LOCATING BUILDINGS IN URBAN AREA FROM GEOMETRICAL INFORMATION

Mr. Chaiyasit Tanchotsrinon

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy Program in Computer Science and
Information Technology
Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University
Academic Year 2016

การระบุตำแหน่งอาคารในเขตเมืองจากสารสนเทศเรขาคณิต

นายชัยสิทธิ์ ตันโชติศรีนนท์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรดุษฎีบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2559

Thesis Title	LOCATING BUILDINGS IN URBAN AREA FROM GEOMETRICAL INFORMATION
By	Mr. Chaiyasit Tanchotsrinon
Field of Study	Computer Science and Information Technology
Thesis Advisor	Assistant Professor Suphakant Phimoltares, Ph.D.
Thesis Co-Advisor	Professor Chidchanok Lursinsap, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Doctoral Degree

.....Dean of the Faculty of Science
(Associate Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

.....Chairman
(Assistant Professor Annupan Rodtook, Ph.D.)
.....Thesis Advisor
(Assistant Professor Suphakant Phimoltares, Ph.D.)
.....Thesis Co-Advisor
(Professor Chidchanok Lursinsap, Ph.D.)
.....Examiner
(Assistant Professor Saranya Maneeroj, Ph.D.)
.....Examiner
(Associate Professor Chatchawit Aporntewan, Ph.D.)
.....External Examiner
(Assistant Professor Ureerat Suksawatchon, Ph.D.)

ชัยสิทธิ์ ตันโชติศรีนนท์ : การระบุตำแหน่งอาคารในเขตเมืองจากสารสนเทศเรขาคณิต (LOCATING BUILDINGS IN URBAN AREA FROM GEOMETRICAL INFORMATION) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. ศุภกานต์ พิมลธเรศ, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ศ. ดร. ชิดชนก เหลือสินทรัพย์, 77 หน้า.

การระบุตำแหน่งของวัตถุทางภูมิศาสตร์อย่างอาคารจากภาพ เป็นงานที่ท้าทายในการประมวลผลภาพ เนื่องจากสามารถประยุกต์ใช้งานได้หลากหลายรูปแบบ เช่น การวางผังเมือง การจัดการภัยพิบัติ ดังนั้นวิทยานิพนธ์ฉบับนี้จึงมุ่งเน้นศึกษาหาวิธีตรวจหาตำแหน่งอาคารบนพื้นฐานของการกราดหาเส้นตรง สำหรับภาพมุมมองทางด้านบนงานวิจัยนี้เริ่มจากบริเวณที่คาดว่าน่าจะเป็นพื้นที่ของอาคารทั้งหมดจะถูกระบุจากภาพทดสอบ เพื่อลดระยะเวลาการประมวลผลภาพใหญ่จะถูกแบ่งออกเป็นภาพย่อยตามบริเวณที่คาดหวัง ส่วนขนาดที่เหมาะสมของภาพย่อยประมาณโดยใช้ฮิสโทแกรมของขนาดบริเวณที่คาดหวัง แล้วองศาของมุมที่เกี่ยวข้องกับบริเวณที่คาดหวังหาได้จากวิธีการแปลงฮัฟ เส้นตรงที่สำคัญทั้งหมดที่สัมพันธ์กับภาพย่อยถูกนำมาใช้เพื่อเป็นการหาเส้นตั้งต้นท้ายที่สุดวัตถุที่มีรูปร่างสี่เหลี่ยมจะถูกตรวจจับโดยขั้นตอนวิธีการกราดเส้นตรง ผลการทดลองแสดงให้เห็นว่าขั้นตอนวิธีที่นำเสนอให้ประสิทธิภาพที่สูงกว่าวิธีการของ Karsli และสามารถให้ความแม่นยำในการระบุตำแหน่งอาคารไม่น้อยกว่า 90% จากอาคารทั้งหมดในภาพ และวิธีการที่ถูกนำเสนอนั้นยังสามารถที่จะนำมาประยุกต์ใช้เพื่อหาอาคารสำหรับภาพทดสอบในมุมมองทัศนคติเนื่องจากสามารถแยกเส้นตรงที่เป็นส่วนประกอบของอาคาร

ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	ลายมือชื่อนิสิต .....
สาขาวิชา	วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ	ลายมือชื่อ อ.ที่ปรึกษาหลัก .....
ปีการศึกษา	2559	ลายมือชื่อ อ.ที่ปรึกษาร่วม .....

5473102123 : MAJOR COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

KEYWORDS: BUILDING DETECTION / CANNY EDGE DETECTION ALGORITHM / LINE SCANNING / HOUGH TRANSFORM / MORPHOLOGICAL OPERATION

CHAIYASIT TANCHOTSRINON: LOCATING BUILDINGS IN URBAN AREA FROM GEOMETRICAL INFORMATION. ADVISOR: ASST. PROF. SUPHAKANT PHIMOLTARES, Ph.D., CO-ADVISOR: PROF. CHIDCHANOK LURSINSAP, Ph.D., 77 pp.

An identification of geographic objects such as buildings from images is a challenging task in image processing, since it can be applied to various applications, e.g. city planning and disaster management. Consequently, an automatic building detection based on line scanning is proposed in this dissertation. For top view image, all possible candidate areas are initially identified on the tested image. To avoid redundant time consuming, the image is chopped into sub-images according to the candidate areas. An appropriate size of the sub-images is approximately estimated by a histogram of candidate area sizes. Then, degrees of angles relevant to the candidate areas are investigated by Hough transform. All significant lines related to the sub-images are subsequently extracted to be used as initial lines. Finally, the rectangle shaped objects are detected by the proposed line scanning algorithm. The experimental results shows that the proposed algorithm can acquire higher performance than Karsli's method, and can achieve at least 90% of accuracy for object based building detection. For perspective view, the proposed algorithm shows that it can be consistently applied to building detection on the tested image, since it can extract lines that are building components.

Department:	Mathematics and Computer Science	Student's Signature .....
		Advisor's Signature .....
Field of Study:	Computer Science and Information Technology	Co-Advisor's Signature .....
Academic Year:	2016	

ACKNOWLEDGEMENTS

I would like to express my immeasurable appreciation and deep gratitude to Assistant Professor Suphakant Phimoltares and Professor Chidchanok Lursinsap, my advisor and my co-advisor, respectively, for their supports, advices, guidance, encouragement, valuable comments, and provisions that are essential to the accomplishment and success of my dissertation. Without their guidance and persistent help, this dissertation would not have been possible.

I would like to express deep gratitude to my dissertation committees for the valuable comments and suggestions that led to the refinement of my dissertation.

I would like to express my sincere thanks to Dean Faculty of Science, and all teachers and staffs of the Department of Mathematics and Computer Science in the Faculty of Science at Chulalongkorn University, for teaching and providing necessary supports and permissions to conduct my dissertation.

I would like to thank the Advanced Virtual and Intelligent Computing (AVIC) Center for their material support in my research, and I also thank all my colleagues at AVIC Center for a lot of useful suggestion, generosity, and encouragement during my education. In addition, I would like to extend my sincere gratitude to everyone who directly and indirectly supports me in accomplishing this dissertation.

Finally, I would like to express my deep sense of gratitude to my family for endless supports with constant love, encouragement, and suggestions that motivated me to accomplish this dissertation.

CONTENTS

Page

THAI ABSTRACT iv
ENGLISH ABSTRACT v
ACKNOWLEDGEMENTS vi
CONTENTS vii
CHAPTER 1 INTRODUCTION 1
1.1. Motivation 1
1.2. Objective 2
1.3. Literature Reviews 3
1.4. Preliminary Study 11
1.5. Compared method (Karsli's method) 12
CHAPTER 2 FUNDAMENTAL KNOWLEDGE 14
2.1. Types of Input data 14
2.2. Canny Algorithm 14
2.3. Hough Transform 16
2.4. Morphological Image Processing 16
2.5. Measurement of binary area properties 17
CHAPTER 3 RESEARCH METHODOLOGY 18
3.1. Top View Images 18
3.1.1. Tested Images 18
3.1.2. Proposed Building Detection 21
3.2. Perspective View Images 49
3.2.1. Tested Images 49
3.2.2. Vertical Line Scanning 50
CHAPTER 4 EXPERIMENTAL RESULTS AND DISCUSSION 54
4.1. Top View Images 54
4.1.1. Results of Building Detection 54
4.1.2. Performance 63
4.2. Perspective View Images 65

Page

4.2.1. Results of Building Detection 65
4.2.2. Performance 70
CHAPTER 5 CONCLUSIONS 71
REFERENCES 73
VITA 77

CHAPTER 1 INTRODUCTION

1.1. Motivation

Even though various types of images such as aerial photographs and satellite images, are utilized in a survey for a long time, their outcomes are quite unsatisfied due to their prices and qualities of the photo resolution. High resolution images have influenced on factor analysis due to the fact that they can provide accurate topography information. Geographical Information obtained from satellite images or aerial photographs are beneficial to various kinds of applications, i.e. natural resource and biodiversity education, cultivated area searching, land use management, forest or watershed management, and change monitoring in abundant forests or degraded forests area. To monitor the changes of the physical characteristics of areas, the photographs taken from several periods are compared with each other. In addition, the satellite data also support an explanation of the locations and their extent. Humans generally assigned districts in several manners such as residential zone, industry zone, and agricultural land, and land uses are regularly changed. Therefore, this research focuses on building objects in the images, which are significantly related to city planning afterwards.

For computer vision and image processing, building detection is a versatile research that it can be efficiently applied to several types of applications, for instance, urban planning, military application, disaster management, and traffic planning. According to preliminary surveys, it found that the building detection methods can be categorized into 3 different manners: man-made, computer-aided or semiautomatic, and automatic system. The building detections based on semiautomatic approach are similar to those based on automatic approach except that they require users to appropriately assign some variables of each condition. To enhance the detection performance, both of automatic and semiautomatic approaches are improved for reducing the effort and time consumption. Consequently, this research paid attention

to produce and develop an automatic building detection system with high performance and reliability.

To achieve a high accuracy of the detections is quite a challenging task. This is because the shapes of buildings are quite complicated and it is hard to distinguish between buildings and other similar objects. In addition, some factors, including image quality, brightness level, and weather conditions, have also considerable influence on a success of building detection. According to the hypothesis, it assumes that the algorithm that can be applied to urban areas might also be applied to rural areas with high performance due to the complexity. Therefore, urban areas are focused on this research.

Urban planning is one of factors used for national development since it is beneficial to organize land uses. In addition, building detection from aerial imagery or satellite imagery is widely used to support geological information. The information is important for making a decision of city development. It supports observation and prediction in density of population in the area. Public utility is adjusted to suit the needs of the society itself. Population is more likely to increase every year but we have limited spaces and resources. Therefore, the building detection is beneficial to the developing city to be systemically, comfortably, securely, good hygienic and environmentally appropriate for optimizing the land uses.

1.2. Objective

The objective of this dissertation is to locate buildings in urban areas from geometrical information. Therefore, an automatic building detection system based on line scanning is proposed for top view and perspective view images.

1.3. Literature Reviews

For many years, various kinds of techniques were applied to building detection in order to enhance the performance. According to a review of A. Mishra et al. [1] on techniques used in building detection and extraction, building detection is a challenging topic because of complexity of background factors, point of shooting angles, different environmental objects and image noises.

To illustrate this point, Singh et al. [2] suggested an object identification method based on normalized difference vegetation index, from multispectral images. Under various image conditions, each specific set of thresholds was compatible for each region so it is a limitation of this method. Pakizeh et al. [3] introduced a semiautomatic building detection by Hough transform and intensity information. With this method, they used a segmentation of aerial images for estimating locations of buildings. Their results gave an acceptable building detection in dense residential zone. However, boundary location of buildings could not be estimated by this method and some parameter values (Dmax, Dmin) had impacted on the detection performance. Additionally, Benedek et al. [4] presented a building extraction based on a probabilistic approach for remotely sensed images. After testing with several aerial image sets, the results of two compared methods were matched, while the method was limited to specific conditions based on tile roof-top and loose residential areas. Also, Sırmaçek et al. [5] suggested a building detection using SIFT keypoints and graph theory for an urban-area. Although the method was robust to scale, rotation, and illumination in addition to show the promising results, it did not perform well in some cases, i.e. low contrast between rooftop and background, dense residential zone, etc.

Wang et al. [6] presented a detection and extraction of gable roofs by using a hierarchical connection graph algorithm for aerial imagery. Cote et al. [7] proposed an automatic extraction of 2-D rooftop footprints for nadir aerial imagery. Its advantages were that it can handle variant reflections and arbitrary illumination. Sırmaçek et al. [8] presented a building detection by a probabilistic framework. This method adopted four feature extractions, including Harris corner detection, gradient-magnitude-based support regions, gabor filtering in different orientation, and accelerated segment test.

Ozgun et al. [9] suggested an automatic building detection for monocular very high resolution optical satellite images. Even though the method could gain the accurate detection, it could not be applied to aerial images. Awrangjeb et al. [10] proposed an automatic extraction of 3-D building roof planes using LiDAR point cloud segmentation. Their results showed that it could successfully remove vegetation and gain highly accurate extractions of buildings and roof planes.

Image segmentation has been utilized in computer vision for serving numerous purposes such as biometric identification, radiology, autopilot systems, urban planning and so on. Various kinds of techniques are used in image segmentation. For instance, thresholding methods, edge-based segmentation, region-based segmentation, color-based segmentation, transform method, and texture analysis etc. Each technique has its own advantages and disadvantages. To illustrate this point, groups of building detection methods based on the same technique are presented below.

For texture analysis, Cetin et al. [11] presented a building detection with textural features and Adaboost for satellite images. Although it gained accurate results under various building types, it required time consuming for computing a set of features by several techniques before feeding to Adaboost. In addition, N. Li et al. [12] presented a texture measure with variogram integrated labeled co-occurrence matrix for building detection.

For watershed, it is a kind of segmentation using mathematical morphology. N. Shorter et al. [13] proposed a watershed algorithm for object segmentation. The objects were measured their characteristics by pixel vegetation index, region solidity, and shadow detection. Besides, R. Gui et al. [14] proposed an individual building extraction using ontological semantic analysis.

Color-based segmentation is a technique for identifying the objects and the regions of interest in the images. Thus, the technique can bring about relevant information to image analysis [15]. Hernadez-Gomez et al. [15] investigated a color segmentation by Lab color space, and the results demonstrated that the optimal combination of color components for the segmentation tasks is ab. Likewise, Shen et

al. [16] presented an image retrieval algorithm with Hu moment invariants and color segment. From the results, the retrieval recall and precision of the method were higher than those of the original method but it could not assign weights for features in accordance with image content. N. M. Kwok et al. [17] studied effective of color space (RGB, YUV, YIQ, HIS, HSV) for segmentation. The results stated that each color space provides different information that might influence on entropy-based segmentation. H. Y. Chong et al. [18] introduced the color space that continued to be robust in spite of spectral changes in illumination. The experimental results were evaluated by using same segmentation method with different color spaces.

Furthermore, a building detection has been proposed and following techniques was utilized to achieve high detection performance.

For corner detection, M. Wang et al. [19] used a combination between mean shift segmentation and scales invariant feature transform, detected corners, for finding building area based corner and the segmentation image. Therefore, the precise boundaries of building roofs was extracted by adaptive windowed Hough transform algorithm. Q. Wang et al. [20] focused on Gable-roof building detection. Hierarchical connection Graph (HCG) was introduced by the authors for detecting and extracting roof-top. In addition, a set of corner sequence was extracted. The HCG Graph represented relationship of paths corners sequence in each building. Therefore, the sequence of optimum corners sequence was chosen by using dynamic programing. The advantage is that the method is flexible for shape and robust corner detection but needs to collect corner templates before operation.

For shadow detection, D. Chaudhuri et al. [21] presented a framework without pre-classification or training data. The framework consisted of an enhancement of the building structure, seed point detection from internal gray variance (IGV) feature, multispeed-based clustering, shadow detection, false alarm reduction and adaptive threshold-based segmentation technique. Some techniques were not designed for automatic building extraction because it required learning algorithm before classification. The Limitation of this method was threshold–based on homogeneity of the roof-top.

For neural network and machine learning, Ghaffarian et al. [22] proposed an automatic building detection by Purposive FastICA algorithm for high resolution Google Earth imagery. The algorithm gained overall high precision performances, and it completely automated without requiring parameter selection. Moreover, it could be effectively applied to other types of images, such as aerial imagery. However, a detection in the case of considerably different colored rooftops was its limitation. Likewise, N. Chandra et al. [23] introduced a building detection for high resolution satellite images using SVM.

Particularly, line and rectangle detections are one of the most powerful techniques that are widely used for detecting the buildings in the images. A significant advantage is that only binary images are required for performing this technique. No color, surface, and/or height information are needed. Thus, a reduction of information needed particularly decreases time consuming. For this point of view, C. Senaras et al. [24] proposed self-supervised decision fusion (SSDF). The system can be assigned training data automatically. The relevant information was extracted for instance vegetation, shadow, directional shadow-object relationship generation, lines and rectangle. Yang et al. [25] introduced a building extraction in towns and villages by texture enhancement. Their results showed that the method could highlight building edges and decrease undesired segmented objects. However, a set of parameters must be tested many times and the real data were not be verified. Additionally, Saeedi et al. [26] presented an automatic building detection method for aerial and satellite images. The method utilized several techniques, including gradient, straight line extraction, line linking and filtering, and image segmentation, for the detection. According to its results, the method could precisely detect small gabled rooftops under various properties of light reflections. However, it was designed for detecting the quadrilateral shape only. Awrangjeb et al. [27] introduced the reconstruction of roofs. In this method, the lines which were roof boundary, were classified using LiDAR data and other data space. NDVI index was used to eliminate vegetation areas, and the entropy data was used to classify the shadow. Furthermore, an application for this kind of technique in image processing was also proposed by Y. Li [28], V. J. M. Fernandes [29], J. Wang [30], and S. Nebiker [31]. Briefly, Table 1 summarizes the building detection methods mentioned above.

