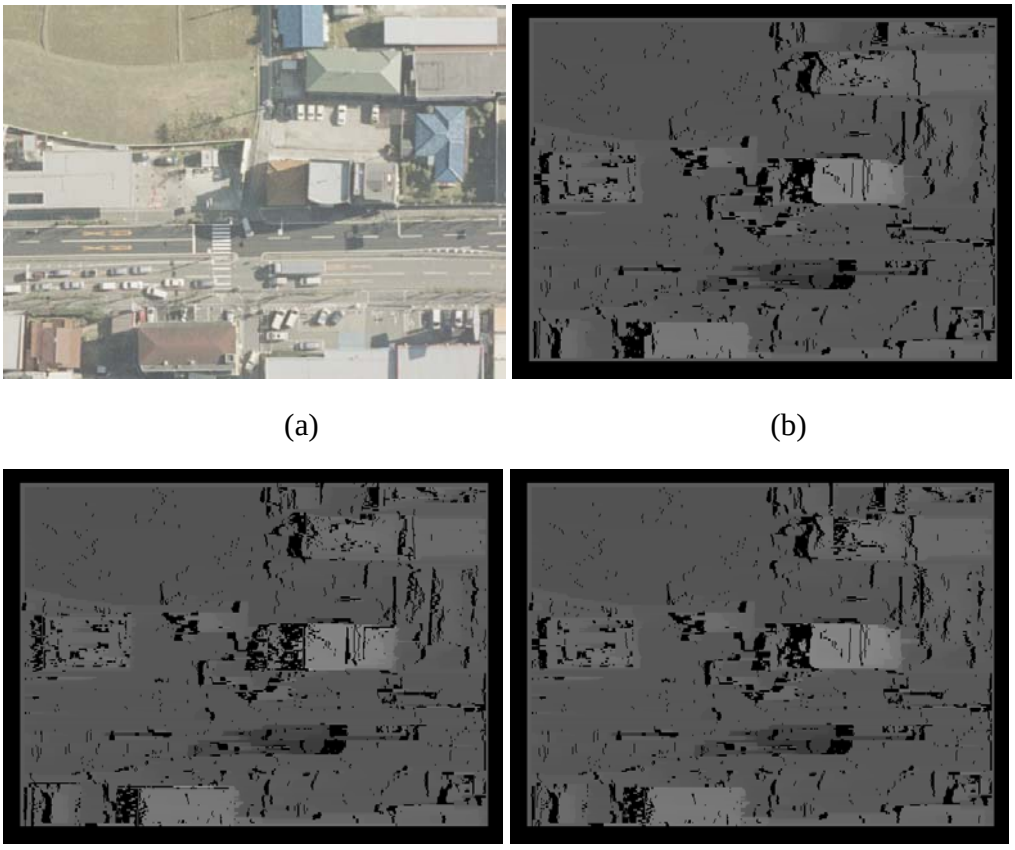


(e) (f)



(g) (h)

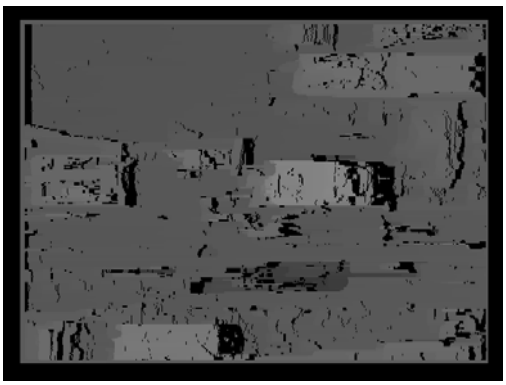
Figure 5.12 Comparison on the reconstructed result between three methods with the test image pair of image block “4_26”. (a) the left input image (e) the right input image (b, f) the reconstructed data from PL-DPSM method (c, g) the reconstructed data from MR-DPSM method (f, h) the reconstructed data from SAR-DPSM method



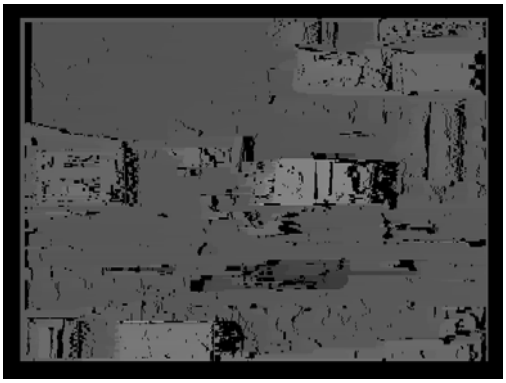
(a) (b)

(c) (d)

