ESTIMATING DAMAGED VOLUMES ON THREE-DIMENSIONAL MODELS OF HISTORIC PAGODAS IN BAGAN AFTER EARTHQUAKE

THIDA AUNG

UNIVERSITY OF COMPUTER STUDIES, YANGON

AUGUST, 2020

Estimating Damaged Volumes on Three-Dimensional Models of Historic Pagodas in Bagan After Earthquake

Thida Aung

University of Computer Studies, Yangon

A thesis submitted to the University of Computer Studies, Yangon
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

August, 2020

ACKNOWLEDGEMENTS

I would like to acknowledge the sincere gratitude to all who get this deep and precious research work done.

First and foremost, I especially thank our Head, **Dr. Mie Mie Thet Thwin**, Rector of University of Computer Studies, Yangon (UCSY), for her valuable guideline and supporting to this research.

Moreover, I would like to willingly thank my supervisor, **Dr. Myint Myint Sein**, Professor of University of Computer Studies, Yangon. She has provided not only a helpful guidance but also a lot of inspiration, motivation and encouragement during my research work. She was a strong influence on instilling inspiration and motivation drive to tide me over the arduous tasks of research activities through all the stages. I have learnt many things about conducting research together with her, and her support has been essential for the success of my work. Without her understanding on my situation, I cannot stand here.

In addition, I would like to thank examiner, **Dr. Aung Nway Oo,** Professor, University of Information Technology for making precious comments and detailed corrections to my thesis and those who are pressing power to improve the end result.

My sincere thanks go to Professor **Dr. Khine Moe Nwe**, Course-coordinator of Ph.D 9th Batch. She guides me whatever I need. I would like to express my respectful gratitude to **Dr. Thin Lai Lai Thein**, Professor, University of Computer Studies, Yangon for her overall supporting throughout of my Ph.D course work and doing research. I would also like to thank all my teachers for their mentoring, encouragement and recommending the thesis.

I would like to express my respectful gratitude to **Daw Aye Aye Khine**, Associate Professor and Head of English Department, University of Computer Studies, Yangon for her overall supporting throughout my Ph.D. course work and doing research. She is always ready for helping me with warmly kindness.

During the years in which I have been doing my research, I am thankful to my friends for this motivating encouragement, for the stimulating discussions about research and many other topics, for being there whenever I needed them.

Last but not least, my parents, especially my father from another life, deserve special thanks for their love, encouragement, care and immense support throughout my research days and also they were always willing to give me moral supports for the whole PhD life.

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

…….. ……..

Date Thida Aung

ABSTRACT

Natural disaster is one of the major threats to cultural heritage in the developing countries. Floods, earthquakes, fires, environmental fatigue or similar long-term climatic effects can cause irreversible damage to cultural heritage. It can completely destroy the entire cultural heritage. In Myanmar, a magnitude of 6.8 earthquake struck west of Chauk nearly Bagan from the depth of 84km on 24th August, 2016. This earthquake destroyed mainly the ancient pagodas in Bagan, Myanmar. Bagan is recognized one of the world heritage status in Asia. Nowadays there are many pagodas approximately 1000 pagodas in Bagan. Therefore the affected areas in historic pagodas were needed to perform the damage assessment on cultural heritage.

This research presents an automatic detection of the damaged volume of the ancient pagodas after earthquake had struck in Bagan. This research emphasizes how many percentages of volume is damaged on the features of the 3D point clouds. 3D point clouds is a collection of data point with (x, y, z) position on space that are produced from different types of units such as 3D software, 3D scanner and so on. The 3D point clouds used in this research are got from the pix4D software. Pix4D software produces the 3D point clouds by operating the UAV images. The UAV images are acquired by taking the aerial photos using the hexacopter. The output point clouds of specific pagoda are mixed with other objects such as trees, other building and so on. The unrequired region needs to be eliminated and the region of interest of pagoda is extracted. The extracting of 3D damaged pagodas is mainly contributed in the proposed system. The proposed system gives the accurate noise removing for damaged pagoda. The information of pre-earthquake volume can be gotten from standardize 2D image to compare the post-earthquake volume. The damaged volume of the historic pagoda can easily be detected. The result for estimating top point of the damaged pagoda is compared with 3D Hough Transform of 3D point clouds. Thought the experimental results of the proposed system, it can be confirmed the proposed approach work adequately.

It is hoped that this research will help the valuable cultural heritage of Myanmar to preserve. The effectiveness of the proposed system can be confirmed through the reasonable experimental outcomes of automatic damaged volume of the

3D historic pagoda. The system can be achieved with UAV images with minimum time consuming and cost.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

Three-dimensional (3D) modeling is a high-tech representation of anything in the digital technology and it concerns with all types of man-made and unman-made objects such as building, tree, vegetation, and some manmade feature belonging to urban area. There has been an increasing demand for 3D visualization of urban area in planning, architecture, engineering and geographic information system (GIS). The 3D models have a variety of applications which include telecom networking, risk management, evacuation capacity evaluation, airport hazard analysis, civilian emergency mitigation, natural disaster management, military surveillance, and statistical geographic localization, analysis of the multi storied buildings on the environment, etc. Nowadays 3D modeling is widely used in preserving the valuable cultural heritage. Natural and man-made threats to tangible cultural heritage is often threatened around the world. Therefore, conservation of cultural heritage must become a critical task in the developed and developing region[22].

The damage or collapse of buildings due to earthquakes is an extremely serious threat to people's lives and property, especially in high density urban areas. After an earthquake, rapid damage assessment is essential for emergency response, rescue work and post-disaster reconstruction. In the past few decades, various types of remote sensing data such as aerial or satellite imagery, lidar and SAR have been widely used to identify, detect and assess natural disasters. Therefore, the precise classification of damaged building types using remote sensing images has become a critical issue. The traditional method of detecting damage in buildings focuses on detecting changes in 2D (i.e. only in the appearance of the image), while the 2D information provided by images is often insufficient and inaccurate for detecting damage in buildings. The detection in damaged building is therefore desirable in the 3D scene’s functionalities. The key idea of 3D building damage detection is detection of 3D changes using a 3D point cloud obtained from aerial images using structure-from-motion techniques (SfM). The approach to building damage detection discussed in the proposed system uses not only the height changes of the 3D pagodas, but also the shape of the pagoda. Therefore, this method completely combines 2D and 3D

information from the real world to detect damaged volume to the pagoda. The results obtained evidenced from a field study, show that this method is feasible and effective in detecting damaged volume of pagodas. The proposed method has also been shown to be easy to use and suitable for rapid damage assessment after natural disasters.

1.1 The Three-dimensional Model for Cultural Heritage

Cultural heritage generally refers to sites, movable and irremovable artifacts, practices, knowledge and other things where groups or societies are old and important and therefore often deserve conscious conservation measures. It always consists only of a selection from the whole cultural repertoire, many of which may not be recognized by a similar conscience [26]. The 3D recreation alludes to catching and duplicating the shape and presence of a subjective article or scene given profundity and shading data. This is an expansive research territory inside the vision field including numerous stages and still open issues. The computerized conservation of social legacy is an exceptionally testing use of 3D reproduction. Social legacy articles and locales enormously contrast from one another and an amplified loyalty of the 3D reproduction is a center prerequisite. The writing on this theme has significantly expanded in the previous years, for the most part because of the assortment of situations and the improvement of new profundity detecting gadgets just as procedures ready to manage this issue. In order to build up a total 3D remaking pipeline, it has been exhaustively contemplated systems identified with this subject and isolated the 3D digitization procedure in four noteworthy diagrams: picture obtaining, see enlistment, work coordination and surface age [18].

Point clouds are dense groups of 3D data points defined by X, Y, and Z coordinates that digitally replicate the project for design and construction. They make up a virtualized representation of the returned laser signal from the scanning line path of the physical representation of the external surface of a part. Every point represents a specific point on the surface, so the closer the points and the more of them, the greater the potential accuracy.

Point cloud files support the design process by providing an actual context where reference objects can be recreated and additional models can be inserted. When

a point cloud is associated with a drawing, it can be used as a drawing guideline, change the display, and apply color styles to differentiate between different functions.

A high degree of accuracy and point cloud density enables a greater flexibility in terms of manipulating point components to change the look and geometry of the point cloud for design, production and aerodynamic considerations and testing. The ability of the software is ensured that the point cloud configuration and makeup are always retained for comparison purposes. Therefore, new information can be gleaned and input for more accurate designs[60].

1.2 Damage Assessment of 3D Model of Cultural Heritage

Today, 3D models of cultural heritage have become an important issue for earth culture researchers around the world. Geotechnical engineering plays an important role in the creation of virtual 3D models. The main geodetic techniques include photogrammetry, remote sensing, geographic information systems, global positioning systems, laser surveying and radar surveying. A large number of geomantic researchers are working in this direction. In emergency assessments, a standardized approach to identify, record and assess hazards and threats allows heritage professionals to compare results and make better decisions. Therefore, standardized formats and sets / types of data fields for data collection need to be developed, defined and implemented [71]. Different people with different backgrounds has different ways of recording and describing damage and threats to heritage and monuments. If we use different recording methods, each will have a different size, criteria, and amount of information, and will likely be incompatible. Analyzing and comparing these different data types is complicated and hinders efficient utilize in management and maintenance decisions[69].

One of the main challenges in carrying out a rapid assessment of damage to cultural heritage is the availability of baseline data and it is needed during the systems that track, compile, analyze and visualize several forms of post-emergency data. The occurrence of a large-scale disaster is complicated so that the disaster emergence response is required as an intermediated phase. It is only after removing and sorting fallen structures and broken debris that the cost of stabilization and restoration can be estimated. This means that the process of assessing damage to cultural heritage must

be mainly divided into two phases as shown in figure 1.1. In phase one, an on-site assessment is needed to estimate costs for salvaging, and mitigating risks to cultural heritage. In phase two, it requires a detailed assessment of the condition of the damaged objects and structural elements in order to estimate the cost of complete maintenance and rehabilitation.

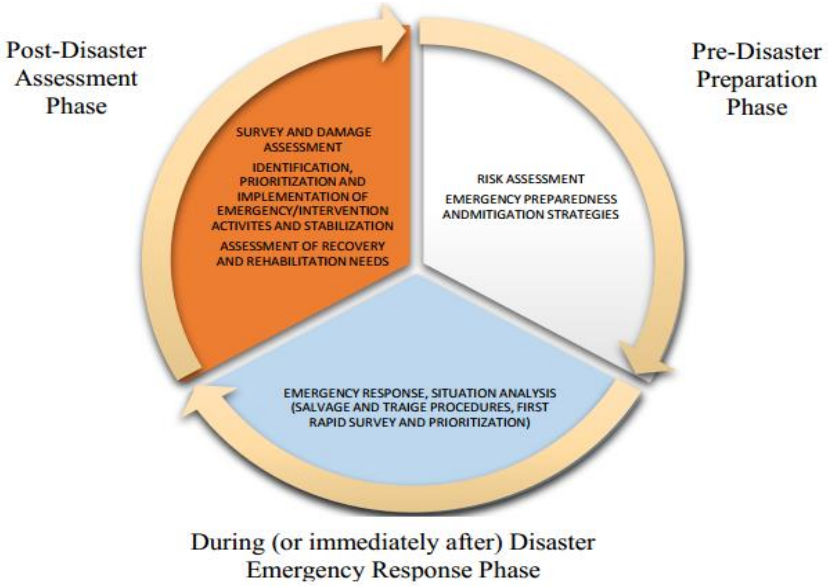


Figure 1.1 Cultural heritage disaster risk management (DRM) cycle

1.3 Fundamental Steps in Automatic Damaged Detection of 3D Model

This proposed system presents an automatic detection of the damaged volume of the historic pagodas when the earthquake had occurred in Bagan at August, 2016. Firstly the proposed system takes approximately 300 aerial photos of the damaged pagoda to generate the 3D point clouds. The point cloud data of the effected pagoda are got by using pix4D software. It is needed to extract the damaged pagoda only by eliminating the trees, buildings and other obstacles around the specific pagoda. After defining ROI of the pagoda, the noise and outlier point clouds are removed and the plane is eliminated. The height measurements of the pagodas are already known and stored in data file and the 2D information of the pagoda is extracted from the 2D image of the pagoda before earthquake. After combing the information of the 2D data and 3D point clouds, the damaged volume of the specific pagoda can be estimated. The proposed system gives not only the rapid damage assessment for the preserving cultural heritage but also the convenient for the user.

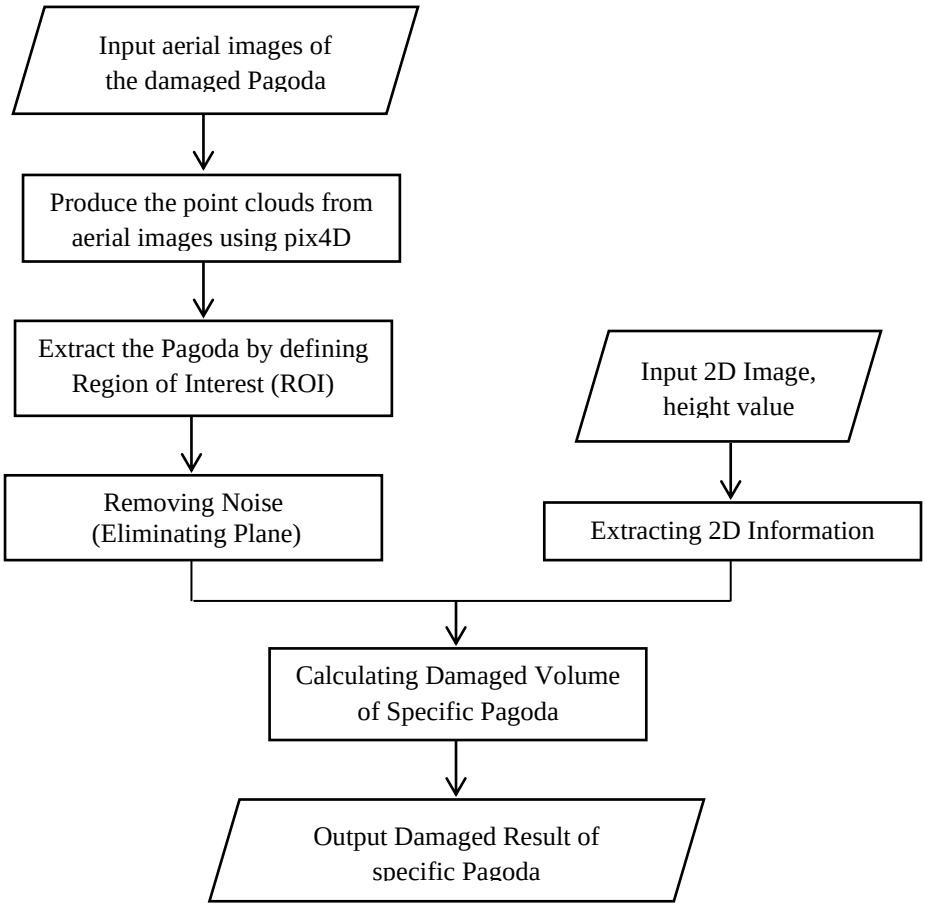


Figure 1.2 Fundamental Steps in Automatic Damaged Volume Detection

1.4 Problem Statements

Preserving the valuable cultural heritage is the important task in Myanmar. Detecting the damaged portion of the historical monuments may cost mainly in many resources such as time, money and effort. Estimating the damaged region of 3D structure from the aerial images is an important problem in many computer vision research areas. The reconstruction of 3D models from the UAV images has become one of the most interesting subjects in both compute vision and computer graphic applications. However, in the past few decades performing the damaged assessment in preserving the cultural heritage were often created manually by users. This process is usually time-consuming and expensive. Therefore, techniques using low-cost data to obtain the damaged assessment of 3D models automatically from the 3D point clouds-based approaches could have significance in 3D application. The information from the 3D historic pagodas with geometric and topologic information are useful for

further interpretation and applications, but the recovering the accurate structure of building in the real world is still an unsolved problem.

In this research, a system has been presented for estimating the damaged volumes on 3D Models of ancient pagodas based mainly on the features of the 3D point clouds' information of the aerial images taken by UAV. The problems addressed in this dissertation include:

- The stages in estimating damaged volume of historic pagodas are required to update all features to the autonomous level.
- It would be interesting task to implement an algorithm for the estimation of all used parameters (e.g., range of one neighbourhood object, outliers threshold in removing noises, polynomial degree, etc.) or make these parameters even adaptive (dynamically changing during the algorithm process).
- The major limitation is the quality of the point cloud though this can be countered by preprocessing step in the proposed system.
- The producing the point clouds using LRF is very expensive and take long time. Transmission through media with refractive indices greater than 1, (e.g. glass) will lead to incorrect depth measurements. The accuracy of an LRF may also be affected by temperature changes.

In this system, the damage estimation of the 3D historic pagodas acquired from aerial images can provide in preserving the cultural heritage with rapid time consuming with high accuracy.

1.5 Motivation

In traditional building damaged detection method, it considers detecting the damage in the affected building after the earthquake. The categorized damage types are trivial damage, severely damage and totally collapse. Therefore, the detection method is focused on the different damaged types using 3D feature of scenes why the real world is a 3D space.

After the earthquake hit, the buildings and pagodas are cracked or damaged. But the whole building or pagoda is not damaged; only the part of building is partially damaged. So it is not quite disappeared from the earth. This fact is pointed out in Figure 1.3. The pagoda is partially destroyed and it is not disappeared in this figure. So the damaged region of the pagoda does not easily calculate from this image. In this case, the traditional 2D Changed Detection technique cannot be used.

Figure 1.3: Aerial Image of Damaged Pagoda

Another motivation is shown in Figure 1.4. In traditional 2D change detection, the damaged region of the building or pagoda is calculated from one side of the image. The damaged region from another side cannot be seen and estimated. Therefore this change detection technique cannot show the accurate and correct result. 3D change detection should be applied for this case.

Figure 1.4: One of the Damaged Pagodas before and after Earthquake

1.6 Objectives of the Thesis

The principal goal of the system is to develop the automatic damage assessment system for 3D historic pagodas after the earthquake using the UAV images. The other objectives of this research are as follows:

(i) To solve the problems of 2D imaging for anatomy overlapping with each other and don't disclose the abnormalities
(ii) To support the works of conservation and maintenance of the these historic pagodas
(iii) To estimate the damaged volume of the historic pagoda accurately
(iv) To produce the quick assessment of the damaged pagodas accurately and correctly
(v) To propose the applicable 3D change detection technique based on the nature of point clouds

1.7 Contributions of the Thesis

There are many developed damaged assessments of 3D buildings after the disasters but damaged assessments for 3D historic Bagan pagodas from aerial images are not developed yet. The main contributions of the research are as follows:

(i) The pagoda's region is only extracted from the 3D complex scene without losing the important data point clouds.
(ii) The efficient noise removing step is implemented using Euclidian distance methods to eliminate the outlier point clouds.
(iii) The whole structure of the pagodas after the earthquake is estimated using the proposed system based on the unlimited number of point clouds.
(iv) The system automatically estimates damage of the 3D pagodas for supporting the rapid damage assessment for the preserving cultural heritage.

1.8 Organization of the Thesis

This thesis is organized with six chapters, including a general introduction, literature review, theory background of proposed 3D modeling method, system implementation, experimental results, and conclusion. The organization of the thesis is as follows:

Chapter 2 presents the related work of the literature review. It deals with related research work in damage detection of 3D pagodas, and many possible approaches to calculate the damaged portion of the 3D buildings are discussed.

Chapter 3 describes the data collection and data preprocessing for the proposed system. Also, the various systems for the producing the 3D model of the pagodas are described.

The 3D geometric methodology and 3D damaged volume method for calculation of the damaged volume of the specific pagoda is introduced in Chapter 4. From this approach, the volume of the damaged pagodas after the earthquake can be easily calculated.

In Chapter 5, the experimental result of the automatic damage detection of the 3D historic pagodas based on the proposed method is discussed and the performance of the system is also mentioned.

Chapter 6 concludes this research paper with the extension work for the future research. A summary of the contribution is discussed in this chapter.

CHAPTER 2

LITERATURE REVIEW

There are many methods and techniques that have been proposed in the area of estimating damaged volume of 3D cultural heritage after earthquake. The previous work in detecting damaged region on 3D object includes several researches that have focused on developing 3D damaged volume detection system partially or completely. These approaches are based on the nature of input data, supervised/unsupervised processing, estimating the damaged result on specific objects. According to different manners, the damaged volume detecting system can be distinguished into automatic methods and interactive methods according to the principle steps that include in the proposed system. In the following, the previous works in this field are reviewed with respect to the proposed system.

2.1 Automatic Damage Detection Methods of Cultural Heritage

In the study of cultural heritage, the characterization and classification of the preservation of the materials that make up a historic building is essential to study its damage. A lot of damage can affect building materials. Recognizing this damage and its evolution over time is an important task in the preservation of cultural heritage. The following reviews are emphasized on the automatic damage detection methods of the cultural heritage.

D. Cerra et al. [14] [15] explained the pioneer experiments of applying fast and robust change detection techniques to sensitive areas of the specific region. A map is created which highlights potentially damaged buildings. This allows professionals to assess in a timely manner the damage caused to the cultural heritage by the images observed. This article proposed to apply a change detection algorithm to solve this problem and to automatically derive a map with the suspected damaged area highlighted. If there is recording errors or geometric distortions due to different recording angles in the image, choose a robust detector to obtain a reliable change map. The points showing the most detailed changes are highlighted, providing a quick set of potential sets of hotspots where more detailed analysis needs to be done. Since destroyed sites are generally displayed as areas where high frequencies dominate, this

method can be improved by integrating the analysis of texture parameters extracted from the image.

D. Abate [1] proposed a methodology for ground monitoring of built heritage using remote sensing techniques and change detection algorithms combined with short range photogrammetry. The final objective is to assess the ability of a simple, practical and instinctive device to generate a corruption map and to identify the areas affected by material and physical changes after a certain time. This methodology helps enthusiastic parents to distinguish, describe and understand the changes occurring on a large scale of manufacturing structures. This framework must test the legitimacy and the unshakable quality of technologies for identifying change for short-term material actions.

F. Bruno et al. [8] [9] [12] introduced 3D change detection and modeling system that based on underwater images. The proposed system detected the state of the archaeological site before and after the cleaning procedure. In this article we have carried out different cleaning experiments using different tools and tools for different types of surfaces (marble, brick, mortar, etc.) affected by different types of biological materials. Describes the process defined for documenting and monitoring the results of colonization. This process includes 1) 3D mapping of the experimental site. 2) Planning of tests and selection of areas to be cleaned. 3) 3D reconstruction of the selected area. 4) Perform the cleaning operation. 5) 3D acquisition of the cleaned area. 6) Comparison of 3D models before and after cleaning the area. 7) Analysis of data on cleaning efficiency.

C. Crespo et al. [20] introduced the use of ground laser scanning intensity data and color information in combination with image classification techniques has been considered to detect damage to granite used in historic buildings. H. Potemkin et al. [55] introduced the Potemkin 3D model, a relatively weak 3D model, but is powerful enough to provide the reconstruction of 3D forms of objects. They said they could use a trained 3D Potemkin model to allow existing detection systems to reconstruct the 3D shape of the objects detected. Chiu et al. explained existing detection methods that can only obtain 2D shapes (or partial 3D information) from detected objects. This was not enough for artificial systems to interact with external objects in 3D space, such as moving a robot arm to grab an object. This is why the reason the 3D shapes of the

objects had been reconstructed using existing detection methods and developed 3D Potemkin models which can be applied to graphic and robotic computer applications. The class model is formed and learned from several parts, labeled as 2D views of instances of the object class from various unadjusted perspectives. No 3D training information is required for the model. Chiu have shown that the robot can estimate the posture of an object using a model and capture the object even when the part captured is not visible in the input image.

2.2 3D Reconstruction of Historical Buildings

Many studies on the reconstruction of buildings using image data have been reported. Researchers looked for ways to reconstruct building models using various sources of image data. Basically, research methods that use image data to reconstruct construction models can be distinguished as using monocular images, stereoscopic images and multiple images [21].

The process of restoring 3D structures from 2D images is the core work of computer vision, and the process of presenting such restored structures is a new challenge in computer graphics. Although there is no general method for deriving models from images, some areas of research have provided results that can be applied to the problems of modeling and rendering architectural scenes. The camera calibration, movement structure, shape of the outline of the silhouette, stereo support and image-based rendering are particularly interesting [6].

Their approach showed that they could reconstruct a 3D model of a scene in about 20 minutes. This time includes the specification of the constraint. In their experiments, model calculations generally took less than a minute to converge on a 1.5 GHz Pentium 4 processor.

2.2.1 Structure from Motion Methods

The 3D reconstruction process begins with a series of image orientations taken from different perspectives, using structures derived from movement techniques. The first phase uses a feature extraction algorithm such as SIFT to extract a set of descriptors through the collection of corresponding images. After that calculate the camera position in the beam adjustment procedure and get the camera calibration

parameters. In the second phase, a high-density passive multi-view stereo concordance algorithm identifies the corresponding points throughout the image sequence. With these known correspondences and the structure of the scene, the 3D world coordinates of each image point can be reconstructed by triangulation. Finding a pair of communication points is not that easy [33]. The scenes are taken from different perspectives, so you can spot false responses due to occlusion, radiant distortion, and perspective distortion. In passive techniques, artificial lighting (such as lamps and projectors) is only used to illuminate the scene and is not used in the triangulation of 3D points. Stereo systems use two digitally synchronized cameras to capture the scene. It is installed on underwater robots [35] or used by divers.

M. Pollefeys et al. [53] [54] aimed to provide a complete system which takes as input a video sequence of static scenes and generates a visual 3D model. Using robust statistics and multiple view relationships, the 3D structure of the observed characteristics and the camera movement and calibration were calculated. In the first step, the characteristics were extracted and followed throughout the sequence. In the second step, stereo matching was used to obtain detailed estimates of the geometry of the observed scene. The resulting model was suitable for both measurement and visualization. The approach presented in this white paper allowed us to capture a realistic virtual model from an image. This approach can also be used to combine virtual objects with real video to generate extended video footage. For this reason, efficient beam fitting techniques have been developed. This beam adjustment was made taking into account the camera calibration to obtain optimal estimates of the metric structure and movement. This approach has worked well with dense depth maps obtained from multiple stereo pairs. An important advantage of the approach compared to the more interactive approach is that it can handle much more complex objects. The downside of their approach was that our approach could not directly capture the photometric properties of the object, but only the combination with the lighting. Therefore, it is not possible to render the 3D model under different lighting. Obtaining a scene representation based on a high quality image required many views of the scene in many directions [60].

Given a 2D projection of a point in the world, its position in 3D space can be anywhere on a radius that extends in a particular direction from the optical center of

the camera. However, if a sufficient number of world projections are observed in several images from different locations, it is possible to mathematically estimate the 3D location of the points and the original location of the camera, up to unknown factors of scale.

Zhang et al. [91] proposed an efficient approach based on the theory of digital photogrammetry by generating coarse 3D city models for purposes. Three sequences of images captured by a digital helicopter video camera, two oblique photographs of the walls of the building, a vertical photograph of the roof of the building, coarse 2D vector data of the building and LIDAR data were used as sources. Using automatic aerial triangulation techniques for high overlapping image sequences, initial values have been provided for the camera parameters for each image. The correspondence between the edges of the building space and its linear features in the sequence of images was determined by a course to fine strategy. The building has been precisely reconstructed by using point hybrid photogrammetry technology. The reconstructed buildings with fine textures superimposed on the DSM and the orthoimages are realistically visualized.

2.2.2 LiDAR and Laser Scanning Methods

LiDAR segmentation is a major problem in the processing of LiDAR data. Several algorithms have been developed to classify LiDAR points. To distinguish ground points from LiDAR point data, morphological filters can be applied assuming the ground points are less than the points of neighboring objects. Another assumption is that the soil is smooth. However, morphological filters are sensitive to noise. A median filter can be used to reduce the effects from a single point of error, the effects of errors cannot be but removed or reduced in the form of point patches. Killian et al. [38] proposed the morphological operator "opening on several levels" was used to maintain the small characteristics while removing the large non-characteristics. Small features can be removed due to the small weight of the window in its method. "Linear prediction" is a method of statistical interpolation. It is used by researchers to generate digital surfaces in the segmentation of LiDAR data.

Vosselman [77] proposed a gradient-based method to exclude non-ground points and modification of the operator of morphological erosion. Sithole [62] modified this method to use different maximum slope thresholds depending on the

local characteristics of the terrain. Several other studies have also been carried out to perform LiDAR segmentation. After the detection of a building area, a 3D building model can be reconstructed from LiDAR points in the detected building area.

Liu et al. [42] reported their work using raw LiDAR data. One method they used was to apply a constant moment and reconstruct a parametric construction model. They concluded that they could derive complex construction models using higher order invariant moments, but these moments were sensitive to noise. In their experiments, the first and second moments were used to derive a gable construction model, including a dormitory on the roof of the building. The building was reconstructed using a general model (polyhedral model). The facets of the construction plan were first detected using a clustering algorithm. To detect the facets of the roof, a 3D Hough transformation was performed on a delauney triangulation mesh generated from LiDAR points on the roof of the building. The density of the LiDAR data used is 5 points / m^2. They assumed that the point distribution was homogeneous to use invariant moments. The non-uniform point distribution introduces a bias into the derived construction model.

Some modern LIDAR systems can capture multi-pulse information, especially the first and last pulses. Alharthy and Bethel [2] reported their work on the first and last data from the pulsed laser scanner. Buildings have little or no reflection at the last pulse, while tree areas are highly reflective due to laser transmission, so use these two pulse reflection data to separate vegetation / trees healthy building. Izadi and Saeedi [34] also reported that other objects, such as cars, were eliminated according to the height and size thresholds. For convenience of calculation, they used the cross correlation between the building area and the model to calculate the main minor directions of the building area. Then the building area was rotated in a horizontal / vertical pose.