Table 1. Summarization of building detection methods.

Author	Technique	Advantages	Disadvantages
Singh et al.	NDVI based segmentation and morphological operation	- High performance classification between man-made regions and natural regions	- Cannot be applied to aerial images - Require specific sets of thresholds for regions
Pakizeh et al.	Hough transform and Intensity information	- Give acceptable building detection in dense residential zone	- Some parameter values (Dmax, Dmin) are impacted on the detection performance - Cannot locate building boundary
Benedek et al.	Robust Masked Point process model	- Perform significantly on tile roof top	- Lack of tested images - Design for tile roof top and loose residential zone
Sırmaçek et al.	Scale Invariant Feature Transform Keypoints and Graph Theory	- Robust to scale, rotation and illumination	- Not perform well in low contrast between rooftop and background - Not suitable for dense residential zone
Saeedi et al.	Combination of Gradient, Straight line extraction, Line linking and filtering, and Image segmentation	- Perform well on Large Quadrilateral building shape	- Lack of database for experiment - Design for detecting the quadrilateral shape only
Ozgun et al.	Fuzzy landscape generation and Grabcut	- Can accurately detect buildings with random sizes and shapes	- Cannot be applied to aerial images
Cetin et al.	Texture features and Adaboost	- Independent of building types	- Use too many techniques for extracting features from the images - Time consuming

Yang et al.	Texture enhancing	- Reduced redundant segmented object	- Need to test the appropriate variables many times for each condition - Lack the verification of real data
Hernandez-Gomez et al.	Color segmentation using CIELab space	- Alternative color segmentation - A desirable algorithm for natural images	- Need more exhaustive test in another condition
Shen et al.	Color Segment and shape moment invariants	- More suitable to human visual perception	- Cannot assign weights for features
Wang et al.	Hierarchical Connection Graph (HCA) Algorithm	- Consider building boundaries - Perform on images that are closed to vertical angle (75°-105°)	- Can be applied on gable-roof buildings only - Lack of tested images
Cote et al.	Level-set Curve Evolution	- Consider various shapes of buildings - Can be applied to flat and gable roof-top	- Over detection building shape in some building condition
Sırmaçek et al.	Probabilistic Framework	- Quickly perform - Flexible when feature extraction method is changed	- No building boundary consideration - Difficult to distinguish between buildings and roads in dense residential zone
Tanchotsrinon et al.	Texture analysis and color segmentation based on neural classification	- More suitable to bright roof top - Autonomic systems	- Some darker roof top might be misclassified
Karsli et al.	GLCM texture metrics and classification of multi-dimension dataset with SVM classifier	Auto generate training and testing data set	Need specific data to process (digital terrain model and digital surface model)
Chuadhuri et al.	Morphology and Internal gray variance	- Automatic systems without pre-classification or training set	- Threshold based method might be suitable for homogeneity rooftops

		- Perform on grayscale image	- Some building boundary is not accurate - Without building shadow the system might be miss detection
Wang et al.	Connetion Graph Algorithm	- Robust to missing corner detection - Boundary and gable were extracted	- Need to collect corner roof template - Design for gable roof only
Ghaffarian et al.	Purposive Fast ICA	- Any building shape can be extracted - Unsupervised classification	- Moderate precision extracted on building boundary - Miss some correct building regions caused by PFICA algorithm - Incorrectly remove small building regions by morphological operations
Kovacs et al.	Orientation based building detection	- Any building shape can be extracted	- Poor precision extracted on building boundary - Data resolution might be effective to classification results - Lack contrast affected to Active contour method
Awrangjeb et al.	Point cloud segmentation	- Good roof plane extraction performance - Can distinguish buildings in dense vegetation environment	- Accuracy limited by LiDAR point density - Not integrate smoothing of extracted segment boundaries - Not perform well on curved roofs
N. Shorter et al.	Watershed segmentation and Pixels Analyses	- Any building shape can be extracted - Perform well on large buildings	- Building boundary cannot be extracted - High rate of false positive - Require several threshold values

N. Li et al.	Combination of Labled Co-Occurrence matrix and Variogram	- Develop a new method for texture analysis	- Different and suspicious window sizes - Pixel-based classification
A. J. Fazan et al.	Snakes and dynamic programing	- Any building shape can be extracted - Clear boundary	- Time complexity - Seed points must be created by human
Awrangjeb et al.	3D reconstruction of building roofs.	- Extract both roof edge and roof ridge - boundary fitting reconstruction	- Require a lot of input data - Hard to detect small buildings

1.4. Preliminary Study

In preliminary study [32], the automatic building detection by texture analysis and color segmentation was proposed. According to the method, it began with filtering, color segmenting, and adjusting contrast of images. The improved images were subsequently extracted for GLCM and SVD features. Then, the feature vectors were fed to MLP for detecting building objects. In testing procedure, twelve images with several interesting conditions, including different building sizes, colors, and shapes, were randomly captured by Google Earth. Although the experimental results were quite satisfied, some limitations should be further considered. For example, the preprocessing steps were too complicated even they required short time consumption. The MLP was manually configured by trial and error, then, the experimental results were varied. The method is suitable for light residential zone, while it has some limitation in detecting the buildings when the color of roof-tops is similar to that of roads. To obtain the better results for practical use, some limitations should be considered and the building detection method should be improved. In the recent survey, all existing methods including the preliminary work aim to locate all buildings in top-down direction. This means all global and local features are extracted simultaneously and then they are used to determine the building locations. In contrary, with the assumption that each building is composed of several lines, the buildings can be determined after all lines forming a polygon are located in bottom-up direction. Since low-level feature like line is considered, the system is expected to detect buildings finer than the existing one. Therefore, the main part of this dissertation focuses on an enhanced building detection method based on edge detection method, polygon extraction based on line detection, shape measuring, and texture-based pattern recognition.

1.5. Compared method (Karsli's method)

This research adopts a method of F. Karsli et al. [33] as a compared method for evaluating the building detection performance. According to the method, they introduced multi-features extraction on very high resolution image multispectral and LiDAR data.

Description of Image data set

According to Karsli's method [33], the test areas were two different regions that were characterized as urban and semi-urban with dimensions of 326 × 351 and 350 × 347 m. Besides, the elevation of the areas was changing between 207 and 333 m. The areas were intentionally selected due to their interesting topographic characters of morphology and slope. In fact, the buildings are in different geometrical structures at various spectral characteristics. Thus, it was almost impossible to automatically extract buildings in the areas when optical sensor data was only used. Additionally, the areas also had another significant characteristics. That is, their semi-urban regions contained the adjacent positions of high trees and buildings, there was no dominating single geometric shape for the types of roofs, and the areas contained several types of roofs, such as flat and gabled. With only LiDAR data, it was also impossible for an automatic building detections.

To avoid the problem mentioned above, Karsli et al. utilized an alternative approach by using a combination of LiDAR and optical sensor data.

Concept of Karsli's Method [33]

Input data can be separated into LiDAR point cloud data and optical image data. Firstly, LiDAR cloud data was used for generating digital terrain model and digital surface model to gain elevation information from objects and terrain surface in the interesting area. Secondly, Optical image data containing color image (RGB) were used

to generate modified normalized difference vegetation index (MNDVI). Therefore, the fusion of two sources of data was divided into images of 5x5 window size which were applied to GLCM algorithm to extracted texture features. They are 26 dimensional features, including 3 RGB bands, 1 slope ratio index, 1 MNDVI index, 1 nDSM elevations, 5 Dissimilarity, 5 Entropy, 5 Homogeneity and 5 Contrast. Therefore, the system were automatic generated training and testing data sets for SVM classifier by using vegetation mask and morphological enhancement (local maximum with rectangularity index and local minimum).

CHAPTER 2 FUNDAMENTAL KNOWLEDGE

2.1. Types of Input data

For building detection, the widely used input data can divided into 3 main categories, as follows.

- **Multispectral image** – it is acquired by satellite data based on various ranges of electromagnetic spectrum. They are used to sense land cover or some characteristics of regions. For example, short wave infrared (wavelength 1.57 to 16.65 μm) senses moisture content of soil and vegetation. In addition, normalize difference vegetation index is a widely used technique for highlighting the vegetation areas.

- **Aerial photography** – it can be taken by planes, helicopters, and drones. This type of images is typically used for constructing a map. Although it contains less information than the others, it is a cost effectiveness.

- **Radar Imagery** – the input data can be divided into Digital Surface Model (DSM) and Digital Terrain model (DTM). DTM is a Digital Elevation Model (DEM) of the shape of ground surface, while DSM is a DEM of the shape of the surface, including infra-structures and trees. In addition, the heights of objects in the images can be identified by this type of images.

2.2. Canny Algorithm

It is a well-known algorithm that was introduced by John F. Canny in 1986, for extracting edges from the images. This Canny algorithm is composed of 3 main procedures, as explained below.

- **Noise reduction**

Gaussian filtering is employed in this edge detector because it is sensitive to noise present in raw unprocessed images. Then, it obtains a slightly blurred original images that is robust to a single noisy pixel.

- **Finding the intensity gradient**

Since each edge of images may point in various directions, the algorithm employs four filters for detecting vertical, horizontal, and two diagonal edges in the blurred images. The operator, such as Sobel and Prewitt, is adopted to yield the first derivative in horizontal and vertical directions. Then, the edge direction angle is rounded to one of four significant angles.

- **Non-maximum suppression**

With image gradient estimation, a search is performed to determine if the magnitude of gradient presumes a local maximum in the gradient direction. Thus, a binary image containing edge points is acquired by this non-maximum suppression.

Advantages and Disadvantages of Canny algorithm

The first advantage is obtained from Gaussian filter so any noises presented in an image can be removed. The second is enhancing the signal regarding noise ratio. The third is higher performance of edge detection, especially in noisy state by applying thresholding method. The effectiveness of Canny algorithm is affected by adjustable parameters. To illustrate this point, small filters are properly used for small or sharp edge detection, while large filters are properly used for larger or smoother edge detection. Time consuming for complicated computation is its critical limitation.

2.3. Hough Transform

Hough transform was proposed by Hough in 1962. It was originally designed to detect straight lines and curves. For the original version, it can be only performed in the case that analytic equations of object borderlines are known. Hence, it was not required for any prior knowledge of region positions. Its significant advantage is a robustness of its segmentation results, namely, the segmentation is not too sensitive to noise or imperfect data.

2.4. Morphological Image Processing

A simple set of structuring elements is illustrated in Figure 1. For an implementation by computers, it is necessary to convert this set of structuring elements to a rectangular array by adding the background elements.

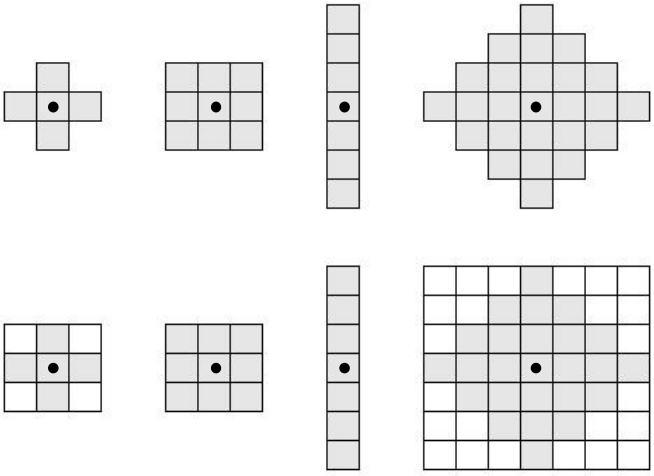


Figure 1 Examples of structuring elements. The first row presents original structuring elements while the second row shows the structuring elements in forms of rectangular array.

The background border should be adequate large for accommodating the entire structuring element when its origin is on the border of the origin set. Besides, the structuring elements should be filled with the smallest possible number of

background elements that are actually required for converting it into a rectangular array.

Opening and Closing

They are based on two primitive operations, which are dilation and erosion. The dilation operation expands the image components, while the erosion operation shrinks them. Opening typically performs a dilation between the binary image and the structuring element, followed by an erosion. In contrast, Closing typically performs an erosion between the binary image and the structuring element, followed by a dilation.

2.5. Measurement of binary area properties

Various kinds of aspects are applied to measure the properties of binary objects. Solidity is a property which is used for measuring the solid of binary objects. It can be computed by finding the area and subsequent dividing with its convex area. The area of binary object is defined by the total number of pixels in the object. Convex area is greater than or equal to the object area, and it is defined by a set of outer pixels that connect to each other like rubber band. In this way, solidity can be indicated binary object solid. In this part, solidity is used to distinguish between man-made and non-man made area, since the solidity should have high values in the man-made areas.

CHAPTER 3 RESEARCH METHODOLOGY

For an improvement of building detection algorithms, a research methodology can be divided into two main parts according to two different points of views of images. The first part is conducted for detecting the buildings in the top view images, and the second part is conducted for detecting the buildings in the perspective view images. Each part is described as follows.

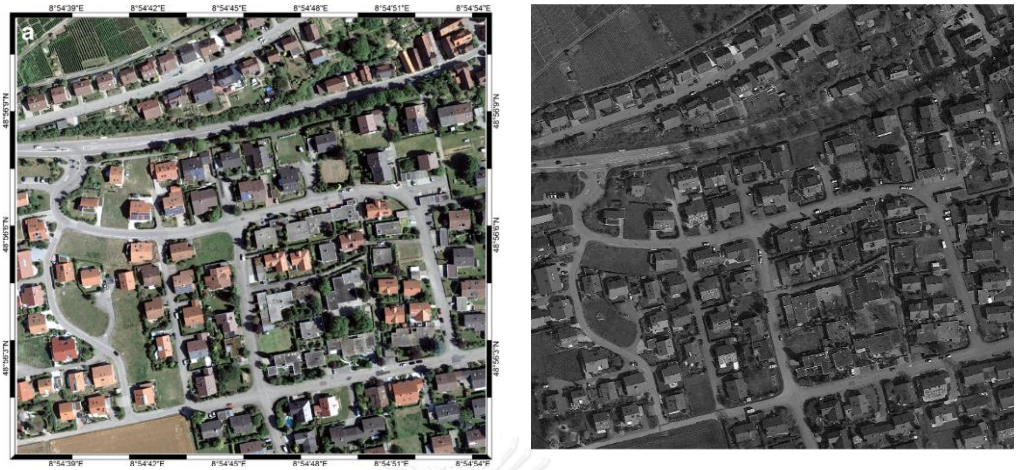
3.1. Top View Images

Initially, top view images were captured by Google earth, and candidate areas of buildings in the images were identified. Then, degrees of angles relevant to the areas were acquired by applying Hough transform. Significant lines were subsequently extracted by rotating the chop images of the areas with the relevant angles. Finally, the rectangle shaped objects were detected as buildings by line scanning algorithm.

3.1.1. Tested Images

To evaluate a performance of the proposed building detection, the first two tested images were shared the same locations as those of Karsli's method [33]. Hence, the top view images located on the coordinates (48°56'9"N, 8°54'39"E) and (48°56'9"N, 8°57'33"E) were captured by Google earth. For this part, these two tested images are named as T1 and T2 as shown in Figure 2 and Figure 3.

The tested image T1 contains 124 buildings while T2 contains 111 buildings.



(a) (b)

Figure 2 First tested images (T1) of (a) Karsli's method [33] and (b) this research.

(a) (b)

Figure 3 Second tested images (T2) of (a) Karsli's method [33] and (b) this research.

To evaluate a performance of the proposed algorithm, the other three top view images captured by Google earth were additionally tested in this part. The tested images T3, T4, and T5 were located on coordinates (48°56'0.61"N, 8°58'5.15"E), (48°55'55.40"N, 8°58'12.31"E), and (48°56'19.96"N, 8°57'38.56"), respectively as shown in Figure 4. T3 contains 122 buildings, T4 contains 138 buildings, and T5 contains 168 buildings.

(a) T3

(b) T4

(c) T5

Figure 4 Tested Images (a) T3 (b) T4 and (c) T5.

3.1.2. Proposed Building Detection

Initially, all possible candidate areas are identified from the tested image. To avoid redundant time consuming, the tested image is chopped into sub-images according to centroids of the candidate areas. An appropriate size of the sub-images is approximately estimated by a histogram of candidate area sizes. Then, degrees of angles relevant to the candidate areas are investigated by Hough transform. All significant lines related to the sub-images are subsequently extracted to be used as initial lines. Finally, the rectangle shaped objects are detected by line scanning algorithm. A flowchart of our proposed building detection is illustrated in Figure 5.