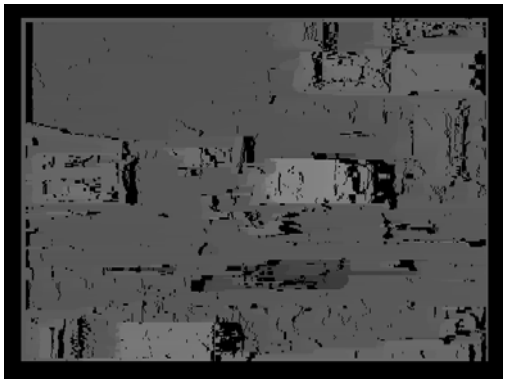
(e)



(f)



(g)



(h)

Figure 5.13 Comparison on the reconstructed result between three methods with the test image pair of image block “6_16”. (a) the left input image (e) the right input image (b, f) the reconstructed data from PL-DPSM method (c, g) the reconstructed data from MR-DPSM method (f, h) the reconstructed data from SAR-DPSM method

For SAR-DPSM method, candidate rectangles are automatically detected but manually corresponded. But for MR-DPSM method, the above two processes have to been fulfilled all by the observers. For 53 pairs out of totally 55 pairs of test images, it takes less time for the observers when perform SAR-DPSM method. Hence SAR-DPSM method needs less manual work than MR-DPSM in 94.55% of all the image pairs. The average working time for all the 55 pairs in SAR-DPSM method is 142.96 seconds while

that for MR-DPSM method is 277.36 seconds. The average reduced percentage on the working time for every image pair is 46.54%. That is to say, we can spend only about half of the working time in MR-DPSM method but still achieve graceful performance by the proposed SAR-DPSM method. In Table 5-4, the working time of the two methods in 10 example image pairs is compared. At the same time, the number of detected candidate rectangles respectively in MR-DPSM and SAR-DPSM are also shown in Table 5-5. The candidate rectangles in MR-DPSM are greatly reduced in SAR-DPSM and then the computing burden can be eased. But the performance does not deteriorate too much according to the above analysis on noise level.

TABLE V-4

MANUAL WORKING TIME IN MR-DPSM AND SAR-DPSM

Block Index	Time (Seconds)		Reduced Percentage
	MR-DPSM	SAR-DPSM	
4_26	390	161	58.7%
5_18	289	203	29.8%
6_6	489	177	63.8%
6_16	369	149	59.6%
7_24	293	39	86.7%
8_15	327	188	42.5%
8_28	343	126	63.3%
9_17	232	153	34.0%
10_5	175	151	13.7%
22_27	199	85	57.3%
Average	310.6	143.2	50.94%

TABLE V-5

NUMBER OF CANDIDATE RECTANGLES IN MR-DPSM AND SAR-DPSM

Block Index	Number of Rectangles	
	MR-DPSM	SAR-DPSM
4_26	12	5
5_18	12	6
6_6	17	6
6_16	13	4
7_24	10	2
8_15	7	5
8_28	14	3
9_17	9	4
10_5	5	4
22_27	6	2

TABLE V-6

TIME COST OF THREE PROCESSES

Block Index	Time (Seconds)		
	Rectangle Detection	DPSM with Rectangles	DPSM without Rectangles
4_26	12.7	108.6	143.2
5_10	13.4	139.6	149.3
7_24	8.1	121.0	150.4
13_24	12.8	146.7	159.2
18_22	13.3	111.8	147.9
Average	12.06	125.54	150.00

Finally, we will discuss the computational complexity of the proposed SAR-DPSM method. All the experiments are carried out under Microsoft Windows Server 2003 SP1 on a NEC Express 5800 PC equipped with Intel Pentium IV 3.0GHz processor and 2.5GB RAM. Except the manual rectangle corresponding process, the proposed SAR-DPSM method include two parts: rectangle detection and DPSM with rectangles. Table 5-6 shows the time cost of the above two processes on 5 example image pairs. In comparison, the time cost of “DPSM without rectangle” module in PL-DPSM is also shown.

Averagely, on the test image with size of 640*480, it takes about 12.06 seconds in rectangle detection. “DPSM with rectangles” in SAR-DPSM spends about 125.54 seconds while “DPSM without rectangles” in PL-DPSM spends about 150.00 seconds. “DPSM with rectangles” has smaller time complexity because with the list of matching line segments, the searching range of matching point for every pixel is reduced.

5.4 Conclusions

The high complexity of the urban scene, such as the crowded houses, the cluttered background and the various building structures, affects the performance of the traditional stereo matching methods especially on the boundaries of building regions. Here we propose a novel rectangle-based stereo matching method, SAR-DPSM to address the above challenge. The candidate rectangles potentially corresponding to every building

region are detected in both left and right images. Then after manual correspondence, the stereo matching is carried out to obtain the 3D data with lower noise level.