LiDAR data have special characteristics and require a specific methodology to process them. For general LiDAR post-processing, Tao and Hu [70] gave an overview of the commonly used algorithms. The point density of LiDAR data used by researchers in the building reconstruction community is very high. In general, studies have been carried out using LiDAR data at a density of around 4 points / m^2 to obtain good construction models. Consequently, the cost of acquiring and processing LiDAR

data is high. Another disadvantage of LIDAR data is their poor morphological quality. Cannot capture sharp line features such as building boundaries. Therefore, if the point density is not high, it is difficult to obtain a high precision construction model only from LIDAR data.

Suresh et al. [67] presented a 3D geospatial visualization of the UCSC campus. They used airborne LiDAR data and digital elevation models. In this article, we will apply photogrammetry technology to build precise 3D navigation and visualization systems for urban environments. 3D terrain height maps are created from digital elevation models. This approach solves the problem of creating a more realistic roof, since the roof contributes significantly to the complexity and visual appeal of the model created.

Zuxun Zhang et al. [95] have provided an efficient approach by generating coarse 3D city models for purposes based on the theory of digital photogrammetry. Three sequences of images acquired by a digital helicopter video camera, two oblique photographs of the walls of the building, a vertical photograph of the roof of the building and coarse 2D and LiDAR vector data of the building are used as sources. It uses automatic aerial triangulation techniques for high overlap image sequences to provide the initial camera parameter values for each image. The correspondence between the edge of the building space and its linear features in the sequence of images is determined by a course to fine strategy. Hybrid point-line photogrammetry technology is used for precise reconstruction of buildings. The reconstructed buildings with fine textures superimposed on the DSM and the ortho images are realistically visualized.

Charalambos Poullis et al. [17] presented a complete and robust system for quickly creating realistic virtual cities from LiDAR data and image sensors. Two important components have been developed for the system. It is a fully automatic technology for extracting 3D models of polygons from LiDAR data and a flexible texture mixing technology for generating photorealistic textures from multiple image data. It is used to segment regions by automatic segmentation and to extract an initial set of roof boundaries which are textualized and refined by a new refinement process based on the classification of the Gaussian mixture model. It applies to linear and

non-linear boundaries. The texture synthesis process is independent of the 3D model and can be applied in different contexts. The result is a standalone 3D texture model.

R. Allegre et al. [3] [4] extended recent work on the dynamic reconstruction of surfaces and presented a method for reconstructing simplified mesh surfaces from large sets of unstructured points. This method consisted of two main components. An efficient selective reconstruction algorithm based on geometric convection which simplifies the set of entry points when reconstructing the surface, and a room which dynamically adjusts or roughens the reconstructed surface according to certain local sampling constraints to update the algorithm. They introduced a new data structure that greatly accelerates the original selective reconstruction algorithm, allowing it to manage point models with millions of sample points. The data structure mixed the KD tree with a Delaunay triangulation of selected points, augmented by a sparse subset of historical sampling points. This design effectively addressed the specific spatial localization problem of the geometric convection algorithm. An off-core implementation of this method is developed. This allowed them to transparently reconstruct simplified mesh surfaces from sets of points that did not fit in the main memory and to update them interactively [27].

Advanced 3D scanning technology allows you to get a set of dense and precise surface sampling points. From a sufficiently dense set of sampling points, a crust algorithm based on the Voronoi diagram and its double Delaunay triangulation can reconstruct a triangular mesh that is topologically valid and converges on the original surface. However, the algorithms are long and are therefore limited in real applications. Strang [63] introduced a non-uniform resampling of the input dataset according to the size of the local entities before the reconstruction based on the fact that dense regions are not necessary in regions without lines for a successful reconstruction. In this way, they accelerated the reconstruction without losing the necessary details.

2.3 Extracting Region of Interest (ROI) from Point cloud Objects

Extracting the Region of Interest (ROI) of the 3D point clouds object in the literature can be grouped on the system’s input to the proposed system. The point clouds of the pagoda that could contain buildings, tree, roads, manmade object, and

other. To extract pagoda only, one requires focusing on ROI regions including these pagodas only while ignoring non-building regions such as trees, roads, cars, and other objects. Most 3D object detection methods extract specified by segmenting the point clouds by defining the region.

S. Xu et al. [82] proposed a robust, precise and efficient method for extracting road edges from 3D mobile LiDAR point clouds has been proposed. The method consists of two steps: 1) extract the candidate sidewalk points based on the proposed new energy function, and 2) shrink the candidate points using the least expensive path model proposed. This system used a representation based on voxels to organize a set of mass points. The volume of each voxel is chosen from the predefined values 0.04 x 0.04 x 0.04 m^3. The intensity of each voxel is used to calculate the sampling density gradient of the point cloud. The algorithm is evaluated under three aspects: robustness, precision and efficiency. To further assess the robustness of this algorithm, this system cannot be extracted in large road environments, including residential areas. This area contains wheelchairs and bicycles, so it may not be possible to extract curbs on the road. There is no information on the sidewalk in these areas. This is a limitation because there is only one large gradient in sampling density in these regions. The candidate points are completely missing. If the missing border area is along a straight road, you can get a full border based on the information nearby. However, if these areas are on intersection roads, borders can be connected using only fixed curves based on prior knowledge.

L. Zhu et al. [94] proposed to identify and evaluate automated and easily applicable GIS procedures for extracting areas such as buildings, vegetation regions, parking areas and streets from LiDAR data using processing tools. The final function has four different steps. Interpolation is firstly used to transfer 3D points to a high-resolution raster. Raster grids with height and intensity are generated. Then, several raster maps (nDSM return difference, gradient and LiDAR intensity map) are combined to produce a multichannel image. Third, the functionality space for this image is created. Finally, a supervised classification of the functionality space is developed. This approach has been demonstrated both in conceptual models and in real, complex studies. In this system, two approaches to classifying LiDAR data have been developed using better image processing methods. The functional classification

based on supervised space seems promising for identifying buildings, trees, sidewalks and grass using several LiDAR products. Two conceptual models have been developed to characterize the space of multidimensional characteristics. One used a standard digital surface model (nDSM) and the other used a derived gradient. This study example was large enough to demonstrate that the image classification approach was a reliable and efficient way to work with software readily available in university research environments.

M. Maghaoumi et al. [45] introduced GemSketch that is an interactive system for extracting 3D geometry from point clouds. Taking advantage of the human interaction modalities in the loop and guided by the image, GemSketch preserved the 3D of generalized cylinders and generalized cubes from point clouds while preserving the scale of the original object and retaining its details. The mesh can be extracted with precision. In addition, GemSketch can perform more robust extractions using multiple views of the same object. GemSketch can process noisy data and extract objects in the event of occlusion or congestion. The system develops a minimal and intuitive user interface that allows calculation methods to overcome limitations caused by occlusion, congestion and other defects. The contribution is an evaluation of this proposed approach for the public RGB-D dataset. However, the proposed system cannot extract more complex objects.

One main problem in LiDAR data has large amount of points, which results in heavy processing time when processing large loads of high point density. X. Hu et al. [32] introduced a simple and fast algorithm with the scan line analysis to automatically extract the build points from LiDAR data. These input data are pre-classified into buildings and other different types with the formal rules. Finally, surface cultivation methods are used to improve the extraction's quality in the preprocessing step. The proposed method produces incorrect results due to incorrect assumptions about the roughness and variegation of the building roof surface.

Tóvári and Pfeifer [73] classified the ground plane extracting methods into surface and into segments filters. Extensive research has shown that all filters work in good performance in a smooth rural landscape, but all fail in complex urban areas and hilly terrain with vegetation region in the specific region. After removing ground

plane, the remaining points are classified in vegetation and buildings by defining DTM and altitude differences from statistical data points.

Haithcoat et al. [28] proposed to get information on the all appearances of point clouds using threshold value to extract tiny objects such as transportation vehicles and trees. Orthogonality is used to simplify the construction footprints. It is started with the standard DSM getting the difference DSM and DTM in order to identify vegetation and construction points. Finally, the footprint of the building is extracted by rectangles aligned with the edges of the border. Frédéricque et al. [25] focused on the return on investment and extracted the buildings' main core. Next, a collection of rectangular rules is generated using the main directions at specific points on the skeleton. Then, an algorithm can repetitively be used to obtain a simplified graph of the rectangle. This provides a representation of the building blocks by a group of rectangles. Another group simultaneously extracts soil, buildings and vegetation [66].

Moussa et al. [48] classified LiDAR data into building, tree and soil segments using rule-based segmentation techniques and refine classification using spectral information obtained from orthorectified CIR images. Zhang et al. [90] used a unautomated classification of airborne LiDAR points with SVM. It first finds flat point cloud surface patches, find combinations of patches to improve classification and detection based on the model. Finally, the limits of the region are drawn. It offered a framework for the buildings' reconstruction and the buildings are gotten by SVM based on geometrical characteristics.

Brenner et al. [11] have shown that the extraction of buildings is resolved in two stages. First, you need to detect the buildings in the data and determine the approximate outline of the building. Second, the return on investment detected requires a geometric reconstruction of the building. Algorithm used to detect buildings because it is based on a robust interpolation. In the construction area detected, the flat roof plates, their boundary polygons and their neighbors are extracted [78]. The grouping of adjacent aircraft has not yet been implemented. The examples presented in this white paper were calculated using LiDAR data from the city of Vienna test site captured by TopoSys.

2.4 Segmenting and Clustering Point Cloud Objects

Semantic segmentation, where pixels are related with semantic tags, is a fundamental research subject in digital image processing. Semantic point cloud segmentation (PCSS) is a 3D structure of semantic segmentation where points normally or irregularly distributed in 3D space are used instead of pixels normally spread in 2D images. Point clouds can be obtained directly from measurable distance sensors or generated from stereo or multi-view images. The recent development of stereo vision procedures and the deployment of various 3D sensors have made the point cloud, basic 3D data easily accessible. Qualified point clouds in image processing connect the virtual world to the real world. More precisely, it generates 2.5D / 3D geometric structures to allow modeling.

B. Yang et al. [84] proposed a new simple and general conceptual framework for segmentation, for example in 3D point clouds. This method, called 3D-BoNet, follows a simple design philosophy of the point-to-point MLP. The frame directly regresses the 3D delimitation frame of each instance in the point cloud while predicting the mask at the point level for each instance. It consists of a backbone network followed by two parallel network branches for 1) the regression of the bounding box and 2) the prediction of the point mask. 3D-BoNet is a one-step, end-to-end training without anchoring. In addition, unlike existing approaches, it does not require any post-processing step such as non-maximum deletion, functionality sampling, and grouping and voting, which makes it extremely efficient in terms of calculation. Extensive experimentation has shown that this approach surpasses existing work on the ScanNet and S3DIS data sets, while increasing the calculation efficiency by a factor of around 10.

Point cloud grouping is an important technique for modeling large point clouds acquired with laser scanners. Point cloud clustering has three clustering approaches: model-based clustering, edge-based clustering, and region-based clustering. In geo-informatics, groupings based on edges and regions are often applied to the modeling of buildings and roads. These approaches use low resolution point cloud data consisting of tens or hundreds of points per square meter, such as aeronautical laser scanning data or data from the on-board mobile mapping system. These approaches also focus on knowledge and geometric limits. Nakagawa et al. [52] introduced to improve region-based point cloud clustering in point cloud integration

modeling. We have also focused on area-based grouping of points to extract polygons from large point clouds, as it is not easy to estimate precise edges from point clouds obtained with laser scanners. For grouping by region, the consensus of random samples (RANSAC) is a good approach for detecting and estimating area. However, to improve the performance of surface estimation from large point clouds, you need to choose a local workspace. In addition, with traditional RANSAC, it is required to consider that a point is inner or outer of the surface region.

H. Kisner et al. [39] introduced a new method and compare the results with these well-known spectral clustering algorithms. The key problems in implementing a spectral clustering method are how to define similarity, how to create a Laplacian graph and how to choose the number of clusters when you have little prior knowledge. The recommended spectral clustering approach is described and evaluated using 3D point clouds. The advantage of this approach is that you do not need any prior knowledge of the number of clusters, even the number of clusters or objects. A new unsupervised learning approach for segmentation of 3D point clouds using spectral grouping with higher eigenvectors in combination with decision trees has been proposed. Learn how to implement a new spectral clustering approach to achieve high quality results. The application of this solution does not require a priori knowledge of the scene and does not require the number of clusters. In some cases, it has been shown that only the threshold of the objective function needs to be adjusted. As a result, spectral grouping techniques are superior to many other grouping algorithms. This approach is very robust for a variety of input data. In addition, the eigenvalues and the eigenvectors are calculated only once and inserted in the segmentation tree, compared to other methods.

C. Teutsch et al. [72] proposed a rapid point cloud grouping method suitable for detecting outliers, segmenting objects, and labeling regions of large multidimensional data sets. The database is a minimal data structure, similar to a KD tree, which can quickly detect connected subsets. The proposed algorithm using this tree structure can be parallelized, further increasing the speed of computation of very large data sets. The given steps are an important part of data preprocessing. They improve the properties of the input data and provide reliable calculations for area measurements, polygon meshes and other visualization techniques. This algorithm can be easily parallelized, further increasing the computational performance of

multicore machines for most applications. Barbu and Luca [7] also provided a content-based indexing method that makes it easier to restore the iris image database.Our indexing model is based on HOG-based image feature extraction which produces high-dimensional feature vectors accessible through SAM. Content-based feature vectors are indexed using the KD tree indexing architecture.

K. Zhang et al. [89] introduce a new K-means algorithm with density constraint. Before treating the point cloud with the K-means algorithm with density constraints, this article defined the density of the point cloud. The density of point clouds can be used to quantify convergence. Finally, the density constrained K-means algorithm is verified by experimental results. Our experiences have shown that an improvement in K-means can notably reduce processing time. Increasing the value of K, the number of clusters, can significantly reduce the computing time of the clustering algorithm. This algorithm has proven to be feasible through experimentation. Euclidean distance based algorithms can drop the algorithm into local optimization, but can be avoided to some extent due to density constraints.

2.5 Denoising 3D Point Clouds

With the rapid development of low-cost sensors such as Kinect cameras and time of flight, it is easy to get a point cloud from a growing community. However, point clouds acquired by these sensors are unavoidable due to noise contamination and include outliers due to sensor limitations, device-specific noise, lighting or reflection properties of surface or scene artifacts. Therefore, to obtain a precise point cloud suitable for further processing, we must perform a denoising operation on the raw point cloud.

J. Zeng et al. [88] proposed small-size manifold model of image patches to a point cloud surface patch, research of self-similar patches and use of the patch manifold in advance to simultaneously eliminate noise. For discrete observations of patches on a collector, a new approach to quantify the similarity between two surface patches of the same size by reconciling the collector dimension calculations defined in the continuous domain with a graph based on Laplacian regularizing patches Propose a measurement discrete patch distance To build noise-resistant graphics.

M. Rakotosaona et al. [57] developed a simple data-driven method to eliminate outliers and reduce noise in unordered point clouds. This approach is based on a deep learning architecture adopted from PCPNet, which has recently been proposed to estimate local 3D shape properties in a point cloud. This method first classified and eliminates the outliers, then estimates a correction vector which projects noisy points on the original clean surface. This approach is efficient and robust for varying amounts of noise and outliers but can handle large point clouds that are densely sampled.

Voxing, a preprocessing step for 3D data is an intensive computing process and causes quantization artifacts. C. Duan et al. [23] proposed an architecture which consumes 3D point clouds directly, without voxelization or transformation for classification and segmentation tasks. This article uses frame ideas and redesigns to estimate normal point vectors in a noisy point cloud and directly eliminate noise in the 3D point cloud via deep learning techniques [10]. It is a new algorithm for denoising a 3D point cloud has been proposed by estimating the point reference plane using a deep learning technique called neural projection algorithm (NPD). The reference plane and the noisy points of the project are estimated directly on the corresponding reference planes, without having to search for points close to each point in the noisy point cloud as in the previous algorithms. The proposed architecture has been validated on real data sets and the proposed architecture surpasses the BF, PDE, GBD and NLD algorithms.

The robot's visual ability plays an important role in behavior and helps to effectively manage the information received. G. Arvanitis [5] presented a real-time method for eliminating outliers and noise in 3D point clouds. These are captured by the robot's optical system with the freedom to use a depth camera. Using this method, the end result of the created 3D object is smoothed, providing an ideal shape for use in other processing techniques such as navigation, object recognition, segmentation. The experiment studies real-world scenarios in which a robot moves while acquiring point clouds in an environment of natural light to reveal unpleasant noise and outliers.

E. Mugner et al. [49] have proposed a method to remove random errors from 3D point clouds. It is based on the estimation of the local geometry descriptor for each point. This method can be applied to any point cloud acquired by a fixed, mobile or airborne LiDAR system. The principle and some results of this method are presented

from various LiDAR systems mounted on UAV. A comparison of low cost and high quality LiDAR systems is performed in the same area and demonstrates the benefits of applying a noise reduction algorithm to LiDAR UAV data. It also demonstrates the effect of noise reduction as a pre-treatment tool for soil classification applications. Finally, some applications of the denoising algorithm are presented in dense point clouds generated by photogrammetry software.

S. M. Haque et al. [30] proposed a point cloud denoising approach that is robust to outliers and can retain 3D functionality on a small scale. Outliers are detected and removed using dissimilarities based on the locations of the points and their corresponding normals. It then uses a robust approach to estimate the locations of surface points in a way designed to preserve crisp and fine 3D characteristics. The effectiveness of the approach is compared to similar methods in the literature by experiments on synthetic and real data, including large-scale 3D reconstructions of heritage monuments

Raw point cloud data generated by the 3D analysis tool contains additional noise from various sources. H. Wenming et al. [79] proposed for taking full advantage of the depth information of unorganized points and the geometric theory of spatial analysis, by applying the method of averaging over the 2D image domain of image denoising theory to 3D point data, the unorganized 3D points A group noise elimination method has been proposed. The noise of point clouds is filtered using an irregular polyhedron based on a limited local neighborhood. This experiment shows that the proposed method successfully eliminates the noise of the point cloud while retaining the characteristics of the point cloud model.

K. Wolff et al. [80] introduced a simple and efficient way to eliminate noise and outliers from these sets of points. The algorithm uses the input image and the corresponding depth map to remove pixels that are geometrically or photometrically incompatible with the colored surface involved by the input [19]. This reduces the smoothing of standard surface reconstruction methods (such as Poisson surface reconstruction) and results in better quality surfaces with more functionality. The algorithm is efficient, easy to implement and robust at varying noise levels.

2.6 Automatic 3D Volume Calculation

A point cloud is a set of point data of a product obtained by a measuring instrument in reverse engineering. With the advantages of being easily accessible, simple and flexible, point clouds have gradually become one of the usual 3D models. Typically, point cloud data contains coordinate information and other topology information, and most 3D models use point cloud data. This allows you to apply 3D modeling to more fields. The calculation of the point cloud volume based on 3D modeling applies to all aspects of society. Many researchers are doing a lot of research to calculate the volume of point clouds.

M. H. Rahman et al. [56] tried to estimate the volume on the basis of a mobile terminal meal impact system, but this paper used this method, but with certain details and results, the accuracy of the calculation and the stability of the volume nature is not high.

Hengwei Xu et al. [31] acquired tree point cloud data as a data source based on a 3D laser scanner and uses a convex hull algorithm with planar dispersion point and an irregular slice segmentation accumulation algorithm to automatically calculate the canopy projection area and canopy volume suggestions for ideas to extract. This method uses a planar convex shell algorithm, where the cross-sectional area is slightly larger than the actual area and increases the calculation error. Z. Yan et al [83] proposed a concave hull method for finding the crown volume of individual trees based on vehicle-vorne LiDAR data. A method was developed to accurately calculate the canopy volume of individual trees from on-board laser scan data (VLS) using a concave shell by the slicing method. Cloud Compare is open source three-dimensional (3D) software for processing point clouds and meshes, used with VLS data to create individual tree segments and tree branches. The thickness and number of slices adjusted to the standard point cloud were adaptively calculated according to the rate of change of the tall zone. Each slice zone is a concave shell algorithm where part of the crown is considered a truncated cone. Next, the entire volume was gotten by adding all sub-volumes.

Jinling et al. [36] acquired 3D point data using the regression algorithm of the Gaussian shape shaping process, use the walking cube algorithm for hexahedral segmentation and object modeling, accumulate small sizes cube and get results. It suggests too complex for undispersed point cloud data.

B. Li et al. [43] enhanced CODSM to calculate the volume of PCO by increasing observations in the opposite direction and, therefore, considering the bidirectional mean Suggested method. This method takes full advantage of the cross shift of random errors and systematic compensation for directional errors, effectively overcomes (or reduces) the effects of random errors and reduces the effects of systematic errors. The cone model and the stone and child base are two typical objects for calculating the volume of PCO using CODSM. The results show that the CODSM has all the advantages inherent in the SM and effectively weakens the volatility of random errors and the direction of systematic errors of the SM. CODSM is a robust configuration upgrade from SM.

Yu Zhi et al. [93] have proposed a method of calculating the volume of a 3D point cloud based on the slice method. First determine the projection plane. Place the projection plane in the projection coordinate system as an XOY plane, establish the projection coordinate system and cut the 3D point cloud along the projection coordinate system in the direction of the Z axis. Then, all the point cloud data for each cutting block is projected onto the XOY plane. Then filter the data points using the scanning algorithm to get the area and volume of the point cloud.

2.7 Summary

This chapter contains five sections for lecture reviews of the proposed system. They are automatic 3D damage detection of cultural heritage, 3D reconstruction of historic building, extracting region of interest, segmenting, and clustering point clouds, denoising 3D point clouds and automatic 3D volume calculation of 3D objects. Each section in this chapter discuss detail reviews comparing state of the arts methods that are important for the proposed system.

Firstly, recognizing the damaged volume after earthquake and its evolution in many years is an important task in the preservation of cultural heritage. The following reviews are emphasized on the automatic damage detection methods of the cultural heritage are been emphasized with the detail explanations. Besides 3D reconstruction of historic monuments are required to detect the damaged region. Therefore, 3D reconstruction is mentioned with comparative discussion using monocular images, stereoscopic images, and multiple images. After this section, the point clouds of the pagoda that could contain buildings, tree, roads, manmade object, and other. It is

required to extract pagoda only, one requires focusing on ROI regions including these pagodas only while ignoring non-building regions. In the section of segmenting, and clustering point clouds, point cloud grouping is an important technique for modeling large point clouds. Point cloud clustering has categorized into three clustering approaches: model-based clustering, edge-based clustering, and region-based clustering. This section explains the comparing results based on three point clouds segmentation methods.

In addition, this chapter described denoising 3D point clouds that is used in the proposed system. 3D point clouds acquired by various sensors are unavoidable due to noise contamination and sensor limitations, device-specific noise, lighting or reflection properties of surface or scene artifacts. Therefore, to obtain a precise point cloud removing outlier is suitable for further processing. The final section of this chapter is automatic 3D volume calculation of 3D objects. Point cloud data contains coordinate information and other topology information. This allows to apply 3D modeling to more fields. The calculation of the point cloud volume based on 3D modeling applies to all aspects of society. A lot of research to calculate the volume of point clouds that are detail mentioned in this section. And next chapter will discuss data collection and point clouds preprocessing methods of the proposed system.

CHAPTER 3

DATA COLLECTION AND DATA PREPROCESSING

Geographic information systems (GIS) are used for location mapping, visualization and analysis. Drones often collect data used by GIS technology. For example, an orthomosaic photo of a drone is modified to avoid distortion. By this way a reliable resource has been gained for creating maps to scale. Mapping tools can create 3D models of maps. This allows data scientists to create predictive models, analyze spatial characteristics, and improve collaboration between internal and field staff. With the help of drones / UAVs, calculating quantities, surveying quantities, storage, gravel pits, landfills or quarries are no longer an expensive task. Significant savings compared to traditional complex measurement methods. The drone captures the data using a camera (photogrammetry) or a LiDAR scanner (laser scanning). Perform very precise 3D surveys in a very short time and at minimal cost. On this basis, precise quantity calculations and quantity readings can be performed and documented. The process is contactless and safe for employees and workers, as there is no need to enter hazardous areas. In addition, UAV theft may occur during operation. This means that you are not interrupted by your daily work. With the rapid deployment and remote access of drones, this technology becomes an important tool for risk assessment and emergency response planning. More recently, the group called for the collection of geographic data to support a humanitarian response plan following the April 2016 earthquake in Ecuador. Emergency response teams can better coordinate rescue and rescue operations by sending drones to disaster areas and collecting geographic data to help assess the situation.

Remote sensing is essential for mangrove research because it saves time and is economical, which can make up for difficult field work, especially in time-consuming locations. Regular monitoring / mapping of mangroves based on remote sensing can provide the true information (and spatiotemporal dynamics) needed for better management. In addition, past and current vegetation maps help reconstruct past events and predict future development scenarios. Historically, satellite and aerial photography have been used to map mangroves. Different types of satellite images acquired with very low to very high resolution (VHR) sensors are relevant for

scientific purposes, such as global mangrove coverage and estimating the local / regional distribution of species. Used. Low-to-medium resolution satellite data (Landsat, Sentinel, etc.) is always available for free, but VHR images are expensive and researchers using them are often limited by budget. In addition, providing cloudless satellite imagery for mangrove mapping is a known difficulty.

Unlike satellite imagery, aerial photography is not affected by clouds. Indeed, the acquisition time (flight of the plane) can be adjusted to local weather conditions. Their spatial resolution varies from sub-meters to meters in centimeters, depending on the flight altitude. Aerial photography is probably the only source of pre-satellite image documentation to identify land use / land cover in the past, and still produces images with optimal spatial resolution for retrospective studies. The importance and usefulness of recently photographed archived aerial photographs for taxonomy at the level of mangrove species is evident in various publications.

In a 3D survey of the area, the actual state of X can be determined to document. On the other hand, the exact delta (comparison of the differences) of the change in volume can be documented by two different flights' time. Complex urban environments include materials of various shapes and surfaces, and buildings can seem indistinguishable from roads, trees and sidewalks. For decades, the extraction of geospatial data for urban areas has been carried out manually on the basis of mono or stereo plots using aerial photographs[44].

As methods evolve, there are more and more ways to collect geographic data using UAV methods. One of these techniques is UAV photogrammetry, which uses a drone equipped with a camera in combination with photogrammetry software to create a 3D model of the ground surface and an orthophoto.

3.1 Data Collection

UAVs have become a popular platform for collecting different types of geospatial data for various mapping, monitoring and modeling applications. Advances in imaging and computer technology allow a wide variety of photogrammetry, computer vision and now end-to-end learning to generate three-dimensional textured (3D) information in the form of digital surface models and field workflows have been introduced for meshes, modified mosaics, CAD models and other models [50]. These

3D products can be used in applications where precision and precision play a key role. Therefore, evaluating and ensuring the performance of a 3D modeling workflow requires extensive testing of the relevant characteristics and reliable and well-founded data. It describes images collected from surface gravel mines by a custom UAV system. This includes additional data that allows the implementation and evaluation of photogrammetric structures or approaches from movements for sparse or dense 3D reconstruction [58].

In this research 6.8 magnitude of earthquake hit at Bagan, the historic city of Myanmar on 24th August 2016. The aim of the proposed system is to analyze the damage area of the ancient pagodas for preserving the culture heritage in ancient city Bagan after the earthquake. It remains over 2200 temples and pagodas still survive to the present day. After hitting the earthquake, about 397 pagodas are damaged and among them, 15 pagodas have huge damaged. The data collection for taking the aerial photos of damaged pagodas was performed by GIS Lab from The University of Computer Studies, Yangon. The UAV system used in the proposed system was a VTOL hexacopter with a calibrated Canon EOS model 550D digital camera to get the high-resolution images for reconstructing effected pagodas. There are about 300 aerial photos for the specific damaged pagodas, and each of which has horizontal and vertical resolutions of 72 dpi. The hexacopter used in taking the aerial photos of Bagan pagodas is illustrated shown in Figure 3.1.

Figure 3.1 Hexacopter Drone

3.1.1 Taking UAV Images

UAVs have gotten extensive consideration in the region of 3D displaying on the grounds that they can be furnished with high-goals cameras that consolidate the upsides of customary ethereal and earthly photogrammetry. The portability and readiness of the UAV to move unreservedly in 3D and it can give numerous high goals pictures with covered region. Modern 3D models are built with cutting edge high-accuracy SfM and MVS pipelines, for example, Pix4D, Bundler and Colmap. These models are of incredible enthusiasm for an assortment of fields, including 3D demonstrating of the city, object review and the utilization of digitized constructing models for documentation of social legacy. The nature of the subsequent 3D model relies upon the prerequisites of the picture-based 3D displaying procedure, for example, the securing of various covering pictures, an adequate gauge between camera sees and the counteraction of impeding light from encompassing hindrances. It emphatically relies upon the flight intend to be followed. For the mapping of practically level and roomy scenes, for example, scenes, flight arranging can be effectively done [51].

A simple grid model or a form of circular flight from the same altitude but can be very complex in dense urban areas composed of different types of artifacts and vegetation. Planning a UAV trajectory in such an area requires considering the surrounding environment and maintaining a safe distance from obstacles, while ensuring that the entire object is captured at close range and under different prospects.

In recent years, general research on planning UAV mapping routes has already focused on automating the generation of optimal flight plans. Automatic flight planning can be classified as a modleless method or a model-based method. The first performs a search tasks in unknown environments by repeatedly updating the model with new measures via selection. These models do not require prior knowledge of the scene. However, it does not generally guarantee complete coverage of the object. Rely on approximate models of hands, scenes and improve models with optimal subsequent flight It is globally optimized. The targets of these research and exploitation approaches are often maximize coverage of target objects or minimize acquisition time or energy [40].