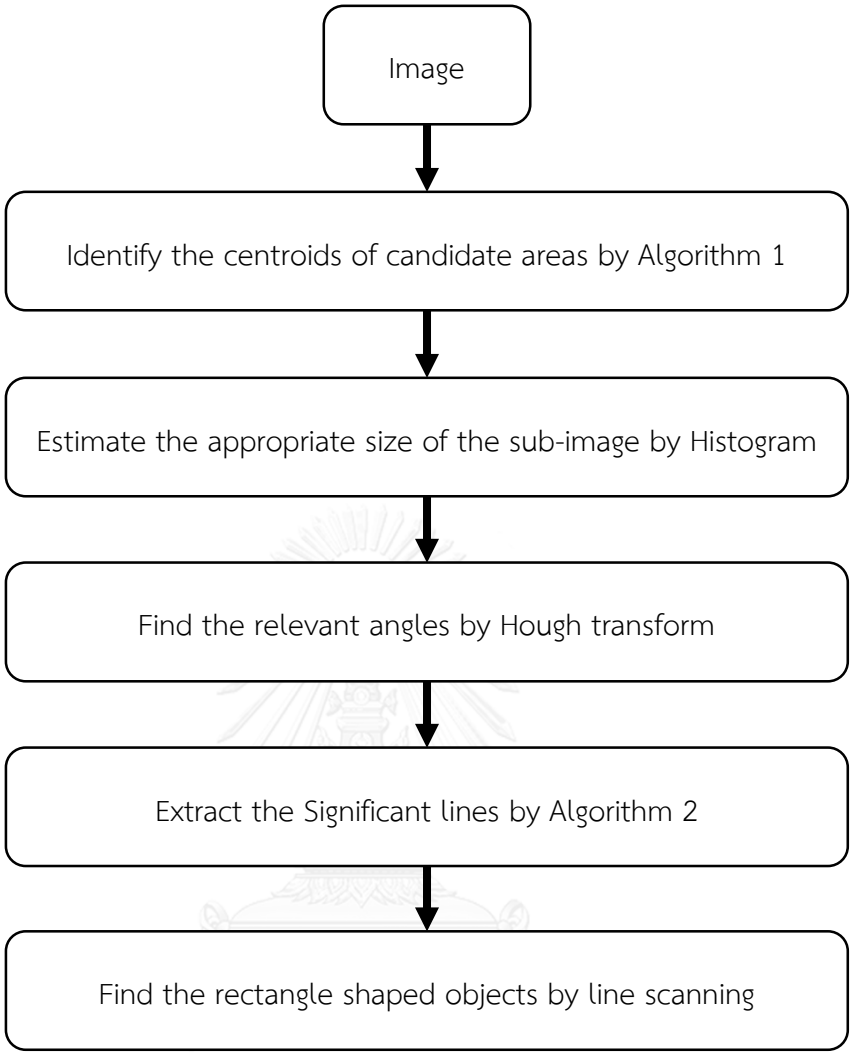


Figure 5 A flowchart of the proposed building detection.

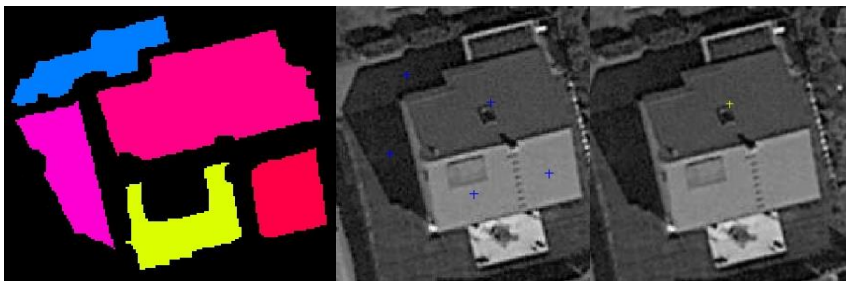
Accordingly, our proposed building detection can be divided into five main procedures, as follows.

3.1.2.1. Identification of Candidate Areas

For each tested image, an original image (RGB) was transformed into a grayscale image (G), and the grayscale image was subsequently transformed to a binary edge image (E) by Canny algorithm. Morphological operations with rectangle-shaped structuring element of size 5 x 5 pixels, were performed on the edge image for connecting nearest edges. Then, sliding-neighborhood operations by total summation were performed. This operation applied the function of total summation to each 5-by-5 sliding block of the edge image. Since buildings have the highly smooth surface of areas that do not naturally occur, the pixels whose the total summation of their sliding blocks are zeros, were chosen. After removing small objects of sizes less than 1,000 pixels, the solidity values of the remaining objects were calculated. The solid property of binary objects can be measured by specifying the proportion of the pixels in the area of the object and that of the convex hull. The values range from 0 to 1. Thus, the objects having the solidity greater than or equal to 0.5 were extracted. To merge nearest objects, the morphological operations with rectangle-shaped structuring element of size 5 x 5 pixels and filling holes were performed. After removing small objects of sizes less than 1,000 pixels, the candidate areas were gained, and their centroids were calculated to be represented as the locations of the areas.

Since the same building can possibly be broken into several areas and performing the proposed method on these high adjacency locations can produce an unnecessary high time consuming, grouping these centroids should be concerned. Therefore, the centroids whose Euclidean distances between them are less than or equal to 100 pixels, were combined in the same group. For each group, only one centroid locating on its center was selected.

Figure 6 demonstrates the candidate areas identified by the same building, and their centroids before and after grouping the adjacency areas.



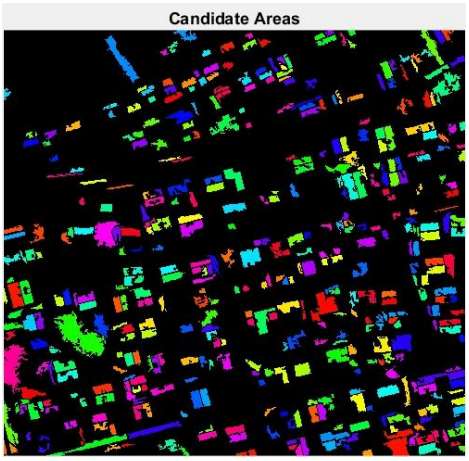
(a) (b) (c)

Figure 6 Examples of (a) candidate areas (b) centroids of the candidate areas and (c) centroids after grouping the adjacency areas.

As a result, the centroids of all groups were gained to be represented as the locations of the candidate areas for the next procedure. Algorithm 1 summarizes a procedure of how to identify the candidate areas of buildings in the top view images.

Algorithm 1: An Identification of Candidate Areas	
1.	Transform an original image (RGB) into a grayscale image (G).
2.	Transform the grayscale image (G) into a binary edge image (E) by Canny algorithm.
3.	Perform morphological closing operations with rectangle-shaped structuring element of size 5 x 5 pixels.
4.	Perform sliding-neighborhood operation by applying the function of total summation to each 5-by-5 sliding block of the image.
5.	Remove small objects of sizes less than 1,000 pixels.
6.	Extract all objects whose solidity values range from 0.5 to 1.
7.	Perform morphological closing operations with rectangle-shaped structuring element of size 5 x 5 pixels.
8.	Fill holes in the objects.
9.	Remove small objects of sizes less than 1,000 pixels.
10.	Calculate centroids of all candidate areas.
11.	Divide the adjacency centroids into groups using the Euclidean distances less than or equal to 100 pixels.
12.	Select the centers of groups for identifying the candidate areas.

Figures 7 to 11 demonstrate candidate areas and their centroids before and after grouping their adjacency areas in the tested image T1 to T5, respectively.

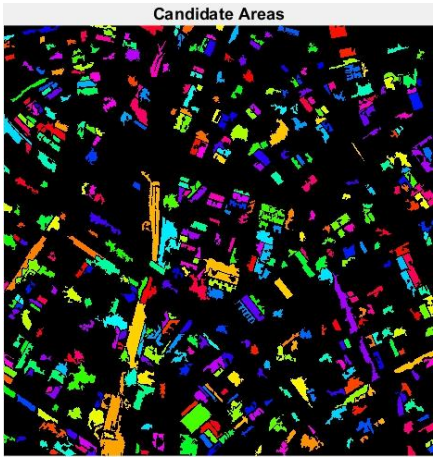


(a)

(b)

(c)

Figure 7 (a) Candidate areas, (b) Centroids of candidate areas before grouping the adjacency areas, and (c) Centroids of candidate areas after grouping the adjacency areas of the tested image T1.

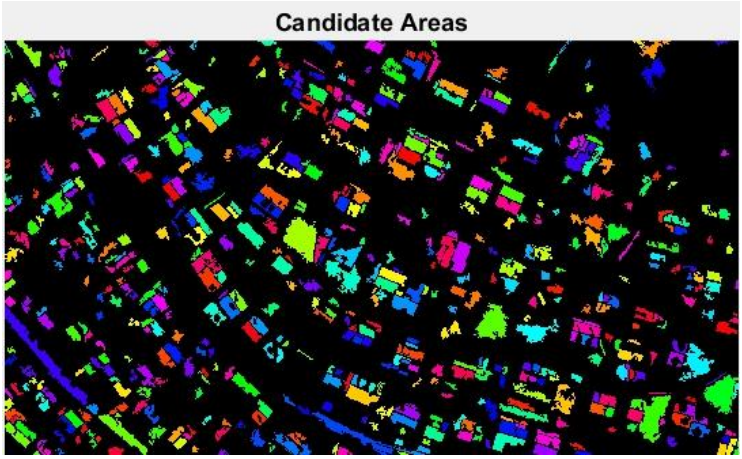


(a)

(b)

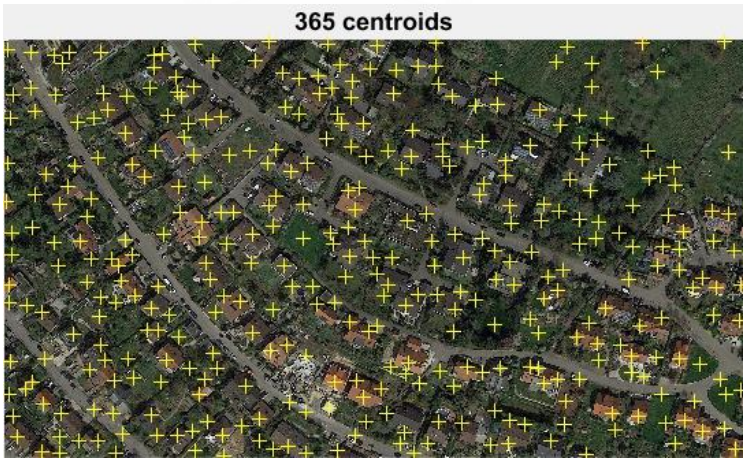
(c)

Figure 8 (a) Candidate areas, (b) Centroids of candidate areas before grouping the adjacency areas, and (c) Centroids of candidate areas after grouping the adjacency areas of the tested image T2.



(a)

(b)

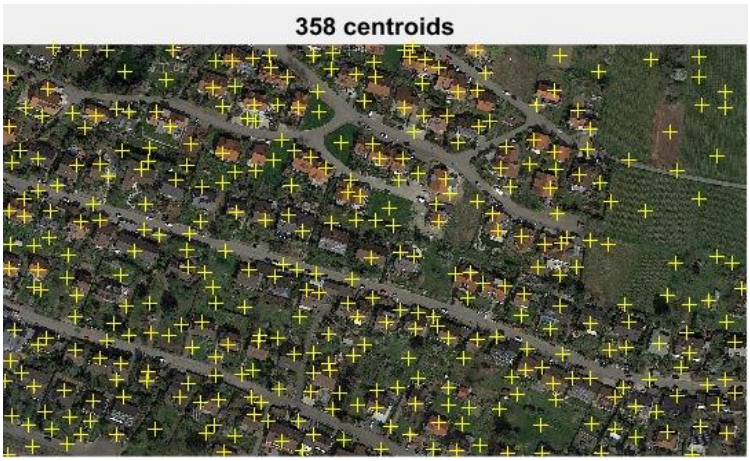


(c)

Figure 9 (a) Candidate areas, (b) Centroids of candidate areas before grouping the adjacency areas, and (c) Centroids of candidate areas after grouping the adjacency areas of the tested image T3.

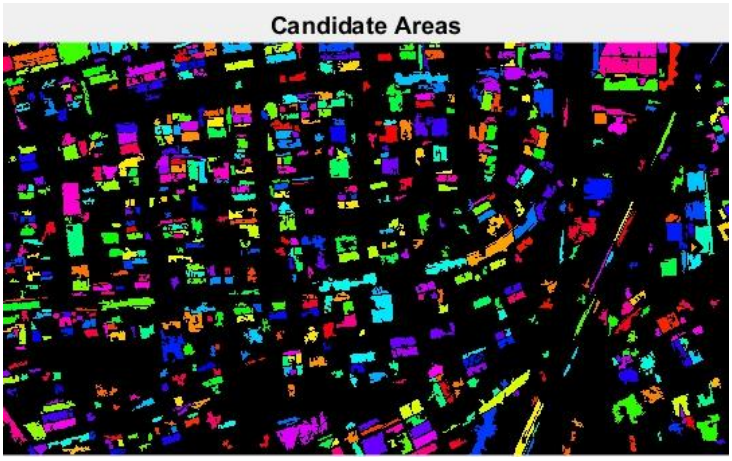
(a)

(b)



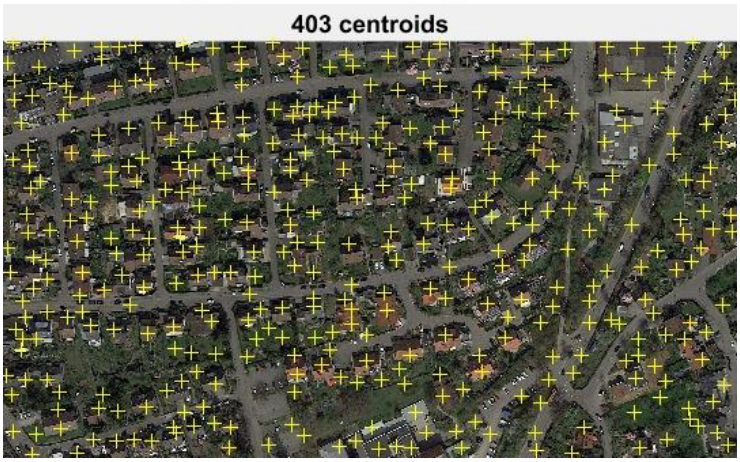
(c)

Figure 10 (a) Candidate areas, (b) Centroids of candidate areas before grouping the adjacency areas, and (c) Centroids of candidate areas after grouping the adjacency areas of the tested image T4.



(a)

(b)



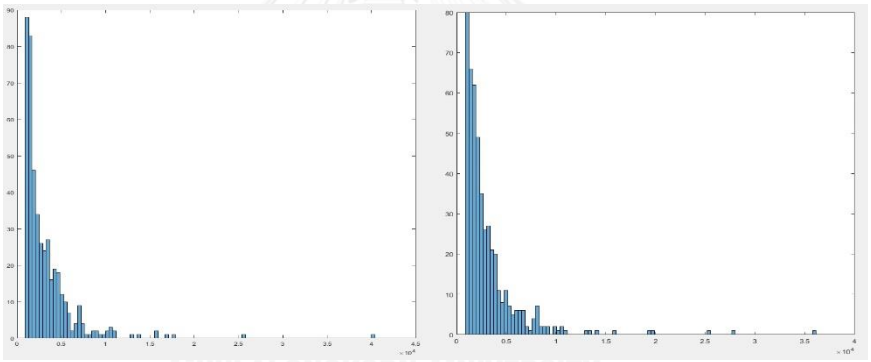
(c)

Figure 11 (a) Candidate areas, (b) Centroids of candidate areas before grouping the adjacency areas, and (c) Centroids of candidate areas after grouping the adjacency areas of the tested image T5.

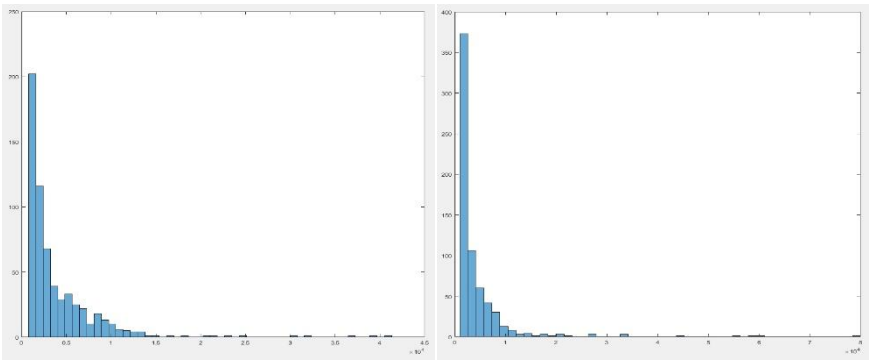
3.1.2.2. Applying Histogram

For scanning the rectangle shaped objects, the tested images were chopped into sub-images according to the acquired centroids. If a size of the sub-images was too large, too many objects appeared on them so degrees of angles that are really relevant to the candidate areas were hard to identify by Hough transform. Then, the significant lines might not be obtained. On the other hand, if a size of the sub-images was too small, it might not cover whole parts of building objects so it affected their shapes. Then, the objects might not be detected by line scanning.

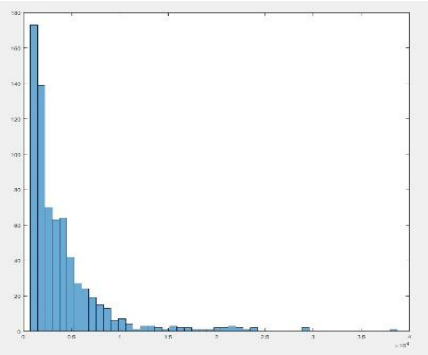
Therefore, a histogram of sizes of all candidate areas was investigated in order to estimate an appropriate size of the sub-images.



(a) T1 (b) T2



(c) T3 (d) T4



(e) T5

Figure 12 Histograms of sizes of all candidate areas extracted by applying Algorithm 1 to the tested images (a) T1 (b) T2 (c) T3 (d) T4 and (e) T5, respectively.

Figure 12 displays the histograms of sizes of candidate areas extracted by Algorithm 1. Among all tested images, the maximum size of found candidate areas was approximately 80,000 pixels. In this research, the rectangle shaped objects were considered. Moreover, the width and height of the sub-images were set to equal values for simplifying the implementation. As a result, the width and height, which were calculated from a square root of the maximum area, were 282.84. The values were further rounded to 300 in order to expand the areas to cover whole components of the building objects. Consequently, for each centroid, the original tested images were chopped into sub-images of sizes 300 x 300 pixels in order to scan for the rectangle shaped objects.

To illustrate this point, Figure 13 displays an example of the sub-image obtained from the centroid number 156 in the tested image T2.