Comparative experiments show that the proposed SAR-DPSM can perform better than PL-DPSM at 85.45% of the entire test data and achieve at most 8.00% improvement among the entire test data compared with PL-DPSM. The MR-DPSM performs better than SAR-DPSM at 87.27% of the entire test data and achieves at most 50.14% improvement among the entire test data due to the prior-knowledge from the human beings. But the proposed SAR-DPSM can save averagely 50.94% manual working time compared with MR-DPSM with graceful performance.

A drawback of the proposed SAR-DPSM method is that it is semi-automatic. Then, how to automatically build correspondence between the rectangles from left and right image will be an important work in the future. Moreover, the accuracy of the candidate rectangle detection greatly interferes with the afterwards stereo matching. How to obtain the candidate rectangles that are better matching with the boundary of building regions is another focus of our future work.

6 Conclusion

In this dissertation, the problem of robust salient contour detection is mainly discussed through the steps of salient edge detection, salient contour segment detection and salient object contour detection. The robust salient contour detection methods are further applied in the 3D building reconstruction problem to verify its effectiveness.

Firstly, a multi-scale transition region based edge detector is proposed. It is constructed based on the intrinsic property of natural images, the existence of “transition region”. The transition region analysis helps to facilitate the edge detection process to detect more precise edge pixels. In order to capture all the texture patterns, we decide to further combine the information from multiple scales. The information integration process over all the scales is carried out on by analyzing the relatively stable distribution characteristics of the real edge pixels in the transition region across scales. The proposed algorithm is evaluated through comparison with a fully automatic eigen-analysis based color edge detector and the well-known Canny edge detector. Experimental results show that compared with the eigen-analysis based color edge detector, the proposed method can produce better edge detection results with better connection and also keep the main information in the image. Under the edge coherence measure, it can be verified that the proposed method performs better than Canny edge detector in general case.

Secondly, in order to capture the salient line segments in the image, a scale invariance based line detector is proposed. Firstly, a refined Nevatia-Babu line detector and is firstly proposed. The refined Nevatia-Babu line detector can finish the line detection with only one step as simply linking the edge pixels with the connectivity property of basic line patterns. In comparison, the traditional Nevatia-Babu line detector is carried out in two steps, e.g. edge linking and contour curve breaking. At the same time, the contour curve breaking involves complex computing of the curve curvature. Because

of the simplicity of the computing, the refined line detector can efficiently detect the line segments in the image even for the image in large size. Experimental results show that the refined line detector is rather efficient and can finish the line segment detection in 20 seconds for an aerial image with size of 3600*3800. In order to suppress the responses from the texture region and also keep the responses from the object contour, a scale invariance based line detector is designed. A multi-scale image pyramid is firstly built by repeatedly smoothing and sub-sampling the input image. Compared with the single scale detector, the multi-scale is a little more time-consuming but can produce better result. Large percentage of the line segments from texture region is suppressed and the extracted line segments are with better continuity. We make comparison between the proposed method and the PPHT method. From the experimental results on the synthetic images, we can find out that proposed method can obtain better line detection results. Firstly, the proposed method can detect line segments in lower redundancy with no redundancy in 3 example images and only 0.67 redundancy level in one example image. But the redundancy levels of the detection results from PPHT method on the same example images are from 2.02 to 23.25. At the same time, the proposed method is much more efficient than PPHT method especially for the image including many line segments. The proposed method can finish the line detection in 3 seconds for an image including 39 line segments while the PPHT spends 80 seconds. Compared with the proposed method, the PPHT method spends at least 2 times and at most 26 times of the processing time of the proposed method.

Thirdly, a CDT graph based contour detection algorithm is proposed. The proposed method adopts the CDT graph and then the computing complexity can be enormously reduced because the possible links are reduced to very low number. At the same time, there is no loss of potential object contours with this simplified CDT graph structure. Since there are only the salient line segments detected by the proposed multi-scale contour segment detection method, we can transfer the perceptual grouping into a two-

class classification problem by classifying all the CDT edges into “contour link” cluster or “non-contour link” cluster. The CDT edges are classified according to the value of their saliency measures. Here a novel saliency measure based on the Snakes model is proposed. The saliency measure includes the consideration both on the Gestalt laws and on the image information. Experimental results show that the proposed method can derive contours with better continuity and saliency than the former method under the condition of cluttered background. On the given 7 experimental images from BSD, the improvement percentage on the overall saliency of the detected contours from the proposed method over the former method is at least 18% and at most 165% for the experimental image. The improvement percentage on the detection rate of the proposed method over the former method is at least 15% and at most 68% for the experimental image.