To generate a 3D point cloud, you must automatically detect thousands of key points on each image. If two key points on two different images are detected as being identical, they will be key points which correspond correctly. The computer extracts the characteristics of the object from all of the corresponding key points and generates a 3D model. The corresponding key points can only be detected in the common area (or the overlap area) between different images. The larger the common space, the more key points there are. The more matched keypoints are found, the more increase accuracy of the 3D model [41].

There are two types of overlap. They are the front overlap (front envelope) and the lateral overlap (side envelope).

The Sidelap refers to the percentage of overlap between the different flight legs. An image of the same size (represented by a rectangle) is captured at each location, as shown in the Figure 3.2. Suppose that the blue point indicates location A and the green point indicates B. The shaded area indicates the overlapping area between images A and B. Therefore, if the shaded area represents 80% of the rectangle, the sidelap will be 80%.

The Frontlap refers to the percentage of overlap between one image and the next, where the two images are taken by the drone when it flies in the same direction. If the shaded area of the image represents 80% of the rectangle, the front cover will be 80%.

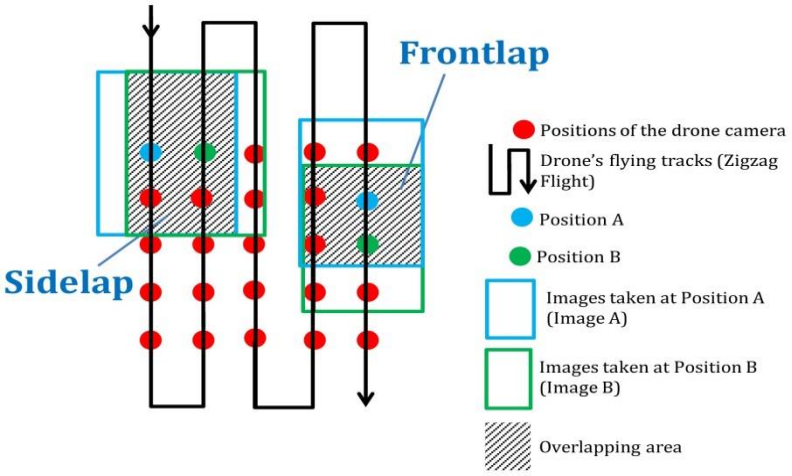


Figure 3.2 Sidelap and Frotlap

Drones without a built-in camera require additional steps before they can be used for photogrammetry (i.e., add and install cameras and gimbal systems). The positive property of this type of system is that it can capture high quality images during each flight test thanks to the additional DSLR camera [46]. Sources of error can occur during the calibration process when taking aerial shots.

3.1.2 Generating 3D Point Clouds

The process of converting aerial images to a 3D point cloud is performed in order to extract objections, detect entities, find corresponding entity points and prepare for other steps. In this research, Pix4D software is used to generate the 3D point clouds. Keypoints detection, keypoints matching, sparse points producing and dense points producing process are in pix4D software.

The structure from motion photogrammetry is often used to define the whole recreation workflow from image sets to denser point clouds of the object. However, SfM refers only to main stages of the workflow, such as point clouds that provide camera parameters. Although some strategies use sparse point clouds as the final product, high-density features matching algorithms such as MVS are often used in the following steps in order to make the point clouds dense. Therefore, the entire process can be named SfM-MVS.

The SfM-MVS process begins by automatically extracting key points (ie, a set of points or pixels with characteristic contrast or texture) from an image. Keypoints are identified in all images and linked (paired) between displayed images. The transformation of SIFT and their variants is the most common algorithm used to identify and match key points in SfM. SIFT generates a digital descriptor for each point in each image. Because the size and orientation of these descriptors are constant, they are useful for identifying points or objects in photos taken from various angles and under various conditions. Then use image geometry and rough reconstruction of the relative positions of these keypoints to check the consistency of the keypoint correspondence.

Given a lot of number of images and keypoints to match, SfM will make global adjustments, including camera poses and parameters, and sparse 3D point

clouds of the scene (the locations of keypoints from different photos that match). Use (i) initialization values obtained from a series of key points in a randomly selected pair and supplementary parameters of camera and pose, and (ii) non-linear adjustments to solve the set adjustment. SfM output is scaled and geo-referenced based on data from ground control points (GCP) and / or cameras or navigation devices on the platform.

Next, the camera poses and parameters obtained from SfM are used to produce large amount of point cloud using MVS algorithm. Before MVS is densified, in order to improve the computational efficiency or feasibility, the images will be grouped according to their position. In this way, the dense point clouds of each cluster (or a group of images) are calculated separately. A dense point cloud with color / spectral information in the input image represents the main output of the SfM-MVS workflow.

3.1.3 Error Estimation of 3D Point Clouds

A 3D point **X** is projected into the images by:

$$m_i = P_i(X) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.1)$$

where P_i contains the internal and external parameters of the camera . The covariance Σ_i of the connection point x_i is linked to the extraction precision. They are displayed in the form of circles (the radius is enlarged for viewing). 2D position of the attachment point x_i and the projected 3D point mid in the image are displayed. The difference between the measured 2D point and the projected 2D point constitutes the reprojection error [64]. This difference is weighted by the 2D covariance Σi and given for an image i as follows:

$$e_i = (m_i - X_i)^T \sum_i^{-1} (m_i - X_i) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.2)$$

A complete estimation equation for the 3D points displayed in the n images is created from the sum of the individual components e of the equation (2).

$$e = \sum_{i=0}^{n} (m_i - X_i)^T \sum_i^{-1} (m_i - X_i) \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (3.3)$$

it is obtained by minimizing e as a function of the 3D point X.

3.2 Data Preprocessing

Data preprocessing has a significant effect on overall system performance, as unreliable samples will likely result in incorrect output. To avoid the effects of noisy and unreliable data, this study applies an effective method of data preprocessing.
Many sensors used in smart vehicles, such as RGBD cameras, LiDARs and stereo cameras, provide driving scene detection information in the form of point clouds. Detection information can be used to process point cloud data to overcome obstacles. So that smart vehicles get fully autonomous driving, detection and tracking, lane edge detection, obstacle path prediction, precise map drawing, real-time positioning and map construction, and other types of intelligent vehicle positioning, and environmental sensing are essential.

In an urban 3D environment, the point cloud obtained by a multiline LiDAR scan is enormous, containing tens of thousands of data points containing a large number of outliers. These factors greatly increase the difficulty of implementing various types of detection and localization algorithms, which has a significant impact on the accuracy and real-time performance of the use of point cloud data. Therefore, there is a need for an appropriate simplification of the point cloud and a preprocessing method to efficiently process such a large amount of data. The purpose of the preprocessing of the point cloud data is to provide point cloud data with weak outliers of noise, a small size of the data and local characteristics important for the segmentation of the ground point cloud, the subsequent processing, the aberrant denoising and the sub-sampling to be done. In this way, the efficiency of the further use of the point cloud data is guaranteed.

In terms of perception of the environment, the results of research on the application of 3D point clouds in pix4D software are particularly important. In 3D environments, the problems of high density, aberrant noise and variance of point clouds are not useful for point cloud data. A method for preprocessing large-scale scattering point clouds has been proposed.

3.2.1 Defining Region of Interest (ROI)

Typically, point clouds are formed by large amounts of data with many redundant coordinates. It is recommended that the point clouds are filtered before improving the information available. This reduces and optimizes data processing and

gives more consistent results after applying these data corrections. Extracting the ROI method returns the points in the region of interest (ROI) in the entry point cloud. Finding the return on investment reduces the processing time of the system. ROI candidate point clouds using the most efficient process often get better deals without sacrificing output quality. In this study, the findPointsInROI method of Matlab 2019 is used to search for points in ROI of whole point clouds region.

index = findPointsInROI (ptCloud, king) returns the points in the region of interest (ROI) in the entry point cloud. The points in the specified ROI are obtained using a search algorithm based on the Kd tree.

A K-D tree is a binary tree where all the leaf nodes are k-dimensional points. All non-leaf nodes can be considered to implicitly generate a division hyperplane which divides space into two parts called half-spaces. The points to the left of this hyperplane are represented by the left subtree of the node, and the points to the right of the hyperplane are represented by the right subtree. The direction of the hyperplane is selected as follows: Each node of the tree is associated with one of the k dimensions and the hyperplane is perpendicular to the axis of this dimension. So, for example, if the "x" axis is selected for a particular division, all points in the subtree that have a value "x" less than the node will appear in the left subtree and a value "x" more large All points with are in the right subtree. In such cases, the hyperplane is defined by the x value of the point, and its normal is in the x axis of the unit.

There are many ways to build a KD tree, because there are many possible ways to select a plane divided along an axis. The standard method of building a k-d tree has the following restrictions:

• Moving down the tree scrolls through the axes used to select the division plane. (For example, in a three-dimensional tree, the root is a plane aligned with x, the children of the root are both aligned with y, the grandchildren of the root are all aligned with z and the great grandchildren of the root are all aligned with X. The grandchildren Rand of the root all have planes aligned on Y if there is a flat plane.

• The points are inserted by selecting the median of the points located in the subtree, relative to the coordinates of the axes used to create the divided plane. (Note the assumption that the complete set of n points is supplied to the algorithm in advance.)

3.2.2 Segmenting 3D Point Clouds

The segmentation method requires partitioning the unorganized point cloud model P into smaller parts so that the overall processing time for P is significantly reduced. A simple approach to clustering data in the Euclidean method can be implemented using a spatial 3D grid subdivision using fixed-width boxes, or more generally an octree data structure [86]. This representation can be built very quickly and is useful if you need a volumetric representation of the occupied space, or if the data in each resulting 3D box can be approximated by different structures. In a more general sense, nearest neighbors are used to implement clustering techniques like flood-filling algorithms [81].

Let's assume a point cloud has been given with a table and objects on top of it. The individual object point clusters lying on the plane. Assuming that a Kd-tree structure is used for finding the nearest neighbors, the algorithmic steps for that would be:

1. *creat a Kd-tree representation for the input point cloud dataset* $\boldsymbol{P}$*;*
2. *set up an empty list of clusters* $\boldsymbol{C}$*, and a queue of the points that need to be checked* $\boldsymbol{Q}$*;*
3. *then for every point* $\boldsymbol{P_i} \in \boldsymbol{P}$*, perform the following steps;*
   - *Add Pi to the current queue* $\boldsymbol{Q}$*;*
   - *For every point* $\boldsymbol{P_i} \in \boldsymbol{Q}$ *do:*
     - *Search for the set* P_k^i *of point neighbors of* P_i *in asphere with radius* $\boldsymbol{r < d_{th}}$
     - *For every neighbor* $\boldsymbol{P_k^i} \in \boldsymbol{P_i^k}$*, check if the point has already been processed, and if not add it to* $\boldsymbol{Q}$*;*
   - *When the list of all point in* $\boldsymbol{Q}$ *has been processed, add Q to the list of cluster* $\boldsymbol{C}$*, and reset* $\boldsymbol{Q}$ *to an empty list*
4. *the algorithm terminates when all points* $\boldsymbol{P_i} \in \boldsymbol{P}$ *have been processed and are now part of the list of point cluster* $\boldsymbol{C}$

3.2.3 Noise Filtering of 3D Point Clouds

Point clouds acquired by RGB-D cameras are necessarily affected by outliers. Not belonging to the outer surface of the object, this is because of different viewing angles, to light resistance, reflection properties of the surface of the object, limits of the sensor. 3D point clouds are simple, flexible and powerful A new primitive

representation of the object, many areas of research such as backward engineering, 3D printing, archeology, virtual reality, medicine and other fields. Point clouds only need to store 3D coordinates, so there is no need to preserve the connectivity of polygonal meshes or the consistency of topologies such as triangular meshes. As a result, point cloud operations improve performance and reduce overhead [29].

However, due to the different viewing angles, the light intensities, the reflection characteristics of the surfaces of objects and the limitations of the sensors, the point cloud data acquired by these RGB-D cameras are necessarily affected by outliers and do not belong to them. These outliers must be effectively eliminated in a real application. Besides, the further operation of the point cloud, such as measurements and surface reconstruction, will be seriously performed.

The reasons for outliers are general varied. General causes are human or equipment mistakes and common or sudden variations in the system's function. In practice, datasets are generally affected by several types of outliers, and whether or not outlier depends on the application. For 3D modeling of cities from aerial photographs, the elimination of outliers is an important preprocessing step to get cleaner datasets for 3D reconstruction of the pagoda [65]. One of the important aspects to consider when designing outlier detection algorithms is the nature of outliers. Outliers is divided into three categories:

- point outlier: a single form of point that is different significantly from the remaining data points
- collective outlier: groups of instances that differ significantly from the rest of the data points, even if each data instances are normal
- contextual outliers: a single instance which is abnormal only in a specific context (such as a spatial or temporal context)

Outliers are the specific type of points that is derived from the main point cloud data sets. The point clouds are required from 3D reconstruction. They are due to image imperfections (such as lens distortion and sensor noise), corresponding ambiguities, camera calibration fault, and exposure estimation procedures and the map camera depth. There are two types of point outliers namely global and local outliers. Global outliers are instances of unique data that deviate considerably from the dataset. A local outlier is significantly different from other nearby data group. Figure 3.3 shows the concept of this global and local outlier. P_1 and P_2 can be easily

detected as global outliers, as these data instances represent significant distances to the remaining points. From a global point of view, P_3 is defined as a regular data instance because it is in the C_2 cluster. P_3 looks like a local outlier because the distance to cluster C_2 has lager space than that in cluster C_2. That is identical to the distance between P_3 and C_2. Finally, the points in cluster C_3 can be known as a global outlier or a regular small group. The question of whether these micro-clusters should be detected as abnormal depends on the application. Most outlier's detection methods cannot capture global and local outliers. However, suitable methods for detecting global outliers are rarely applicable to the local detection of outliers (see Figure 3.3). In general, detecting local outliers is more difficult than detecting global outliers. Firstly, defining the locality is an important and often unclear task, especially when the data represent clusters of varying densities. Second, if that spatial support contains nearby outliers or different distributions of regular data instances, the statistical properties of the data instances will be greatly affected.

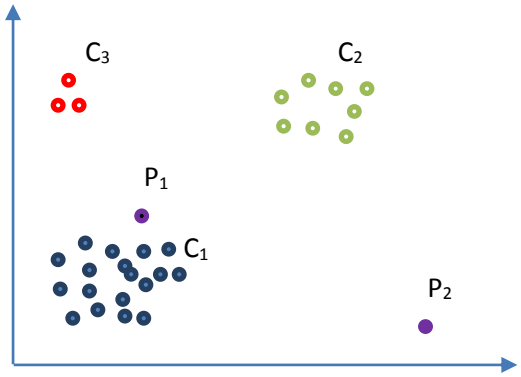


Figure 3.3 Different Types of Point Outliers

3.2.4 Ground Plane Detection

Plan detection is a prerequisite for various tasks. In computer vision, most widely a known method of plan detection is the random sampling consensus algorithm (RANSAC) [11]. This algorithm has been proven to successfully detect planes in 2D and 3D space. It is reliable even with high outliers. The principle of the RANSAC algorithm is to find the best plane to approximate set of 3D points. Select a random number of points (Note: when a plane is approximated, it calculates a polynomial function (enough to select three random points) of the corresponding surface plan. Then find all the points in the original cloud surface obtained according to a given threshold. Repeat these steps N times. Each time the retrieved result is compared with the last recorded result when the new results are better corresponding to many points in

the point cloud), replacement of the results recorded by new results (see Figure 3.4). This principle assumes that the ground plane corresponds to the largest plane of the dataset [68].

RANSAC uses a voting system to find the best results. The data elements in the dataset are used to vote on one or more models. The implementation of this voting system is based on two assumptions: the noisy characteristics do not systematically vote on a single model (a small number of outliers) and are enough to agree with a good model. There are features (fewer missing data). The RANSAC algorithm essentially comprises two steps repeated iteratively.

In the first step, a subset sample containing the minimum number of data elements is randomly selected from the input data set. The fitted model and the corresponding model parameters are calculated using only the elements of this subset sample. The radix of the sample subset is a minimum value enough to determine the parameters of the model.

In the second step, the algorithm checks which elements of the data set correspond to the model instantiated by the estimated parameters of the model obtained in the first step. A data element is considered an outlier if it does not correspond to the adjustment model instantiated by a set of model parameters estimated within an error threshold which defines the maximum deviation because of noise.

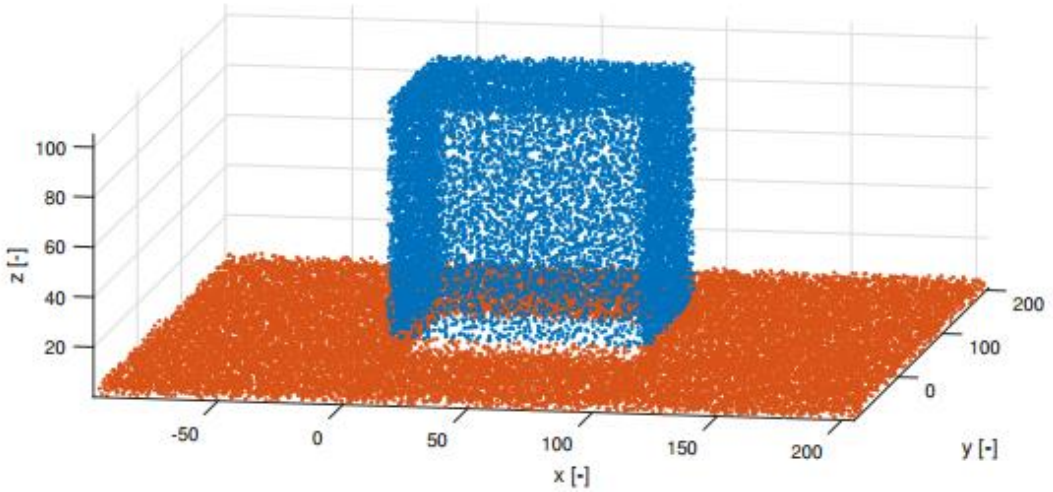


Figure 3.4 Ground Plane Detection

3.3 Summary

In this chapter, there are two main sections to describe data collection and data preprocessing of the proposed system. In data collection section, taking UAV images, generating 3D point clouds and estimation of the errors of 3D point clouds are included. Nowadays, UAVs have become a popular trend for collecting different types of geospatial data for various mapping, monitoring and modeling applications. The UAV system used in the proposed system was a VTOL hexacopter with a calibrated Canon EOS model 550D digital camera in order to get the high-resolution images. That is why there is no affected by cloud like the satellite images and it gives the better resolution images for reconstruction the earthquake affected pagodas. After taking UAV images of the damaged pagodas, the Pix4D software is used to generate the 3D point clouds. The Pix4D software mainly used by SfM_MVS workflow. A dense point cloud with color / spectral information in the input image represents the main output of the SfM-MVS workflow. After that, the next subsection explains the error estimation of the 3D point clouds generated by Pix4D software.

In data preprocessing section, the purpose of the preprocessing of the point cloud data is to provide point cloud data with noises, the important characteristics for the segmentation of the ground point cloud and for the further processing in the proposed system. The proposed system needs the only pagoda's point clouds so that it needs to extract the points in the region of interest (ROI) in the entry point cloud using the ROI methods. After defining the ROI of the pagoda, the segmentation method requires partitioning the unorganized point clouds into smaller parts so that the overall processing time for P is significantly reduced. Besides, point clouds acquired by RGB-D cameras are necessarily affected by outliers. Therefore, the segmentation method are designed depend on the nature of the input 3D point clouds. The final subsection is eliminate the ground plane of the pagoda. In ground plane detection, the RANSAC algorithm is to find the best plane to approximate set of 3D points. The next chapter will explains 3D geometry and 3D volume estimation that is used in proposed system.

CHAPTER 4

3D GEOMETRY AND 3D VOLUME ESTIMATION

A point cloud is a set of data points defined in a specific coordinate system. For example, in a 3D coordinate system, a point cloud can define the shape of a physical, real or created system. Point clouds are used to create 3D modeling in a variety of disciplines and other models used in 3D modeling, such as medical imaging, architecture, 3D printing, manufacturing, games 3D and various virtual reality (VR) applications. This chapter presents the details of the nature of 3D geometry. It also deals with the nature of 3D point clouds in 3D structures.

4.1 The 3D Geometry

The 3D transformation is an extension of the 2D transformation. 2D uses two coordinates, x and y, while 3D uses three coordinates x, y and z. For 3D images and objects, a 3D transformation is required. These are translation, scaling and rotation. These are also called elementary transformations represented using matrices. More complex transformations are managed using 3D matrices. 2D can display 2D objects. Bar graphs, pie charts, graphs, etc. However, 3D is used to represent more natural objects [44].

The selection of these axial systems is optional, but it must be taken with the systems used in modern CG software packages. A main case happens when transforming 3D points onto a 2D plane. 2D planes usually have a left-hand axis system. The 3D system has three axes x, y, z. There are two types of orientation for the 3D coordinate system. Right and left-handed systems. In a right-handed system, the right thumb points in the positive Z direction and the left system thumb points in both negative directions. This becomes apparent when you look at the perspective projection [24]. In this article, the right-hand system is shown in Figure 4.1, and the point P is displayed with the coordinates.

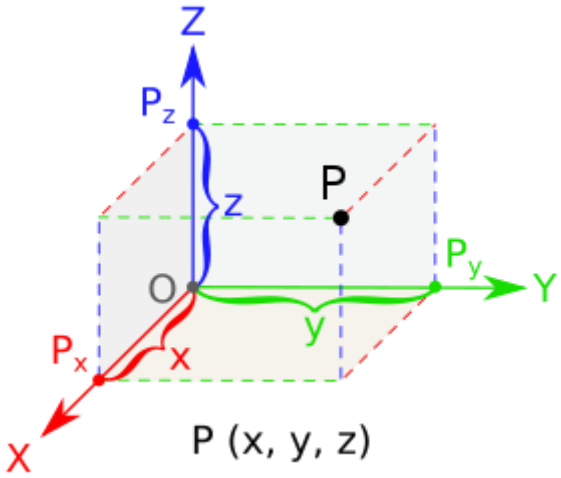


Figure 4.1 A 3D Coordinate System Showing the Coordinates of a Point P

In three-dimensional space, points are represented by positions on the x, y and z plots shown in Figure 4.2. They are each perpendicular to each other. This is like interpreting 2D coordinate geometry where each point is represented by only two coordinates.

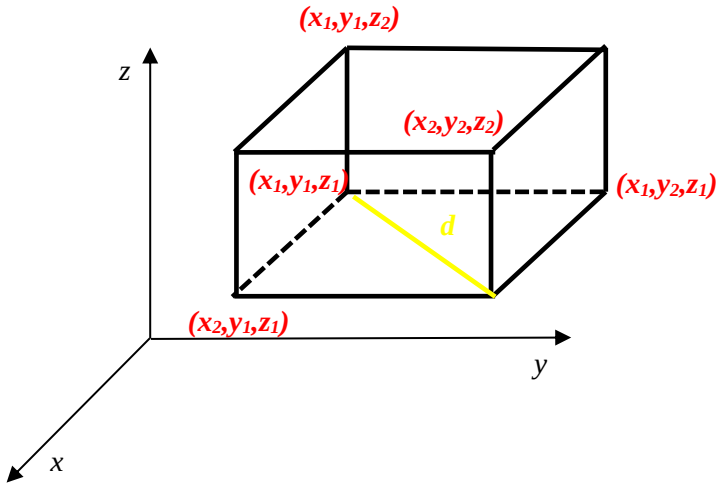


Figure 4.2 Distance between two 3D points

The separation between two points can be gotten by using the Pythagorean Theorem. Distances $\Delta x = x_2-x_1$ and $\Delta y = y_2-y_1$, so the distance d between P_1 and P_2 is shown below:

$$\mathbf{d} = \sqrt{\Delta x^2 + \Delta y^2} \qquad \ldots\ldots\ldots\ldots\ldots\ldots \qquad (4.1)$$

The Cartesian plane x-y gives a mechanism for converting a pair which has linked variables into a graphic form. The variables x and y are used to mention the equation, for example y = 3x+2. Each value of x and y can be placed on the horizontal axis. This set of points forms a familiar line related to the equation m, y = mx+c. The axis of the independent variable x is horizontal, and the dependent variable y is vertical axis from convection. The axes intersect 90 ° at a point called the origin. The point P on the Cartesian plane is represented as (x, y). Mathematical functions and geometric forms can be observed as a list of coordinates in the program [75] [76].

4.2 The Three Dimension Projection

3D projection is mapping mechanism 3D points to a 2D plane. Since most current methods of graphic data display are based on two-dimensional supports in a plane (information on the pixels of several bit planes), 3D projection is mainly useful in computer graphics, in engineering and drawing. The well-known types of projections are:

(1) Orthographic Projection

(2) Perspective Projection

(3) Weak perspective Projection

4.2.1 Orthographic Projection

Orthographic (or orthographic) is a way to represent a 3D object in two dimensions. Orthographic projection is a small set of transformations commonly used to display profiles, details, or accurate measurements of 3D objects. General categories of orthographic projections are plans, sections, bird's-eye views and elevations. The orthographic scale projection provides a linear approximation of the perspective projection. This can be applied to small objects far from the observer and close to the optical axis. It is similar with the other projection lines that are symmetrically projected with the plane and all the planes of the scene are displayed on the display surface in an affine transformation[24].

A simple orthographic projection on the plane z = 0 can be defined by the matrix

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.2)$$

For each point $v=(v_x,v_y,v_z)$, the transformed point would be

$$P_V = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ 0 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.3)$$

It is often more practical to use homogeneous coordinates. The above transformation can be expressed for homogeneous coordinates like

$$P_v = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.4)$$

For each homogeneous vector $v=(v_x,v_y,v_z,1)$, the transformed vector would be

$$P_V = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ 1 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ 0 \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.5)$$

4.2.2 Perspective Projection

When the human eye watches the scene, distant objects appear smaller than distant objects they really exist. This is called perspective. Orthographic projection ignores this effect and allows precise measurements, while perspective projection offers a smaller view of distant objects and provides additional realism. Perspective projection requires a more complex definition than orthographic projection. The conceptual help to understand how this projection works is to imagine a 2D projection as if you were looking through the viewfinder of the camera. The position, orientation and field of view of the camera control the behavior of the

projection transformation. A perspective projection is directed to the image plane along a line from a single point called the center of the projection. This means that if the object is far from the center of the projection, the projection will be small and if it is close, the projection will be large [87].

The simplest perspective projection uses the origin as the center of the projection and z = 1 as the image plane. The simplest perspective projection uses the origin as the center of the projection and z = 1 as the image plane. The functional form of this transformation $x' = x/z; y' = y/z,$ can be expressed in homogeneous coordinates as follows:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ w_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.6)$$

After executing the matrix, the homogeneous component w_c is generally not equal to 1. Therefore, to remap to the real plane, a homogeneous or perspective division is performed by dividing each component by w_c.

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \frac{1}{w_c} \begin{bmatrix} x_c \\ y_c \\ z_c \\ w_c \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.7)$$

4.2.3 Weak Perspective

Weak perspective projection uses the same principles as orthographic projection, but requires to specify a scale factor. Therefore, the closest objects appear larger in the projection, and vice versa. It can be considered as a hybrid of orthographic and perspective projections, described as a perspective projection where the individual point depths Z_i are replaced by constant mean depths Z_{ave}, or simply orthographic and scaling [92].

Therefore, a weak perspective model approximates perspective projection using a simpler model, similar to pure orthographic projection (not scaled). If the depth of

the object along the line of sight is small compared to the distance from the camera and the field of view is small, this is a reasonable approximation. Under these conditions, all the points of the 3D object are at the same Z average distance from the camera and there is no significant error in the projection (compared to the model in full perspective).

4.3 Projective Transformation

All image acquisition systems, including human or machine vision systems, by their nature carry out a certain transformation from real 3D space to local 2D space. Finding the parameters of such a transformation is fundamental to describe an acquisition system.

For most cameras, the models describing the spatial transformations are based on central or parallel perspective projections. Linear parallel projection is the simplest approach. However, what you see with a real camera is only an approximation. Thus, parallel projections are linear, but can only be justified if the object observed is very close to the camera. A better approach to describe the behavior of real optics is obtained in a space of higher dimension of coordinates called homogeneous, by using a transformation of projection in perspective which can be described by linear equations [61].

Geometric transformations play an important role in generating images of 3D objects using these transformations. The position of an object in relation to other objects can be easily expressed. Perspectives can change quickly or objects can move relatively to each other.

4.3.1 Translation

Object movement from one place to another. Perform translation using translation vectors. There are three vectors in 3D, not two. These vectors are in the x, y, and z directions. Use T_x to represent the transformation in the x direction. The transformation is along the y direction and is represented by Ty. Use T_z to represent movement in the z-direction.

If P is a point with coordinates in three directions (x, y, z), after translation, the coordinates will be after translation (x^1 y^1 z^1). T_x T_y T_z is the translation vector in the x, y and z directions shown in Figure 4.3, respectively [37].