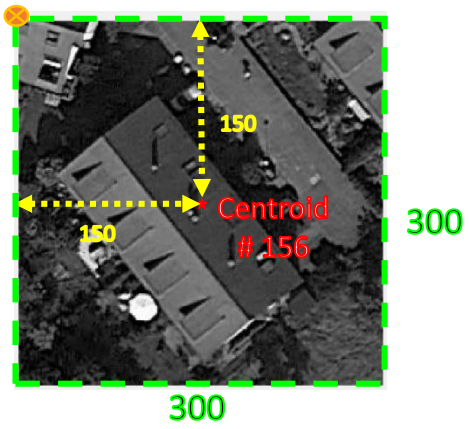


Figure 13 The sub-image of centroid number 156 in the tested image T2.

However, the sizes of sub-images might occasionally be less than 300 pixels if the centroids were located near borders of the tested images.

3.1.2.3. Applying Hough Transform

Instead of whole sub-images, the areas closing to the centroids can give the degrees of angles that were more relevant to the candidate areas. Therefore, the half-size sub-images whose their centers were located on the centroids, were applied to Hough transform for extracting their relevant angles.

Each half-size of the grayscale sub-image was transformed into binary edge image by Canny algorithm. The edge image was applied to Hough transform in order to compute Standard Hough Transform (SHT) for locating Hough peaks. This part specified the maximum number of Hough peaks as 10. Then, ten degrees of angles were gained. In some cases, the duplicate angles might occur and they resulted in an unnecessary high time consuming so only unique angles were included. Since the buildings can be in both horizontal and vertical orientations, the degrees of angles that were perpendicular to the unique angles, were also relevant to a success of the proposed building detection. Therefore, the unique angles and their perpendicular angles were used as relevant angles of the sub-images. Figure 14 displays the unique angles of the half-size sub-image in the tested image T2 at centroid number 156 after applying Hough transform.

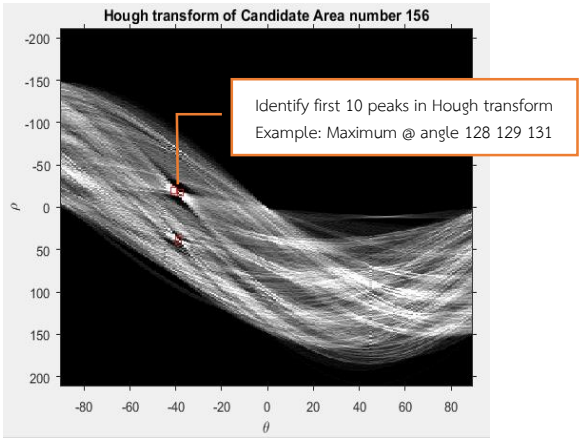
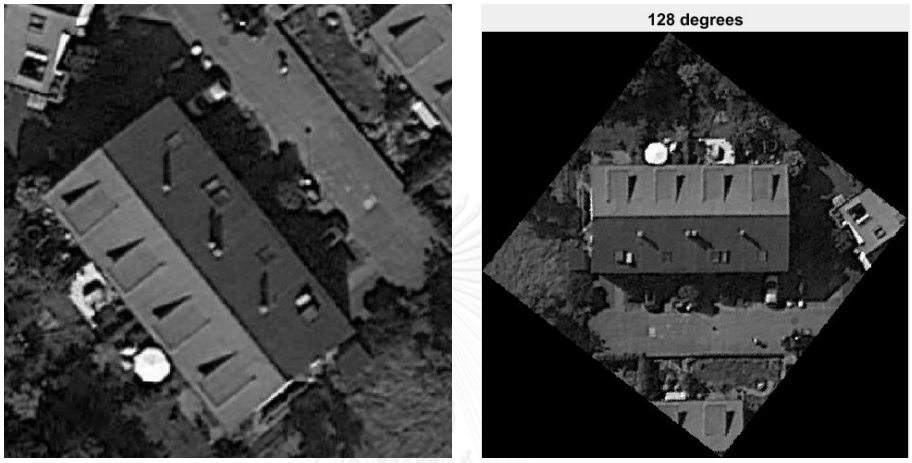


Figure 14 Unique angles of the half-size sub-image in T2 at centroid number 156 after applying Hough transform.

3.1.2.4. Significant Lines

For each relevant angle, the grayscale sub-image was rotated by θ_j in clockwise direction using nearest neighbor interpolation for assigning intensity values to the spatially transformed pixels. An example of sub-image rotation is shown in Figure 15.



(a) (b)

Figure 15 An example of (a) the grayscale sub-image of centroid number 156 and (b) the sub-image after rotation in clockwise direction by 128 degrees.

Since too many noises in the grayscale image could produce too many unnecessary lines in the edge image, this might confuse the proposed building detection to gain false positive results. To avoid this problem, 2D Gaussian smoothing kernel with standard deviation (σ) of 1 was applied to the rotated sub-images for removing the noises.

Figure 16 displays the sub-image of the centroid number 156 (I_{156}) before and after filtering with 2D Gaussian smoothing kernel with standard deviation (σ) of 1. It showed that the intensity values of the blurred image were smoother than those of the original image.

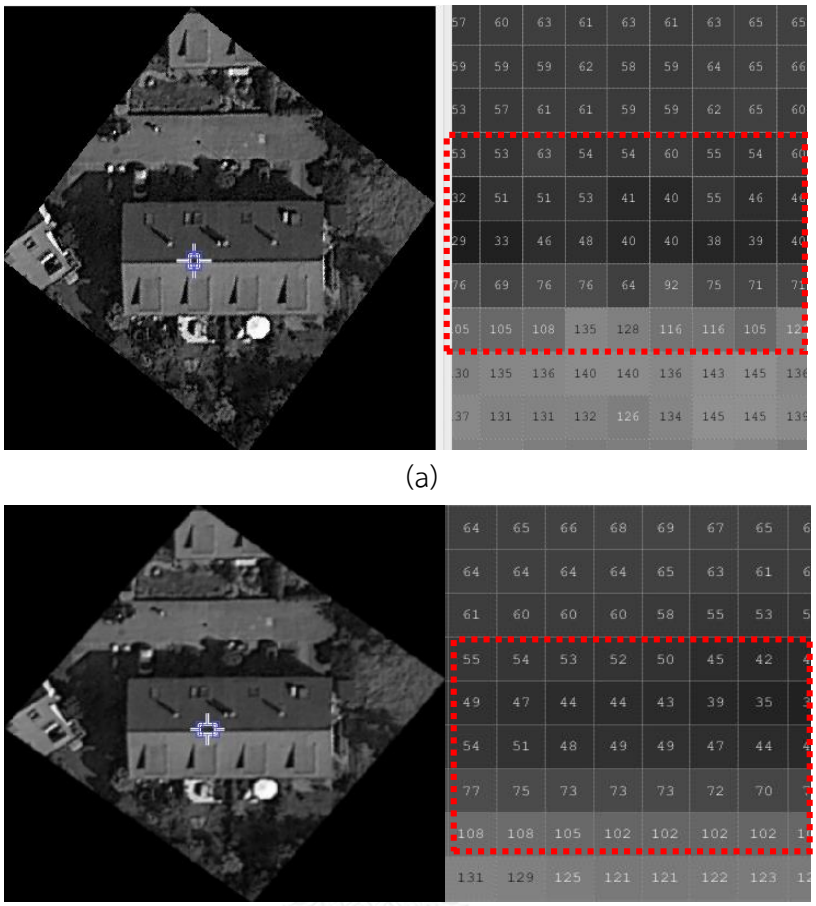
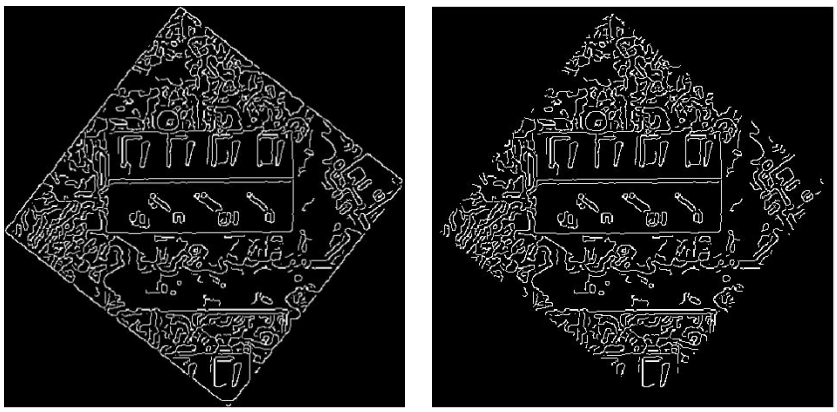


Figure 16 The rotated sub-image of centroid number 156 (a) before and (b) after filtering with 2D Gaussian smoothing kernel with standard deviation (**σ**) of 1.

After that, the grayscale sub-image was transformed into binary edge image by Canny algorithm. In this step, it can be noted that all elements of the edge image were logical and their values were either true or false. Since the boundary lines of the rotated sub-image were automatically added to the binary edge image by Canny algorithm, a process of removing the boundary lines was required to avoid incorrect detection that might occur. For this point of view, Figure 17 illustrates the rotated and filtered edge image of centroid number 156 before and after removing the boundary lines.



(a) (b)

Figure 17 The edge image of the centroid number 156 (I_{156}) after rotation in clockwise direction by 128 degrees, filtering, and edge detection. (a) Before removing the boundary lines (b) After removing the boundary lines.

The edges of buildings should have continuously long lengths so that they could cover the whole building components. In fact, the lines appeared in the edge image might be broken and discontinuous due to several reasons. For instance, if some parts of buildings were covered by trees, tree shadows, and/or other objects, these edges might be broken and discontinuous so their lengths were less than they should be. Since the buildings of sizes greater than or equal to 50 x 50 pixels were focused on this research, the lines of lengths greater than or equal to 25 pixels that were a half of the desired length were interested and considered as the significant lines.

Let a matrix R containing all logical elements of the rotated edge image obtained from rotation in clockwise direction by the angle θ_k, 2D Gaussian filtering, Canny algorithm, and boundary line removing. The size of matrix R is n x m that was corresponded with the size of the rotated edge image. All elements of the matrix R were logical which could be either true or false. Let V_i be a vector containing all logical elements of the matrix R at row i, where i = 4, 5, ..., n-3. For each row of the matrix R, the line of maximum length was extracted. If its length was greater than or equal to 25 pixels, logical OR operation was performed on the vector of the row and its adjacency. This operation was required for connecting lines to gain the possible

maximum lengths that they should have. Let O_i be a vector containing logical elements resulting from the OR operation. The line containing the line of maximum length was extracted as the significant line of the row. Note that one significant line was gained for each row. This process was continued until it reached to the last relevant degrees of angles. Algorithm 2 explained the procedure of how to extract the significant lines.

The significant lines were subsequently used as initial lines for detecting the rectangle shaped objects by the proposed line scanning.

Algorithm 2: Significant Line Detection

1. Let **S** be an empty set.
2. **For** each angle θ_k **do**
3. Rotate the grayscale sub-image in clockwise direction by the angle θ_k.
4. Filter the sub-image with a 2D Gaussian smoothing kernel with σ of 1.
5. Transform the sub-image into binary edge image by Canny algorithm.
6. Remove boundary lines of the rotated edge image.
7. Let matrix **R** containing all logical elements of the rotated edge image of sizes n x n, where $n \in I^+$ and $n \leq 300$.
8. Let $\mathbf{V_i}$ be a vector containing all logical elements of the matrix R at row i, where i = 4, 5, ..., n-3.
9. **For** each vector $\mathbf{V_i}$ in matrix **R do**
10. Let L_{max} be the line of maximum length and $|L_{max}|$ be a length of L_{max}.
11. Find the line L_{max} in the vector $\mathbf{V_i}$.
12. **If** $|L_{max}| \geq 25$ pixels **Then**
13. vector $\mathbf{O_i} = \mathbf{V_{i-3}}$ OR $\mathbf{V_{i-2}}$ OR $\mathbf{V_{i-1}}$ OR $\mathbf{V_i}$ OR $\mathbf{V_{i+1}}$ OR $\mathbf{V_{i+2}}$ OR $\mathbf{V_{i+3}}$.
14. Let L_{sig} be the line containing L_{max} and $|L_{sig}| \geq |L_{max}|$.
15. Find the line L_{sig} in the vector $\mathbf{O_i}$.
16. Add L_{sig} to **S**.
17. **EndIf**
18. **EndFor**
19. **EndFor**

The significant lines of the sub-image in T2 at centroid number 156 from all six relevant degrees of angles is demonstrated in Figure 18.

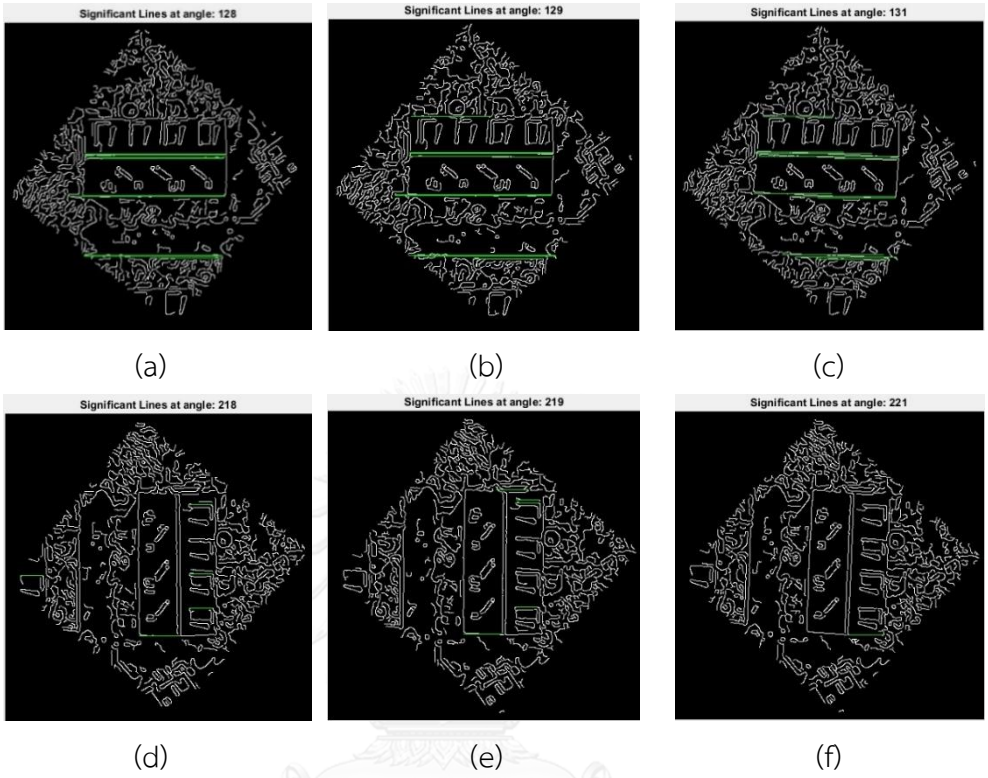


Figure 18 Significant lines of the sub-image of centroid number 156 at (a) 128 degrees (b) 129 degrees (c) 131 degrees (d) 218 degrees (e) 219 degrees and (f) 221 degrees.

3.1.2.5. Rectangle Detection

As previously mentioned, the concept of line scanning was adopted for detecting the rectangle shaped objects in the image. For this part, the proposed line scanning algorithm consists of four main procedures, as follows.

1. Specify a range of points to perform line scanning.
2. Specify a desired threshold.
3. Perform line scanning from the initial point in the range to a desired direction in the binary edge image by continuing performing logical OR operations between the vector of the initial point and that of its next point in the range until the point is out of range and/or the specific threshold is met.
4. If the specific threshold is met, it means there exists the line in the direction. Then, the detected point is collected.

According to this concept, Algorithm 3 proposed all procedures of line scanning.

Algorithm 3: Line Scanning

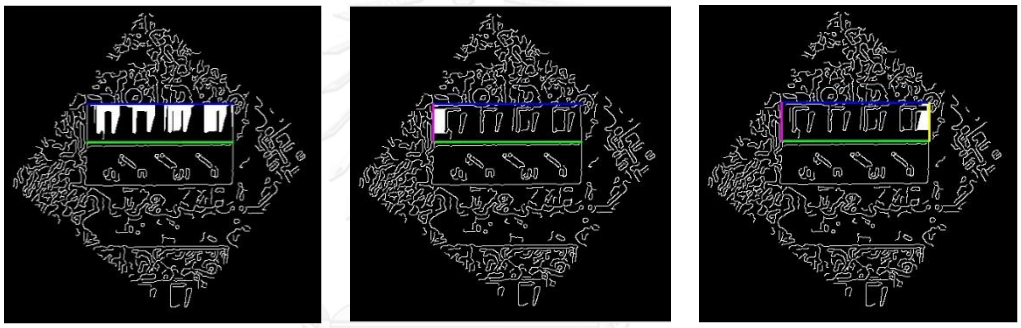
1. Let matrix E containing all logical elements in the binary edge image of size n x n, where n ≤ 300.
2. Suppose that the significant line locates on matrix E at row s and columns from a to b, where 1 ≤ s, a, b ≤ n.
3. Let R be a vector of size b-a+1 whose all elements are initialized by false.
4. Let N_R be the total number of true in the vector R.
5. Let t be a scalar value of desired threshold.
6. Let vector S = $[s_i]$ containing points for scanning, where m is a size of vector S and i = 1, 2, ..., m.
7. Let V_{s_k} be a vector containing all elements of matrix E at row s_k and columns from a to b.
8. k = 1
10. **While** k ≤ m AND N_R < t **do**
11. $R = R \text{ OR } V_{s_k}$
12. $k = k + 1$
14. **EndWhile**
15. Let d be a point of detected lines.
16. If $N_R \geq t$ then
17. d = s_k
18. **EndIf**

In fact, the rectangles are composed of four lines, including top, bottom, left, and right lines, jointed together. Consequently, this algorithm was utilized to scan for the lines in these four directions in order to detect the rectangle shape buildings in the images.