At last, an application field of salient contour detection is discussed, that is, 3D building reconstruction in the aerial image. The high complexity of the urban scene affects the performance of the traditional stereo matching methods especially on the boundaries of building regions. The proposed SAR-DPSM method can well address the above challenge. The stereo image pairs are matched through the list of the matching line segments from the matching rectangles. The proposed SAR-DPSM method can achieve better accuracy in at most 85.45% of the entire test data compared with the PL-DPSM method. Compared with proposed SAR-DPSM method, MR-DPSM method can achieve in at most 87.27% of all the data. In addition, the proposed SAR-DPSM method can achieve at most 8.00% improvement among the entire test data compared with PL-DPSM, while MR-DPSM method can achieve at most 50.14% improvement compared with proposed SAR-DPSM method. MR-DPSM can achieve better performance than the proposed SAR-DPSM in most cases. But the automatic rectangle contour detection in the proposed method helps to reduce the manual working time. Compared with MR-DPSM, SAR-DPSM can save averagely 50.94% manual working time. Therefore, the proposed

method can save a lot of human working time with graceful performance.

In conclusions, we discuss how to detect salient image features such as edges, contour segments and contours in order to detect the interested objects in the image. The proposed algorithms can detect salient features robustly especially under the cluttered background scene compared with former methods.

Acknowledgement

First of all, I would like to thank my supervisor, Professor Satoshi Goto, for his guidance and instructions. His insight guides me in the whole process of my research. At the same time, I would like to express my gratitude to Associate Professor Takeshi Ikenaga and Professor Yoshimura for their concise and accurate advice to me.

I would like to also give special thanks to Mr. Hideo Shimazu, Mr. Makoto Iwata, Mr. Hiroyuki Yagyu, Mr. Toru Miyazaki and Mr. Hirokazu Koizumi from System Technology Laboratory, NEC System and Technologies, Ltd. for their generous technical support and instructions during my research.

Thanks to the excellent research collaborators I have, Mr. Yang Song, Mr. Xianghui Wei, Mr. Yibo Fan, Mr. Zhenxing Chen, Mr. Yao Ma, Mr. Jun Wang, Ms. Wen Ji, Mr. Yan Zhuang, Mr. Tianruo Zhang and so on. The discussions with you are fruitful and important to me. Thank you very much. Thanks to all the members in the research lab, I enjoy the time with you.

Finally, I would like to express my thanks to my family and friends, for their support all the time. This doctor dissertation would not have been possible without them. Thanks to my parents and my sister for your love, support and understanding. I can not walk so far without you. Thank you very much.

Reference

[1] J. Gray, “What’s Next? A Few Remaining Problems in Information Technology”, Turing Award Lecture, 1998.

[2] D. Marr, “Vision: A Computational Investigation into the Human Representation and Processing of Visual Information”, W.H. Freeman, New York, 1982.

[3] J. Mundy, T. Binford, T. Boult, A. Hanson, R. Beveridge, R. Haralick, V. Ramesh, C. Kohl, D. Lawton, D. Morgan, K. Price, T. Strat, “The Image Understanding Environment Program”, in Proc. CVPR’92, pp. 406-416, Jun. 1992.

[4] C. Kohl, J. Mundy, “The Development of the Image Understanding Environment”, in Proc. CVPR’94, pp. 443-447, Jun. 1994.

[5] G. R. Bradski, “The OpenCV Library”, Dr. Dobb’s Journal, pp. 120-125, Nov. 2000.

[6] “Semantic Robot Vision Challenge (SRVC) Research Competition”, http://www.semantic-robot-vision-challenge.org/

[7] J. Bigun, “Vision with Direction: A Systematic Introduction to Image Processing and Computer Vision”, Springer, 2006.

[8] L. Itti, C. Koch and E. Niebur, “A Model of Saliency-Based Visual Attention for Rapid Scene Analysis”, IEEE Trans. on PAMI, vol. 20, no. 11, pp. 1254-1259, 1998.

[9] M. K. Kapadia, G. Westheimer and C. D. Gilbert, “Spatial Distribution of Contextual Interactions in Primary Visual Cortex and in Visual Perception”, Journal of Neurophysiology, vol. 84, no. 4, pp. 2048-2062, 2002.

[10] J. J. Knierim and D. C. van Essen, “Neuronal Responses to Static Texture Patterns in Area V1 of the Alert Macaque Monkey”, Journal of Neurophysiology, vol. 67, no. 4, pp. 961-980, 1992.

[11] C. Grigorescu, N. Petkov and M. A. Westenberg, “Contour and Boundary Detection Improved by Surround Suppression of Texture Edges”, Image and Vision

Computing, vol. 22, pp. 609-622, 2004.