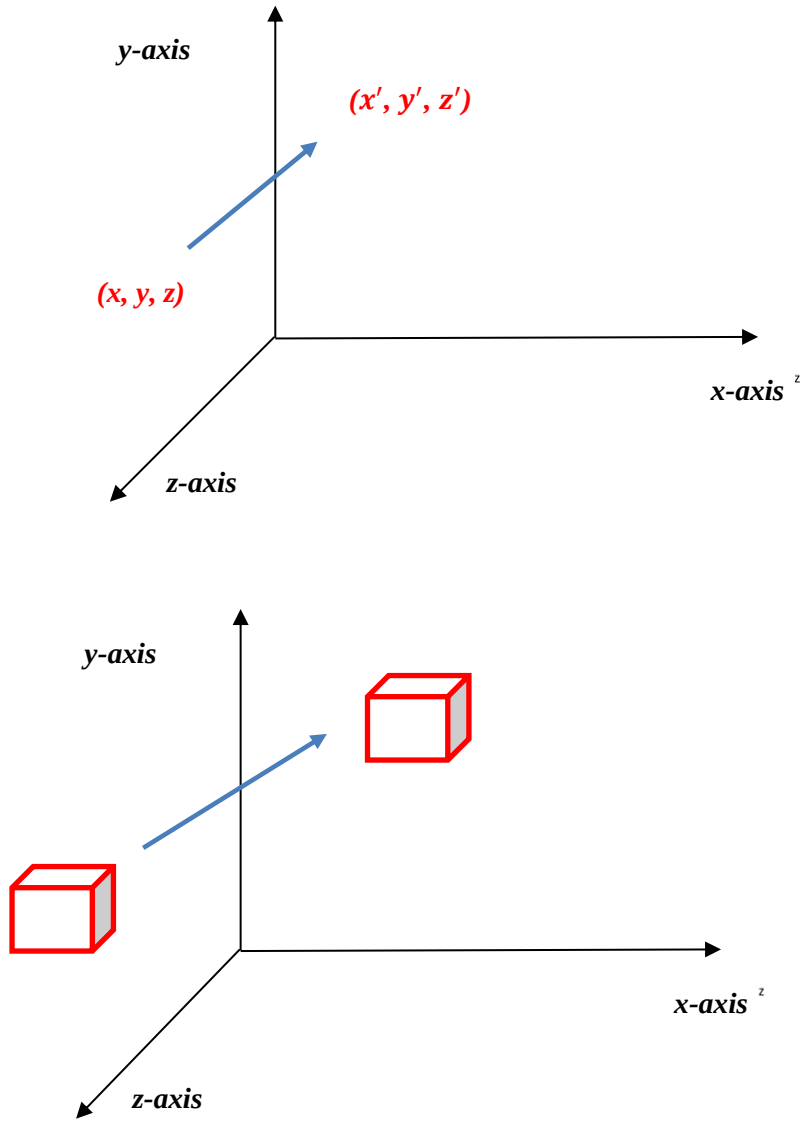


Figure 4.3 Translation

$$x' = x + T_x \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.8)$$

$$y' = y + T_y \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.9)$$

$$z' = z + T_z \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.10)$$

Matrix for translation is shown below:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ T_x & T_y & T_z & 1 \end{bmatrix} or \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.11)$$

Point shown in Figure 4.3 is (x, y, z). It becomes (x^1,y^1,z^1) after translation. T_x T_y T_z are translation vector.

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots \quad (4.12)$$

4.3.2 Scaling

The object can be resized using scaling conversion. During the scaling process, the dimensions of the object are enlarged or compressed. Scaling can be achieved by multiplying the original coordinates of the object by a scale factor to get the desired result. Suppose that the original coordinates are X, Y, Z and the scale factors are (S_x, S_y, S_z) respectively. S_x means scaling factor in x- direction; S_y means scaling factor in y-direction and S_z is scaling factor in z-direction.

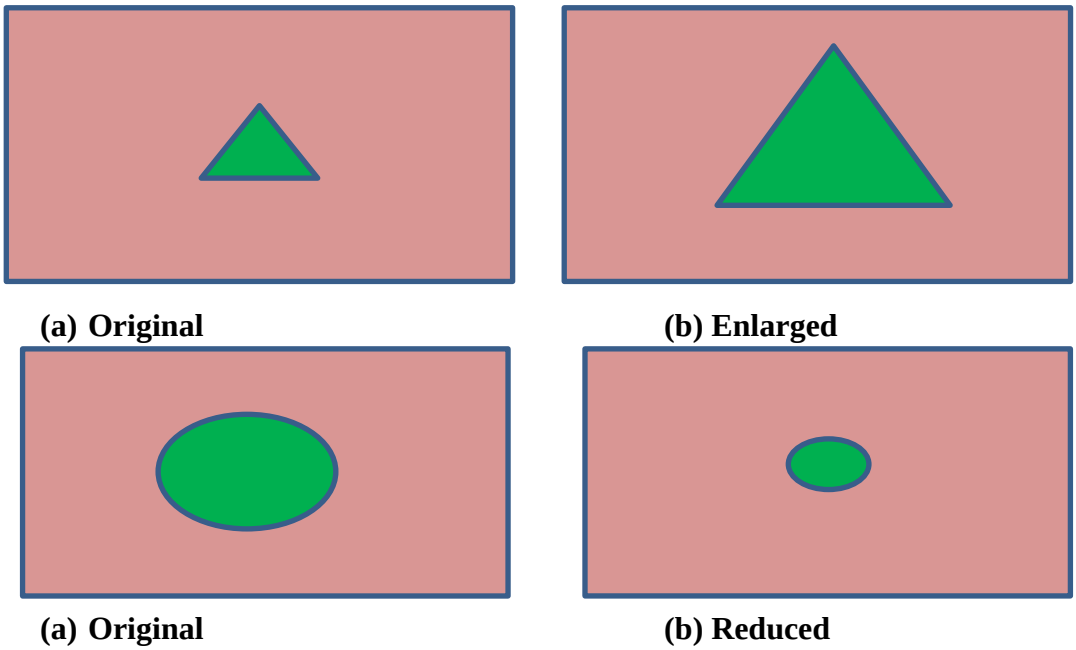


Figure 4.4 Illustration of scaling

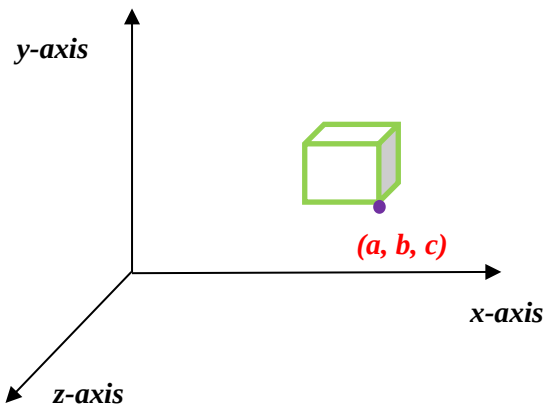
The scaling matix is shown below:

$$\begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (4.13)$$

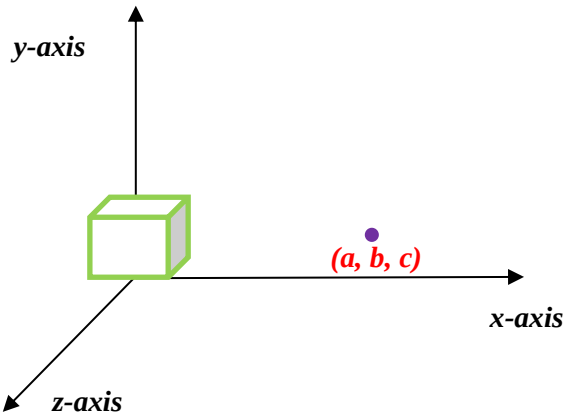
Here are the steps performed when scaling an object at fixed points (a, b, c). It can be expressed as follows:

1. Convert the fixed point to the origin
2. Scale the object relative to the origin
3. Move the object to its original position.

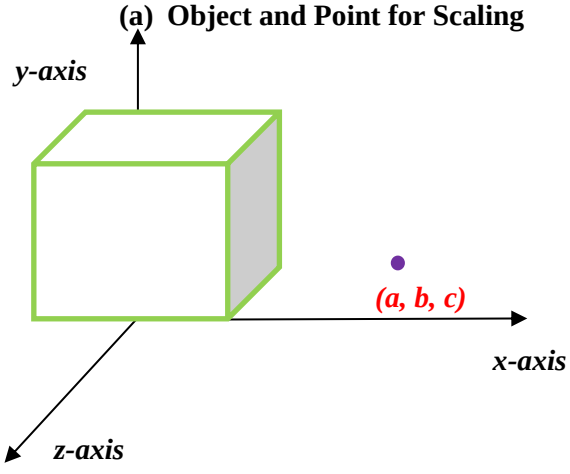
In Figure 4.5, Figure (a) point (a, b, c) is shown, and object whose scaling is done as shown in steps in Figure (b), Figure (c) and Figure (d).



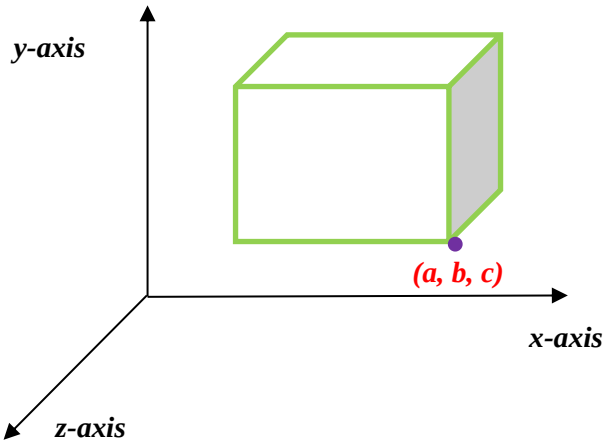
(a) Object and Point for Scaling



(b) Object transfer at Origin



(c) Scaling of Object



(d) Object shift to Original position

Figure 4.5 Steps of Scaling at Fix Point

4.3.3 Rotation

The Three-dimensional rotation is a complex transformation, and the simplest form involves the rotation of a point around three axes shown in Figure 4.6. The rotation is measured clockwise around the axis, looking at the origin along the axis.

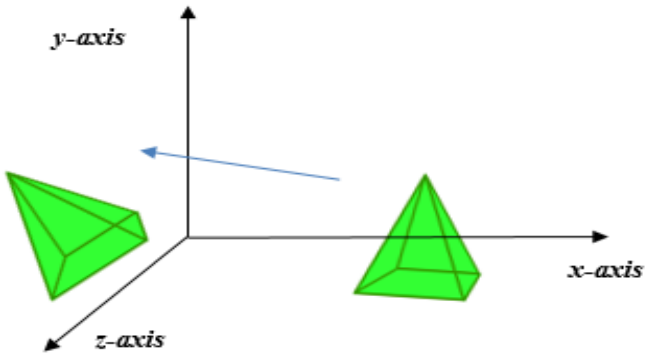


Figure 4.6 Rotation about y-axis

The rotation of a point about the Z axis by an angle θ, *X* axis by an angle α *and Y* axis by an angle β can be defined as follows:

Rotation of a point (X, Y, Z) about the X-axis by an angle α is computed as

$$\begin{bmatrix} X_X \\ Y_X \\ Z_X \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots \quad (4.14)$$

Similarly, rotation about Y-axis by an angle β is computed as

$$\begin{bmatrix} X_Y \\ Y_Y \\ Z_Y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \ldots\ldots\ldots\ldots \quad (4.15)$$

and rotation about Z-axis by an angle γ is computed as

$$\begin{bmatrix} X_Z \\ Y_Z \\ Z_Z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 & 0 \\ -\sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \text{..... (4.16)}$$

Finally, the geometric transformation of the point can be obtained by cascading the transformations defined by by T, S, R_X, R_Y and R_Z, and by multiplying the vector representing the homogeneous coordinates of the point.

4.4 3D Volume Estimation

The input of the algorithm is a Point XYZ type point cloud in pix4D output format. It is the most basic and most used type. Point clouds only contain 3D xyz coordinate information. This format is chosen because it contains the type of information or contact details necessary for processing and calculation [13][16]. In digital geometry, an alpha shape, or α-shape, is a family of simple linear curves by pieces in the Euclidean plane associated with a finite set of point forms. They were originally defined by Edelsbrunner, Kirkpatrick & Seidel (1983). The alpha form associated with a set of points is a generalization of the concept of convex shell. In other words, all the convex shells are alpha forms, but all the alpha forms are not convex shells. This is a very ambiguous concept, there are probably many possible interpretations, and the alpha form is one of them. Alpha shapes can be used to reconstruct shapes from a dense, unorganized set of data points. In fact, the alpha form is bounded by a border, which is a linear approximation of the original form.

Alpha shapes are formed from the boundaries of the alpha complex. The alpha complex itself is a Delaunay triangulation sub-complex for a particular set of points. For a particular set of points in space, you can define a family of alpha shapes ranging from very coarse points (convex shell) to very fine points. The "alpha" parameter specifies the level of refinement. The higher the alpha, the coarser the adjustment and the finer the alpha. The level of sophistication required to match the volume of an alpha form with the volume of the original dataset to which it adapts can be taken as a

measure of the complexity of the form. The more complex objects are volumes that converge.

To understand intuitively the alpha form, imagine that the volume is composed of a large quantity of ice cream sprinkled with "pieces" of solid chocolate (point clouds). Use an ice cream scoop to remove all areas of the ice cream so that the spoon can be reached without touching the chocolate pieces. This includes the area inside the block and effectively isolates the cavities from the volume. Ultimately, this results in a block of ice cream with solid "shards" surrounded by both convex and concave surfaces. If the solid "pieces" are joined by straight lines rather than curves, this forms the "alpha form" characteristic of a particular set of "pieces" of chocolate and scoops of ice cream. In this example, "Alpha" is the radius of the scoop. Larger spoons result in the removal of less ice cream and the "rough" alpha form.

The adjustment resulting from the alpha form can include one volume or several smaller volumes. Therefore, as the alpha decreases, the accuracy of the approximation changes from a convex shell to a finer approximation as more regions are suppressed by smaller spheres. Finally, the radius of the sphere is reduced to the point where no point intersects and no alpha shape is created. Convex shells and "coarse" alpha shapes occupy more volume than the underlying object. On the other hand, the very fine alpha-shaped volumes are smaller than the original structure. At certain “optimal” refinement levels, the volume of the alpha form is equal to the volume of the sample, and it is this “optimal” alpha that is based on the complexity metric of the 3D form.

At first, it may seem easy to estimate the volume of a point cloud. The structure and general formula of the 3D cube are illustrated below.

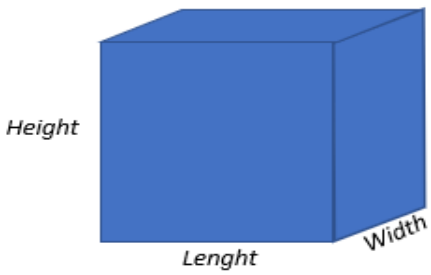


Figure 4.7 Structure of 3D Cube

$$Volume = Length * Height * Depth \quad \ldots\ldots\ldots\ldots \quad (4.17)$$

Then, if the 3D objects have different shapes, they can be filled with cubes or other geometric shapes using a known volume equation, and their sum is the total volume.

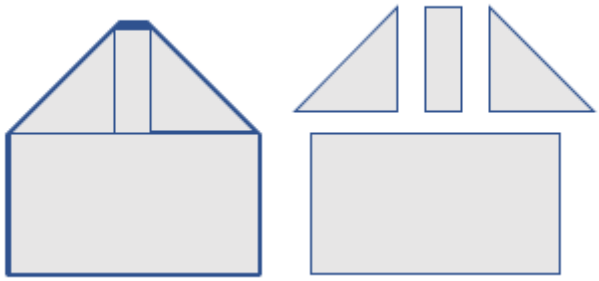


Figure 4.8 Estimating Volume on Complicated 2D Shape by Cutting Area in Pieces

However, in a real 3D example, there is a complex set of 3D shapes and 3D point clouds, which is a real drawback. Find the shape of the point cloud and divide it into several parts so that the volume can be calculated.

There are two approaches to get around a point cloud: convex and concave. Convex means that the shape does not bend inward. Two dots cannot be connected inside a shape by passing outside the shape. Depression is the opposite. It is curved inward, allowing you to draw a point between two points inside that pass through the outside.

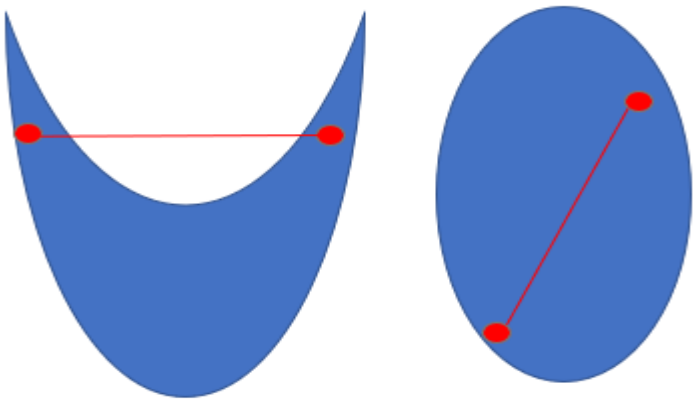


Figure 4.9 Concave shape (left) and Convex Shape (right)

The convex contours are a relatively simple question: take a convex shape around all the points with a minimum volume (or 2D surface). However, estimating the contour of a concave surface is difficult. The downside is that multiple concave contours are possible for the same point cloud, and there is no mathematical way to define the best contour.

Since concave contours allow inward lines, the smallest concave contour shape is likely to find a shape with almost zero area at this point cloud, which is greatly underestimated. On the other hand, the convex contours are a serious overestimation. To overcome the difficulty of concave surfaces, an alpha-shaped property has been developed.

Draw triangles between the points so that there is no overlap between these triangles (this is called Delauney triangulation). This gives you a convex shell (this is the official term for what we called the outline).

All these triangular borders have a fixed length. Because the convex hull is too large, you need to remove some of these triangular boundaries to create a better hull. The exact selection criteria are: Draw a circle as wide as possible around each limit of the triangle. When selecting a border

1. This circle contains no other point
2. The radius of this circle is smaller than the alpha

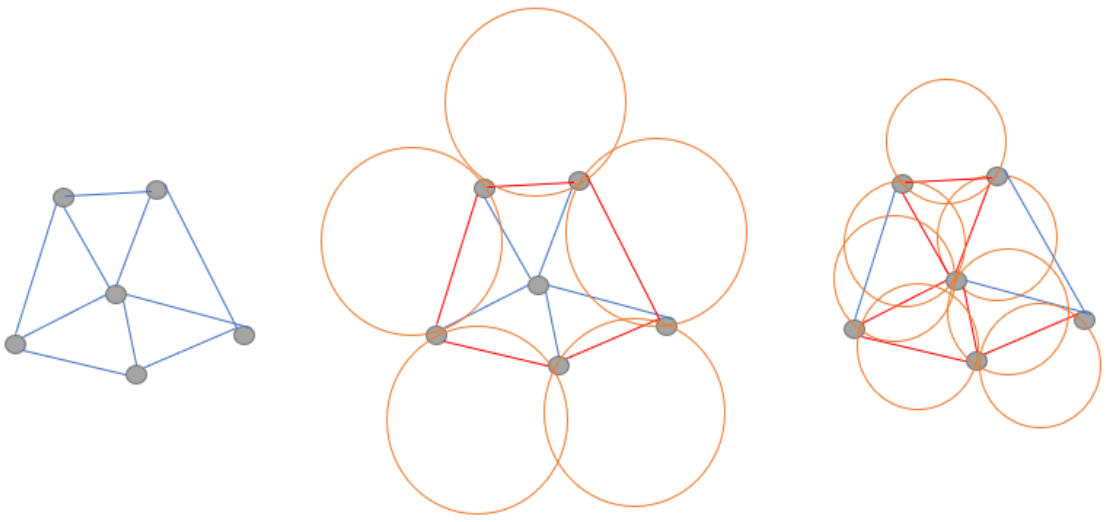


Figure 4.10: Finding Alpha Shape

4.5 Summary

This chapter discuses about 3D geometry and 3D volume estimation that is used in the proposed system. This chapter has four main sections. They are 3D geometry, 3D projection, projective transformation, and 3D volume estimation. This chapter mentions in detail the nature of 3D coordinate system of point clouds. Therefore, most current methods of graphic data of two-dimensional supports in a plane 3D projection is mainly useful in computer graphics, in engineering and drawing. The three basic types of 3D projection are orthographic Projection, perspective Projection and weak perspective Projection. These three types of projection are detail explained in this chapter. Besides, geometric transformations play an important role in generating images of 3D objects using these transformations. The projection transformations of an object are translation, scaling, and rotation are detail discussed in section 4.3. The final section is 3D volume estimation. This section explains how to estimate the volume based on the nature of 3D point clouds produced from Pix4D software. The Next chapter 5 will discuss experimental results of the research after implementation.

CHAPTER 5

DESIGN IMPLEMENTATION AND EXPERIMENTAL RESULTS

This chapter presents how the proposed system works. The experimental results of estimating damaged volume for the 3D historic pagoda from aerial images are implemented with MATLAB R2019a. The experiments were performed on core i7-4790CPU with 3.6GHz, 32G RAM, 16G VGA. The system is processed step by step as shown in Figure 5.1.

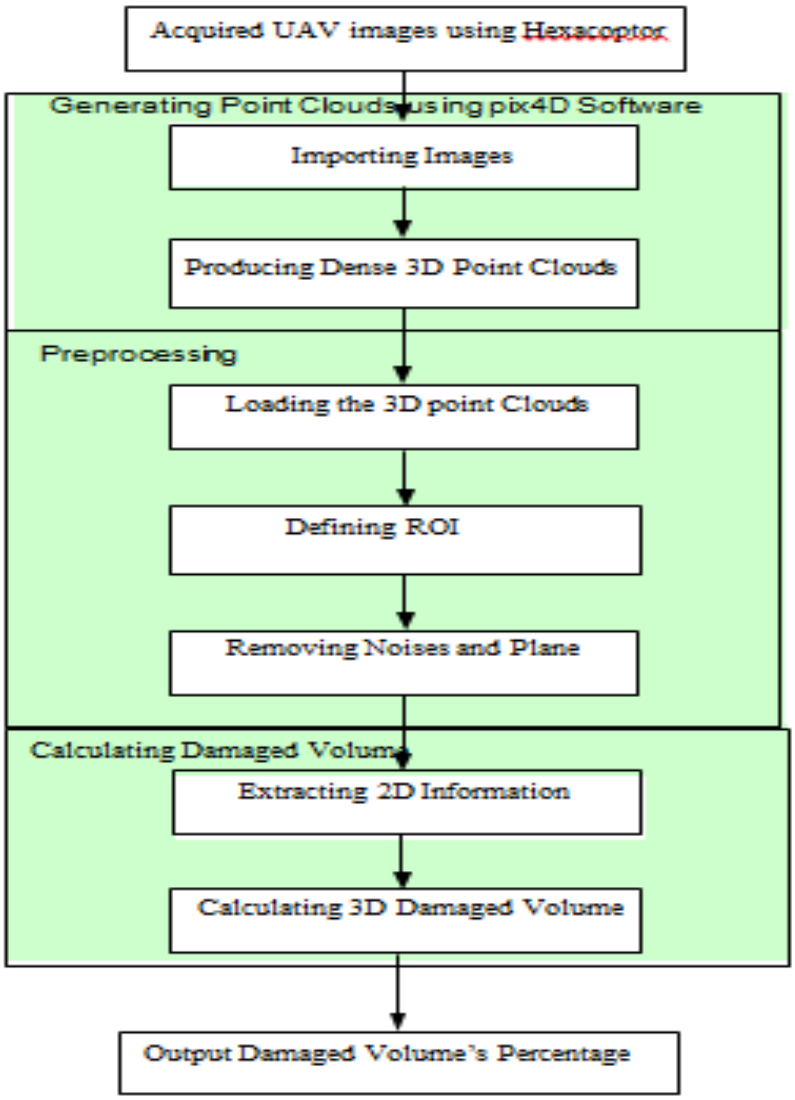


Figure 5.1 Implementation Design of Proposed System

In the proposed 3D modeling system, there are three main parts: generating the point clouds, preprocessing stage containing defining ROI, removing noises and plane; the final stage is calculating 3D damaged volume. The detailed steps of the proposed system are as follow:

Step 1: Taking the Aerial Photos using UAV

Step 2: Eliminating the photos

Step 3: Producing the point clouds of the pagoda using pix4D software

Step 4: Defining the ROI of the specific pagoda

Step 5: Removing the noises and eliminating Plane

Step 6: Extracting 2D Data from 2D Image

Step 7: Calculating Damaged Volume of Specific Pagoda

5.1 Implementation of Proposed System

The experiments of the proposed system are tested on five pagodas which have been done to evaluate the proposed 3D damaged volume detection system. This section explains study area and data acquisition, how the detailed steps of the proposed system are implemented to give the damaged volume results.

5.1.1 Study Area and Data Collection

Bagan, the ancient city, the archeological house of Myanmar, is the study area (see Fig. 5.2) where occurred magnitude of 6.8 earthquake on August 24, 2016. Bagan is in the Mandalay Region of Myanmar. It was established between the 11^{th} and 13^{th} centuries. It has over 10,000 Buddhist temples, pagodas, and monasteries. It remains of over 2200 temples and pagodas still survive to the present day. After hitting the earthquake, about 397 pagodas are affected and among them, 15 pagodas have been damaged severely. The data collection for taking the aerial photos of damaged pagodas was performed by GIS Lab from The University of Computer Studies, Yangon. The UAV system used for the proposed system was a VTOL hexacopter designed and manufactured by SAL Engineering and equipped with a calibrated Canon EOS model 550D digital camera. There are about 300 aerial photos

for the specific damaged pagodas and each of which has horizontal and vertical resolutions of 72dpi.

Figure 5.2 Location of Bagan, Myanmar

5.1.2 Producing Point Clouds using pix4D software

The best software for 3D reconstruction would be Structure-from-Motion (SFM) based methods. The challenges with flying platforms would be vibrations and video stability but techniques exist to deal with these to a great extent. Pix4D software by default would generate relatively sparse point clouds but with surface fitting and strong priors-based reconstruction that can improved the results greatly. Structure from motion (SfM) is a technique that can be used for obtaining topographic data from digital imagery that is widely used in the geographic information system. There are several software options for processing photographs to obtain 3D point clouds containing XYZ coordinates and RGB color index.

5.1.3 Defining the ROI of the specific pagoda

That point cloud datasets contain varying point densities. Therefore, the sparse outliers deduce the system's result even more. Some of the outlier noise can be solved by using statistical analysis on each 3D point cloud's neighborhood and dropping out the unnecessary points cloud. The unnecessary points can be calculated on the computation of the distribution of points in the input dataset. This step performs on the properties of the 3D point clouds and the number of the point clouds of the specific pagoda. The ROI is defined the range of X-plane of inputting point clouds of the specific pagoda in the proposed system.

In the proposed system, the X-plane interval range of the pagoda is mainly input for finding ROI step. The further steps of the proposed system are mainly dependent on the ROI output of the damaged pagodas. If ROI of damaged pagodas is inconsistent, the next step can be processed. The X-plane interval range of 3D point clouds of the damaged pagoda varies from -221.62 to 271.56. The X-plane interval range can be varied on the drone's configuration settings when the aerial images are taken. The X-plane's limits are chosen placing the various thresholding values with many experiments based on the numbers of point clouds and maximum X-limit. Finally, the proposed system distinguishes three ROI regions. The three ROI regions are classified as follows:

Region (1) if (xLimit < 96) roi = [-20, 20; -20, 20; -Inf, Inf];

Region (2) elseif ((xLimit > 96) && (xLimit < 150)) roi = [-35, 35; -35, 35; -Inf, Inf];

Region (3) elseif (xLimit >= 150) roi = [-50, 50; -50, 50; -Inf, Inf];

5.1.4 Removing Noises and Plane

The proposed system removes noises by segmenting the point clouds into clusters with a minimum Euclidean distance between points from different clusters. The proposed system sets the threshold values (maxDistance = 0.2 and distThreshold = 0.17) to maximum distance when many tasking with different threshold values. This step assigns an integer cluster label to each point in the point cloud and returns the labels of all points. This step next fits a plane to a point cloud that has a maximum

allowable distance from an inlier point to the plane. The function returns a geometrical model that describes the plane. The main steps for eliminating plane and outlier noises are shown in follow:

```
[~, ~, outliers] = pcfitplane(pcSelect, maxDistance, referenceVector);

ptCloudWithoutGround = select(pcSelect, outliers, 'OutputSize', 'full');

[labels, ~] = pcsegdist(ptCloudWithoutGround, distThreshold);
```

The MSAC algorithm is a variant of the RANSAC algorithm which is explained in Chapter 3. After this stage, the point clouds are reduced to half of the original point clouds of the pagoda. Table 5.1 shows the comparison of original point clouds and point clouds after removing noises and plane. The image deteriorates due to noise and disturbance. If there are blurs or other distortions in the images, they will be removed. Therefore, the noise filtering process plays an important role in the image preprocessing step. Unnecessary images are eliminated, which reduces the execution time to generate point clouds. Examine some of the standard noise formats and the different ways to remove or reduce the effect on the image.

Tabel 5.1 Reduced Percentage of number of Point Clouds

No.	Name of Pagoda	No. of Original Point Clouds	No. of Point Clouds after extracting ROI	Reduced %
1.	A Naunt Phat Late Pagoda	7 683 845	2 553 410	66.77%
2.	Ta Yoke Pye Pagoda	11528 186	3 153 763	72.64%
3.	Gu Pyaunt Gyi Pagoda	6 662 288	903 107	86.45%
4.	Min Ga Lar Zay Tee Pagoda	13 124 274	5 270 746	59.84%
5.	Su La Ma Ni Pagoda	8 279 211	2 993 983	63.84%
6.	Say Ta Nar Gyi Pagoda	20 244 339	6 241 256	69.17%
7.	Ywar Haung Gyi Pagoda	9 571 758	1 657 174	82.69%

5.1.5 Extracting 2D Information

The proposed system performs extracting 2D information from 2D images after removing the noises and plane of the point clouds. In this step, we need to find the standardize 2D images of the historic Bagan pagodas. Many 2D pagodas are taken in different views and different angles. This kind of images cannot give the accurate pixel information to determine the height and width of the pagodas. Therefore, this step seems easy but takes the long time to find the standardize 2D image of pagodas. The drawing image and 2D image of the pagodas are acquired from the other research book.