3.1.2.5.1. Detecting Rectangles

Since the detected significant lines could be top, middle, or bottom edges of the building objects, the rectangles might be located above and/or below these lines. For the sub-image of T2 at centroid number 156, the significant line number 2 was chosen to demonstrate the line which the rectangles located above and below it.

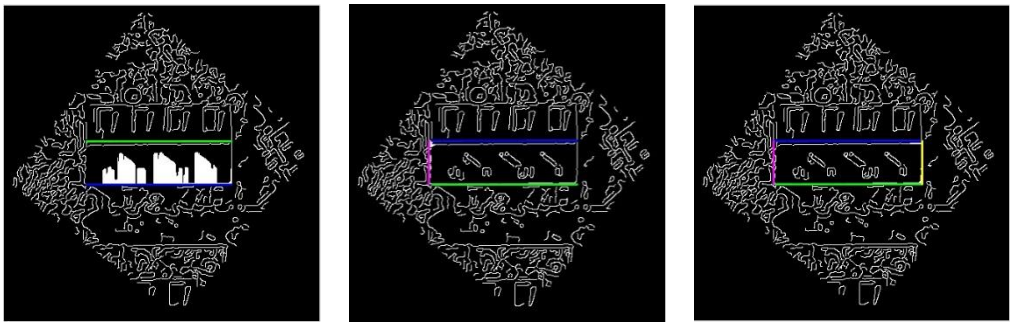
For the rectangles located above the significant lines, line scanning to top, left, and right directions were required. The demonstration of line scanning to these three directions from the significant line number 2 of the sub-image at centroid number 156 after rotation in clockwise direction by 128 degrees were shown in Figure 19.



(a) (b) (c)

Figure 19 Line scanning to (a) top (b) left and (c) right directions from the significant line number 2 of the sub-image at centroid number 156 after rotation in clockwise direction by 128 degrees.

Likewise, line scanning to bottom, left, and right directions were required for locating the boundaries of rectangles below the significant lines. The demonstration of line scanning to these three directions from the significant line number 2 of the sub-image at centroid number 156 after rotation in clockwise direction by 128 degrees were shown in Figure 20.



(a) (b) (c)

Figure 20 Line scanning to (a) bottom (b) left and (c) right directions from the significant line number 2 of the sub-image at centroid number 156 after rotation in clockwise direction by 128 degrees.

3.1.2.5.2. Merging Rectangles

In a case that the significant lines had either rectangles above or below them, the detected rectangles were simply collected. Otherwise, if the significant lines had both of rectangles above and below them, merging the detected rectangles above the lines with those below the lines was then required. This process could be depicted as shown in Figure 21.

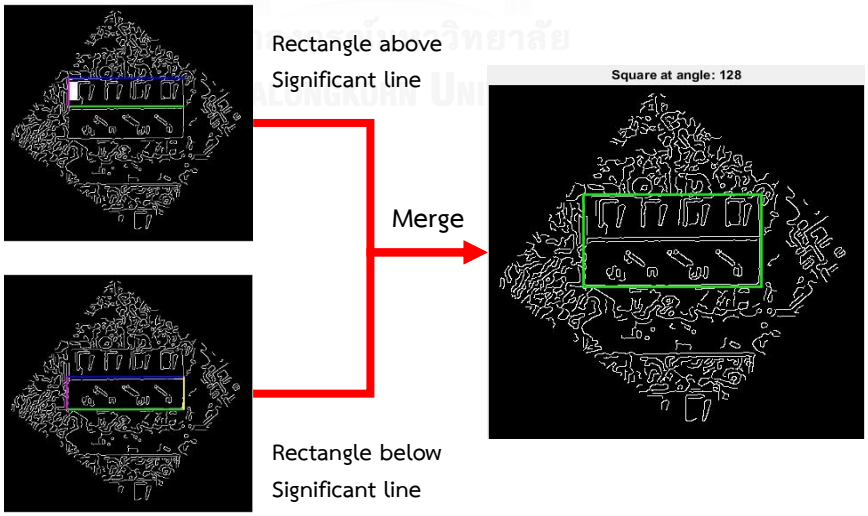


Figure 21 Merging rectangles above and below the significant line number 2 of the sub-image at centroid number 156 after rotation in clockwise direction by 128 degrees.

Figure 22 illustrated the detected rectangles of the sub-image in T2 at centroid number 156 after rotation by all six relevant degrees of angles and applying to Algorithm 3.

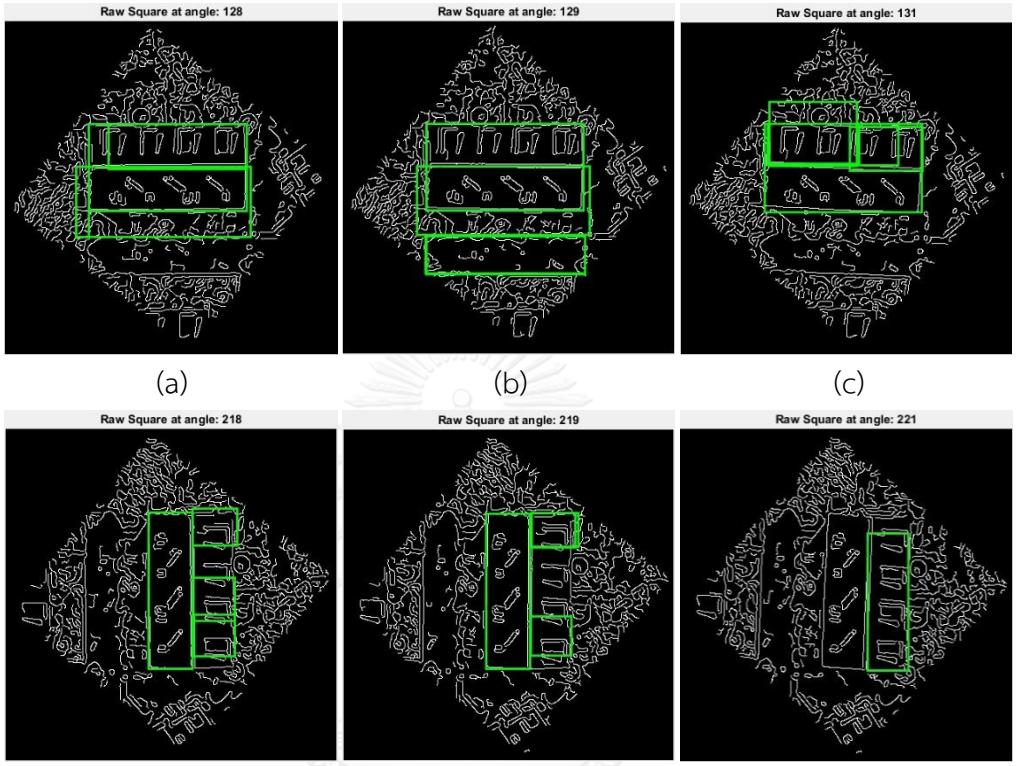


Figure 22 Detected rectangles of the sub-image at centroid number 156 after rotation by all six relevant degrees of angles.

3.1.2.5.3. Filtering Rectangles

Since this part focused on the buildings of sizes at least 50 x 50 pixels, the detected rectangles of sizes less than 2,500 pixels were then removed. In addition, the number of true elements in the detected rectangles was also beneficial to be used to filter the rectangles because buildings is one of man-made objects so they should contain smooth intensity in the grayscale image and should also have the very high number of false elements in them after transforming into binary edge image. Thus, the number of true elements in them should be very small. To avoid the noises at the

borders of the detected rectangles, this part excluded areas of 5 pixels from all borders, as demonstrated in Figure 23.

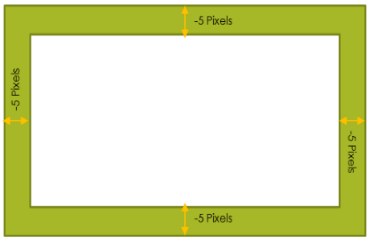
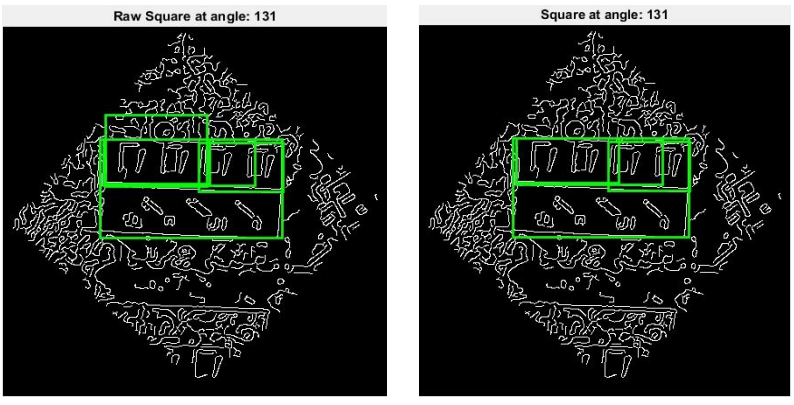


Figure 23 Detected rectangles presenting the excluded areas in green and the filtered areas in white.

Therefore, these two criteria were used to filter the detected rectangles.

1. Remove the rectangles of sizes less than 2,500 pixels.
2. Remove the rectangles that have a percentage of true elements in the filtered areas greater than 10%.

To illustrate this point of view, Figure 24 displayed the detected rectangles of the sub-image in T2 at centroid number 156 after rotation by 131 degrees before and after filtering.



(a) (b)

Figure 24 The detected rectangles of the sub-image at centroid number 156 after rotation by 131 degrees (a) before and (b) after filtering.

Besides, Figure 25 displayed the detected rectangles of the sub-image in T2 at centroid number 156 after rotation by all six relevant degrees of angles and filtration.

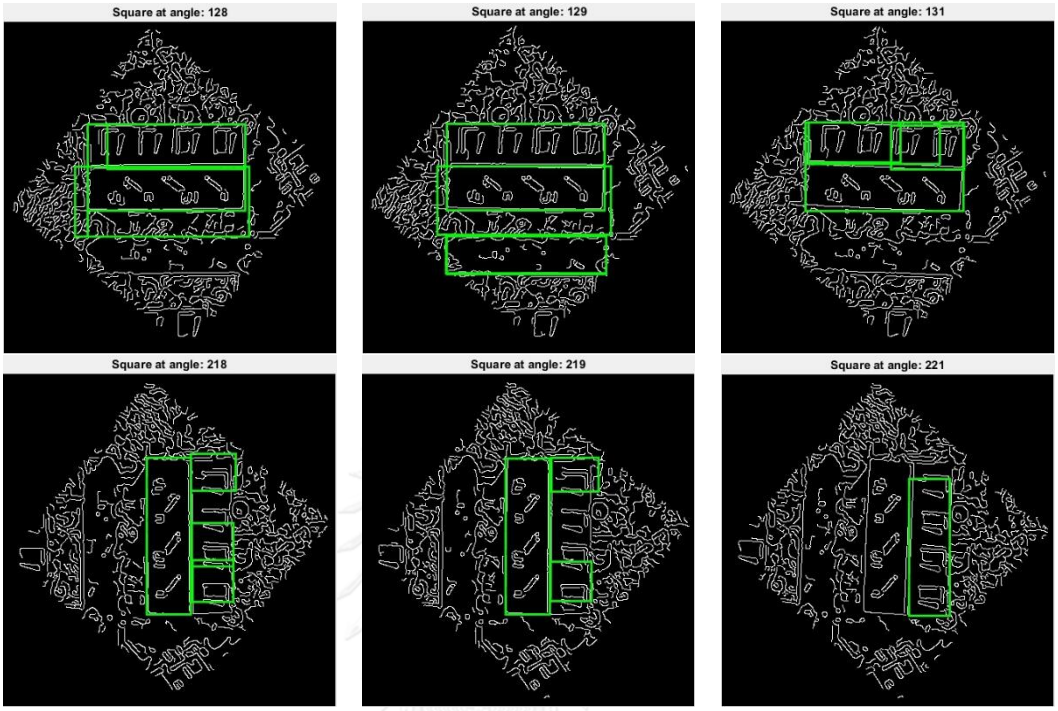


Figure 25 Detected rectangles of the sub-image at centroid number 156 after rotation by all six relevant degrees of angles and filtration.

After filtration, the detected rectangles were contained in binary image, where true represented areas of detected rectangles and false represented non-detected areas. The binary image was subsequently rotated by the same degrees of angles, in which it was previously rotated, in reverse direction to be turn it back to the original orientation.

3.1.2.5.4. Gaining Reliable Rectangles

For each degrees, one binary image, which was a logical matrix containing digit 1s in detected rectangular areas and 0s in non-detected areas, was obtained. Thus, the results from all different degrees were combined by adding these matrix together.

Figure 26 displayed a color image of the sub-image at centroid number 156 gained by adding the result matrices from all six degrees of angles. Each color represented each summation value.

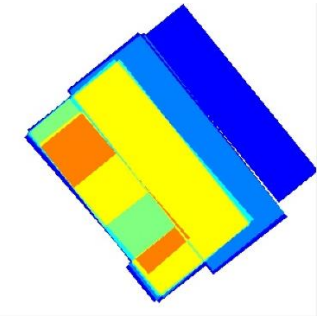


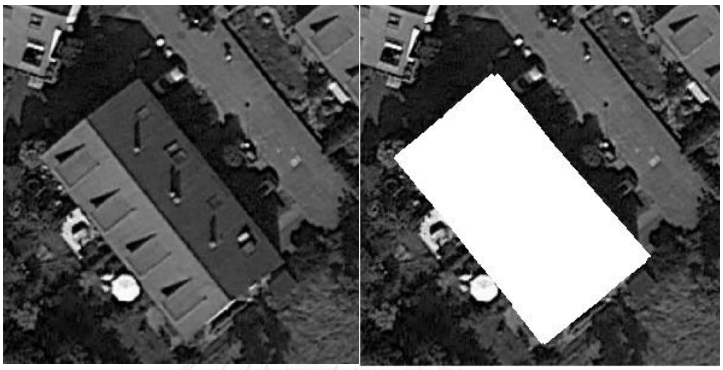
Figure 26 Color image of the sub-image at centroid number 156 gained by adding the result matrices of all relevant degrees.

Since the relevant angles of this part included both of the degrees extracted by Hough transform and their perpendicular degrees, it increased a possibility of areas to be detected by line scanning algorithm. Therefore, the building objects should be detected more than one time among different degrees of angles if there were buildings located at the areas. After adding the matrices of all degrees, the obtained summation values could be used to represent a reliability level of the detected areas. For example, if the summation values was five, it meant that the areas were detected as rectangular shape objects or buildings, five times by five degrees of angles.

Thus, this part empirically used 50% of the maximum summation values as our final result of detected rectangular areas. For instance, if the maximum summation value was six, all detected areas containing their summation values greater than or equal to three were extracted as the buildings. However, when the maximum summation were less than or equal to two, the rectangular areas in the sub-images might be hard to detect. It could be caused by several reasons, such as broken lines, too much noises, or degrees of angles irrelevant to the objects. In that case, all detected areas were results.

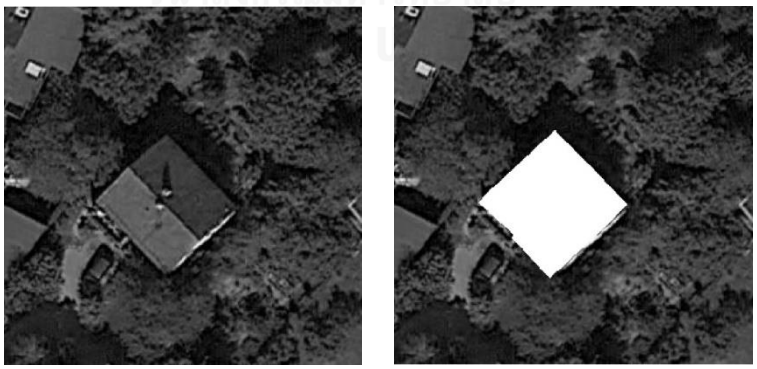
Besides, it had a possibility to be gained small areas of objects by specifying the range of summation values to be accepted. Therefore, filtering with object size was required for removing the small areas that cannot be buildings. As previously

mentioned, buildings of size 50 x 50 pixels were focused on this part. In fact, the whole buildings might practically not be gained because their shapes were not rectangle exactly or they were renovated. Thus, the expected size of 50 pixels would be slightly reduced to 40 pixels, and the buildings of sizes less than 1,600 pixels were, then, removed. Figure 27 and Figure 28 displayed the detected buildings obtained from the proposed line scanning algorithm, in the sub-images of different centroids.



(a) (b)

Figure 27 The sub-image of centroid number 156. (a) Original sub-image (b) Detected building resulted from the proposed line scanning.