[12] S.C. Yen and L. H. Finkel, "Salient Contour Extraction by Temporal Binding in a Cortically-Based Network", Advances in Neural Information Processing Systems, vol. 10, pp. 915-921, 1997.

[13] V. Vonikakis, A. Gasteratos and I. Andreadis, “Enhancement of Perceptually Salient Contours Using a Parallel Artificial Cortical Network”, Biological Cybernetics, vol. 94, pp. 192-214, 2006.

[14] I. Rock and S. Palmer, “The Legacy of Gestalt Psychology”, Scientific Am., pp. 84-90, 1990.

[15] S. C. Zhu, “Embedding Gestalt Laws in Markov Random Fields”, IEEE Trans. on PAMI, vol. 21, no. 11, pp. 1170-1187, 1999.

[16] T. Tversky, W. S. Geisler and J. S. Perry, “Contour Grouping: Closure Effects are Explained by Good Continuation and Proximity”, Vision Research, vol. 44, pp. 2769-2777, 2004.

[17] L. Loss, G. Bebis, M. Nicolescu and A. Skourikhine, “Perceptual Grouping Based on Iterative Multi-scale Tensor Voting”, in Proc. ISVC’06, pp. 870-881, 2006.

[18] L. R. Williams and K. K. Thornber, “A Comparison of Measures for Detecting Natural Shapes in Cluttered Backgrounds”, Int. J. Computer Vision, vol. 34, no. 2/3, pp. 81-96, 2000.

[19] S. Mahamud, K. K. Thornber, L. R. Williams, “Segmentation of Salient Closed Contours from Real Images”, in Proc. ICCV’99, pp. 891-897, 1999.

[20] D. Martin, C. Fowlkes, D. Tal and J. Malik, “A Database of Human Segmented Natural Images and Its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics”, In Proc. ICCV’01, vol. 2, pp. 416-423, 2001.

[21] F. J. Estrada and A. D. Jepson, “Perceptual Grouping for Contour Extraction”, In ICPR’04, vol. 2, pp. 32-35, Aug. 2004.

[22] F. J. Estrada and J. H. Elder, “Multi-Scale Contour Extraction Based on Natural

Image Statistics", in Proc. CVPRW'06, pp. 183-183, Jun. 2006.

[23] J. Wang, T. Ikenaga and S. Goto, "Scale Invariance Based Salient Contour Extraction", J. Institute of Image Electronics Engineer of Japan, vol. 36, no. 5, pp. 710-720, 2007.

[24] R. Nevatia and K. R. Babu, "Linear Feature Extraction and Description", Computer Graphics and Image Processing, vol. 13, pp. 257-269, 1980.

[25] P. Kahn, A. Winkler, C. Y. Chong, "Perceptual Grouping as Energy Minimization", In Proc. ICSMC'90, pp. 542-546, Nov. 1990.

[26] J. S. Stahl and S. Wang, "Convex Grouping Combining Boundary and Region Inormation", in Proc. ICCV'05, vol. 2, pp. 946-953, Oct. 2005.

[27] S. Krishnamachari and R. Chellappa, "Delineating Bildings by Gouping Lines with MRFs", IEEE Trans. Image Processing, vol. 5, no. 1, pp. 164-168, 1996.

[28] A. Massmann, S. Posch, G. Sagerer and D. Schluter, "Using Markov random fields for contour-based grouping", in Proc. ICIP'97, vol. 2, pp. 207-210, 1997.

[29] G. Guy and G. Medioni, "Inferring Global Perceptual Contours from Local Features", In Proc. CVPR'93, pp. 786-787, New York, USA, Jun. 1993.

[30] H. Wang and J. Oliensis, "Salient Contour Detection Using a Global Contour Discontinuity Measurement", In Proc. CVPRW'06, pp. 190-190, Jun. 2006.

[31] J. Princen, J. Illingworth and J. Kittler, "A Hierarchical Approach to Line Extraction", In Proc. CVPR'89, pp. 92-97, Jun.1989.

[32] H. K. Hartline, "Inhibition of Activity of Visual Receptors by Illuminating Nearby Retinal Areas in the Limulus Eye", Federation Proc., vol. 8, pp. 69, 1949.

[33] A. Rosenfeld and M. Thurston, "Edge and Curve Detection for Visual Scene Analysis", IEEE Trans. on Computers, pp. 562-569, May. 1971.

[34] J. Canny, "A Computational Approach to Edge Detection", IEEE Trans. on PAMI, vol. 8, no. 6, pp. 679-698, 1986.

[35] M. Tabb and N. Ahuja, "Multiscale Image Segmentation by Integrated Edge and

Region Detection", IEEE Trans. on Image Processing, vol. 6, no. 5, pp. 642-655, 1997.