5.1.6 Calculating Damaged Volume of Specific Pagoda

The proposed system performs the following steps for calculating damaged volume of the pagoda based on the 2D image's information and 3D point clouds' properties. The boundary function is an implementation of alpha shapes. Using alpha shapes, a set of points can be assigned a polygon by using a set of circles of a specific radius: imagine an arbitrary shape drawn around the points and proceed to remove as much of this shape as possible using circles of a specific radius. It needs to continue as long as possible, without enclosing any points. A small radius will mean more "material" can be removed, a larger radius means less "removal", i.e. a small radius creates a close-cropped shape whereas an infinite radius recreates a convex hull of the set. The points determined to be edge points are then connected with straight edges. This can create hollow areas inside the point set.

MATLAB has an alphashape() function which calculates alphashapes with all possible alpha radii giving different shapes. This is used in the boundary function. boundary() workflow:

(1) Create alphashape

(2) Find critical alpha radius, needed to create a single region for alpha shape

(3) Extract all alphavalues that create unique shapes above this critical value

(4) Use the shrink factor (S) to select a single alpha value to use.

(5) Set the threshold for filling in holes in the alphashape.

(6) Return the indices of the original point cloud corresponding to the vertices of this alphashape.

5.2 Calculating Damaged Volume Using 3D Hough Transform

Hough (1962) implemented the Hough Transform algorithm for the detection of parametrized objects, mainly for lines and circles in digitalized images. Now, 3D Hough Transform focuses on the detection of the lines in 3D point clouds of the 3D historic Bagan Pagodas. The original Hough transform is operated on the 2D images. The extended 3D Hough transform contributes to the three-dimensional point clouds. The point clouds are given to the 3D Hough transform as the input to find all possible straight lines passing through the points. Each point has been voting for all lines in the parameter space [47].

The direction of the line can be calculated from θ for altitude (elevation) and ϕ for horizontal orientation (azimuth) as shown in Figure 5.3 and Equation 5.1. To overcome the aniti-parallel directional vectors for same line let the restrictions of angle ranges for θ and ϕ be $0 \leq \theta \leq \pi/2$ and $-\pi \leq \phi \leq \pi$. For eliminating the redundancy through aniti-parallel vector pairs in the (x,y)-plane ($b_z = 0$) is removed with $b_y \geq 0$ if $b_z = 0$ and $b_x = 1$ if $b_y = b_z = 0$.

$$\vec{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \begin{bmatrix} \cos\phi\cos\theta \\ \sin\phi\cos\theta \\ \sin\theta \end{bmatrix} \quad \cdots\cdots\cdots\cdots\cdots \quad (5.1)$$

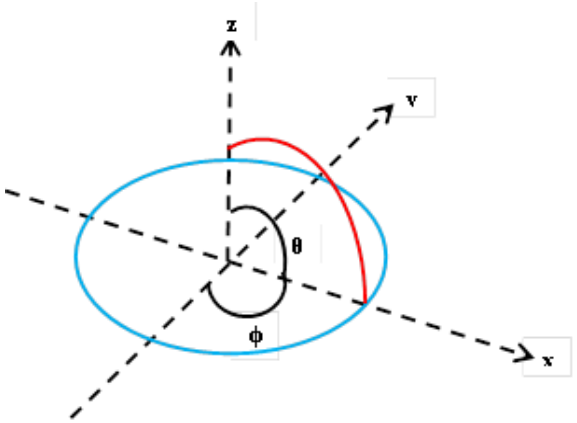


Figure 5.3 ϕ and θ as azimuth (blue) and elevation (red)

Algorithm: Hough Transform

---

input : point cloud $X = \{\vec{x}_1, \dots, \vec{x}_n\}$,

direction vectors $B = \{\vec{b}_1, \dots, \vec{b}_{N_1}\}$,

x'-discretization $X' = \{x'_1, \dots, x'_{N_2}\}$,

y'-discretization $Y' = \{y'_1, \dots, y'_{N_3}\}$

output: voting array A of size N_1 x N_2 x N_3

$A(\vec{b}_i, x'_j, y'_k) \leftarrow 0 \; for \; all \; \vec{b}_i, x'_j, y'_k$

for $\vec{x} \in X$ **do**

for $\vec{b}_i \in B$ **do**

$(\boldsymbol{x}', \boldsymbol{y}') \leftarrow$ computed after Equation (2)

$(x'_j, y'_k) \leftarrow$the nearest neighbor to $(\boldsymbol{x}', \boldsymbol{y}')$ from X' x Y'

$A(\vec{b}_i, x'_j, y'_k) \leftarrow A(\vec{b}_i, x'_j, y'_k) + 1$

end

end

While finding the lines among the point clouds, sometimes some failure can become out. These are

1. There is no enough memory for the 3D point clouds or Hough space accumulator array,
2. the point cloud contains many identical points
3. it takes place overflow in an accumulator array counter.

The results of the 3D Hough Transform is the lines in the vector form with the following information;

npoints = n, a = (a_x, a_y, a_z), b = (b_x, b_y, b_z)

where n is the number of points that are lying on the same straight line, $\vec{a}$ = (a_x, a_y, a_z) is the anchor point that lies inline, and $\vec{b}$ = (b_x, b_y, b_z) is the line direction. The detected lines are returned in decreasing order of the Hough space vote counts as in the following example for Mingalar Zay De Pagoda:

npoints=2970007, a = (-13.796632,-13.791075, -13.785058), b = (0.577281, 0.577351, 0.577418)

npoints=4989, a=(9.489355,4.441291,3.902953), b=(0.590895, 0.558838, 0.581845)

npoints=4422, a=(7.817226, 10.019059, 13.326892), b=(0.566138, 0.603609, 0.561377)

npoints=3781, a=(12.581729, 13.519197, 7.481504), b=(0.564424, 0.573193, 0.594033)

The straight lines through the point cloud of the Mingalar Zay De pagoda is shown in the result figure. The candidate straight lines should be needed to determine the top intersection points by eliminating the noisy lines. Zedi's top intersection vertex can be gotten from finding the point joining the candidate straight lines.

Now consider that the top intersection (x, y, z) can be calculated as follows: Let l_1 and l_2 is the two lines with vector forms;

$$l_1 = Pt_1 + aVec_1, \quad l_2 = Pt_2 + bVec_2$$

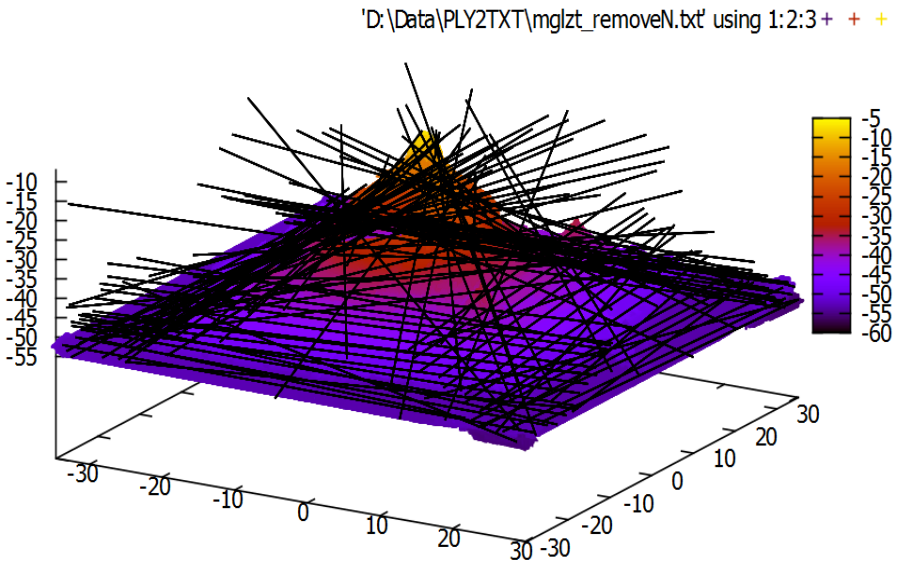


Figure 5.4 Finding Candidate Lines of Mingalar Zay De Pagoda

where Vec_1 and Vec_2 are the director vectors and pp_1 point which belongs to l_1; Pt_2 point which belongs to l_2, respectively.

<=> $Pt_1 + aVec_1 = Pt_2 + bVec_2$ => $aVec_1 = (Pt_2 - Pt_1) + bVec_2 x\ Vec_2$ in the two sides with cross product : $a(Vec_1 \times Vec_2) = (Pt_2 - Pt_1) \times Vec_2$

once "a" can be replaced it in the 1st equation to get the point of intersection (x,y,z). This equation is proof with the following calculation;

Take $Pt_1 = (1,0,0)$ $Vec_1 = (2,3,1)$ and

$Pt_2 = (0,5,5)$ $Vec_2 = (5,1, -3)$

$a(Vec_1 \times Vec_2) = a(-10,-11,-13)$ and

$(Pt_2 - Pt_1) \times Vec_2 = (-1,5,5)x(5,1,-3) = (-20,-22,-26)$

Since the latter is 2(-10,-11,-13), we have a = 2, and the intersection point is

$Pt_1 + 2Vec_1 = (1,0,0) + (4,6,2) = (5,6,2)$

To find b and check the answer;

$Pt_2 + bVec_2 = (0,5,5) + b(5,1,-3)$ which will equal (5,6,2) when b=1.

After estimating top intersection point, width, height, and length of the individual pagoda are estimated from the original reforming shape of the pagoda and also collected from the ground truth data. Some geographic information is available on the historical database of the Ministry of Cultural and Religious Affairs, Myanmar Damaged percentage can be calculated by filling pixels on the reforming shape. The damaged percentage is calculated on the number of filling pixels and the remaining pixels. Equation 5.2 is used to determine the damaged percentage of the historic pagodas.

$$DamagedPercentage = \left(\frac{numberOfDamagedPixels}{numberOfTotalPixels} \right) \cdot 100\% \quad \cdots\cdots\cdots\cdots\cdots\cdots \quad (5.2)$$

5.3 Experiments and Results

The experiments of the proposed system are tested on ten pagodas which have been done to evaluate the proposed 3D damaged volume detection system. The aerial images taken by hexacopter drone are about 300 images. The aerial images are inputted into pix4D software and then the 3D point clouds are generated. The point cloud dataset consisting 10.4 million points. The proposed system estimates inputting the point clouds.

5.3.1 Experiment 1 for Mingalar Zay De Pagoda

In experiment 1, the damaged volume of Mingalar ZayDe Pagoda is calculated. The total 312 aerial images of Mingalar Zay De Pagodas are taken and inputted to pix4D software to generate the point clouds. The generation of the point clouds from 2D aerial images takes the longer processing time about 90 minutes. Figure 5.5 shows the aerial photos of Mingalar Zay De Pagodas after earthquake.

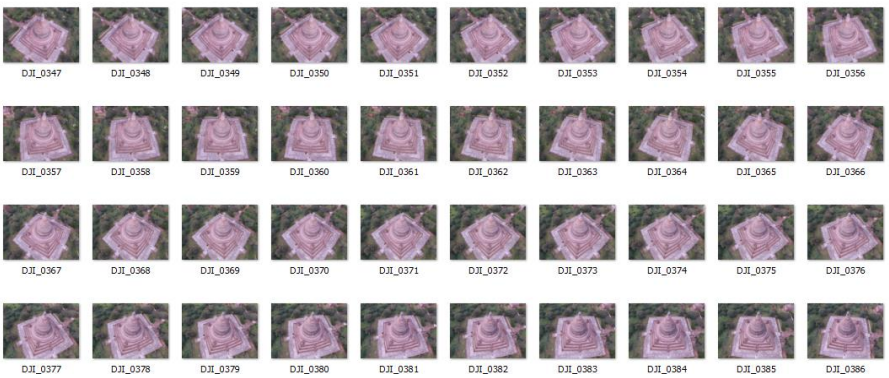


Figure 5.5 Aerial Images of Mingalar Zay De Pagoda

The pix4D software takes the longer processing time but it gives the accuracy of 3D reconstruction than other 3D reconstruction software. The total 13 124 274 point cloulds of Mingalar Ze De Pagoda are prepared to detect the damaged volume. The Figure 5.6 shows the point clouds file (PLY file) of Mingalar Ze De Pagoda.

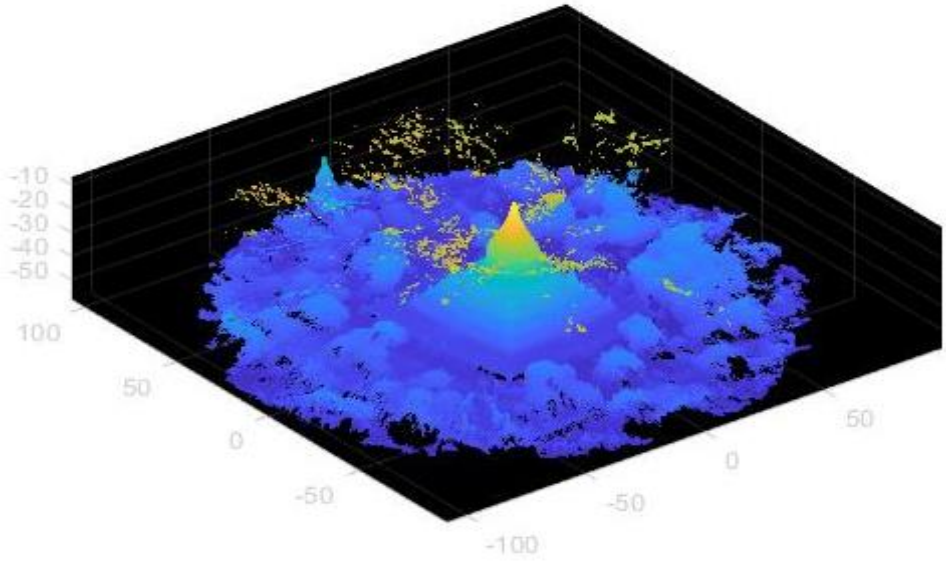


Figure 5.6 Point Clouds of Mingalar Zay De Pagoda

Objects such as roads, cars, trees, and buildings are common to a typical urban aerial image. But, in the proposed system, only pagoda is needed to extract. The small region which may not contain in pagoda are removed by defining ROI (region of interest). Therefore, it defined the region of interest of the pagoda. Figure 5.7 is the point clouds file of Mingalar Zay De Pagoda after defining the respective ROI.

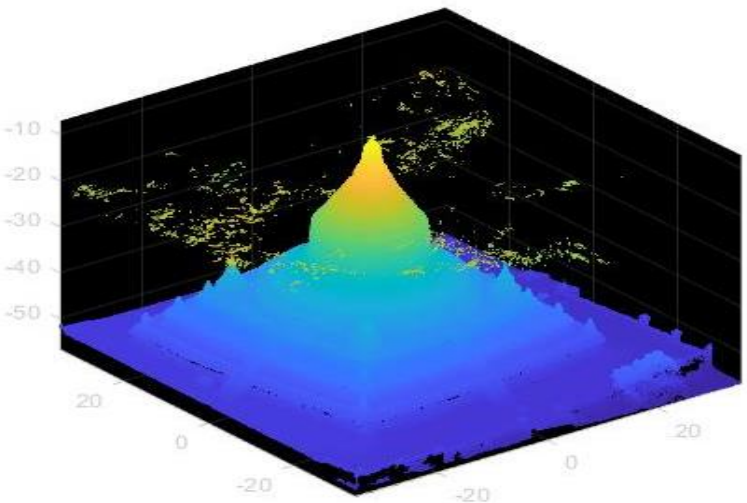


Figure 5.7 Defining ROI for Experiment 1

After extracting the ROI of Mingalar Ze De pagoda, noises are included in the extracted pagoda region. So, the noises point clouds are needed to remove. The plane of the pagoda is not required to calculate the damaged volume. Figure 5.8 is the extracted pagoda only after removing noises and plane.

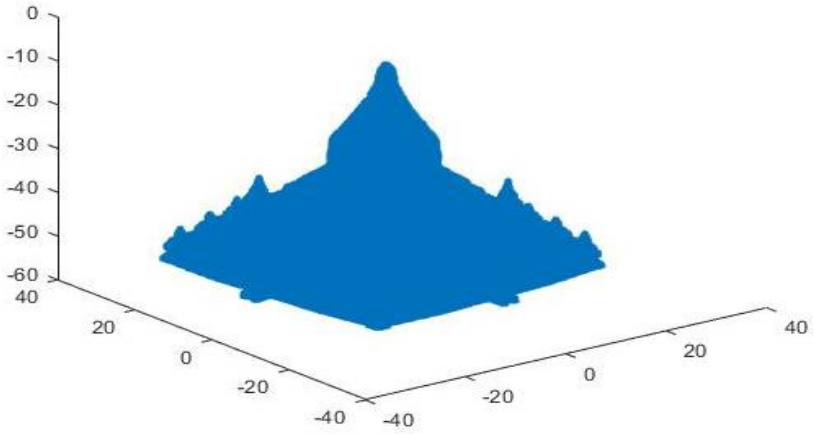


Figure 5.8 Removing Noises and Plane for Experiment 1

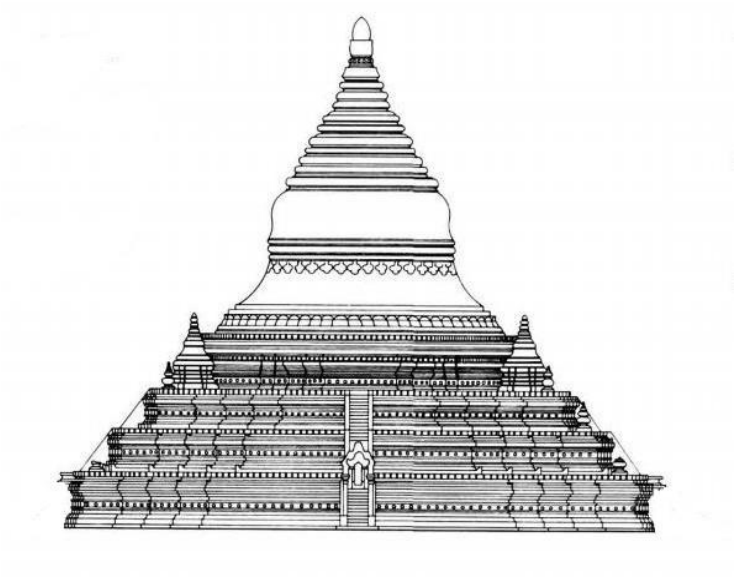


Figure 5.9 Extracting 2D Information for Experiment 1

The proposed system needs 2D information of the pagoda to calculate the damaged volume. The pixel information of height and width can be extracted from 2D image of Minglar Zay De pagoda. Figure 5.9 is 2D image structure of Mingalar Zay De pagoda. After that, the damaged volume is calculated with the detailed steps

mentioned in Section 5.1.6. The comparison of pagoda before and after earthquake is shown in Figure 5.10.

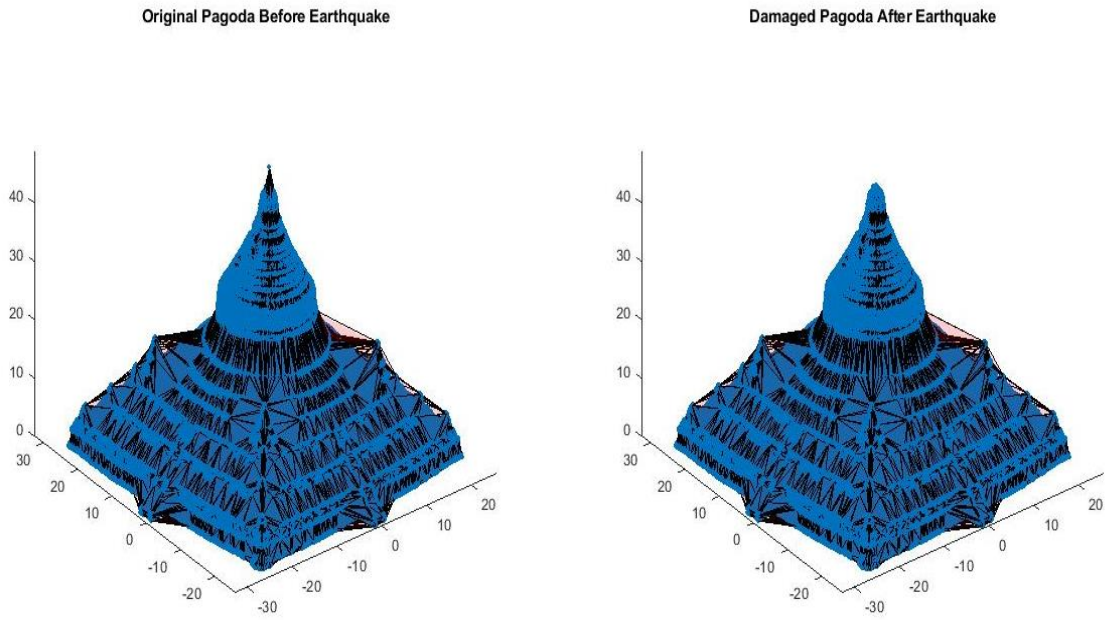


Figure 5.10 Damaged Volume for Experiment 1

The Mingalar Zay De pagoda has no damaged after earthquake. But the proposed system gives the damaged percentage about 2%. The result of estimated damaged volume is shown in following pie chart.

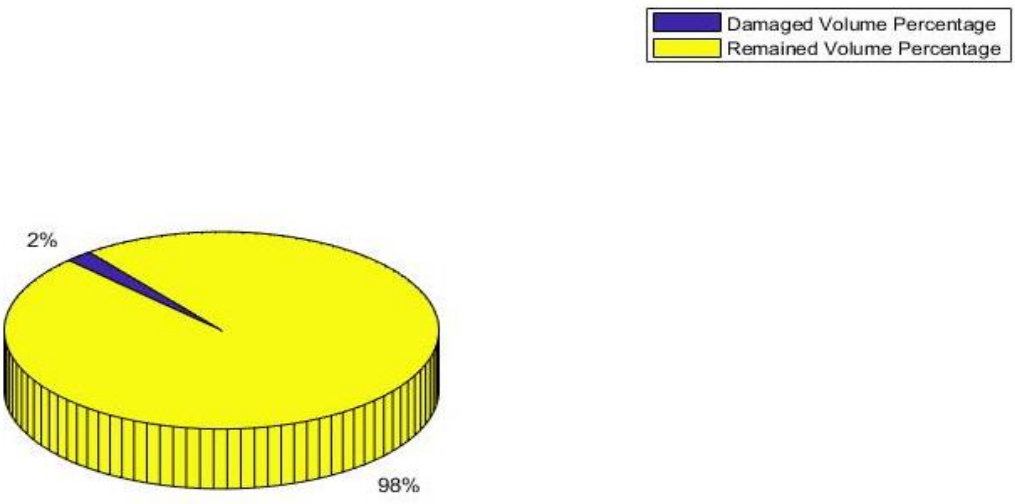


Figure 5.11 Pie Chart of Damaged Volume Percentage for Experiment 1

Figure 5.12 shows the processing time of each step in the proposed system. In each step, the running time of removing noises and plane takes longer time because

clustering the point clouds and eliminating the plane operates on each 3D point of the pagoda.

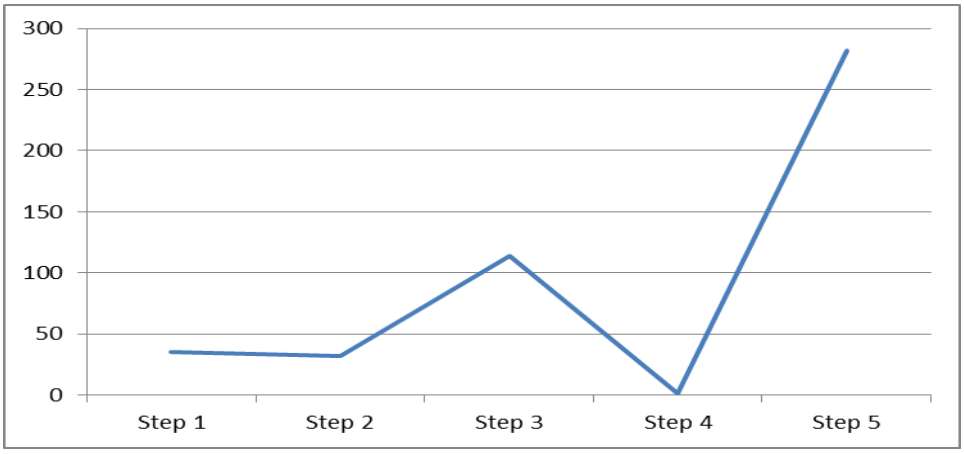


Figure 5.12 Computation Time for each Steps of Experiment 1

5.3.2 Experiment 2 for Law Ka Htike Pan Pagoda

In experiment 2, the damaged volume of Law Ka Htike Pan Pagoda is calculated. The total 180 aerial images of Law Ka Htike Pagodas are taken and inputted to pix4D software to generate the point clouds. The generation of the point clouds from 2D aerial images takes the longer processing time about 50 minutes. Figure 5.13 shows the aerial photos of Law Ka Htike Pagodas after earthquake.

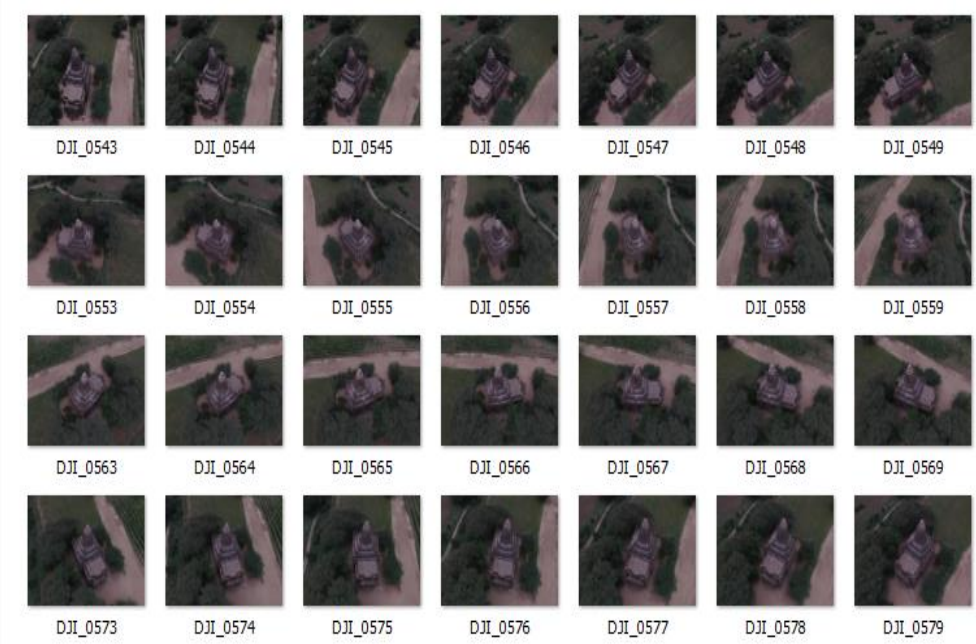


Figure 5.13 Aerial Images of Law Ka Htike Pan Pagoda

The pix4D software takes the longer processing time but it gives the accuracy of 3D reconstruction than other 3D reconstruction software. The total 8726285 point cloulds of Law Ka Htike Pagoda are prepared to detect the damaged volume. The Figure 5.14 shows the point clouds file (PLY file) of Law Ka Htike Pagoda.

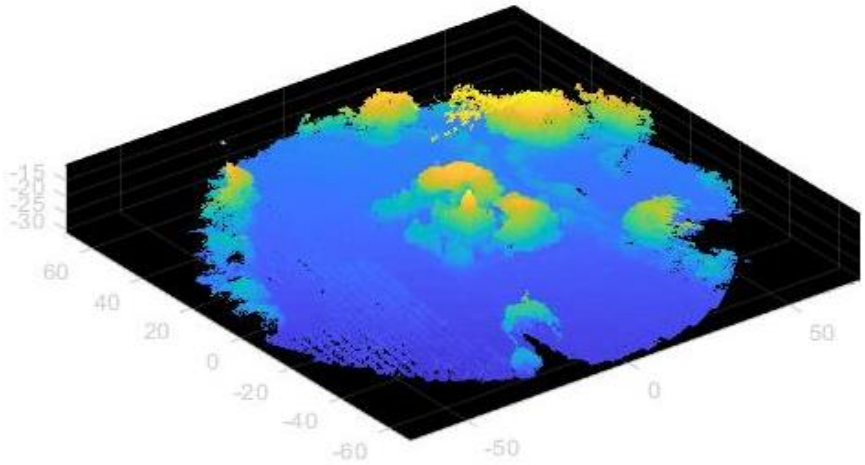


Figure 5.14 Point Clouds of Law Ka Htike Pan Pagoda

Objects such as roads, cars, trees, and buildings are common to a typical urban aerial image. But, in the proposed system, only pagoda is needed to extract. The small region which may not contain in pagoda are removed by defining ROI (region of interest). Therefore it defined the region of interest of the pagoda. Figure 5.15 is the point clouds file of Law Ka Htike Pagoda after defining the respective ROI.