(a) (b)

(c)

(d)

(e)

(f)

(g)

(h)

(i)

(j)

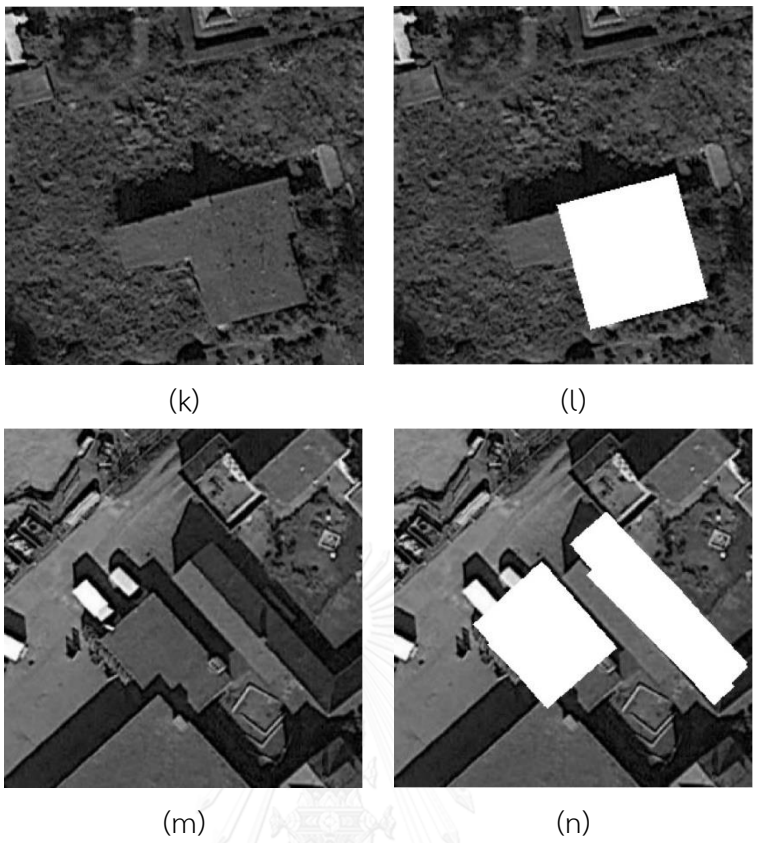


Figure 28 The sub-image of different seven centroids.
Left column is represented for the original sub-images and
right column is represented for the detected buildings obtained by the proposed line scanning.

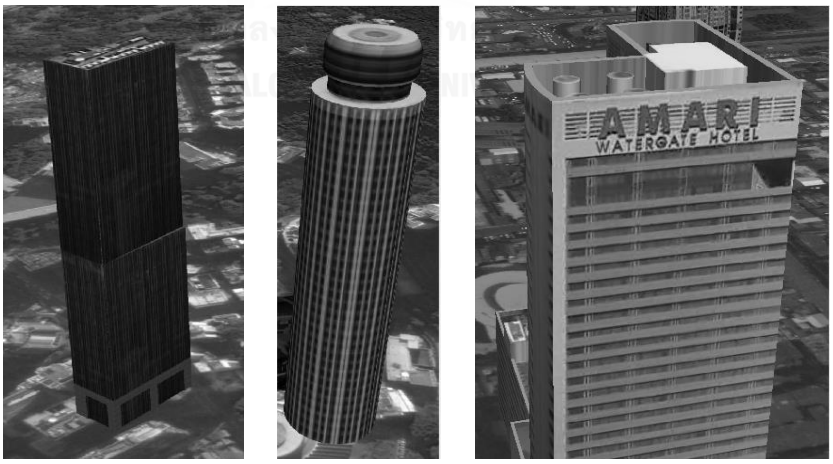
Since the sub-images were chopped according to the identified centroids, it required a procedure for combining the line scan results of sub-images in the original tested images. In addition, if the detected areas obtained from adjacency centroids were very close to each other, they can be merged to one area. To avoid this problem, the number of detected objects in the sub-images was counted before and after merging the line scan results. If the number of detected objects after merging was greater than or equal to the number of those before merging, the merged result was turned back to its location in the original image.

3.2. Perspective View Images

Likewise, the proposed line scanning can be further applied to perspective view images for detecting the buildings. For perspective view images, the buildings are mainly in a vertical orientation, and the heights of them are quite high. For this reason, the vertical lines are focused to be used to detect the buildings in the images. Therefore, this part of dissertation applies the concept of line scanning to develop a vertical line scanning algorithm for detecting the buildings in perspective view images.

3.2.1. Tested Images

In order to evaluate the performance on building detection, perspective view images were captured by Google earth. For preliminary study, three tested perspective view images each of which contained only one building were used to test and develop the vertical line scanning algorithm. They were named as P1, P2, and P3 as shown in Figure 29. Furthermore, the tested images P4 and P5 located on coordinates (13°43'19.71"N 100°32'17.57"E) and (22°16'22.08"N 114°10'10.59"E), were captured for a testing procedure. Both images P4 and P5 are shown in Figure 30.



(a) (b) (c)

Figure 29 Tested Perspective View images (a) P1 (b) P2 and (c) P3.

(a)

(b)

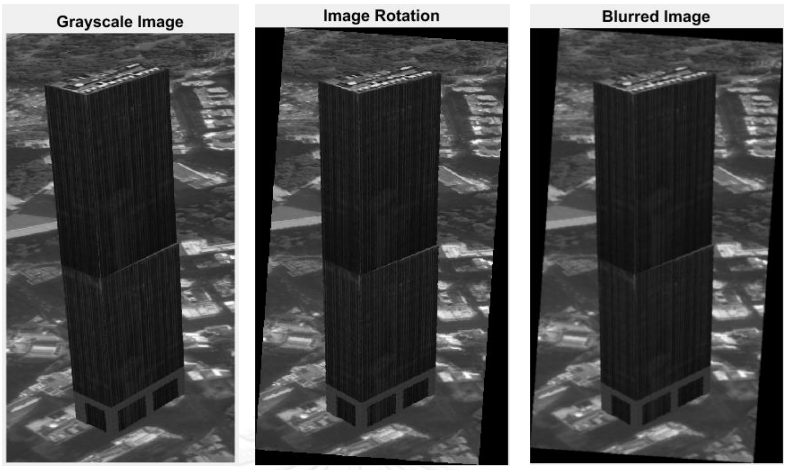
Figure 30 Tested Perspective View images (a) P4 and (b) P5.

3.2.2. Vertical Line Scanning

Likewise, the procedures of the vertical line scanning algorithm are quite similar to those of the line scanning algorithm used in top view images due to the fact that these two algorithms are based on the same concept. However, there are some procedures that are slightly different. The details of the vertical line scanning algorithm are as follows.

Since this research part focused on the vertical lines, each tested image was rotated by specific degrees of angles, which ranged from 15 degrees in counter clockwise direction to 15 degrees in clockwise direction, with step size of 1 degree.

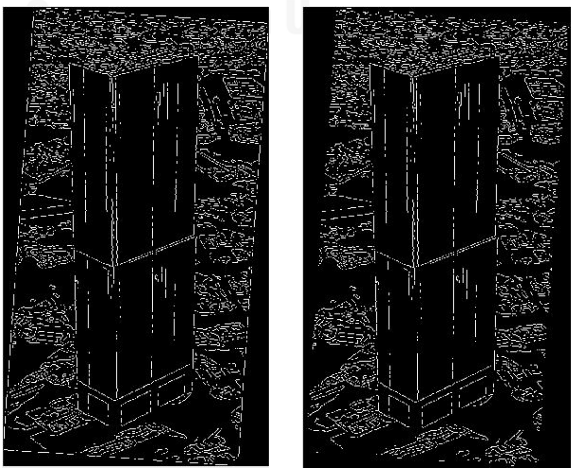
For each angle, the grayscale image in perspective view was rotated by a degree, and filtered with a 2-D Gaussian smoothing kernel with **σ** of 1, as demonstrated in Figure 31.



(a) (b) (c)

Figure 31 The tested image P1 (a) Grayscale image (b) Rotated image and (c) Filtered Image

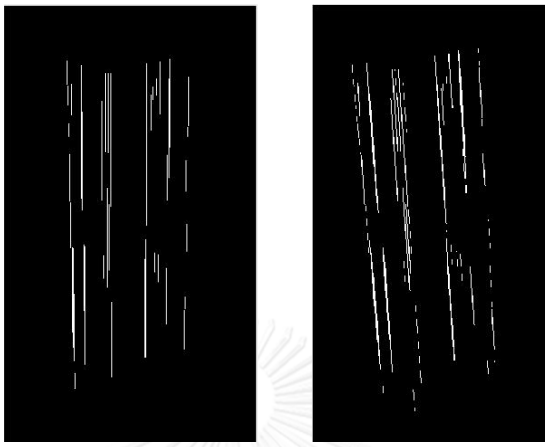
The rotated and filtered image was transformed into a binary edge image by Canny algorithm, followed by removing the boundary lines of the image, as demonstrated in Figure 32.



(a) (b)

Figure 32 The rotated and filtered image P1 (a) before removing boundary lines and (b) after removing boundary lines.

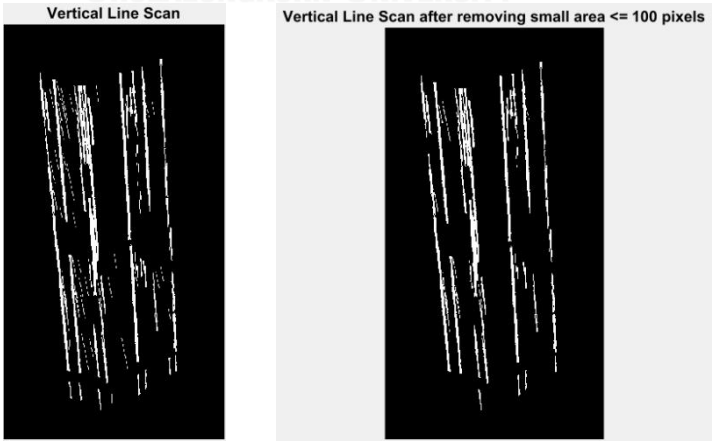
From the edge image, the vertical lines whose lengths were greater than or equal to 15 pixels, were extracted, and the image was rotated by the same degree in the reverse direction so that it turned back to its original orientation, as displayed in Figure 33.



(a) (b)

Figure 33 (a) The rotated image of P1 with extracted vertical lines (b) The rotated image of P1 with extracted vertical lines after rotation in the reverse direction by 4 degrees.

With vertical line scanning algorithm, the vertical lines of the image among all specific range of degrees were obtained, and small areas less than or equal to 100 pixels were subsequently removed, as displayed in Figure 34.



(a) (b)

Figure 34 Results of the tested image P1 by (a) vertical line scanning (b) vertical line scanning and small area removing.

Algorithm 4 describes a procedure of how to extract the vertical lines from the perspective view images by line scanning.

Algorithm 4: Vertical Line Scanning on Perspective view image

1. **For** each angle Θ_k **do**
2. Rotate the grayscale image by Θ_k.
3. Filter the image with a 2-D Gaussian smoothing kernel with σ of 1.
4. Transform to binary edge image by Canny algorithm.
5. Remove boundary lines of the rotated edge image.
6. Let matrix M be the rotated edge image with n rows and m columns.
7. Let vector V_j contain all logical elements of the matrix M at column j where j = 3, 4, ..., m-2.
8. **For** each vector V_j in matrix M **do**
9. Let L_{max} be the line of maximum length and $|L_{max}|$ be a length of L_{max}.
10. Find the line L_{max} in the vector V_j.
11. If $|L_{max}| \geq 15$ pixels Then
12. vector $O_j = V_{j-2}$ OR V_{j-1} OR V_j OR V_{j+1} OR V_{j+2}.
13. Let L_{sig} be the line containing L_{max} and $|L_{sig}| \geq |L_{max}|$.
14. Find the line L_{sig} in the vector O_j.
15. Add L_{sig} to the rotated edge images.
16. **EndIf**
17. **EndFor**
18. Rotate the image by Θ_k in the reverse direction and store the image.
19. **EndFor**
20. Merge all images obtained from all different degrees of angles.
21. Remove small areas less than 100 pixels by image opening.

CHAPTER 4 EXPERIMENTAL RESULTS AND DISCUSSION

The experimental results are demonstrated and discussed in this chapter. They are separately summarized according to two different views of images, as follows.

4.1. Top View Images

4.1.1. Results of Building Detection

The results of the proposed line scanning are compared with those of Karsli's method in order to evaluate the performance of building detection.

4.1.1.1.Karsli's method

Figure 35 displayed the tested images T1 and T2 and their results obtained by Karsli's method, as referred in [33]. Their results are displayed in binary images in which black areas represents the detected areas of buildings and white areas represents the non-detected areas. According to these images, they show that the locations and shapes of the buildings of the results are quite similar to those of the tested images. Since they evaluated their performances by pixel based building detection, their results could only display the detected building areas but the total number of buildings and the locations of buildings could not be identified.

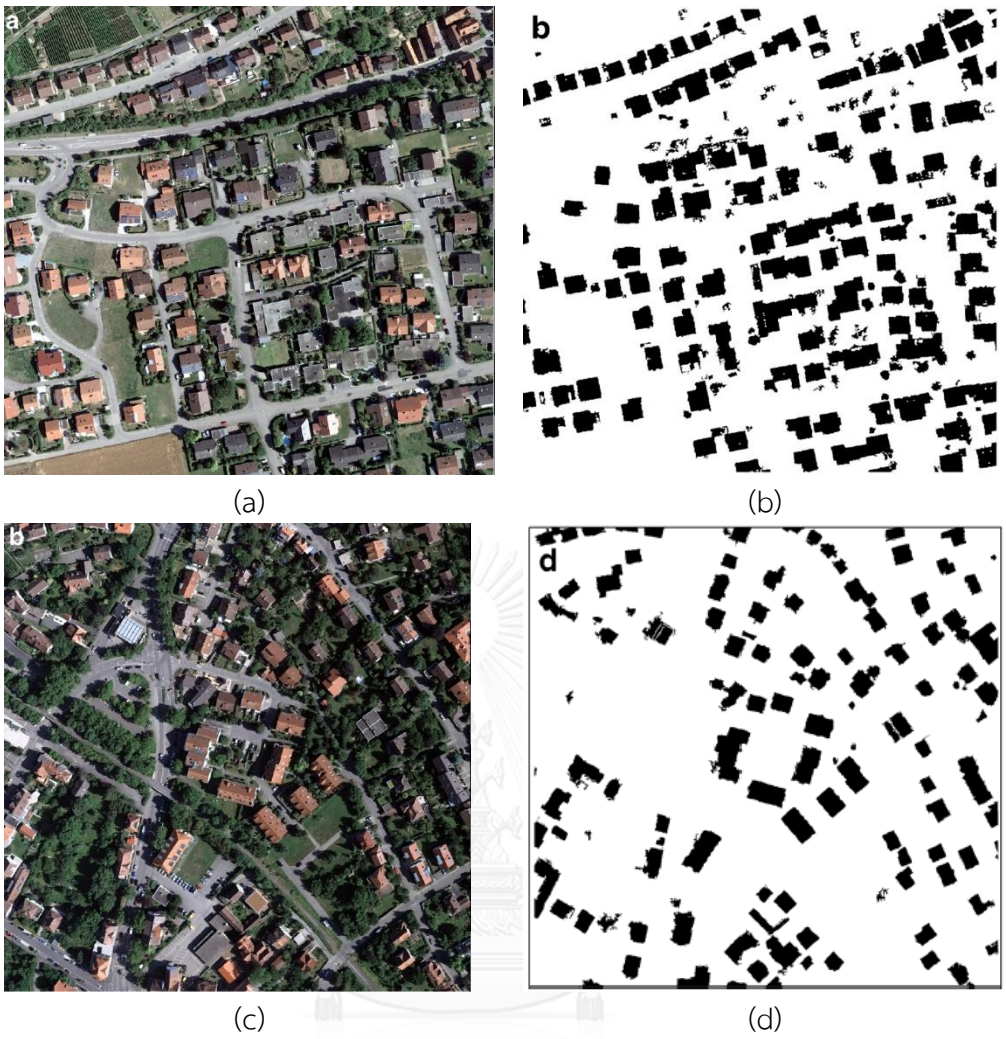


Figure 35 (a) Tested Images T1 (b) Result of building detection in T1 by Karsli's method [33] (c) Tested Images T2 (d) Result of building detection in T2 by Karsli's method [33].

4.1.1.2. Proposed Line Scanning

Figure 36 displayed the tested images T1 and T2 and their results of building detection by the proposed line scanning. From these results, each block of white areas represents each detected building objects, and the objects are mainly overlapped with locations and shapes of the buildings in the tested images. It signifies that a detection of rectangle shape objects by the proposed line scanning can be efficiently applied to detect the buildings in the top view images. In addition, the total number of detected

buildings and their locations can be identified by this proposed method since it is an object based building detection.

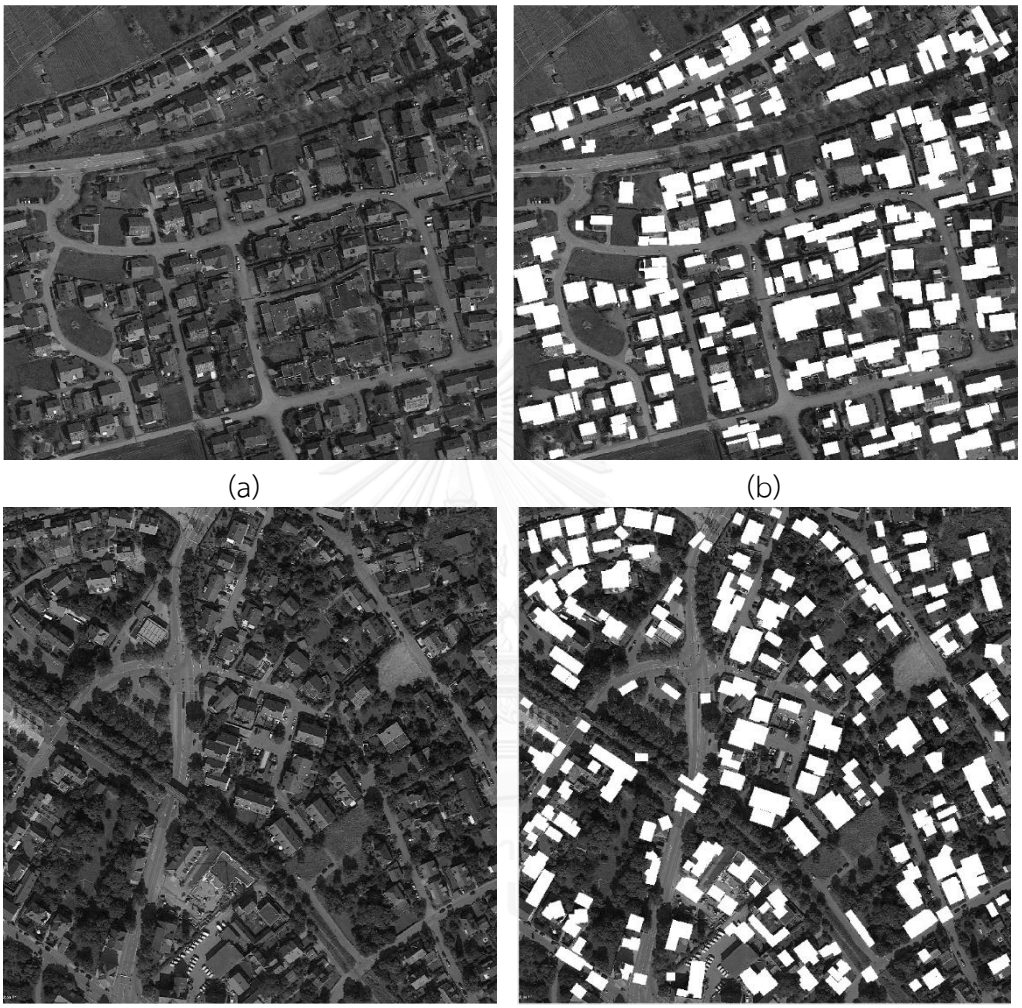


Figure 36 (a) Tested Images T1 (b) Result of building detection in T1 by the proposed line scanning (c) Tested Images T2 (d) Result of building detection in T2 by the proposed line scanning.