[36] T. Lindeberg, "Edge Detection and Ridge Detection with Automatic Scale Selection", Int. J. Computer Vision, vol. 30, no. 2, pp. 117-156, 1998.

[37] D. Marr and E. Hildreth, "Theory of Edge Detection", In Proc. Royal Society of London, Series B, Biological Sciences, pp. 187-217, Feb. 1980.

[38] F. Bergholm, "Edge Focusing", IEEE Trans. on PAMI, vol. 9, no. 6, pp. 726-741,1987.

[39] J. H. Elder and S. W. Zucker, "Local Scale Control for Edge Detection and Blur Estimation", IEEE Trans. on PAMI, vol. 20, no. 7, pp. 699-716, 1998.

[40] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion", IEEE Trans. on PAMI, vol. 12, no. 7, pp. 629-639, 1990.

[41] J. Scharacanski and A. N. Venetsanopoulos, "Edge Detection of Color Images Using Directional Operators", IEEE Trans. on CSVT, vol. 7, no. 2, pp. 397-401, 1997.

[42] T. Kanade, "Image Understanding Research at CMU", In Proc. IUW'87, vol. 2, pp. 32-40, 1987.

[43] P. E. Trahania and A. N. Venetsanopoulos, "Color Edge Detection Using Vector Order Statistics", IEEE. Trans. on Image Processing, vol. 2, no. 2, pp. 259-264, 1993.

[44] M. A. Ruzon and C. Tomasi, "Edge, Junction, and Corner Detection Using Color Distributions", IEEE Trans. on PAMI, vol. 23, no. 11, pp. 1281-1295, 2001.

[45] A. Koschan and M. Abidi, "Detection and Classification of Edges in Color Images", Signal Processing Magazine, Special Issue on Color Image Processing, vol. 22, no. 1, pp. 64-73, 2005.

[46] T. Lindeberg, "Scale-Space Theory in Computer Vision", Kluwer Academic, Netherlands, 1994.

[47] C. Bredfeldt and D. Ringach, "Dynamics of Spatial Frequency Tuning of Macaque V1", J. Neuroscience, pp. 1976-1984, 2002.

[48] G. D. Joshi and J. Sivaswamy, "Multi-scale Approach to Salient Contour Extraction", In Proc. ICCR'05, pp. 186-193, 2005.

[49] X. Ren and J. Malik, "A Probabilistic Multi-scale Model for Contour Completion Based on Image Statistics", In Proc. ECCV'02, vol. 1, pp. 312-327, 2002.

[50] G. Sharma, H. J. Trussell, "Digital Color Imaging", IEEE Trans. on Image Processing, vol. 6, no. 7, pp. 901-932, 1997.

[51] C. Galamhos, J. Matas, J. Kittler, "Progressive probabilistic Hough Transform for Line Detection", Proc. of CVPR'99, vol. 1, pp. 560, Jun. 1999.

[52] Y. Liu, T. Ikenaga, S. Goto, "A Fully Automatic Approach of Color Image Edge Detection", In Proc. SMC'06, pp. 1609-1612, Taiwan, 2006.

[53] X. Zhang, Z Song, Y. Wang and H. Wang, "Color Quantization of Digital Images", In Proc. PCM'05, LNCS 3768, pp. 653-664, 2005.

[54] X. Zhang, H. Wang, C. Gao and Y. Wang, "Color Distribution Evenness and Its Application to Color-Texture Segmentation", In Proc. ICME'07, pp. 1655-1658, Jul. 2007.

[55] L. Kitchen and A. Rosenfeld, "Edge Evaluation Using Local Edge Coherence", IEEE Trans. on SMC, vol. 11, no. 9, pp. 597-605, 1981.

[56] James Allan Heather and Xuedong Yang. "Spatial decomposition of hough transform", The 2nd Canadian Conference on Computer and Robot Vision, pp. 476-482, 2005.

[57] B. Sumengen and B. S. Manjunath, "Multi-Scale Edge Detection and Image Segmentation", In Proc. EUSIPCO'05, Sep. 2005.

[58] W. Li, V. Piech, C. D. Gilbert, "Contour Saliency in Primary Visual Cortex", Neuron, vol. 50, pp. 951-962, 2006.

[59] S. Wang, T. Kubota, J.M. Siskind, J. Wang, " Salient Closed Boundary Extraction with Ratio contour", IEEE Trans. PAMI, vol. 27, no. 4, pp 546-561, 2005.

[60] L.D. Cohen, "Multiple Contour Finding and Perceptual Grouping Using Minimal

Paths”, vol. 14, pp. 225-236, Jour. Mathematical Imaging and Vision, 2001.