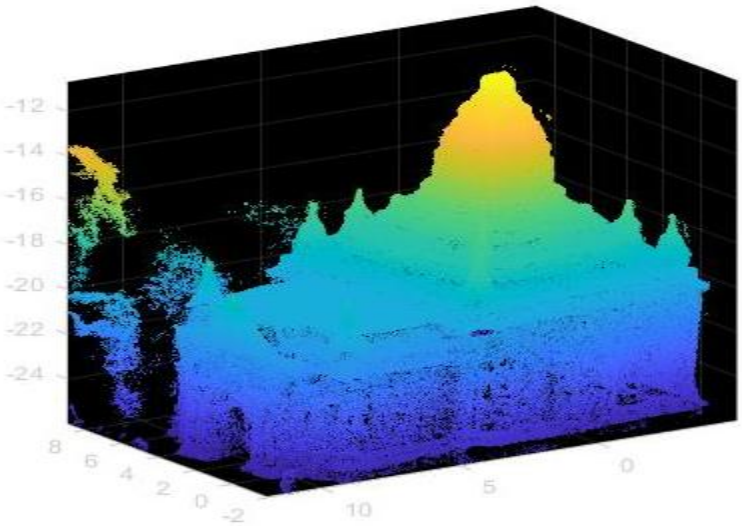


Figure 5.15 Defining ROI for Experiment 2

After extracting the ROI of Law Ka Htike pagoda, noises are included in the extracted pagoda region. So the noises point clouds are needed to remove. The plane of the pagoda is not required to calculate the damaged volume. Figure 5.16 is the extracted pagoda only after removing noises and plane.

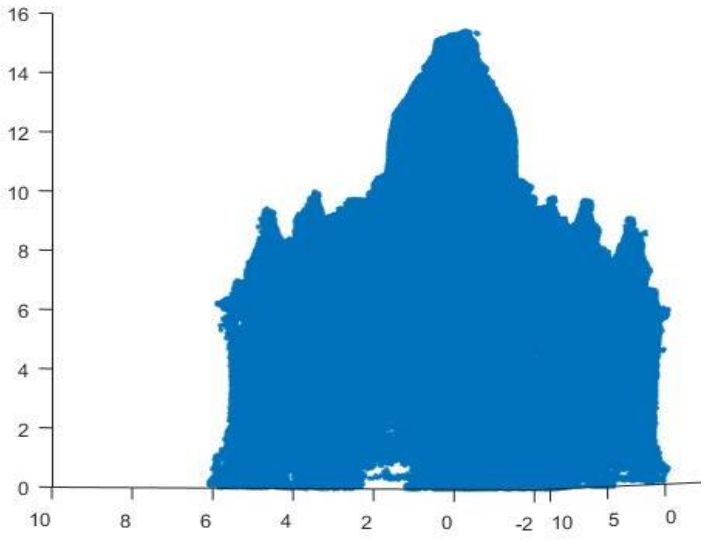


Figure 5.16 Removing Noises and Plane for Experiment 2

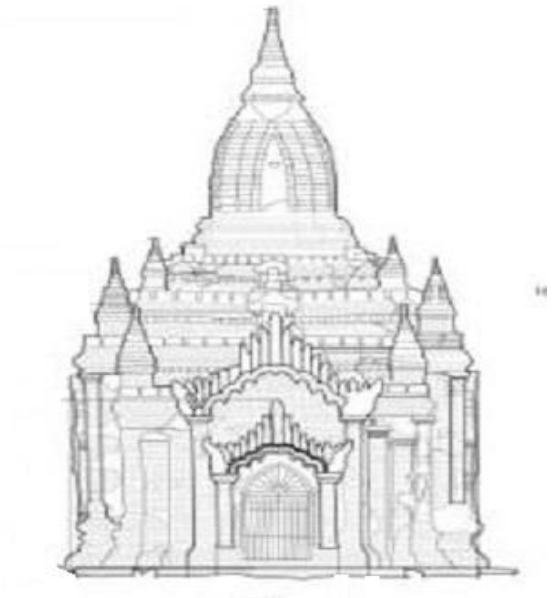


Figure 5.17 Extracting 2D Information for Experiment 2

The proposed system needs 2D information of the pagoda to calculate the damaged volume. The pixel information of height and width can be extracted from 2D

image of Law Ka Htike Pan pagoda. Figure 5.17 is 2D image of Law Ka Htike Pan pagoda. After that the damaged volume is calculated with the detailed steps mentioned in Section 5.1.6. The percentage of the damaged volume of Law Ka Htike Pan pagoda is shown in Figure 5.18.

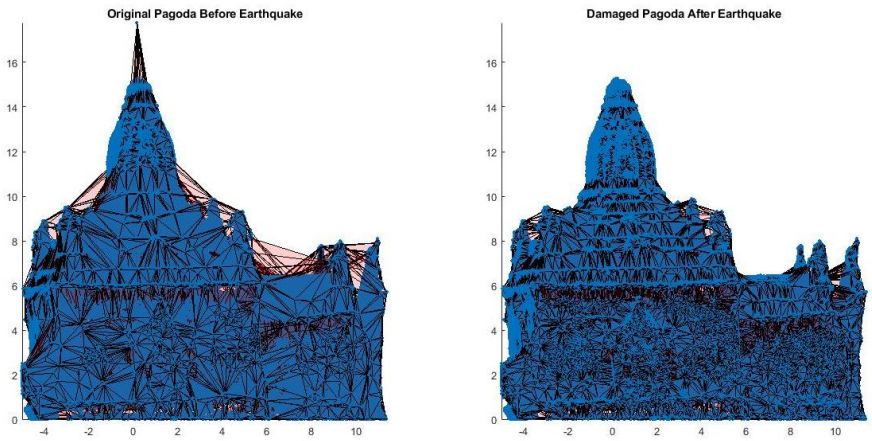


Figure 5.18 Damaged Volume Comparisons for Experiment 2

The Law Ka Htike Pan pagoda has damaged after earthquake. But the proposed system gives the damaged percentage about 7%. The result of estimated damaged volume is shown in following pie chart.

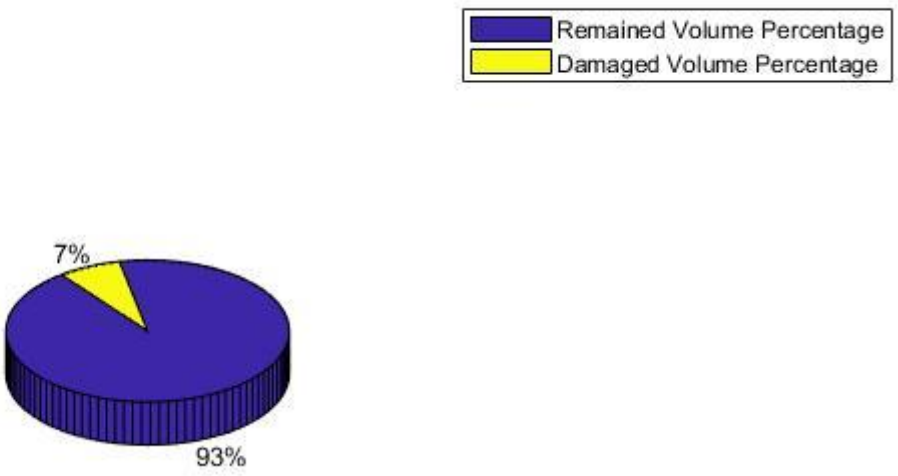


Figure 5.19 Pie Chart of Damaged Volume Percentage for Experiment 2

Figure 5.20 shows the processing time of each step in the proposed system. In each step, the running time of removing noises and plane takes longer time because

clustering the point clouds and eliminating the plane operates on each 3D point of the pagoda.

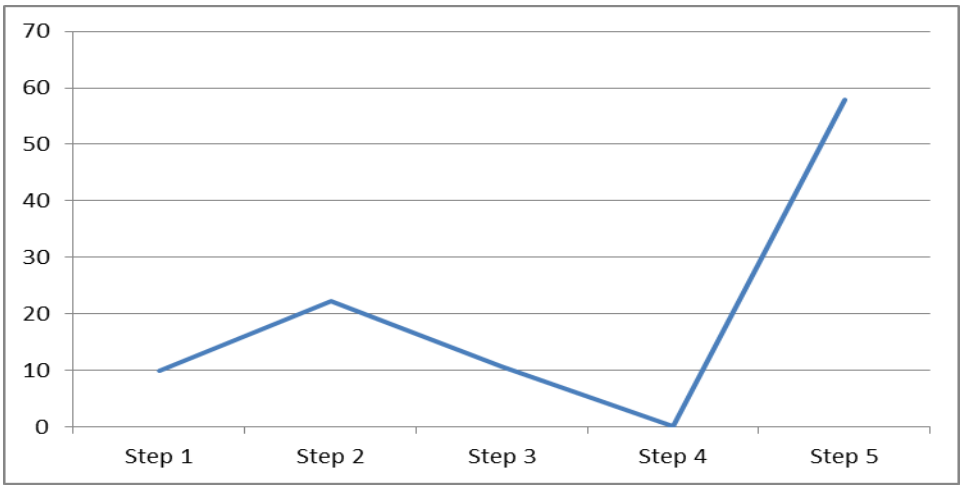


Figure 5.20 Computation Time for each Steps of Experiment 2

5.3.3 Experiment 3 for That Byin Nu Pagoda

In experiment 3, the damaged volume of That Byin Nu Pagoda is calculated. The total 407 aerial images of That Byin Nu Pagodas are taken and inputted to pix4D software to generate the point clouds. The generation of the point clouds from 2D aerial images takes the longer processing time about 110 minutes. Figure 5.21 shows the aerial photos of That Byin Nu Pagodas after earthquake.

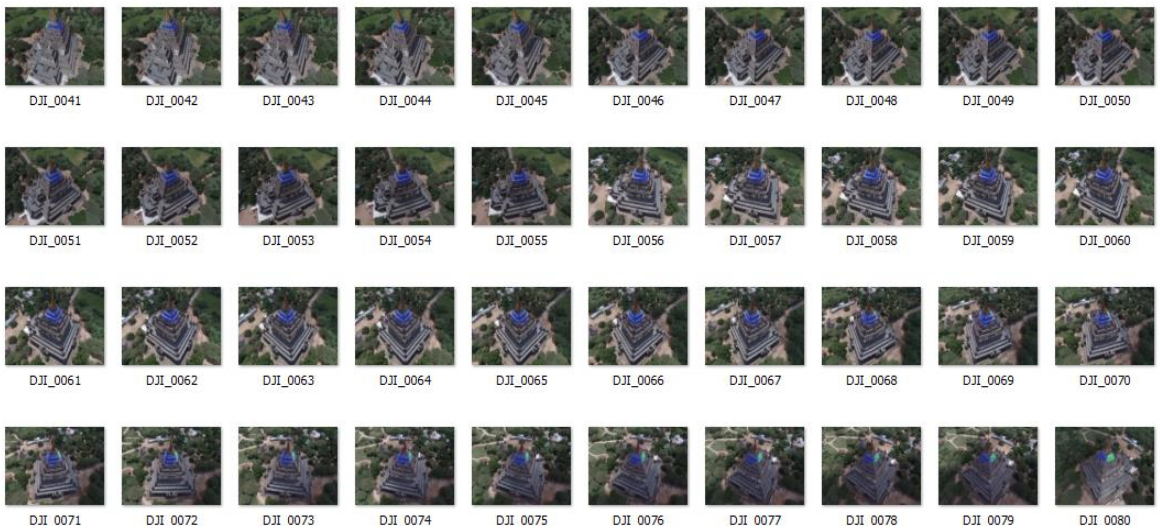


Figure 5.21 Aerial Images of That Byin Nu Pagoda

The pix4D software takes the longer processing time but it gives the accuracy of 3D reconstruction than other 3D reconstruction software. The total 25719197 point cloulds of That Byin Nu Pagoda are prepared to detect the damaged volume. The Figure 5.22 shows the point clouds file (PLY file) of That Byin Nu Pagoda.

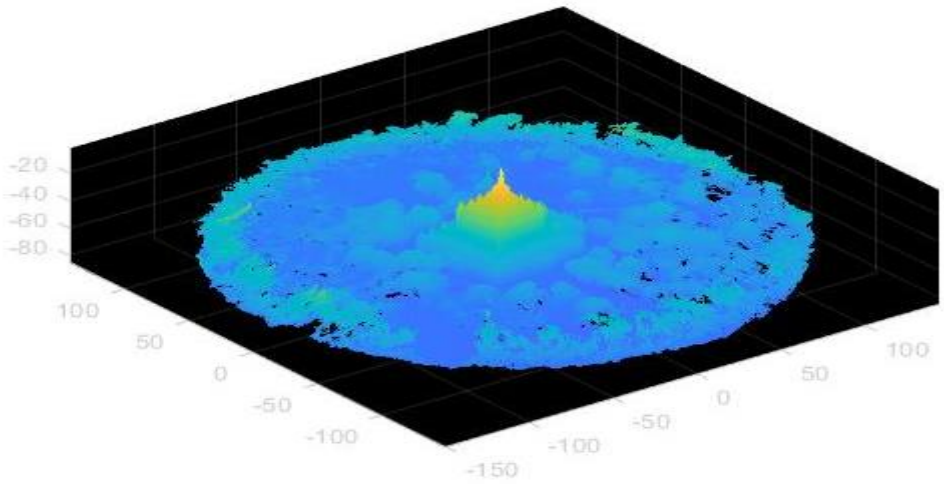


Figure 5.22 Point Clouds of That Byin Nu Pagoda

Objects such as roads, cars, trees, and buildings are common to a typical urban aerial image. But, in the proposed system, only pagoda is needed to extract. The small region which may not contain in pagoda are removed by defining ROI (region of interest). Therefore, it defined the region of interest of the pagoda. Figure 5.23 is the point clouds file of That Byin Nu Pagoda after defining the respective ROI.

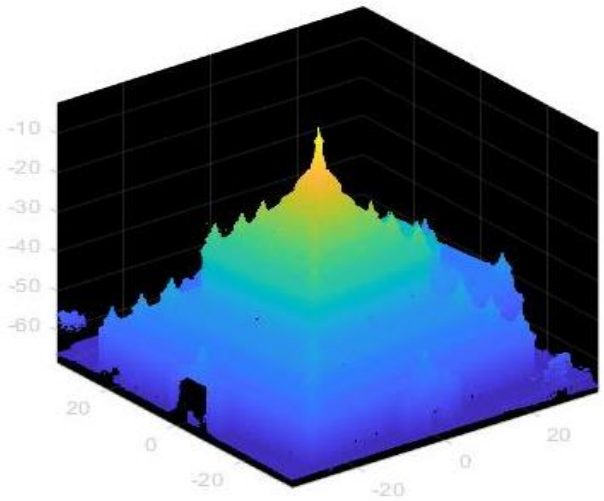


Figure 5.23 Defining ROI for Experiment 3

After extracting the ROI of That Byin Nu pagoda, noises are included in the extracted pagoda region. So the noises point clouds are needed to remove. The plane of the pagoda is not required to calculate the damaged volume. Figure 5.24 is the extracted pagoda only after removing noises and plane.

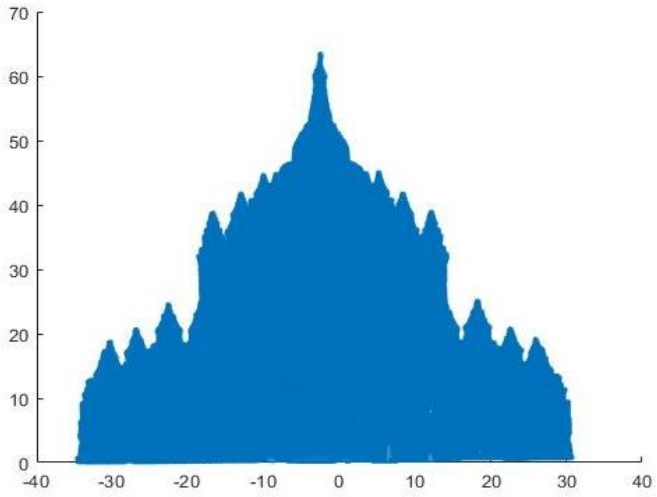


Figure 5.24 Removing Noises and Plane for Experiment 3

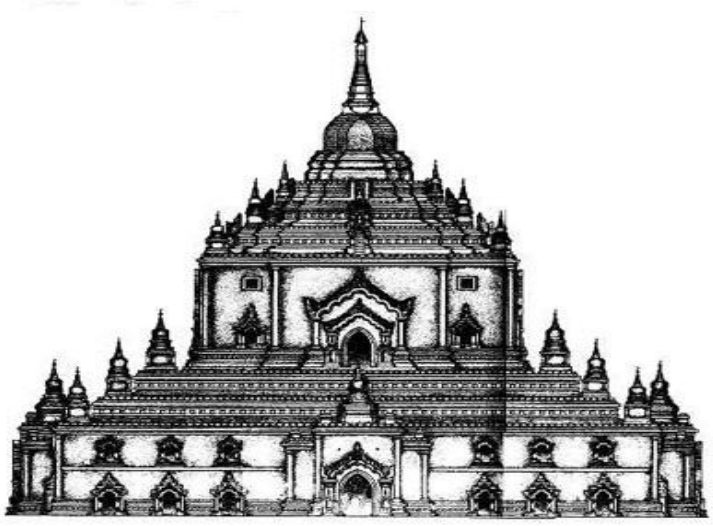


Figure 5.25 Extracting 2D Information for Experiment 3

The proposed system needs 2D information of the pagoda to calculate the damaged volume. The pixel information of height and width can be extracted from 2D image of That Byin Nu pagoda. Figure 5.25 is 2D image of That Byin Nu pagoda. After that the damaged volume is calculated with the detailed steps mentioned in Section 5.1.6. The before and after damaged portion of That Byin Nu pagoda is shown in Figure 5.26.

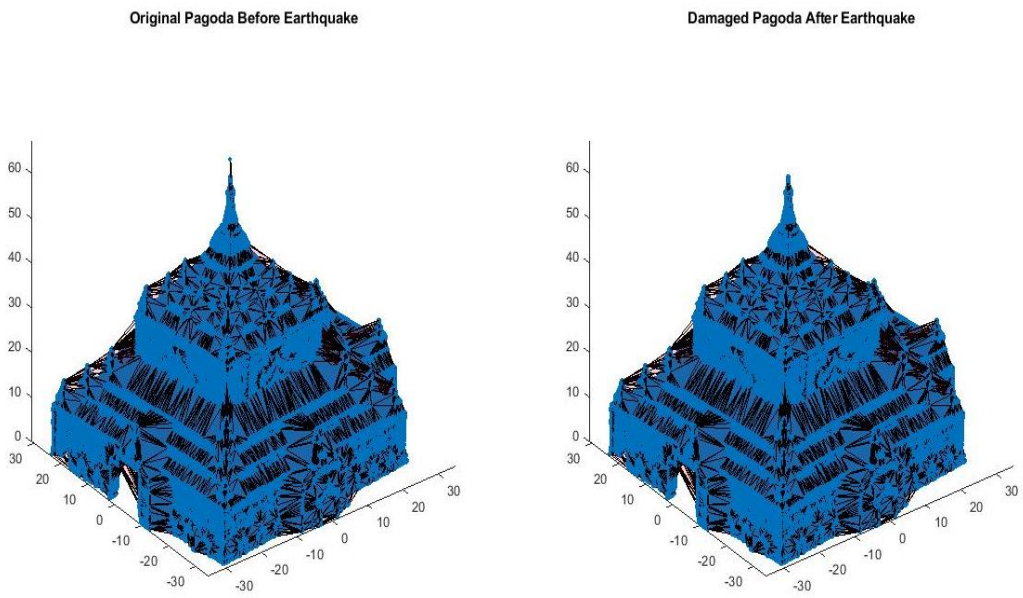


Figure 5.26 Damaged Volume Comparisons for Experiment 3

The That Byin Nu De pagoda has no damaged after earthquake. But the proposed system gives the damaged percentage about 3%. The result of estimated damaged volume is shown in following pie chart.

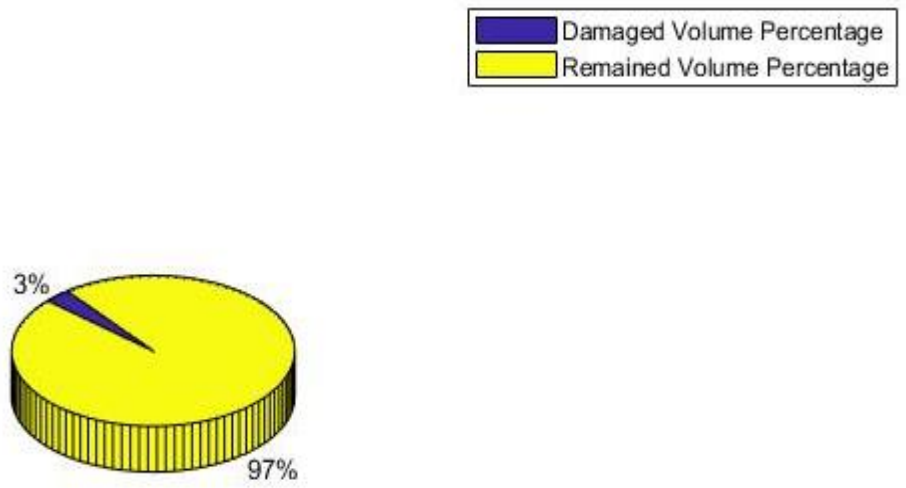


Figure 5.27 Pie Chart of Damaged Volume Percentage for Experiment 3

Figure 5.28 shows the processing time of each step in the proposed system. In each step, the running time of removing noises and plane takes longer time because clustering the point clouds and eliminating the plane operates on each 3D point of the pagoda.

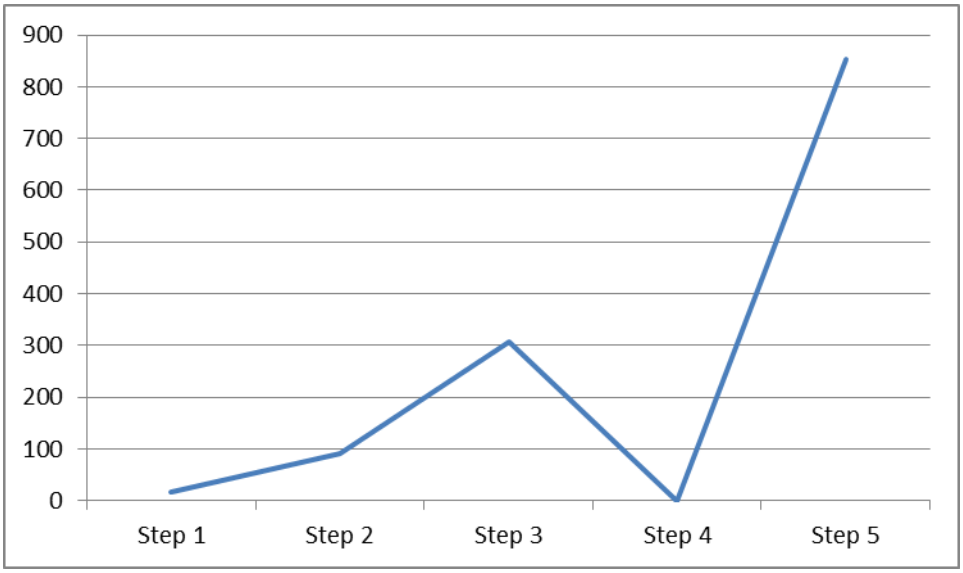


Figure 5.28 Computation Time for each Steps of Experiment 3

5.3.4 Experiment 4 for Sula Ma Ni Pagoda

In experiment 4, the damaged volume of Sula Ma Ni Pagoda is calculated. The total 295 aerial images of Sula Ma Ni Pagodas are taken and inputted to pix4D software to generate the point clouds. The generation of the point clouds from 2D aerial images takes the longer processing time about 90 minutes. Figure 5.29 shows the aerial photos of Sula Ma Ni Pagodas after earthquake.

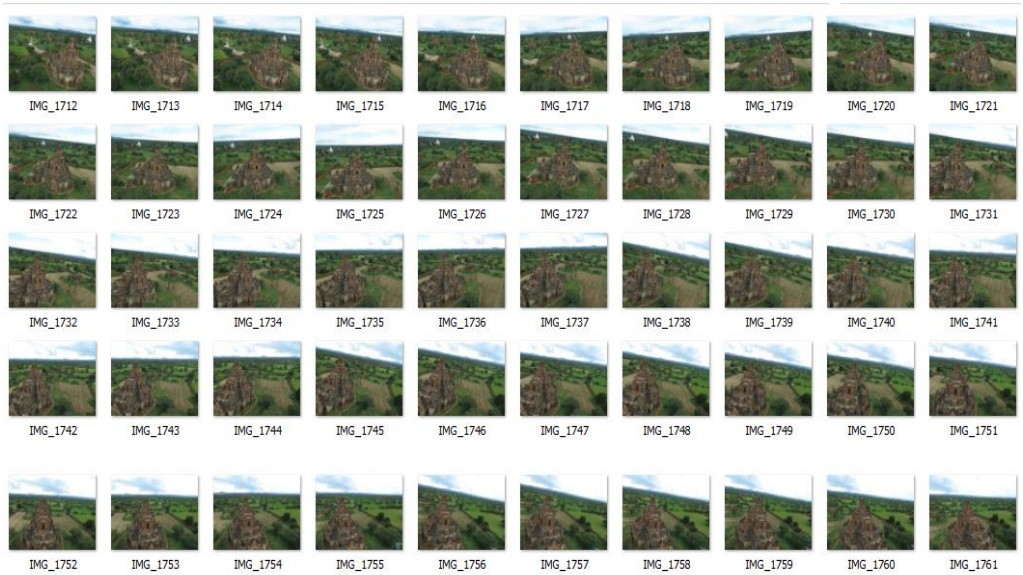


Figure 5.29 Aerial Images of Sula Ma Ni Pagoda

The pix4D software takes the longer processing time but it gives the accuracy of 3D reconstruction than other 3D reconstruction software. The total 8 279 211 point cloulds of Sula Ma Ni Pagoda are prepared to detect the damaged volume. The Figure 5.30 shows the point clouds file (PLY file) of Sula Ma Ni Pagoda.

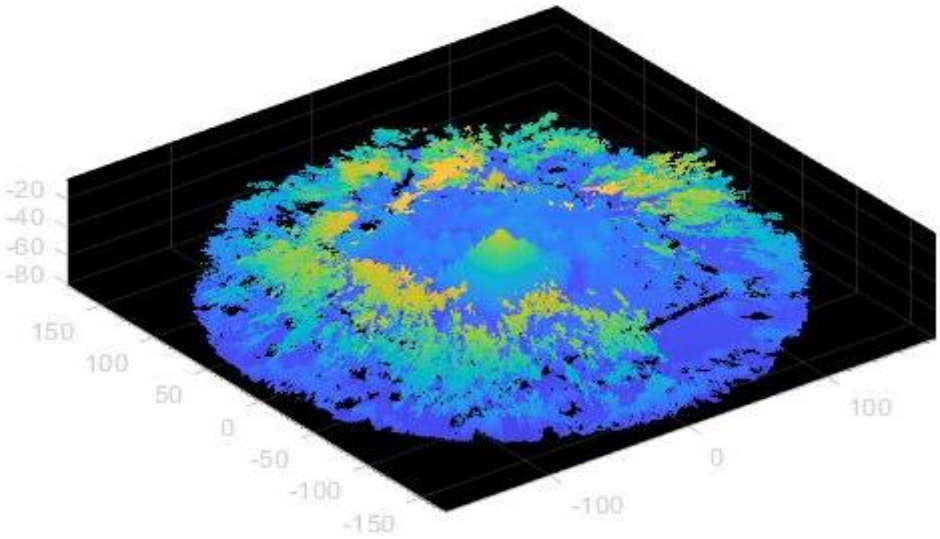


Figure 5.30 Point Clouds of Sula Ma Ni Pagoda

Objects such as roads, cars, trees, and buildings are common to a typical urban aerial image. But, in the proposed system, only pagoda is needed to extract. The small region which may not contain in pagoda are removed by defining ROI (region of interest). Therefore, it defined the region of interest of the pagoda. Figure 5.31 is the point clouds file of Sula Ma Ni Pagoda after defining the respective ROI.

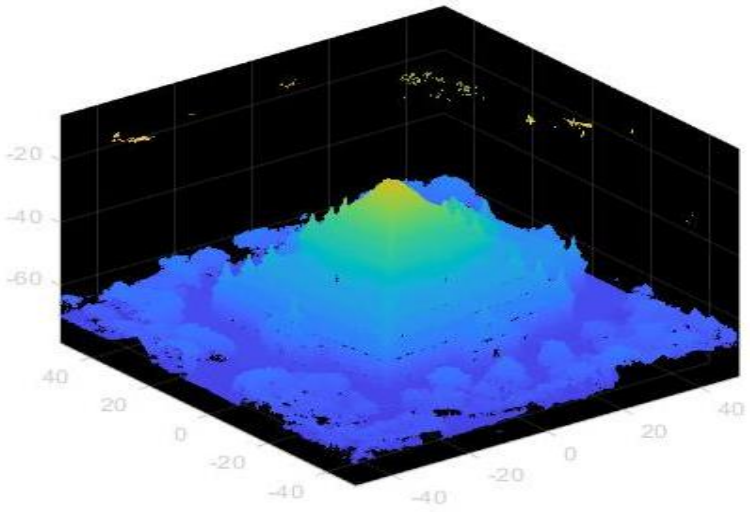


Figure 5.31 Defining ROI for Experiment 4

After extracting the ROI of Sula Ma Ni pagoda, noises are included in the extracted pagoda region. So the noises point clouds are needed to remove. The plane of the pagoda is not required to calculate the damaged volume. Figure 5.32 is the extracted pagoda only after removing noises and plane.