In order to demonstrate the compared results between Karsli's method and the proposed line scanning, Figure 37 and Figure 38 display some interested issues of the compared results between these two methods in the tested image T1 and T2, respectively. From these figures, they demonstrate some examples of the missing areas in blue ellipses, the merged buildings in red ellipses, and the false positives in the green ellipses.

In case of missing areas, the buildings locating on the left side of the result in T1 by Karsli's method are almost all missed. Some buildings are missed the whole parts of them and some buildings are missed only some part. Likewise, the results of building detection by the proposed line scanning displays some missing buildings. This may be caused by shapes of buildings, broken lines, or others. For example, if the buildings locating near a border of the image, their shapes may not be rectangle exactly due to missing some parts of them. As a result, line scanning will not detect them because their shapes are not rectangles. Particularly, Figure 38 displays many missing buildings found in the result of building detection in T2 by Karsli's method, as shown by the blue ellipses. In contrast, only the small number of missing buildings are found in the result of T2 by the proposed line scanning.

In case of merged buildings, the results of building detection in T1 and T2 by Karsli's method displays the greater number of merged buildings than those of the proposed line scanning. This is because Karsli's method aims at extracting the building areas for pixel based building detection while the proposed method is designed for object based building detection. Thus, it can extract the building objects in the images by additional defining the criteria for merging the objects in the adjacency locations.

In case of false positive, Karsli's method used the height information from the LiDAR data as one of criteria for differentiating the building areas from the non-buildings. Thus, some false positives are found in their results since many trees may have height values closed to the buildings so it is hard to differentiate between them, and, then, tree areas are misclassified as building areas. For the proposed line scanning,

false positives are also found in its results but it can be noticed that they are mainly some rectangle shaped areas, such as roads or green yards.

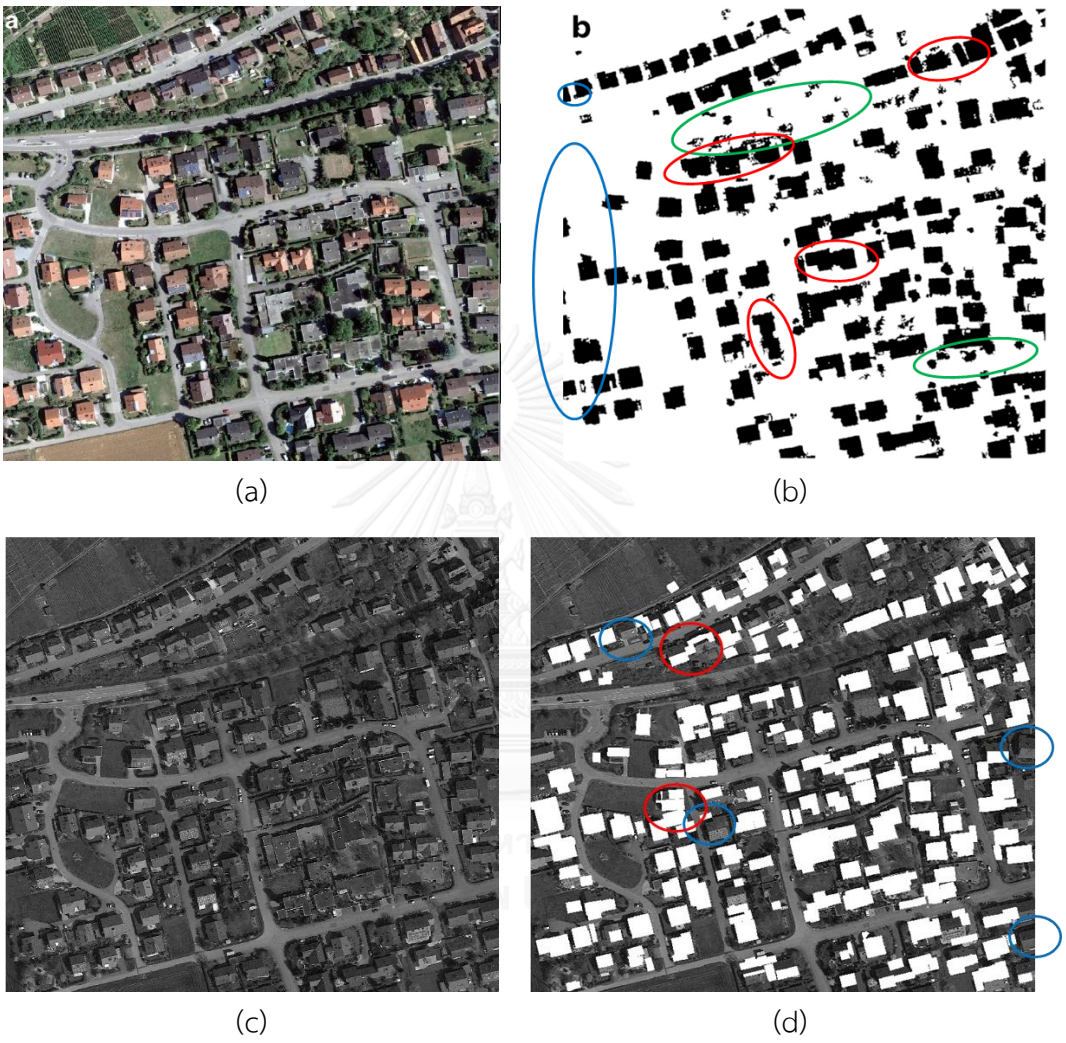
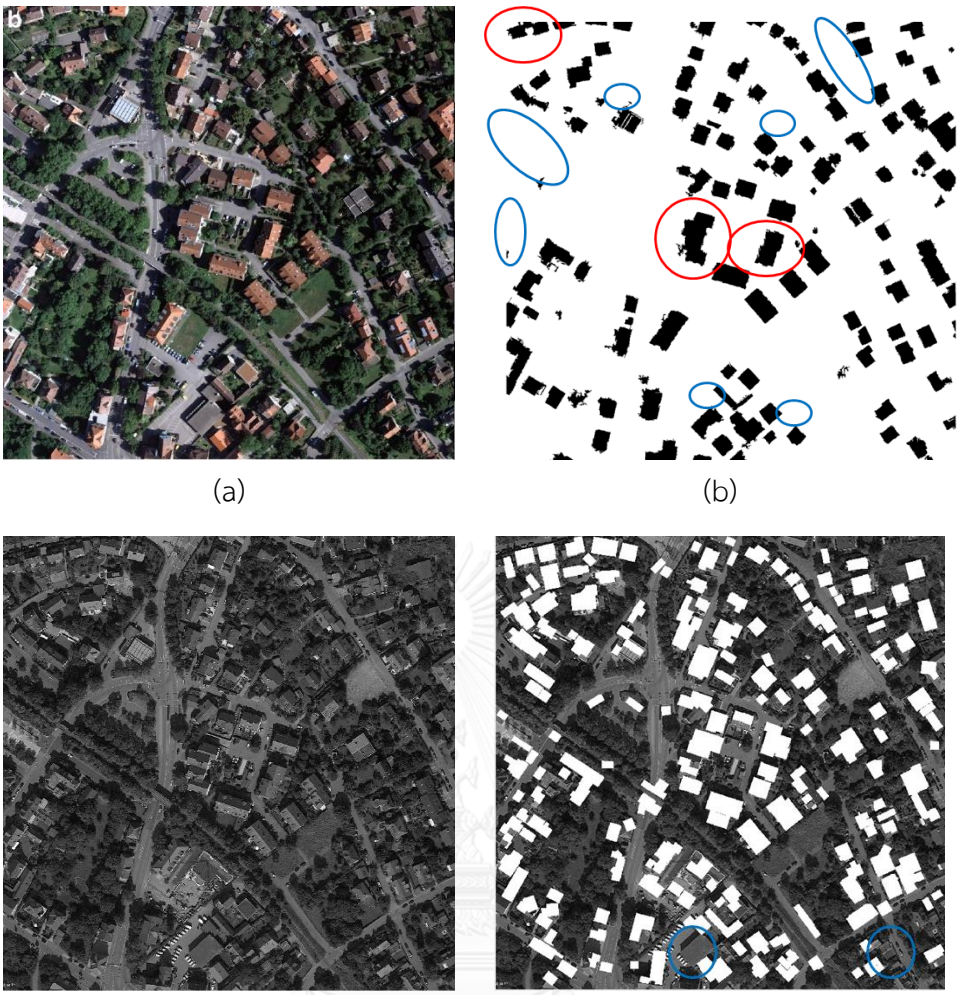


Figure 37 Compared results (a) T1 and (b) Result of building detection in T1 by Karsli's method (c) T1 and (d) Result of building detection in T1 by the proposed line scanning.



(a) (b)

(c) (d)

Figure 38 Compared results (a) T2 and (b) Result of building detection in T2 by Karsli's method (c) T2 and (d) Result of building detection in T2 by the proposed line scanning.

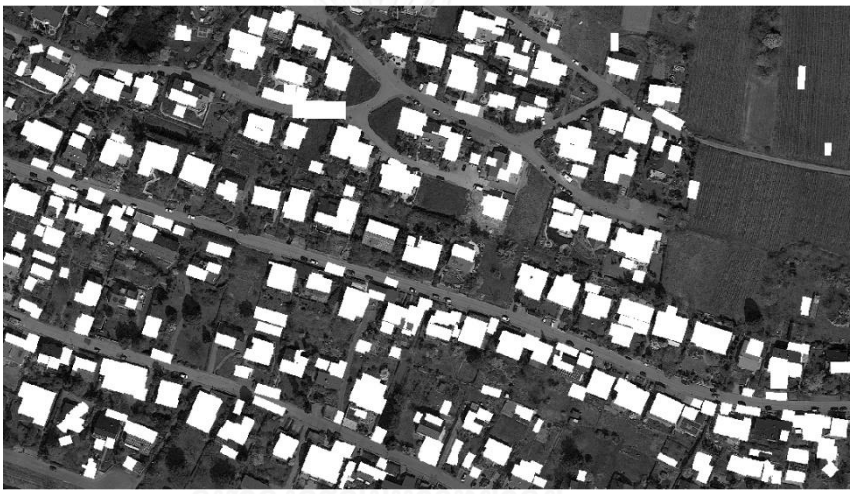
Additionally, three tested images T3, T4, and T5 are also used to evaluate the performance of building detection of the proposed line scanning, and their results are demonstrated in Figure 39, Figure 40, and Figure 41, respectively. From the experimental results, they show the same direction as the previous two tested images T1 and T2.

(a)

(b)

Figure 39 (a) Tested Image T3 (b) Result of building detection in T3 by line scanning.

(a)



(b)

Figure 40 (a) Tested Image T4 (b) Result of building detection in T4 by line scanning.

(a)

(b)

Figure 41 (a) Tested Image T5 (b) Result of building detection in T5 by line scanning.

4.1.2. Performance

For this part, the objects that are located on the same positions of the buildings in the tested images are determined as detected buildings. For each tested image, the number of detected buildings is divided by the total of buildings for computing accuracy. The accuracies of the results of building detection by Karsli's method and those of the proposed line scanning are shown in Table 2.

Table 2 Accuracy of building detection between Karsli's method and the proposed line scanning algorithm

Image	Method	Total number of buildings	The number of detected buildings	Accuracy
T1	Proposed method	124	117	94.35%
	Karsli's method	124	112	90.32%
T2	Proposed method	111	101	90.99%
	Karsli's method	111	93	83.78%
T3	Proposed method	122	114	93.44%
T4	Proposed method	138	126	91.30%
T5	Proposed method	168	154	91.67%

From the experimental results, they show that the accuracies of the proposed line scanning are higher than those of Karsli's method, in the tested images T1 and T2. It can be implied that the proposed line scanning can achieve higher performance than Karsli's method in detecting buildings in the tested images. Additionally, the proposed method can successfully achieve at least 90% of accuracy in all five tested images. Thus, it signifies that the proposed line scanning is an efficient method for detecting the buildings in the top view images.

In addition to the performance between these two methods, some issues should be considered, as follows.

In aspect of input, Karsli's method uses an optical image (RGB) and LiDAR systems (digital terrain model and digital surface model) as the input for their method. This input contain various kinds of specific information that need surveyors, time and cost to acquire them. Moreover, large memory is also required for storing the information. In contrast, the proposed line scanning requires only grayscale images captured by Google earth as its input. This image type contains only intensity information and the Google earth is a widely used program that anyone can access.

In aspect of method, Karsli's method uses texture analysis and feature extraction techniques with support vector machine classification for their building detection. Therefore, time consuming is required for training and masking processes. In contrast, both of feature extraction and training process are not required for the proposed line scanning. Thus, any prior knowledge and learning process are not required before testing.

In aspect of performance, Karsli's method measures the performance of their results by pixel based building detection. As previously mentioned, it can only extract the building areas in the images but it cannot identifies the number of buildings and their locations, while the proposed line scanning is designed for object based building detection so it can identify the total number of the buildings as well as their locations in the images.

In fact, all detected objects from the proposed line scanning can be building or non-building objects. In this research, true positives and false positives are considered, as revealed in Table 3. For this part, true positive represents the detected objects locating on the building areas of the tested images, while false positive represents the detected objects locating on the non-building areas.

Table 3 The number of true positives and false positives of the results of building detection by the proposed line scanning

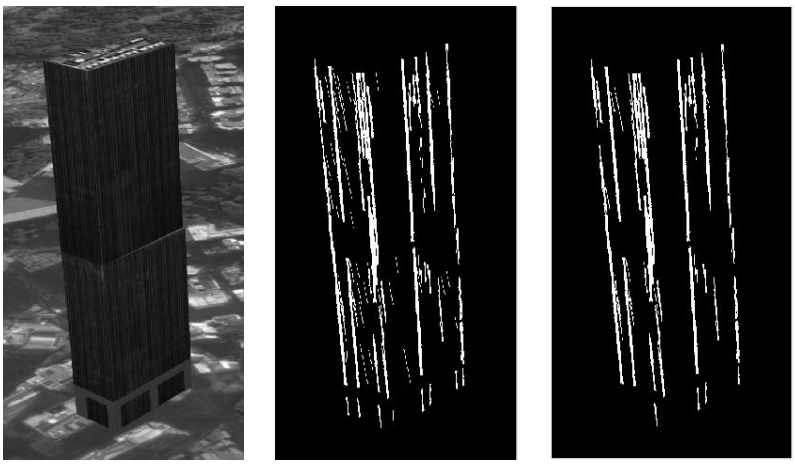
Image	The number of Detected Objects	TP	FP
T1	148	128	20
T2	161	124	37
T3	200	147	53
T4	208	150	58
T5	235	180	55

4.2. Perspective View Images

4.2.1. Results of Building Detection

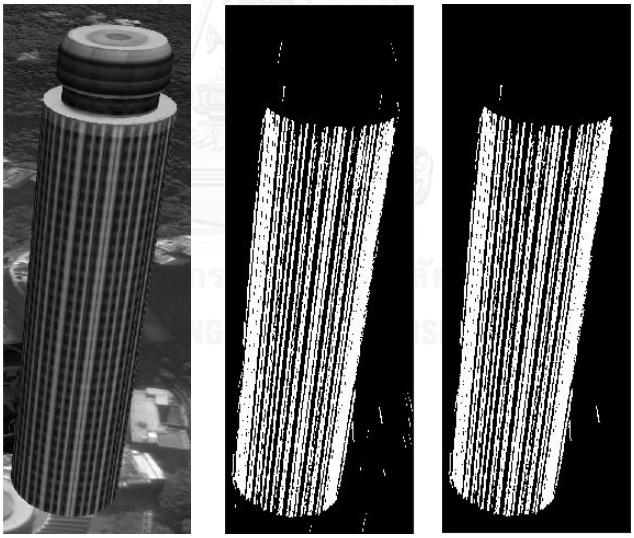
Similarly, the proposed line scanning can be subsequently applied to building detection in perspective view images. As previously mentioned, the buildings in perspective view images are mainly in a vertical orientation, and they are very high. Therefore, the vertical lines are focused, and a vertical line scanning algorithm is developed to detect the vertical lines that are relevant to the buildings.

Figure 42, Figure 43, and Figure 44 demonstrate the results obtained from vertical line scanning algorithm of tested images P1, P2, and P3, respectively. From these images, it displays that the algorithm can extract the vertical lines related to the buildings, including rectangle shaped buildings (P1 and P3) and tube shaped buildings (P2). Thus, it can be implied that the proposed line scanning can efficiently extract the vertical lines related to the buildings, independent of shapes.