[61] A. Shashua and S. Ullman, “Structural Saliency: The Detection of Globally Salient Structures Using a Locally Connected Network”, Proc. of 2nd Int’l Conf. Computer Vision, Clearwater, FL, 1988.

[62] T. Alter and R. Basri, “Extracting Salient Contours from Images: An Analysis of the Saliency Network”, Proc. IEEE Conf. Computer Vision and Pattern Recognition, San Francisco, CA, pp. 13-20, 1996.

[63] Y. V. Venkatesh, S. K. Raja, N. Ramaya, “ Multiple Contour Extraction From Graylevel Images Using an Artificial Neural Network“, vol. 25, no. 4, pp. 892-899, 2006.

[64] A. Zlotnick, “Locating Corners in Noisy Curves by Delineating Imperfect Sequences”, Technical Report CMU-CS-88-199, Carnegie Mellon University, Dec. 1988.

[65] D. Lagunovsky, S. Ablameyko, “Straight-line-based primitive extraction in grey-scale object recognition”, Pattern Recognition Letters, Vol. 20, No. 10, pp. 1005-1014, 1999.

[66] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints" , Int. J. Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.

[67] F. Attneave, "Some Informational Aspects of Visual Perception", Psychological Review, vol. 61, pp. 183-193, 1954.

[68] P. J. Burt, “Fast Filter Transforms for Image Processing”, Computer Graphics and Image Processing, vol. 16, pp. 20-51, 1981.

[69] S.M. Konishi, A.L. Yuille, and J.M. Coughlan, "A Statistical Approach to Multi-Scale Edge Detection", Image and Vision Computing, vol. 21, pp. 1-10, 2003.

[70] D. M. McKeown, “Toward Automatic Cartographic Feature Extraction”, Mapping and Spatial Modelling for Navigation, L. F. Pau, ed., NATO ASI series, vol. F65, pp. 149-180, 1990.

[71] W. Jacobs. “Robust and efficient detection of salient convex groups”. IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 18, No. 1, pp. 23-37, Jan. 1996.

[72] D. Martin, C. Fowlkes, J. Malik, "Learning to Detect Natural Image Boundaries Using Local Brightness, Color and Texture Cues", IEEE Trans. PAMI, vol. 25, no. 5, pp. 530-549, 2004.

[73] J. Shotton, A. Blake, and R. Cipolla, "Contour-Based Learning for Object Detection", In Proc. ICCV, vol. 1, pp. 503-510, 2005.

[74] A. Thayananthan, B. Stenger, P.H.S. Torr and R. Cipolla, "Shape Context and Chamfer Matching in Cluttered Scenes", In Proc. CVPR, vol. 1, pp. 127-133, 2003.

[75] A. Yilmaz, X. Li and M. Shah, "Contour-Based Object Tracking with Occlusion Handling in Video Acquired Using Mobile Cameras", IEEE Trans. PAMI, vol. 26, no. 11, pp. 1531-1536, 2004.

[76] K. Hariharakrishnan and D. Schonfeld, "Fast Object Tracking Using Adpative Block Matching", IEEE Trans. Multimedia, vol. 7, no. 5, pp. 853-859, 2005.

[77] T. Gevers, "Robust Segmentation and Tracking of Colored Objects in Video", IEEE Trans. CSVT, vol. 14, no. 6, pp. 776-781, 2004.

[78] X. Li, "Contour Adaptive Image Coding", In Proc. ICASSP, vol. 2, pp. 197-200, 2005.

[79] R.A. Lane and N. A. Thacker, “Tutorial: overview of stereo matching research”, Tina Memo, vol. 1, pp. 1-10, 1994.

[80] G. Wei and G. Hirzinger, “Intensity and feature based stereo matching by disparity parameterization“, in Proc. ICCV’98, pp. 1035-1040, Jan 1998.

[81] T. Poggio, V. Torre and C. Koch, “Computational vision and regularization”, Nature, vol. 317, no. 26, pp. 314-319, 1985.

[82] M. Okutomi and T. Kanade, “A locally adaptive window for signal matching”, Intl. J. Computer Vision, vol. 7, no. 2, pp. 143-162, 1992.

[83] L. Di Stefano, M. Marchionni and S. Mattocia, “A fast area-based stereo matching algorithm”, Image and Vision Computing, vol. 22, no. 12, pp. 983-1005, 2004.

[84] P.F. McLauchlan, “Recovery of textured surfaces using stereo vision”, PhD Thesis, AIVRU, University of Sheffield, 1990.

[85] S. Hongo, N. Sonehara and I. Yoroizawa, “Edge-based binocular stereopsis algorithm: a matching mechanism with probabilistic feedback”, Neural Networks, vol. 9, no. 3, pp. 379-395, 1996.