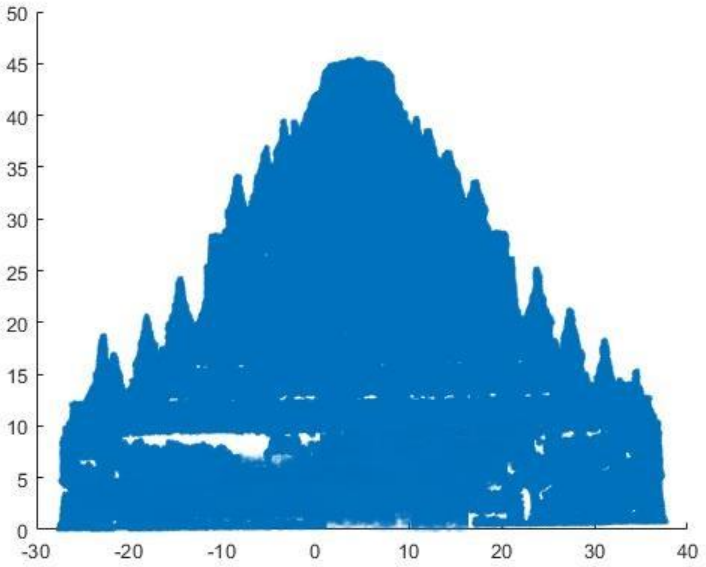


Figure 5.32 Removing Noises and Plane for Experiment 4

Figure 5.33 Extracting 2D Information for Experiment 4

The proposed system needs 2D information of the pagoda to calculate the damaged volume. The pixel information of height and width can be extracted from 2D image of Sula Ma Ni pagoda. Figure 5.33 is 2D image of Sula Ma Ni pagoda. After that the damaged volume is calculated with the detailed steps mentioned in Section 5.1.6. The before and after damaged portion of Sula Ma Ni pagoda is shown in Figure 5.34.

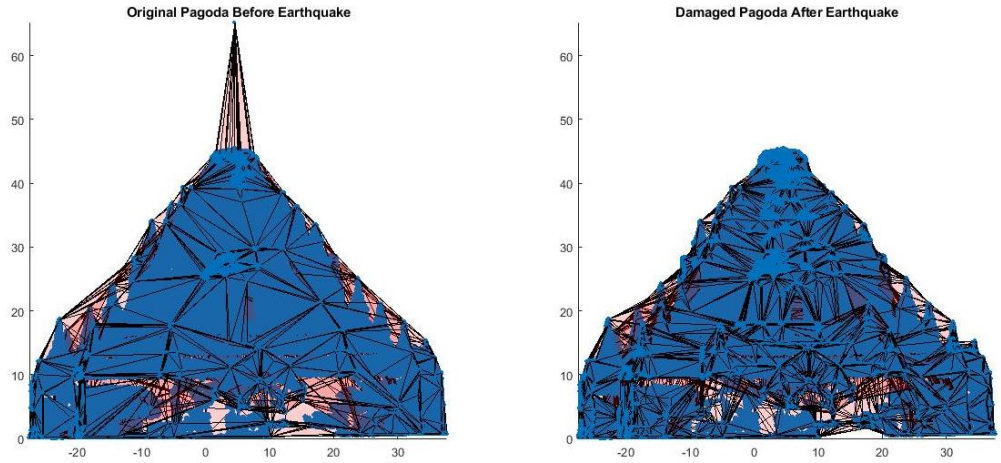


Figure 5.34 Damaged Volume Comparisons for Experiment 4

The Sula Ma Ni pagoda has damaged after earthquake. The proposed system gives the damaged percentage about 14%. The result of estimated damaged volume is shown in following pie chart.

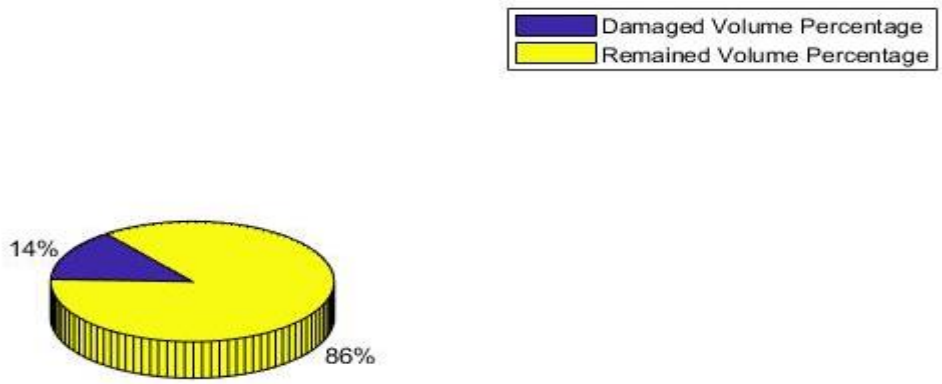


Figure 5.35 Pie Chart of Damaged Volume Percentage for Experiment 4

Figure 5.36 shows the processing time of each step in the proposed system. In each step, the running time of removing noises and plane takes longer time because

clustering the point clouds and eliminating the plane operates on each 3D point of the pagoda.

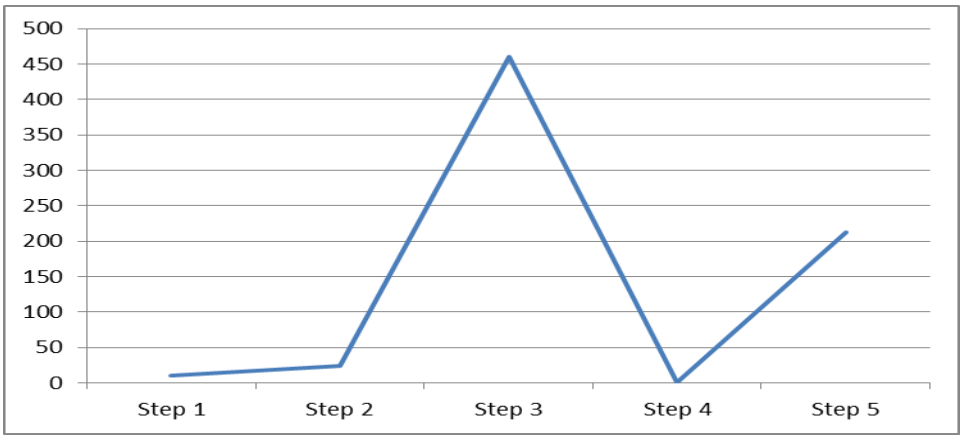


Figure 5.36 Computation Time for each Steps of Experiment 4

5.3.5 Experiment 5 for Ga Dot Palin Pagoda

In experiment 5, the damaged volume of Ga Dot Palin Pagoda is calculated. The total 229 aerial images of Ga Dot Palin Pagodas are taken and inputted to pix4D software to generate the point clouds. The generation of the point clouds from 2D aerial images takes the longer processing time about 50 minutes. Figure 5.37 shows the aerial photos of Ga Dot Palin Pagodas after earthquake.

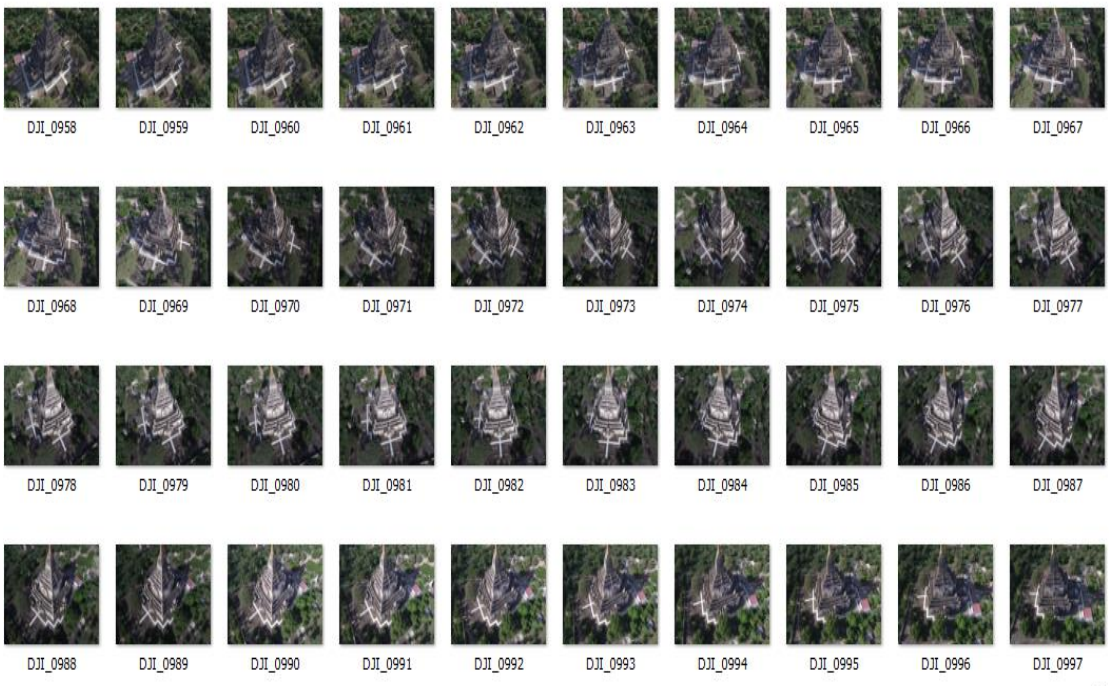


Figure 5.37 Aerial Images of Ga Dot Palin Pagoda

The pix4D software takes the longer processing time but it gives the accuracy of 3D reconstruction than other 3D reconstruction software. The total 11751907 point cloulds of Ga Dot Palin Pagoda are prepared to detect the damaged volume. The Figure 5.38 shows the point clouds file (PLY file) of Ga Dot Palin Pagoda.

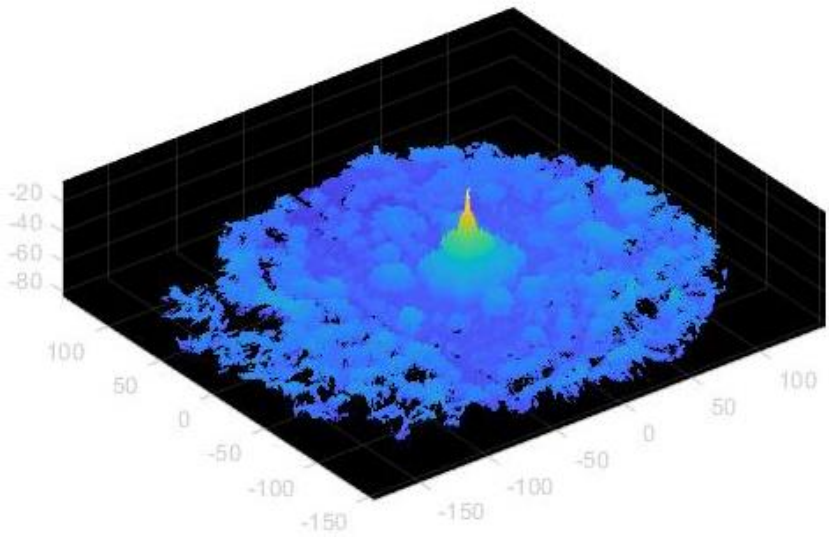


Figure 5.38 Point Clouds of Ga Dot Palin Pagoda

The Objects such as roads, cars, trees, and buildings are common to a typical urban aerial image. But, in the proposed system, only pagoda is needed to extract. The small region which may not contain in pagoda are removed by defining ROI (region of interest). Therefore, it defined the region of interest of the pagoda. Figure 5.39 is the point clouds file of Ga Dot Palin Pagoda after defining the respective ROI.

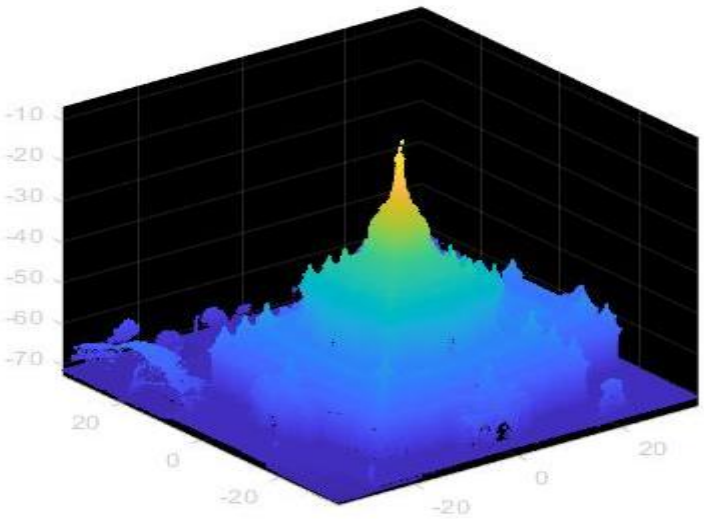


Figure 5.39 Defining ROI for Experiment 1

After extracting the ROI of Ga Dot Palin pagoda, noises are included in the extracted pagoda region. So the noises point clouds are needed to remove. The plane of the pagoda is not required to calculate the damaged volume. Figure 5.40 is the extracted pagoda only after removing noises and plane.

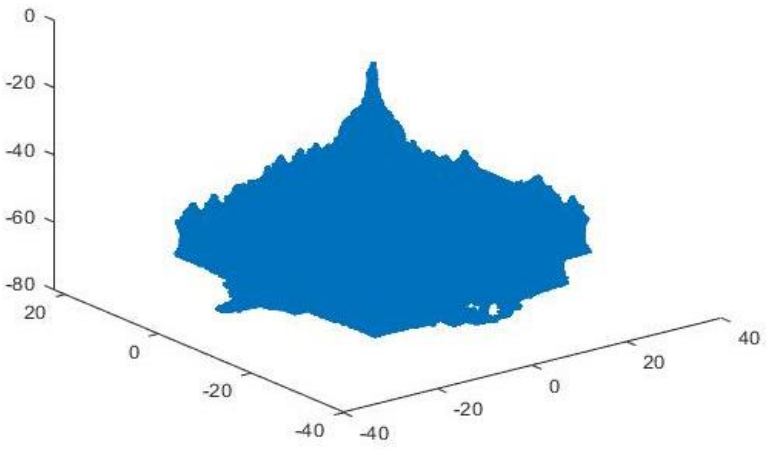


Figure 5.40 Removing Noises and Plane for Experiment 5

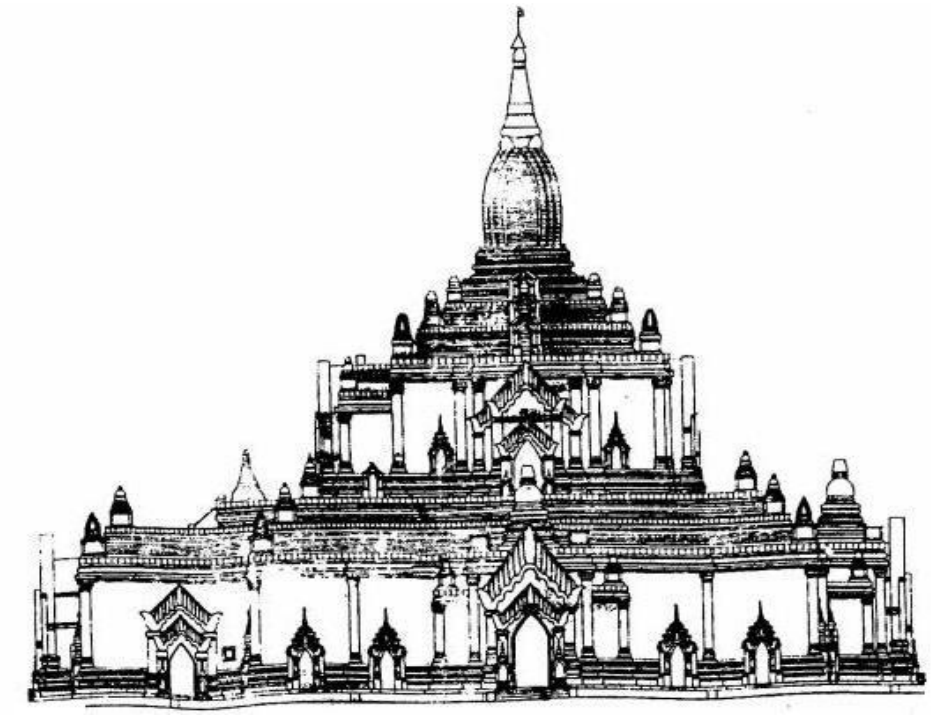


Figure 5.41 Extracting 2D Information for Experiment 5

The proposed system needs 2D information of the pagoda to calculate the damaged volume. The pixel information of height and width can be extracted from 2D

image of Ga Dot Palin pagoda. Figure 5.41 is 2D image of Ga Dot Palin pagoda. After that the damaged volume is calculated with the detailed steps mentioned in Section 5.1.6. The before and after damaged volume of Ga Dot Palin pagoda is shown in Figure 5.42.

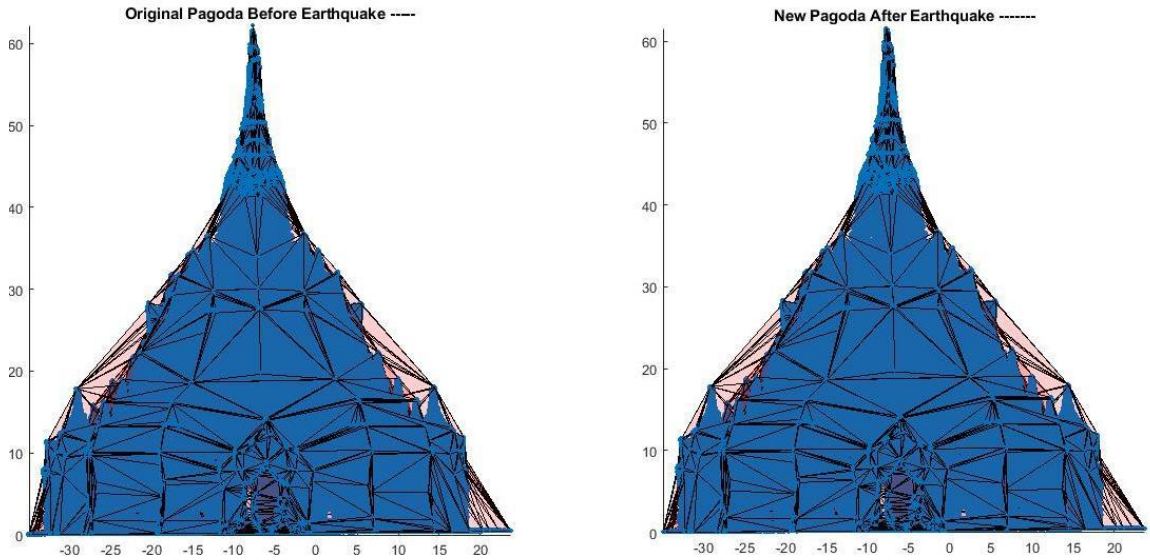


Figure 5.42 Damaged Volume Comparisons for Experiment 5

The Ga Dot Palin pagoda has no damaged after earthquake. The proposed system gives the damaged percentage about 0%. The result of estimated damaged volume is shown in following pie chart.

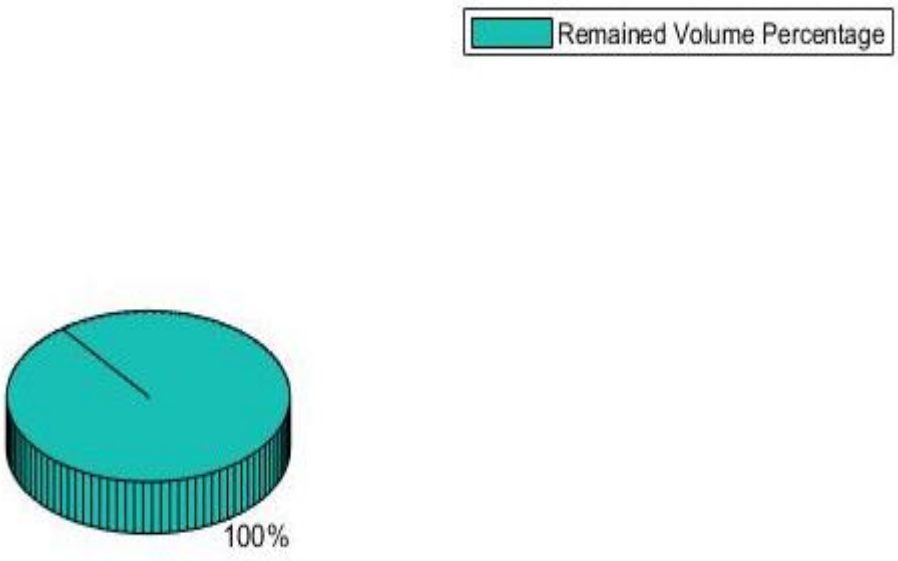


Figure 5.43 Pie Chart of Damaged Volume Comparisons for Experiment 5

Figure 5.44 shows the processing time of each step in the proposed system. In each step, the running time of removing noises and plane takes longer time because clustering the point clouds and eliminating the plane operates on each 3D point of the pagoda.

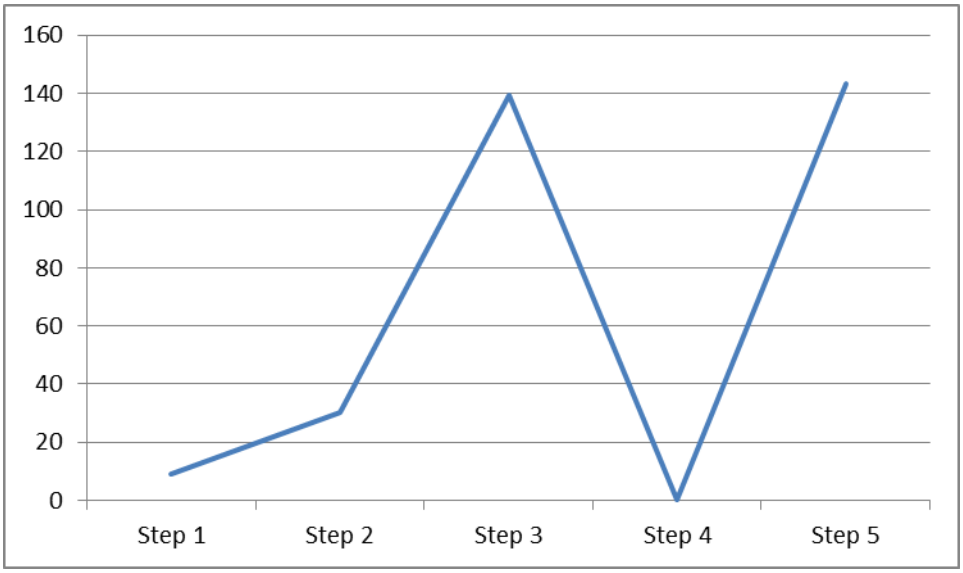


Figure 5.44 Computation Time for each Steps of Experiment 5

5.4 Performance Analysis

The proposed method has been tested on the damaged and undamaged pagodas after the earthquake. The processing time of the proposed system mainly depends on the number of point clouds produced from pix4D software. According to the statistics, the 3D damaged volume detection method can be confirmed with the number of point clouds without losing the original data. In this statistic, the running time of the defining ROI, the removing noises and plane, the extracting 2D information, and the calculating 3D damaged volume are included. The analysis is performed by using the five experiments of damaged pagodas with approximately 10 million-point clouds. Figure 5.45 shows the average running time of the five experiments tested with proposed system. According to the analysis, the greater number of 3D point clouds, the processing time will also be increased. Among the running time of each step of proposed system, removing noises and plane takes long time because the system determines what each point cloud in input data set is in outlier or in plane.

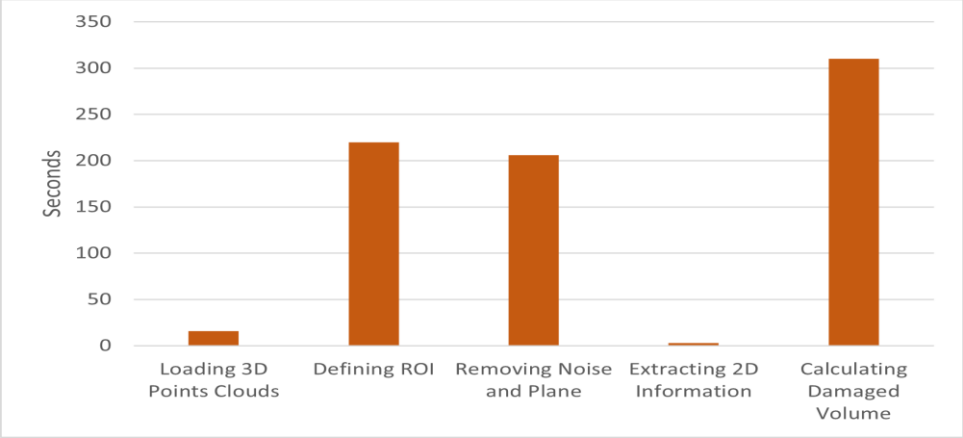


Figure 5.45 Average Processing Time of the Proposed System

The processing time of the proposed method is compared with calculating 3D damaged volume using 3D Hough Transform mentioned in section 5.2. There are five stage in 3D Hough Transform system. It has loading point clouds, defining ROI of pagoda with manual clicking four corner points, removing noises except plane, finding 3D candidate lines and calculating damaged volume finding filling points clouds. The 3D Hough Transform gives the candidate lines to estimate the top intersection point and after that, the 3D damaged volume is calculated using the equation 5.1. In 3D Hough Transform, finding the candidate straight lines stage takes long time to process. The following bar chart (Figure 5.46) shows the average running time of five experiments of the damaged pagodas with 3D Hough Transform.

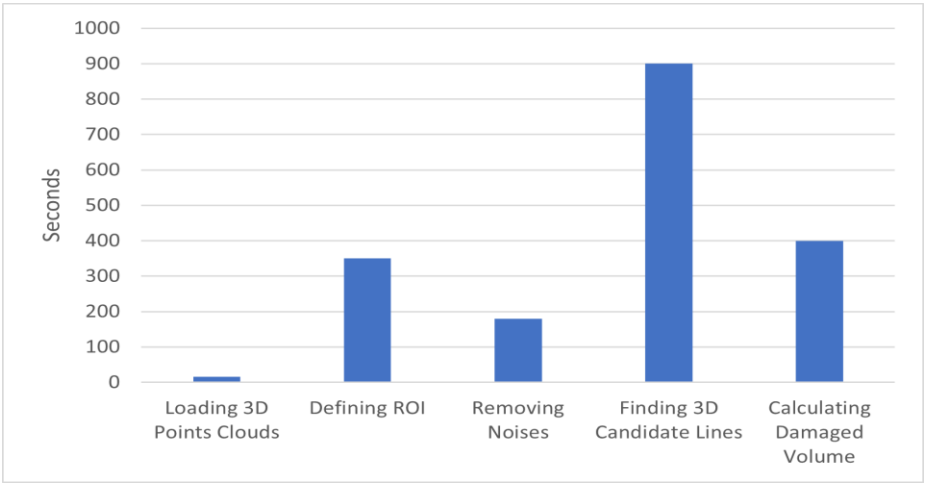


Figure 5.46 Average Processing Time of 3D Hough Transform

The proposed method for estimating the damaged volume of the 3D historic pagoda is firstly implemented in Myanmar. This proposed system is mainly

contributed to detecting the damaged portion of pagoda based on the 3D structure of the historic pagoda after the earthquake. By comparing comparison of processing time of the proposed system with 3D Hough Transform, the average processing time of the proposed system has less effort than 3D Hough Transform system. The following table (Figure 5.47) shows the comparison of processing times of both systems.

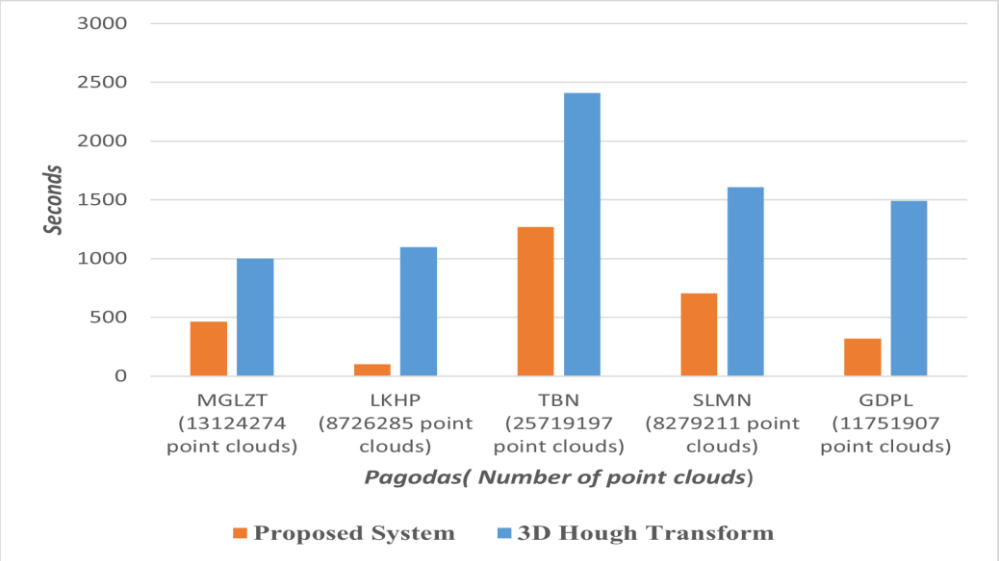


Figure 5.47 Comparison Result of Processing Time

While comparing 3D Hough Transform and the proposed system, the general workflow of both systems is slightly same, but the internal processing behavior is different. In the proposed system, the removing noises and plane step only extract the pagoda region eliminating unnecessary outliers. This step can reduce about 50% percent of the original point clouds. Therefore, this step makes the better accuracy and the makes whole system fast. The calculating 3D damaged volume step in proposed system combine 2D information extracted from 2D image; thus, the processing time of this step is faster than finding 3D candidate lines in 3D Hough Transform. In 3D Hough Transform, the finding 3D candidate line process on each point of the inputted point clouds, so it takes the long time, and it reduces only outlier points except plane. Therefore, the number of inputted point clouds is more increased in 3D Hough Transform. The greater number of point clouds makes the longer processing time of the system. The proposed system processes on the extracted pagoda only and it does not operate on each point clouds. It process on the range of the inputted point clouds. Therefore the proposed system has the least processing time than 3D Hough Transform.