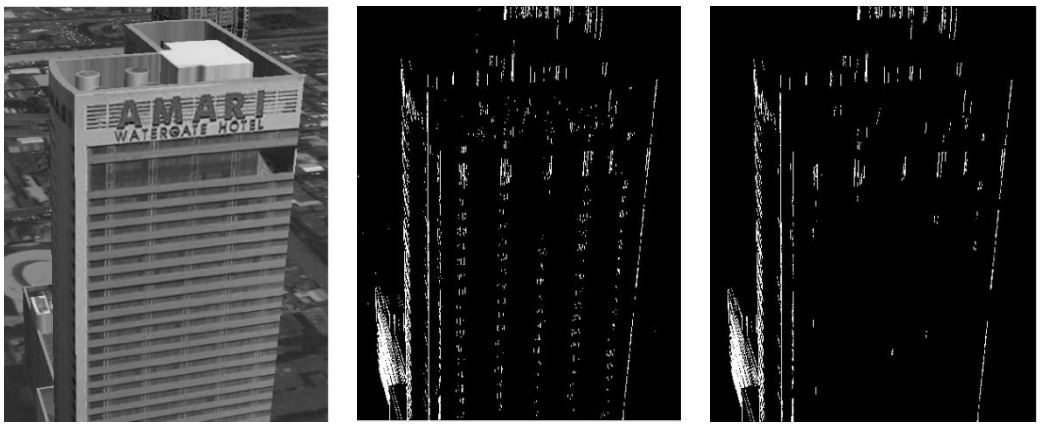
(a) (b) (c)

Figure 42 (a) Tested Image P1 (b) Result of P1 after vertical line scanning (c) Result of P1 after vertical line scanning and small area removing.



(a) (b) (c)

Figure 43 (a) Tested Image P2 (b) Result of P2 after vertical line scanning (c) Result of P2 after vertical line scanning and small area removing.

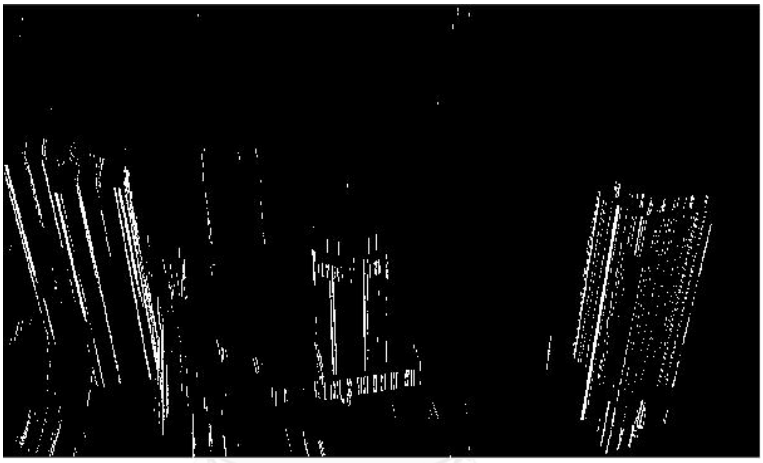


(a) (b) (c)

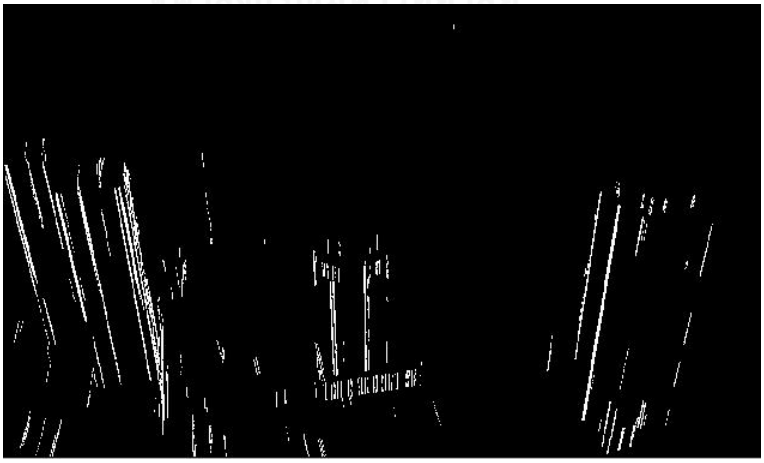
Figure 44 (a) Tested Image P3 (b) Result of P3 after vertical line scanning (c) Result of P3 after vertical line scanning and small area removing.

After testing with greater sizes of images which contain various kinds of buildings, Figure 45 and Figure 46 demonstrate the results gained by the vertical line scanning algorithm of tested perspective view images P4 and P5, respectively. The results show that it can extract vertical lines that are related to all buildings in the images. In the image P4, the locations of buildings close to each other so the extracted lines are really closed. Thus, it is required a procedure for clustering the lines of the same buildings together. The information of intensity, length, and position of these lines may be utilized for distinguishing the buildings to which these lines belong. Since the buildings in the image P5 quite far from each other, it gains the good and clear result. The lines belonging to each building are grouped obviously.

(a)



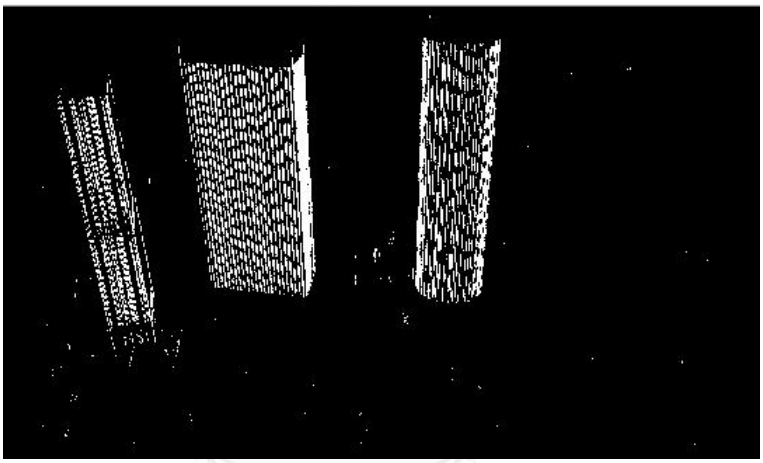
(b)



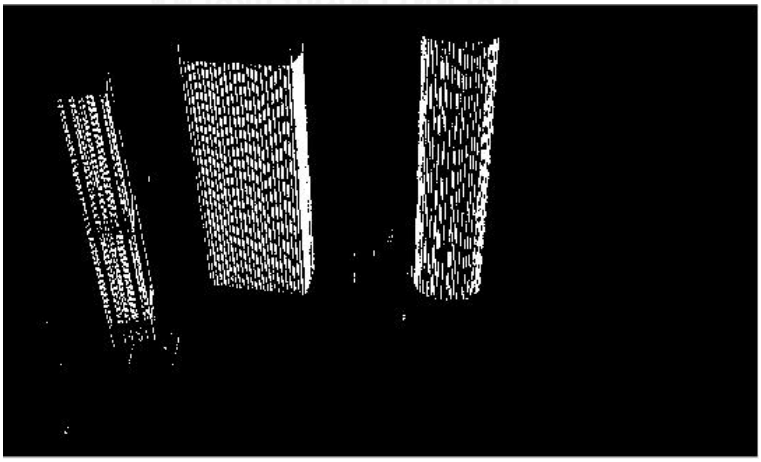
(c)

Figure 45 (a) Tested Image P4 (b) Result of P4 after vertical line scanning and (c) Result of P4 after vertical line scanning and small area removing.

(a)



(b)



(c)

Figure 46 (a) Tested Image P5 (b) Result of P5 after vertical line scanning and (c) Result of P5 after vertical line scanning and small area removing.

4.2.2. Performance

To evaluate the performance of vertical line scanning algorithm, the total number of buildings in the tested images and the number of detected buildings are counted. For this part, the buildings to which the extracted lines belong, are considered as the detected buildings, and the results are revealed in the Table 4. According to the table, it illustrates that the vertical line scanning can extract the lines related to the buildings with high performance. However, the clustering of these lines is required for identifying the buildings, as previously mentioned.

Table 4 The number of buildings and detected buildings gained by the vertical line scanning, in the perspective tested images

Image	Total number of buildings	The number of detected buildings
P1	1	1
P2	1	1
P3	1	1
P4	5	5
P5	4	3

CHAPTER 5 CONCLUSIONS

An identification of geographic objects such as buildings and roads from images is a challenging task in image processing research since it can be applied to various kinds of applications. For instance, city planning, land valuation, disaster management, and cartographic mapping. Consequently, the objective of this dissertation is to locate buildings in urban area from geometrical information, and an automatic building detection based on line scanning has been proposed. For this dissertation, the images with two different points of views are interested so the experiment can be separately summarized according to each type of images.

For top view, all possible candidate areas are initially identified on the tested image. To avoid redundant time consuming, the image is chopped into sub-images according to the candidate areas. An appropriate size of the sub-images is approximately estimated by a histogram of candidate area sizes. Then, degrees of angles relevant to the candidate areas are investigated by Hough transform. All significant lines related to the sub-images are subsequently extracted to be used as initial lines. Finally, the rectangle shaped objects are detected by the proposed algorithm. The experimental results show that the proposed line scanning algorithm can acquire higher performance than Karsli's method, and can achieve at least 90% of accuracy for object based building detection.

For perspective view, the proposed algorithm is subsequently applied to a vertical line scanning in order to detect the vertical lines that are building components, since the buildings in this view are very high and mainly in a vertical orientation. With this algorithm, the tested image is rotated by specific degrees of angles, filtered with a 2D Gaussian smoothing kernel, and transformed into edge image by Canny algorithm. Then, the vertical lines are extracted and the edge image is subsequently rotated back to its original orientation, followed by removing small areas. The experimental results show that the proposed line scanning can efficiently extract the vertical lines that are building components, independent of building shapes. Additionally, a clustering analysis is required for identifying a group of these lines to an individual building, in order to identify the number of buildings and their locations in the images.

Due to the fact that all buildings are composed of at least four main boundary lines, the algorithms based on line and shape detections are the significant approaches to a success of building detection. The proposed line scanning algorithm will be further improved to achieve the high performance on building detection for both top and perspective view images.

REFERENCES

1. Mishra, A., A. Pandey, and A.S. Baghel. *Building detection and extraction techniques: A review*. in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*. 2016.
2. Singh, D., et al. *Building Extraction from Very High Resolution Multispectral Images using NDVI based Segmentation and Morphological Operators*. in *The Proceedings of 2012 Students Conference in Engineering and Systems (SCES)*. 2012. Allahabad, Uttar Pradesh: IEEE.
3. Pakizeh, E. and M. Palhang. *Building Detection from Aerial Images Using Hough Transform and Intensity Information*. in *The Proceeding of 2010 18th Iranian Conference on Electrical Enginerring (ICEE)*. 2010. Isfahan, Iran: IEEE.
4. Benedek, C., X. Descombes, and J. Zerubia. *Building Detection in a Single Remotely Sensed Image with a Point Process of Rectangles* in *The Proceddings of 20th International Conference on Pattern Recognition(ICPR)*. 2010. Istanbul: IEEE.
5. Sırmaçek, B. and C. Ünsalan, *Urban-Area and Building Detection Using SIFT Keypoints and Graph Theory.* GEOSCIENCE AND REMOTE SENSING, 2009. **47**(4): p. 1156-1167.
6. Wang, Q., et al., *A hierarchical Cennection Graph Algorithm for Gable-Roof Detection in Aerial Image.* IEEE Geoscience and Remote Sensing Letters, 2010. **8**(1): p. 177-181.
7. Cote, M. and P. Saeedi, *Automatic Rooftop Extraction in Nadir Aerial Imagery of Suburban Regions Using Corners and Variational Level Set Evolution.* IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, 2013. **51**(1).
8. Sırmaçek, B. and C. Ünsalan, *A Probabilistic Framework to Detect Buildings in Aerial and Satellite Images.* IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, 2011. **49**(1): p. 211-221.
9. Ozgun, A., C. Senaras, and B. Yuksel, *Automated Detection of Arbitrarily Shaped Buildings in Complex Environments From Monocular VHR Optical Satellite Imagery.* Geoscience and Remote Sensing, 2013. **51**: p. 1701-1717.
10. Awrangjeb, M. and C. Fraser, *Automatic Segmentation of Raw LIDAR Data for Extraction of Building Roofs.* Remote Sensing, 2014. **6**(5): p. 3716.
11. Cetin, M., U. Halici, and Ö. Aytekin. *Building detection in satellite images by textural features and Adaboost*. in *The Proceedings of 2010 IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS)*. 2010. Istanbul: IEEE.
12. Li, N., et al., *A Novel Technique Based on the Combination of Labeled Co-Occurrence Matrix and Variogram for the Detection of Built-up Areas in High-Resolution SAR Images.* Remote Sensing, 2014. **6**(5): p. 3857.
13. Shorter, N. and T. Kasparis, *Automatic Vegetation Identification and Building Detection from a Single Nadir Aerial Image.* Remote Sensing, 2009. **1**(4): p. 731.
14. Gui, R., et al., *Individual Building Extraction from TerraSAR-X Images Based on Ontological Semantic Analysis.* Remote Sensing, 2016. **8**(9).

15. Hernandez-Gomez, G., et al. *Natural Image Segmentation Using the CIELab Space*. in *The Proceedings of International Conference on Electrical, Communications, and Computers (CONIELECOMP)*. 2009. Cholula, Puebla: IEEE.
16. Shen, G., L. Jiang, and G. Zhang. *An Image Retrieval Algorithm Based on Color Segment and shape moment invariants*. in *The Proceedings of Second International Symposium on Computational Intelligence and Design*. 2009. Changsha: IEEE.
17. Kwok, N.M., Q.P. Ha, and G. Fang. *Effect of Color Space on Color Image Segmentation*. in *2009 2nd International Congress on Image and Signal Processing*. 2009.
18. Chong, H.Y., S.J. Gortler, and T. Zickler, *A perception-based color space for illumination-invariant image processing.* Acm Transactions on Graphics, 2008. **27**(3).
19. Wang, M., S. Yuan, and J. Pan. *Building detection in high resolution satellite urban image using segmentation, corner detection combined with adaptive windowed Hough Transform*. in *2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS*. 2013.
20. Wang, Q.C., et al., *A Hierarchical Connection Graph Algorithm for Gable-Roof Detection in Aerial Image.* Ieee Geoscience and Remote Sensing Letters, 2011. **8**(1): p. 177-181.
21. Chaudhuri, D., et al., *Automatic Building Detection From High-Resolution Satellite Images Based on Morphology and Internal Gray Variance.* Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2016. **9**(5): p. 1767-1779.
22. Ghaffarian, S. and S. Ghaffarian, *Automatic building detection based on Purposive FastICA (PFICA) algorithm using monocular high resolution Google Earth images.* Isprs Journal of Photogrammetry and Remote Sensing, 2014. **97**: p. 152-159.
23. Chandra, N., J.K. Ghosh, and A. Sharma. *A cognitive based approach for building detection from high resolution satellite images*. in *2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Spring)*. 2016.
24. Senaras, C. and F.T.Y. Vural, *A Self-Supervised Decision Fusion Framework for Building Detection.* Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2016. **9**(5): p. 1780-1791.
25. Yang, G., et al. *Building Extraction in Towns and Villages Based on Digital Aerial Image by Texture Enhancing*. in *The Proceedings of 2010 18th International Conference on Geoinformatics*. 2010. IEEE.
26. Saeedi, P. and H. Zwick. *Automatic Building Detection in Aerial and Satellite Images*. in *The Proceedings of 10th International Conference on Control, Automation, Robotics and Vision (ICARCV 2008)*. 2008. Hanoi: IEEE.
27. Awrangjeb, M., C. Zhang, and C.S. Fraser, *AUTOMATIC RECONSTRUCTION OF BUILDING ROOFS THROUGH EFFECTIVE INTEGRATION OF LIDAR AND MULTISPECTRAL IMAGERY*. ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci., 2012. **I-3**: p. 203-208.

28. Li, Y., et al., *An improved building boundary extraction algorithm based on fusion of optical imagery and LIDAR data.* Optik, 2013. **124**(22): p. 5357-5362.
29. Fernandes, V.J.M. and A.P. Dal Poz, *A Markov-Random-Field Approach for Extracting Straight-Line Segments of Roofs From High-Resolution Aerial Images.* Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2016. **9**(12): p. 5493-5505.
30. Wang, J., et al., *An Efficient Approach for Automatic Rectangular Building Extraction From Very High Resolution Optical Satellite Imagery.* Ieee Geoscience and Remote Sensing Letters, 2015. **12**(3): p. 487-491.
31. Nebiker, S., N. Lack, and M. Deuber, *Building Change Detection from Historical Aerial Photographs Using Dense Image Matching and Object-Based Image Analysis.* Remote Sensing, 2014. **6**(9): p. 8310.
32. Tanchotsrinon, C., S. Phimoltares, and C. Lursinsap. *An autonomic building detection method based on texture analysis, color segmentation, and neural classification*. in *2013 5th International Conference on Knowledge and Smart Technology (KST)*. 2013.
33. Karsli, F., et al., *Automatic building extraction from very high-resolution image and LiDAR data with SVM algorithm.* Arabian Journal of Geosciences, 2016. **9**(14).

APPENDIX

VITA

Name Mr. Chaiyasit Tanchotsrinon

Education

- 2009 to 2011, Master of Science in Computer Science and Information (English Program), Department of Mathematics, Faculty of Science, Chulalongkorn University
- 2004 to 2008, Bachelor of Engineering Program in Survey Engineering and Geographic Information, Department of Survey Engineering and Geographic Information, Faculty of Engineering, Kasetsart University
- 1992 to 2004, Assumption College