[86] Y. Ohta and T. Kanade, “Stereo by intra- and inter-scaline search using dynamic programming”, IEEE Trans. on PAMI, vol. 7, no. 2, 1985.

[87] S.H. Lee and J.J. Leou, “A dynamic-programming approach to line segment matching in stereo vison”, Pattern Recognition, vol. 27, no. 8, pp. 961-986, 1994.

[88] N. M. Nasrabadi, “A stereo vision technique using curve-segments and relaxation matching”, IEEE Trans. on PAMI, vol. 14, no. 5, pp. 566-572, 1992.

[89] V.V. Vinod and S. Ghose, “Point matching using asymmetric neural networks”, Pattern Recognition, vol. 26, no. 8, pp. 1207-1214, 1993.

[90] J.P.P. Starink and E. Backer, “Finding point correspondences using simulated annealing”, Pattern Recognition, vol. 28, no. 2, pp. 231-240, 1995.

[91] H. S. Lim and T. O. Binford, “Stereo correspondence: A hierarchical approach”, in Proc. Image Understanding Workshop, 1987.

[92] G.Q. Wei, W. Brauer and G. Hirzinger, “Intensity-based and gradient based stereo matching using hierarchical Gaussian Basis Functions”, IEEE Trans. on PAMI, vol. 20, no. 11, pp. 1143-1160, 1998.

[93] J. Wang, T. Miyazaki, H. Koizumi, M. Iwata, J. Chong, H. Yagyu, H. Shimazu, T. Ikenaga and S. Goto, “Rectangle Region Based Stereo Matching for Building Reconstruction”, Journal of Ubiquitous Convergence Technology, Nov. 2007 (Accepted, to be published)

[94] J. Wang, S. Goto, K. Kunieda, M. Iwata, H. Koizumi, H. Shimazu, T. Ikenaga,

"Geometric Primitives Detection in Aerial Image", In Proc. ICCI’06, Beijing, China, 2006.

[95] J. Wang, T. Ikenaga, S. Goto, K. Kunieda, M. Iwata, H. Koizumi, H. Shimazu, "A New Mutliscale Line Detection Approach for Aerial Image with Complex Scene", In Proc. APCCAS’06, Singapore, PP. 1970-1973, 2006.

[96] J. Wang, M. Iwata, H. Koizumi, H. Shimazu, S. Goto, T. Ikenaga, “Multi-scale Fragemented Edges Grouping for Monocular Building Extraction”, the 20th IEICE Workshop on Circuits and Systems in Karuizawa, pp. 557-562, Karuizawa, Japan, April 2007.

Publications

International Journal

[1] **J. Wang**, T. Ikenaga and S. Goto, “Scale Invariance Based Salient Contour Extraction”, J. Institute of Image Electronics Engineer of Japan, vol. 36, no. 5, pp. 710-720, 2007.

[2] **J. Wang**, T. Miyazaki, H. Koizumi, M. Iwata, J. Chong, H. Yagyu, H. Shimazu, T. Ikenaga and S. Goto, “Rectangle Region Based Stereo Matching for Building Reconstruction”, Journal of Ubiquitous Convergence Technology, Nov. 2007.

International Conference

[1] **J. Wang**, Y. Zhang, S. Goto, "Surface curvature based automatic human face feature extraction", In Proc. ISPACS’05, Hongkong, China, PP.69-72, 2005.

[2] **J. Wang**, S. Goto, K. Kunieda, M. Iwata, H. Koizumi, H. Shimazu, T. Ikenaga, "Geometric Primitives Detection in Aerial Image", In Proc. ICCI’06, Beijing, China, 2006.

[3] **J. Wang**, T. Ikenaga, S. Goto, K. Kunieda, M. Iwata, H. Koizumi, H. Shimazu, "A New Mutliscale Line Detection Approach for Aerial Image with Complex Scene", In Proc. APCCAS’06, Singapore, PP. 1970-1973, 2006.

[4] **J. Wang**, K. Kunieda, M. Iwata, H. Koizumi, H. Shimazu, T. Ikenaga, S. Goto, "Multi-resolution Analysis Based Salient Contour Extraction", In Proc. ISPACS’06, Tottori, Japan, PP. 689-692,2006.

Domestic Conference

[1] **J. Wang**, M. Iwata, H. Koizumi, H. Shimazu, S. Goto, T. Ikenaga, “Multi-scale Fragemented Edges Grouping for Monocular Building Extraction”, the 20th IEICE Workshop on Circuits and Systems in Karuizawa, pp. 557-562, Karuizawa, Japan,

April 2007.