5.5 Summary

This chapter emphasizes on the design implementation and experimental results of the proposed system. The first section is implementation of the research process. In this section, the six subsections are detail explained. Each subsection explains the ways how to collect the required aerial images is taken by UAV; the information about study area, Bagan; how to produce 3D point clouds using Pix4D software, how to define the ROI on the specific pagoda depend on the X-plane range of input 3D point clouds; the elimination of outliers and plane; extracting 2D information from standardized images and how to calculate damaged volume of 3D pagodas. The second section is to find the damaged volume of 3D pagodas using 3D Hough Transform. In this section how 3D Hough Transform is used to find damaged volume of historic pagodas. Experimental results of proposed system are carried out with five pagodas in the third section. In this section, the detail steps of the proposed system are discussed with relevant output diagram. After that section, the next section is performance analysis or comparative processing time results of 3D Hough Transform algorithm and the proposed section. The next chapter will be discussed about conclusion and future work of the proposed system.

CHAPTER 6

CONCLUSION AND FUTURE WORK

In conclusion, the system emphasizes on automatic damage detection of 3D historic Bagan Pagodas with 3D Model after the earthquake. How the system builds on the nature of 3D data were explained in Chapter 2. The data collection steps and the methods for producing 3D point clouds from UAV images are explained in detail. The extracting ROI for 3D pagodas and segmenting the point clouds using Euclidean methods can be seen in Chapter 3. Chapter 4 explains the importance of removing noises and plane for the proposed system. Moreover, the extracting 2D information from the 2D images is described. The 3D damaged volume calculation is performed using the 2D and 3D information. The proposed system achieves complete automatic damaged volume detection system from the UAV images and based on the structure of 3D point clouds with experimental results in Chapter 5. Moreover, after the earthquake the automatic damaged volume can be easily detected with time-saving and human resources.

6.1 Summary of Research Contributions

In automatic 3D damaged volume detection system, a new approach for 3D volume calculation from point clouds is proposed. In this dissertation, the proposed system has three basic steps. The first step is producing the point clouds, the second one is preprocessing the 3D point clouds and the final step is calculating the damaged volume of 3D pagodas. The input data, point clouds, are produced by applying the structure from motion method on UAV images. The aerial photos of the specific pagodas are collected using the hexacopter drone in Bagan. The pix4D software is used to generate dense 3D point clouds. Before inputting the aerial images to pix4D software, some images are eliminated by comparing the captured time and duplicate features on the successive images. This process makes running time fast in producing the dense point clouds in pix4D software. After this stage, the proposed system defines the ROI for specific pagoda based on the properties of the 3D structure of the pagodas and this has been detail explained in chapter 3. Noise removing and plane removing are the important phases to estimate the exact damaged volume of the

pagoda. After getting only 3D structure of the damaged pagoda, 2D information is extracted from the 2D imaged. The automatic damaged volume is calculated combining the measurement from 2D and 3D information of the specific pagoda. The proposed system developed the automatic 3D damaged volume calculation system that can overcome the challenges in rapid damaged assessment after the earthquake with rapid time consuming.

All the experiments are performed on aerial images of damaged pagodas in Bagan after earthquake. According to the result of five experiments, the processing time of the proposed system is depending on the number of point clouds of the pagoda. If the point clouds are fewer, the processing time of the proposed system is faster. The system can solve the difficulties from the removing noises and plane of 3D pagodas that can affect the accuracy of the whole system. Furthermore, the exact information from 2D image is important for determining the volume of the pagoda before the earthquake that is essential for 3D damaged volume detection. In this work, the system can estimate the volumes of the damaged pagodas after earthquake with the accurate and correct assessment.

6.2 Research Outcomes

In this section, a brief history of this research and progress are presented according to the listed publications. This research started with a study in how to process the change detection of the building extraction in urban area which detects the urban region growing rate see in publication [1]. Moreover, the study on the analyzing the disaster and the damaged rate of the affected region after the disaster is continued in publication [2]. The change detection system is mainly depended on the input data. The time series input data is rare for our country. The change detection on increase rate or decrease rate of Yangon city can be calculated by using the Landsat time series images in publication [3]. The structure of damaged pagoda before earthquake can be estimated by using the vanishing point. The damaged volume can be calculated by comparing the structures of the historic pagodas before and after earthquake found in publication [4]. The 3D reconstruction of historic pagodas is important for automatic damaged volume detection system. The importance of 3D reconstruction in damaged volume detection system can be seen in publication [5]. The Hough transformation is implemented to find 3D objects in large amount of 3D

point clouds. This gives each point in the input data that will vote on a sinusoidal surface in the Hough space. The automatic damaged volume detection of 3D historic pagoda can be seen in publication [6].

6.3 Advantages of Proposed System

The advantages of the proposed method for 3D damaged volume detection over the other damaged volume detection methods are as follows:

- The proposed system is pioneered for 3D damaged volume detection of historic pagodas after earthquake in Bagan, Myanmar.
- The proposed system is intended for supporting the rapid damage assessment for the preserving cultural heritage.
- Using proposed system not only reduces the time complexity but also is easy and convenient for user.
- The corresponding problems cannot affect on the proposed system as the other existing system used multiple methods.
- The processing time of the proposed system is less than that of the other 3D damaged volume detection systems.
- The proposed system can handle the unlimited number of 3D point clouds.

6.4 Limitations of Proposed System

The limitations of this work are that it cannot estimate the accurate damaged volume of historic pagodas. The damaged percentage of the pagoda mainly depends on the output, 3D point clouds, from pix4d software. The proposed system has determined the volume of the damaged pagoda mainly on the volume of the rectangle shape. The structure of the historic pagodas in Bagan varies on the archeological aspects such as the established year, the ages of the pagodas and so on. Some pagodas have the complex structure. It means that the base part is rectangle and the upper part is small rectangle shape and then next upper part is cylinder shape. The proposed system cannot give the accurate result for the complex structures.

6.5 Future Extensions

Preserving the cultural heritage of a country is important to develop the quick assessments for maintaining the valuable heritages. This research work provides for preserving the cultural heritage of the ancient pagodas. In this research work, the first stage is for recovering the original structure of a pagoda by taking the damaged images using UAV; the second part is producing the required point clouds using the open source pix4D software and the third part is estimating the damaged volume using nature of the ancient pagodas' structures. The accuracy of the proposed system mainly depends on the second stage, the production of the point clouds from pix4D software. The reconstruction of the damaged pagoda should be implemented with own contribution by combining with the structure from motion and reconstruction algorithm from multiple images. The proposed system calculates the damaged pagoda using the general volume calculation of rectangle shape. Unfortunately, the historic pagodas in Bagan have complex archeological structure. Therefore, the volume calculations for these complex structures are needed to give the exact damaged volume accuracy.

LIST OF ACRONYMS

2D	Two-Dimensional
3D	Three-Dimensional
BF	Bilateral Filter
CAD	Computer Aided design
CIR	Color-Infrared
CODSM	Co-Opposite-Direction Slicing Method
CPU	Central Processing Unit
dpi	Dots per inch
DRM	Disaster Risk Management
DSLR	digital single-lens reflex camera
DSM	Digital Surface Model
DTM	Digital Terrain Model
EOS	Electro-Optical System
GBD	Graph-based denoising
GHz	Giga Hertz
GIS	Geographic Information System
HOG	Histogram of Oriented Gradients
KD/K-D	K-dimensional
LiDARs/LIDAR	Light Detection and Ranging
LRF	Local Reference Frame
MLP	Multilink Protocol
MVS	Multi-view Stereo
nDSM	Normalized Digital Surface Model
NLD	Non-local Denoising
PCO	Point Cloud Object
PCSS	Semantic point cloud segmentation
PDE	Partial Differential Equations
RAM	Random Access Memory
RANSAC	RANdom SAmple Consensus
RGB-D	Red Green Blue Depth

ROI	Region Of Interest
SAM	Spatial Access Methods
SfM	Structure from Motion
SIFT	Scale-invariant Feature Transform
SM	Slicing Method
SVM	Support-vector Machines
UAV	Unmanned Aerial Vehicle
VGA	Video Graphics Array
VHR	Very High Resolution
VLS	Video Laser Scan
VTOL	Vertical Take-off and Landing

AUTHOR'S PUBLICATION

[P1] T. Aung and M. M. Sein, "Change Detection of the Building areas in Urban Regions", in Proceedings of 14th International Conference on Computer Applications (ICCA2016), 124-130, Yangon, Myanmar, 2016.

[P2] T. Aung and M. M. Sein, "Analyzing the Disaster", in Proceedings of 10th International Conference on Genetic and Evolutionary Computing (ICGEC2016), 238-246, Fujian, China, 2016.

[P3] T. Aung and M. M. Sein, " Detection the Urban Changing Areas of Yangon City Using Landsat Time series Images", 15th International Conference on Computer Applications (ICCA2017), Yangon, Myanmar, February 2017.

[P4] T. Aung and M. M. Sein, "Detecting the 3D Damaged Area of Historic Pagodas Using Vanishing Point", in Proceedings of the16th International Conference on Computer Applications (ICCA2018), 111-114, Yangon, Myanmar, 2018.

[P5] T. Aung and M. M. Sein, "Reconstruction of Ancient Pagodas", in Proceedings of IEEE 7th Global Conference on Consumer Electronics (IEEE GCCE), 523-524, Nara, Japan, 2018.

[P6] T. Aung and M. M. Sein, "Estimating Damaged Volume of Historic Pagodas in Bagan after Earthquake using 3D Hough Transform", International Journal on Advanced Science, Engineering and Information Technology (IJASEIT), Volume 10, No. 1, 90-97, February, 2020.

BIBLIOGRAPHY

[1] D. Abate, “Built-Heritage Multi-temporal Monitoring through Photogrammetry and 2D/3D Change Detection Algorithms”, Studies in Conservation, 2018.

[2] A. Alharthy and J. Bethel, “Building extraction and reconstruction from LIDAR data”, Proceedings of ASPRS annual conference, pp.18-26, 2002.

[3] R. Allegre, Chaie and S. Akkouche, “A Dynamic Surface Reconstruction Framework for Large Unstructured Points Sets”, In Proc: of IEEE/Eurographics Symposium on Point Based Graphics, pp.17-26, 2006.

[4] R. Allègre, R. Chaine and S. Akkouche, “A flexible framework for surface reconstruction from large point sets”, Computers & Graphics, Volume 31, Issue 2, Pages 190-204, 2007.

[5] G. Arvanitis, A. S. Lalos, K. Moustakas and N. Fakotakis, "Real-Time Removing of Outliers and Noise in 3D Point Clouds Applied in Robotic Applications", Springer International Publishing, LNAI 10459, pp. 11–19, 2017.

[6] A. Ballabeni, F. I. Apollonio, M. Gaiani and F. Remondino "Advances in Image Pre-Processing to Improve Automated 3D Reconstruction", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-5/W4, 2015.

[7] T. Barbu and M. Luca, "Content-based iris indexing and retrieval model using spatial acces methods", International Symposium on Signals, Circuits and Systems (ISSCS), 10.1109/ISSCS.2015.7203970, 2015.

[8] G. Bianco, F. Bruno, M. Muzzupappa and A. Gallo, "3D Reconstruction and virtual exploration of submerged structures: a case study in the underwater archaeological site of Baia (Italy)", The 13th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST, 2012.

[9] G. Bianco, A. Gallo, F. Bruno, and M. Muzzupappa, "A Comparison Between Active and Passive Techniques for Underwater 3d Applications ", Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XXXVIII-5/W16, 357-363, 2011.

[10] A. Bienert, C. Hess, H.-G. Maas and G. V. Oheimb, "Voxel-Based Technique to Estimate the Volume of Trees from Terrestrial Laser Scanner Data", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-5, 2014.

[11] C. Brenner, "Towards Fully Automatic Generation of City Model", International Society for Photogrammetry and Remote Sensing (ISPRS), Vol.13, 2000.

[12] F. Bruno, A. Gallo, B. D. Petriaggi and P. Caputo, "3D Documentation and Monitoring of the Experimental Cleaning Operations in the Underwater Archaeological Site of Baia (Italy)", Institute of Electrical and Electronics Engineers Conference, October 2013.

[13] A. Carreira, R. Ventura and J. Gaspar, "Volumetrics - Measuring free volumes", Instituto Superior Tecnico / UTL, Lisbon, Portugal,2013.

[14] D. Cerra, J. Tian, V. Lysandrou and S. Plank, "Automatic Damage Detection for Sensitive Cultural Heritage Sites", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLI-B5, 2016.

[15] D. Cerra, S. Plank, V. Lysandrou and J. Tian, "Cultural Heritage Sites in Danger—Towards Automatic Damage Detection from Space", Journal of Remote Sensing, 2016.

[16] W. C. Chang, C. H. Wu, Y. H. Tsai and W. Y. Chiu, "Object Volume Estimation Based on 3D Point Cloud", Industrial Technology Research Institute, Taiwan, 2017.

[17] P. Charalambos and Y. Suya, "3D Reconstruction of Urban Areas", International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission, IEEE, 2011.

[18] F. Chiabrando, G. Sammartano, A. Spanò and G. Semeraro, "Multi-Temporal Images and 3D Dense Models for Archaeological Site Monitoring in Hierapolis Of Phrygia (TR)", Archeologia e Calcolatori 28.2, 469-484, 2017.

[19] P. Chmelara, L. Beran and L. Rejfek "The Depth Map Construction from a 3D Point Cloud", International Conference on Mechanics and Industrial Engineering (ICMIE), 2016.

[20] C. Crespo, J. Armesto, D. González-Aguilera and P. Arias, "Damage Detection on Historical Buildings Using Unsupervised Classification Techniques", International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVIII, Part 5, Commission V Symposium, Newcastle upon Tyne, UK. 2010.

[21] P. E. Debevec, C. J.Taylor and J. Malik, "Modeling and Rendering Architecture from Photographs: a hybrid geometry- and image-based approach", 23rd annual conference on Computer graphics and interactive techniques, pp 11–20, 1996.

[22] H. K. Dhonju, W. Xiao, J. P. Mills and V. Sarhosis, "Share Our Cultural Heritage (SOCH): Worldwide 3D Heritage Reconstruction and Visualization via Web and Mobile GIS", International Journal of Geo-Information, 2018.

[23] C. Duan, S. Chen and J. Kovacevic, "3D Point Cloud Denoising via Deep Neural Network Based Local Surface Estimation", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)", 2019.

[24] D. Fleet, T. Pajdla, B. Schiele and T. Tuytelaars, "Computer Vision", Proceeding in 13th European Conference, Zurich, Switzerland, Part IV, 2014.

[25] B. Frédéricque, S. Daniel, Y. Bédard, N. Paparoditis, “Populating a building multi-representation data base with photogrammetric tools: Recent progress”, ISPRS Journal of Photogrammetry and Remote Sensing, 441_460, 2008.

[26] L. Gomes, P. Bellon and L. Silva, "3D Reconstruction Methods for Digital Preservation of Cultural Heritage: A Survey", Pattern Recognition Letters, 2014.

[27] C. Guo, R. Zhuang, Y. Jie, K. R. Choo and X. Tang,"Secure Range Search Over Encrypted Uncertain IoT Outsourced Data", IEEE Internet of Things Journal, Volume: 6, Issue: 2, 2019.

[28] T.L. Haithcoat, W. Song, J.D. Hipple, “Building footprint extraction and 3D reconstruction from LiDAR data”, Proceedings of the Remote Sensing and Data Fusion over Urban”, pp. 74_78, Italy, 2001.

[29] F. Han, J. S. Jin, M. Wang, W. Jiang, L. Gao and L. Xiao, "A review of algorithms for filtering the 3D point cloud", Signal Processing: Image Communication, 103–112, 2017.

[30] S. M. Haque and V. M. Govindu, "Robust Feature-Preserving Denoising of 3D Point Clouds", Fourth International Conference on 3D Vision, 2016.

[31] X. Hengwei, F. Zhong-ke, S. Zhi-fang, “An automatic extraction algorithm based on 3D laser point cloud data of single tree canopy projection area and canopy volume”. Spectroscopy and Spectral Analysis, vol. 34, pp.465-470, 2014.

[32] X. Hu and L. Ye, "A Fast and Simple Method of Building Detection from LIDAR Data Based On Scan Line Analysis", Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume II-3/W1, 2013.

[33] J. Iglhaut, C. Cabo,S. Puliti, L. Piermattei, J. O’Connor, J. Rosette,"Structure from Motion Photogrammetry in Forestry: a Review", Current Forestry Report, 2019.

[34] M. Izadi and P. Saeedi, "Height estimation for buildings with complex contours in monocular satellite/airborne images based on fuzzy reasoning", IEEE International Conference on Image Processing, 2010.

[35] C. Jia, T. Yang, C. Wang, B. Fan and F. He, "A New Fast Filtering Algorithm for A 3D Point Cloud Based on RGB-D Information", PloS one, 2019.

[36] N. Jinling, Z. Linpeng, “Study on the Method of Volume Measurement of Irregular Volume Based on Scattered Data Points”, The 14th National Conference on Image Graphics, pp.42-46, 2008.

[37] S. K. Jumlesha, S. B. Khadar, C. Satyanarayana and M. V. Ramanaiah, "Automatic Urban Feature Extraction using Mathematical Morphology", The International Journal of Engineering Research and Applications (IJERA), ISSN: 2248-9622, Vol.2, Issue 3, pp. 221-225, 2012.

[38] J. Kilian, N. Haala, and M. Englich, "Capture and evaluation of airborne laser scanner data", International Archives of Photogrammetry and Remote Sensing (IAPRS), 1996.

[39] H. Kisner and U. Thomas , "Segmentation of 3D Point Clouds using a New Spectral Clustering Algorithm Without a-priori Knowledge", In Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP) - Volume 4: VISAPP, pages 315-322, 2018.

[40] T. Koch, M. Körner and F. Fraundorfer, "Automatic and Semantically-Aware 3D UAV Flight Planning for Image-Based 3D Reconstruction", Jounal of Remote Sensing, 2019.

[41] K. Kraus and N. Pfeifer, "Determination of terrain models in wooded areas with airborne laser scanner data", Journal of Photogrammetry & Remote Sensing, Vol.53, pp.193-203, 1998.

[42] X. Liu, R. Cao, Y. Zhang, Z. Zhao, " Roof Plane Extraction from Airborne LiDAR Point Clouds", International Journal of Remote Sensing, 38:12, 3684-3703, DOI: 10.1080/01431161.2017.1302112 , 2017.

[43] B. Li, X. Bi, C. Peng, Y. Chen, X. Zhao and C. Yang, "Calculating Point Cloud Object Volume using CoOpposite-Direction Slicing Method", Preprints 2019, 2019110253 (doi: 10.20944/preprints201911. 0253.v1).

[44] Y. C. Liu and T. S. Huang, "Determination of the Camera Location from 2D to 3D Line and Point Correspondences", IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 12, pp. 28-37, 1990.

[45] M. Maghaoumi, J. J. LaViola Jrr, K. Desingh and O. C. Jenkins, "GemSketch: Interactive Image-Guided Geometry Extraction from Point Clouds", IEEE International Conference on Robotics and Automation (ICRA), 2018.

[46] F. Mancini, M. Dubbini, M. Gattelli, F. Stecchi, S. Fabbri and G. Gabbianelli, "Using Unmanned Aerial Vehicles (UAV) for High-Resolution Reconstruction of Topography: The Structure from Motion Approach on Coastal Environments", Remote Sensing. 5. 6880-6898. 10.3390/rs5126880, 2013.

[47] J. Manuel, D. Christoph and P. Regina,"Hough Parameter Space Regularisation for Line Detection in 3D", International Conference on Computer Vision Theory and Applications (VISAPP), pp. 345-352, 2016.

[48] A.Moussa, N. El-Sheimy, "A new object based method for automated extraction of urban objects from airborne sensors data", Melbourne, Australia, 2012.

[49] E. Mugner and N. Seube, "Denoising of 3D Point Clouds", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-2/W17, 2019.

[50] A. Murtiyoso and P. Grussenmeyer, "Point Cloud Segmentation And Semantic Annotation Aided By Gis Data For Heritage Complexes ", Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLII-2/W9, 523-528, 2019.

[51] A. Murtiyoso, M. Koehl a, P. Grussenmeyer a, T. Freville, "Acquisition and Processing Protocols For UAV Images: 3D Modeling Of Historical Buildings Using Photogrammetry", Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-2/W2, 2017.

[52] M. Nakagawa, "Point Cloud Clustering Using Panoramic Layered Range Image", Recent Applications in Data Clustering, 2018.

[53] M. Pollefeys, L. V. Gool, M. Vergauwen, K. Cornelis, F. Verbiest and J. Tops, "3D recording for archaeological fieldwork", IEEE Computer Graphics and Applications, vol. 23, no. 3, pp. 20-27, 2003.

[54] M. Pollefeys and L. V. Gool, "From Images to 3D Models", Publishing of ACM, Vol. 45, Issue 7, 2002.

[55] H. Potemkin, Chiu, L. P. Kaelbling and T.Lozano-Pèrez, "Automatic class-specific 3D reconstruction from a single image. In CSAIL Technical Reports, 2009.

[56] M. H. Rahman, Q. Li, M. Pickering, M. Frater, D. Kerr, C. Bouchey and E. Delp, "Food volume estimation in a mobile phone based dietary assessment system", Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference, pp:988-995, 2012.

[57] M. Rakotosaona, V. L. Barbera, P. Guerrero, N. J. Mitra and M. Ovsjanikov, "PointCleanNet: Learning to Denoise and Remove Outliers from Dense Point Clouds", Major Revision from Eurographics Conference, 2019.

[58] M. Ruwaimana, B. Satyanarayana, V. Otero, A. M. Muslim, A. Syafiq and S. Ibrahim, "The Advantages of using Drones over Space-borne Imagery in the Mapping of Mangrove Forests", PLoS One, 2018.

[59] A. Saxena, M. Sun and A. Y. Ng, "Learning 3D Scene Structure from a Single Still Image". In Pattern Analysis and Machine Intelligence, IEEE Transactions, Vol.31, pp. 824–840, 2009.

[60] S. P. Singh, K. Jain and V. R. Mandla, "Image Based 3D City Modeling: Comparative Study", International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol.XL-5, pp.23-25, 2014.

[61] S. P. Singh, Kamal Jain and V. Ravibabu Mandla "Virtual 3D City Modeling: Techniques and Applications", International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-2/W2, Istanbul, Turkey, 2013.

[62] G. Sithole, "Filtering of Laser Altimetry Data using A Slope Adaptive Filter", International Archives of Photogrammetry and Remote Sensing (IAPRS), Vol. 34, 3W4, 2001.

[63] G. Strang, "Linear Algebra and its Application", 3rd Edition, Saunders HBJ, 1988.

[64] C. Strecha, "Pix4D - Error Estimation", February 2014.

[65] C. Stucker, A. Richard, J. D. Wegner and K. Schindler, "Supervised Outlier Detection in Large-Scale MVS Point Clouds For 3D City Modeling Applications", Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume IV-2, 2018.

[66] J. Su and L. L. Tang, "Shape Estimation from 3D Point Clouds", "Intelligent Data Analysis and Its Applications, Advances in Intelligent Systems and Computing, Volume 1, 39, 2014.

[67] K. Suresh, K. Lodha, A. Andrew and A. Bickett, “3D Geospatial Visualization of the UCSC Campus”, ASPRS 2005 Annual Conference, “Geospatial Goes: Global: From Your Neighborhood to the Whole Planet”, pp. 7-11, 2005.

[68] B. C. Z. T˚umov´a, "Object Volume Calculation from Large Noisy Point-Clouds", Faculty of Electrical Engineering, Czech Technical University in Prague, Thesis, May 2018.

[69] A. Tandon, "Post-disaster Damage Assessment of Cultural Heritage: Are We Prepared?", 18th Triennial Conference, Copenhagen, 2017.

[70] V. Tao and Y. Hu, “A Review of Post-processing Algorithms for Airborne LIDAR Data”, Proceedings of ASPRS Annual Conference, pp.23-27, 2001.

[71] H. Teleb and G. Chang, "Data glove integration with 3D virtual environments", International Conference on Systems and Informatics (ICSAI2012), 2012.

[72] C. Teutsch, E. Trostmann and D. Berndt, “A parallel point cloud clustering algorithm for subset segmentation and outlier detection”, Proceedings of SPIE - The International Society for Optical Engineering, 2011.

[73] D. Tóvári, N. Pfeifer, “Segmentation based robust interpolation-a new approach to laser data filtering”, IAPRS, 2005.

[74] A. Vafadari, G. Philip and R. Jennings, "Damage Assessment And Monitoring Of Cultural Heritage Places In A Disaster And Post-Disaster Event – A Case Study Of Syria",The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-2/W5, 2017.

[75] J. Vince, "Cartesian Coordinates", In: Mathematics for Computer Graphics. Springer, pp 23-29, London, 2001.

[76] J. Vince, "Essential Mathematics for Computer Graphics fast", Springer, London, doi.org/10.1007/978-1-4471-3685-9, 2001.

[77] G. Vosselman, “Slope based filtering of laser altimetry data”, IAPRS, Vol.33, 2000.

[78] U. Weidner, and W. Förstner, “Towards automatic building extraction from high resolution digital elevation models”, Journal of Photogrammetry & Remote Sensing, Vol.50, pp. 38-49, 1995.

[79] H. Wenming, L. Yuanwang and W. Peizhi, "Algorithm for 3D Point Cloud Denoising", Third International Conference on Genetic and Evolutionary Computing, 2009.

[80] K. Wolff, C. Kim, H. Zimmer, C. Schroers, M. Botsch, O. S. Hornung and A. S. Hornung, "Point Cloud Noise and Outlier Removal for Image-Based 3D Reconstruction", Fourth International Conference on 3D Vision (3DV), 2016.

[81] Y. Xie, J. Tian and X. X. Zhu, "A Review of Point Cloud Semantic Segmentation", IEEE Geoscience and Remote Sensing Magzine, 2019.

[82] S. Xu, R. Wang and H. Zheng, "Road Curb Extraction from Mobile LiDAR Point Clouds", IEEE Transactions on Geoscience and Remote Sensing, Volume: 55, Issue: 2, 2017.

[83] Z. Yan, Rufei Liu, Liang Cheng, Xiao Zhou, X. Ruan and Y. Xiao, “A Concave Hull Methodology for Calculating the Crown Volume of Individual Trees Based on Vehicle-Borne LiDAR Data”, Journal of Remote Sensing, 2019.

[84] B. Yang, J. Wang, R. Clark, Q. Hu, S. Wang, A. Markham and N. Trigoni, "Learning Object Bounding Boxes for 3D Instance Segmentation on Point Clouds", 33rd Conference on Neural Information Processing Systems, Canada, 2019.

[85] M. Yang and S. Gan, "Research on Denoising Algorithms Based on Point Cloud Data of Terrestrial 3D Laser Scanner", Revista de la Facultad de Ingeniería U.C.V., Vol. 32, N°11, pp. 894-899, 2017.

[86] Y. Yu, J. Li, H. Guan and C. Wang, "Semiautomated Extraction of Street Light Poles from Mobile LiDAR PointClouds", IEEE Transactions on Geoscience and Remote Sensing, March 2015.

[87] F. Yu, H. Luo, Z. Lu and P. Wang, "Three-Dimensional Model Analysis and Processing", Advanced Topics in Science and Technology in China Book Series, 2010.

[88] J. Zeng, G. Cheung, M. Ng, J. Pang and C. Yang, "3D Point Cloud Denoising Using Graph Laplacian Regularization of a Low Dimensional Manifold Model", IEEE Computer Vision and Pattern Recognition, 2019.

[89] K. Zhang, W. Bi, X. Zhang, X. Fu, K. Zhou and L. Zhu, "A New K-means Clustering Algorithm for Point Cloud", International Journal of Hybrid Information Technology, Vol.8, No.9, pp.157-170, 2015.

[90] Z. Zhang, J. Wu and Y. Zhang, "Multi-View 3D Model Generation with Image Sequences", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 14, 2014.

[91] Y. Zhang, Z. Zhang, J. Zhang and J. Wu, "3D Building Modelling with Digital Map, Lidar Data and Video Image Sequences", The Photogrmmetric record, Volume 20, Issue 111, pp 285-302, 2005.

[92] B. Zhao, "AIFD Based 2D Image Registration to Multi-View Stereo Mapped 3D Models", Neural Process Letter 48, 1261–1279, 2018.

[93] Y. Zhi, Y. Zhang, H. Chen, K. Yang and H. Xia, "A Method of 3D Point Cloud Volume Calculation Based on Slice Method", International Conference on Intelligent Control and Computer Application, 2016.

[94] L. Zhu,D. P. Lusch and R. Shi, "Feature extraction from 3D LiDAR point clouds using image processing methods", Proceedings of SPIE - The International Society for Optical Engineering, 2011.

[95] Z. Zuxun, W. Jun, Z. Yong and Z. Yongjun, "Multi-view 3D City Model Generation with Image Sequences", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol.34, 2002.